Université Paris Cité

École doctorale de sciences mathématiques de Paris centre

Thèse de doctorat

Discipline : Mathématiques

présentée par

Emmanuel Rauzy

Perspectives on an effective theory of groups

---

dirigée par Andrzej Zuk

Soutenue le 24 Juin 2022 devant le jury composé de :

Mme Laura Ciobanu	Heriot-Watt University	Examinatrice
M. Vincent Guirardel	Université de Rennes 1	Rapporteur
Mme Delaram Kahrobaei	City University of New York	Examinatrice
M. Mathieu Hoyrup	LORIA	Examinateur
M. Henry Wilton	Cambridge University	Rapporteur
M. Andrzej Zuk	Institut de mathématiques de Jussieu-PRG	Directeur

Institut de mathématiques de
Jussieu-Paris Rive gauche
UMR 7586.
Batiment Sophie Germain
8 place Aurélie Nemours
75013 Paris

Université de Paris.
École doctorale de sciences
mathématiques de Paris centre.
Boîte courrier 290
4 place Jussieu
75 252 Paris Cedex 05

Son cerveau était rempli de relations mathématiques, d'équations différentielles, de lois des probabilités, de la théorie des nombres. [...]
Et dans le même temps, vivaient dans sa tête et le bruit des feuilles dans les arbres et le clair de lune, et la bouillie de sarrasin au lait, et le ronflement du feu dans le poêle, et des bribes de mélodies, et des aboiements de chiens, et le sénat de Rome, et les bulletins du *Sovinformburo*, et la haine de l'esclavage, et l'amour pour les graines de potiron.

Vassili Grossman*, Vie et Destin.*

Remerciements

Je tiens d'abord à remercier mon directeur, Andrzej Zuk, dont le premier enseignement a été : « En mathématique, il faut faire ce que l'on aime ». J'ai essayé de ne pas trop déroger à cette règle.

Je remercie ensuite Vincent Guirardel et Henry Wilton, pour leurs lectures attentives de ce manuscrit, je suis très heureux et honoré de l'intérêt qu'ils ont bien voulu porter à mon travail.

Je remercie Laura Ciobanu, Delaram Kahrobaei et Mathieu Hoyrup, qui m'ont fait l'honneur d'accepter de faire partie de mon jury.

Je remercie aussi les membres de l'équipe d'algèbres d'opérateurs, en particulier mon tuteur Pierre Fima, mais aussi François Le Maître, Romain Tessera, George Skandalis.

Je remercie les doctorants de l'imj-prg, en particulier Kevin, Wu Yilin, Amandine et Oussama, avec qui nous avons passé de bon moments à Bilbao et à St. Jacut, Gabriel, Leonardo, Antoine, Tommaso, Jérémie, Mingkun, Macarena, Pierre, Théo, Elie, Jérome (dont je n'ai toujours pas compris le sujet de thèse), ainsi que la nouvelle génération de doctorants, qui, évidemment, sont encore très naïfs : Alexandre, Tal, Razvan, Juan.

Je remercie ensuite mes amis de l'ENS, dont beaucoup, ayant commencé leurs thèses un an avant moi, m'ont ouvert la voie : Laura, Alice et Jean, Malo, Gwenaël, Théo et Marion, Alban, et Marie, Clara, Théotime, Arthur, Jean, Solène et Damien, Auguste et Solène.

Je remercie particulièrement Pierre-Louis, qui, à Bilbao, à un moment où je n'avais pas encore trouvé mes repères, m'a montré, par son enthousiasme et son énergie, la joie que pouvait être la recherche.

Merci à Luca, qui a été au Japon pour me ramener de la liqueur de yuzu, à Assaf, qui m'a appris tout ce que je sais de latex.

Merci à Dorian et à Léo, à Adrian, je remercie encore Adrien, Clara, Romain, Antoine, Léon, et Réouven.

Je remercie ma famille, dont mes soeurs Éliette et Élise, ma grand-mère Monique, mon père Jean-Baptiste, ma belle-mère Léa, ma mère Irène, qui a beaucoup relu mon anglais, mon beau-père Dominique, ils m'ont suivi pendant toute cette thèse, jalonnée de confinements et de fatigues pré- et post-virus.

Finalement, je remercie Camille, sans qui faire des thèses, etc, ne servirait pas à grand-chose.

Résumé

Cette thèse rassemble les résultats obtenus dans quatre articles achevés entre 2020 et 2021 qui portent sur la théorie de la calculabilité sur les groupes.

Y est d'abord décrite une nouvelle approche pour l'étude des problèmes de calculabilité portant sur des groupes qui ne sont pas nécessairement de présentation finie. Cette approche repose principalement sur deux idées. Tout d'abord, l'idée que les présentations récursives ne sont pas des descriptions de groupes suffisamment puissantes pour servir comme base à une théorie des problèmes de décisions qui soit intéressante, idée justifiée par l'obtention des théorèmes de Rice et de Rice-Shapiro pour les groupes donnés par des présentations récursives. La deuxième remarque fondamentale sur laquelle est basée notre approche est la description d'une caractérisation algorithmique des groupes de présentation finie. Cette caractérisation prouve que les descriptions de groupes obtenues en termes d'algorithmes ne sont pas nécessairement des descriptions plus faibles que les autres, puisque la donnée d'une présentation finie peut aussi bien être vue comme la donnée d'algorithmes. On donne différentes directions de recherches qui suivent de ces remarques préliminaires.

Le deuxième axe de cette thèse est l'étude des groupes dont les quotients finis peuvent être énumérés. La caractérisation algorithmique des groupes de présentation finie nous montre que cette famille de groupes constitue une généralisation algorithmique naturelle de la famille des groupes de présentation finie. On montre d'abord que la propriété d'avoir les quotients énumérables est indépendante, pour les groupes résiduellement finis, de la propriété d'avoir le problème du mot résoluble. On donne des exemples de groupes bien connus dont les quotients finis peuvent être énumérés. On utilise finalement l'étude des énumérations de quotients finis pour donner un groupe résiduellement fini, avec le problème du mot résoluble, qui ne peut pas se plonger dans un groupe résiduellement fini de présentation finie, répondant ainsi à une question due à Olga Kharlampovich, Alexei Myasnikov et Mark Sapir.

Le dernier axe de cette thèse est l'étude des propriétés de groupes qui sont décidables à partir d'algorithmes donnant la solution au problème du mot. On relie ce problème à la théorie de l'analyse calculable, il correspond exactement à l'étude de l'analyse calculable markovienne sur l'espace topologique des groupes marqués. On montre que l'espace des groupes marqués est un espace polonais qui n'est pas un *espace polonais effectif.* Il constitue le premier exemple d'un tel espace qui soit un exemple naturel, c'est-à-dire qui ne soit pas un espace introduit uniquement pour ses mauvaises propriétés algorithmiques. Le principal problème ouvert dans la théorie de l'analyse calculable sur les groupes est le problème de la continuité : est-il vrai que toute fonction calculable définie sur l'espace des groupes marqués y est continue ?

Mots clés : Calculabilité ; Espace des groupes marqués ; Théorème de plongement de Higman ; Groupes de présentation finie.

Abstract

In this thesis, we present results about the theory of decision problems for groups that were obtained in four articles submitted between 2020 and 2021.

We describe a new approach to the theory of decision problems for infinitely presented groups, which relies on an algorithmic characterization of finitely presented groups, and on the remark that recursive presentations are descriptions that are too weak to be the basis of an interesting theory, remark justified by a Rice Theorem for recursively presented groups.

We then study residually finite groups whose finite quotients can be effectively enumerated, as this class of groups constitutes a particular algorithmic generalization of the class of finitely presented groups. We relate this to the search of a Higman Embedding Theorem for residually finite groups, we give necessary conditions for a group to embed in a finitely presented residually finite group that are more restrictive than previously known conditions, answering a question of Olga Kharlampovich, Alexei Myasnikov and Mark Sapir.

Finally, we study decision problems for groups described by word problem algorithms, and show that this corresponds to the study of Markovian computable analysis on the space of marked groups. We provide the first systematical method to build, given a group property that satisfies some appropriate conditions, a finitely presented group with solvable word problem in which the finitely generated subgroups that satisfy the chosen property cannot be recognized. We also prove that the space of marked groups is a Polish space, but not an effective Polish space, it constitutes the first natural example of such a set. Because of this, none of the results of computable analysis that serve to prove that computable functions are continuous can be applied to the space of marked groups. We thus ask whether all computable functions defined on the space of marked groups must be continuous. The link between this problem and the theory of Kolmogorov complexity is made explicit.

Key words: Computability theory; Space of marked groups; Higman's Embedding theorem; Finite presentations of groups.

Contents

Introduction (Introduction Française)

Cette thèse porte sur l'étude des problèmes de décisions sur les groupes, c'est-à-dire sur la théorie située à mi-chemin entre la théorie des groupes et la théorie de la calculabilité. Y sont rassemblés les résultats obtenus dans quatre articles, la présente introduction en français contient un résumé du contenu de ces articles.

Remarques et problèmes concernant les descriptions algorithmiques de groupes. Lorsque Max Dehn introduit, en 1911, les problèmes de décisions en théorie des groupes, c'est uniquement pour étudier les groupes de présentation finie, son idée est que les solutions aux problèmes du mot et de l'isomorphisme permettront d'établir une théorie générale des groupes de présentation finie. Pourtant, si l'on veut développer une théorie de la calculabilité sur les groupes de la manière la plus générale possible, il faudra au contraire être capable de travailler avec des groupes qui ne sont pas de présentation finie, puisque de nombreux exemples de groupes qui ont, du point de vue de la calculabilité, de très bonnes propriétés, ne sont pas de présentation finie.

Le premier problème que l'on se pose ici est donc de décrire un peu la théorie de la calculabilité sur les groupes, telle qu'elle aurait été développée si elle n'avait pas eu pour but premier l'étude des groupes de présentation finie -qui sont exactement, rappelons-le, les groupes fondamentaux des variétés fermées, cette remarque nous pousse à dire que le premier moteur de la théorie des problèmes de décisions pour les groupes fut *la topologie*.

On va donc dans cette partie décrire quelques pistes pour le développement d'une véritable théorie algorithmique des groupes.

La première étape de ce programme est de tenter de supprimer le concept de présentation récursive.

En effet, dans cette partie de la thèse, on emploie le terme *algorithme d'énumération du neutre* pour désigner un algorithme qui énumère toutes les manières d'écrire l'élément neutre dans un groupe marqué fixé. Cet algorithme donne en fait la présentation récursive maximale pour ce groupe, celle où toutes les relations sont énumérées. On préfère parler d'algorithme d'énumération du neutre plutôt que de présentation récursive à cause de la confusion possible (et dont on peut citer des exemples dans la littérature) entre la présentation elle-même, couple générateurs - relations, et un algorithme qui énumère ces relations. Ainsi, lorsque l'on parle d'un groupe « donné par une présentation récursive », on parle en fait d'un groupe donné par un algorithme qui énumère cette présentation, et plusieurs algorithmes peuvent correspondre à une même présentation.

On établit tout d'abord la proposition suivante :

Proposition. *Le groupe trivial ne peut être distingué du groupe d'ordre deux lorsque ces groupes sont décrits par des algorithmes d'énumération du neutre.*

Cette proposition peut être étendue à tout couple de groupes, à condition que l'un des deux soit un quotient strict du deuxième. Cela permet d'établir les théorèmes de Rice et de Rice-Shapiro pour les groupes de présentation récursive :

Théorème. *Les seules propriétés décidables pour les groupes décrits par algorithmes d'énumération du neutre sont les propriétés triviales : la propriété vide, et celle satisfaite par tous les groupes.*

Théorème. *Si P est une propriété semi-décidable pour les groupes décrits par algorithmes d'énumération du neutre, alors il existe une suite récursivement énumérable de groupes de présentation finie telle que P consiste exactement en l'ensemble des quotients des groupes de cette suite.*

Ces résultats montrent que la description d'un groupe par un algorithme d'énumération du neutre est trop faible pour permettre de baser une théorie intéressante sur ce genre de description.

On pourrait, à ce moment précis de notre travail, penser que les descriptions de groupes en termes d'algorithmes ne donneront jamais des descriptions contenant assez d'information pour obtenir des résultats intéressants. On va pourtant ici montrer que ça n'est pas le cas, en donnant une caractérisation algorithmique des groupes de présentation finie.

Pour cela, on doit d'abord définir ce qu'on appelle un algorithme de reconnaissance des quotients marqués.

Définition. Étant donné un groupe G et une famille génératrice finie S de G, un algorithme de reconnaissance des quotients marqués de G pour la famille S est un algorithme qui, étant donné un algorithme d'énumération du neutre pour un groupe H par rapport à une famille génératrice S' et une surjection $f : S \to S'$, va s'arrêter si et seulement si la fonction f peut être étendue en un morphisme de groupe allant de G vers H.

On montre alors le théorème important suivant :

Théorème. *Un groupe admet un algorithme d'énumération du neutre et un algorithme de reconnaissance des quotients marqués si et seulement si c'est un groupe de présentation finie. De plus, il existe une procédure effective pour, étant donnée une présentation finie d'un groupe, obtenir ces deux algorithmes pour ce groupe, et réciproquement, il existe une procédure qui, étant donnés un algorithme d'énumération du neutre et un algorithme de reconnaissance des quotients marqués pour un groupe, ces deux algorithmes étant définis par rapport à une même famille génératrice, produit une présentation finie de ce groupe par rapport à cette famille génératrice.*

Ce théorème ne permet en fait pas d'obtenir de nouveaux résultats concernant les groupes de présentation finie. Son but est plutôt de permettre de définir et d'étudier différentes généralisations algorithmiques du concept de groupe de présentation finie.

On va ainsi décrire une hiérarchie des descriptions algorithmiques de groupes, qui est définie ainsi : les problèmes de décision qui portent sur les éléments d'un groupe fixé (problèmes du mot, de conjugaison, de l'ordre, etc.) sont les problèmes d'ordre 0. Un algorithme qui donne la solution à un problème d'ordre 0 dans un groupe fournit de l'information sur ce groupe, et lorsque cette information est suffisante pour définir uniquement ce groupe (c'est le cas pour les problèmes du mot, de conjugaison, de l'ordre, etc.), cet algorithme constitue alors une description d'ordre 0 du groupe qu'il définit.

On peut alors définir les problèmes d'ordre 1 : ce sont les problèmes qui portent sur des groupes décrits par des descriptions d'ordre 0. Et lorsque la solution à un tel problème donne une description d'un groupe, cette description est une description d'ordre 1. C'est le cas par exemple de l'algorithme de reconnaissance des quotients marqués, qui est une solution au problème d'ordre 1 « reconnaître les quotients d'un groupe donné », lorsque ces quotients sont donnés par la description d'ordre 0 constituée d'un algorithme d'énumération du neutre.

Les problèmes d'ordre 2 sont alors les problèmes portant sur les groupes décrits par des descriptions d'ordre 1, etc.

Le théorème de caractérisation des groupes de présentation finie montre que les présentations finies sont des descriptions d'ordre 1, et le problème de l'isomorphisme est ainsi un problème d'ordre 2.

La description de cette hiérarchie est informelle, et je n'ai pour le moment pas réussi à obtenir des définitions précises correspondant aux notions décrites plus haut. Son caractère informel ne prive pourtant pas cette hiérarchie de tout intérêt. En effet, en introduisant une dualité entre les notions de « descriptions de groupes en termes d'algorithmes » et de « problèmes portant sur les groupes », et puisque les problèmes de décisions pour les groupes sont toujours étudiés au sein de différentes classes de groupes, cette hiérarchie nous invite naturellement à l'exploration des descriptions de groupes relatives à certaines classes de groupes.

On introduit donc le concept d'*algorithme de reconnaissance des quotients marqués relatif à une classe de groupes*. Là où l'algorithme de reconnaissance des quotients marqués donnait la solution au problème « reconnaître (tous) les quotients d'un groupe donné », un algorithme de reconnaissance des quotients marqués relatif à une certaine classe de groupes donnera la solution au problème « reconnaître les quotients d'un groupe donné dans la classe choisie ». Dans certains cas, cette notion n'apporte rien de nouveau : par exemple dans le cas des variétés de groupes, où on ne fait que retrouver la notion bien connue de présentation finie dans la variété. On dit d'un algorithme de reconnaissance des quotients marqués relatif à une classe de groupes qu'il est *élémentaire* si l'ensemble des groupes qu'il reconnaît peut s'écrire comme l'ensemble des groupes qui satisfont un nombre fini et fixé de relations choisies. Les algorithmes non-élémentaires sont ceux qui fournissent une généralisation purement algorithmique du concept de présentation finie.

On clôt ce chapitre avec les résultats suivants :

Théorème. *Tout algorithme de reconnaissance des quotients marqués relatif à une variété définie par un nombre fini d'identités est élémentaire.*

Théorème. *Le groupe de l'allumeur de réverbères admet un algorithme de reconnaissance des quotients marqués relatif à l'ensemble des groupes finis qui n'est pas élémentaire.*

Sur les groupes de type fini dont les quotients finis sont effectivement énumérables. Le cas de l'allumeur de réverbères, dont l'algorithme de reconnaissance des quotients marqués relatif aux groupes finis est non-élémentaire, est très intéressant, et nous pousse à développer, dans le troisième chapitre de cette thèse, l'étude des groupes dont les quotients finis sont effectivement énumérables.

Cette étude est principalement motivée par l'algorithme de McKinsey, qui est l'algorithme qui permet de montrer que tous les groupes résiduellement finis de présentation finie ont le problème du mot résoluble. La méthode employée par cet algorithme est, étant fixé un élément d'un groupe donné, d'énumérer tous les quotients finis de ce groupe, à la recherche d'un groupe fini dans lequel l'image de l'élément est non-triviale. Plusieurs variations de cet algorithme ont été étudiées, pour les groupes dont les classes de conjugaisons sont *séparées*, c'est-à-dire fermées dans la topologie profinie, et pour les groupes dont les sous-groupes de type fini sont séparés.

Mais l'étude des groupes dont les classes de conjugaisons ou les sous-groupes sont fermés dans la topologie profinie ne s'est pas restreinte aux groupes de présentation finie, ce qui a engendré des fautes dans plusieurs articles : les auteurs oublient que, pour utiliser l'algorithme de McKinsey dans des groupes qui ne sont pas de présentation finie, il faut au préalable avoir donné une méthode d'énumération des quotients finis pour ces groupes.

Dans ce chapitre sont donnés des exemples de groupes dont on peut énumérer les quotients finis (produits en couronnes, groupes ayant des présentations de Lysionok, etc). On montre surtout que les propriétés « avoir les quotients finis énumérables » et « avoir le problème du mot résoluble » sont indépendantes pour les groupes résiduellement finis :

Théorème. *Il existe un groupe résiduellement fini avec le problème du mot résoluble dont on ne peut pas énumérer les quotients finis.*

Théorème. *Il existe un groupe résiduellement fini dont on peut énumérer les quotients finis, mais qui n'a pas le problème du mot résoluble.*

Ces deux théorèmes sont obtenus à partir d'une construction, due à Dyson, d'une famille de groupes indexés par des parties de $\mathbb{Z}$, et pour lesquels différentes propriétés d'un groupe peuvent être lues dans les propriétés de la partie de $\mathbb{Z}$ correspondante, en particulier on peut dire exactement quand ces groupes sont résiduellement finis, ont le problème du mot résoluble, les quotients finis énumérables, etc. Ces groupes sont obtenus comme *doubles* du groupe de l'allumeur de réverbères, c'est-à-dire en faisant un produit libre amalgamé de deux copies de l'allumeur de réverbères, en identifiant un sous-groupe du premier allumeur de réverbères avec sa copie présente dans le deuxième allumeur de réverbères.

Il n'y a pas de théorème de plongement de Higman pour les groupes résiduellement finis avec le problème du mot résoluble. Un corollaire du théorème de McKinsey, que les groupes résiduellement finis de présentation finie ont le problème du mot résoluble, est le fait qu'il existe des groupes résiduellement finis qui sont de présentation récursive mais qui ne peuvent pas se plonger dans des groupes résiduellement finis de présentation finie. En effet, un des premiers résultats de Dyson obtenu grâce à la construction citée ci-dessus était de construire un groupe résiduellement fini, de présentation récursive, mais n'ayant pas le problème du mot résoluble (un tel groupe a d'ailleurs été obtenu indépendamment par Meskin dans [**80**]). Comme « avoir le problème du mot résoluble » est une propriété dont les sous-groupes de type fini d'un groupe donné héritent, ces groupes ne peuvent pas se plonger dans un groupe résiduellement fini de présentation finie.

Ces considérations nous amènent à faire le lien avec le théorème de plongement de Higman, qui donne une caractérisation des sous-groupes des groupes de présentation finie, en prouvant que tout groupe de présentation récursive se plonge dans un groupe de présentation finie.

La formulation la plus naïve d'un hypothétique théorème de plongement de Higman pour les groupes résiduellement finis, qui serait « tout groupe résiduellement fini de présentation récursive se plonge dans un groupe résiduellement fini de présentation finie », est ainsi invalidée par le théorème de McKinsey.

Une formulation très proche, permettant d'éviter les contre-exemples de Dyson et Meskin, serait : « Tout groupe résiduellement fini avec le problème du mot résoluble se plonge dans un groupe résiduellement fini de présentation finie ». Dans l'article [**60**], Kharlampovich, Myasnikov et Sapir demandent s'il est possible que le théorème ainsi énoncé soit vrai.

Dans le Chapitre 4 de cette thèse, on montre que ce n'est pas le cas. En effet, on exhibe une propriété des groupes résiduellement finis de présentation finie, qui se transmet à leurs sous-groupes, et qui n'est pas automatiquement acquise aux groupes ayant le problème du mot résoluble.

Un groupe est dit *effectivement résiduellement fini* s'il existe un programme qui, étant donné un élément non-trivial de ce groupe, peut produire un quotient fini de ce groupe dans lequel l'image de l'élément est non-triviale.

Cette propriété est plus faible que celle d'avoir les quotients finis énumérables, on demande en fait qu'un groupe ait, non pas tous ses quotients finis énumérables, mais suffisamment de quotients énumérables, pour que l'on puisse distinguer tous ses éléments dans des quotients finis. Contrairement à la propriété d'avoir les quotients finis énumérables, cette propriété se transmet aux sous-groupes.

Le principal résultat de ce chapitre est la construction d'un groupe résiduellement fini avec le problème du mot résoluble mais qui n'est pas effectivement résiduellement fini.

Théorème. *Il existe un groupe résiduellement fini avec le problème du mot résoluble mais qui n'est pas effectivement résiduellement fini. En particulier, ce groupe ne peut pas se plonger dans un groupe résiduellement fini de présentation finie.*

On conjecture bien entendu que la condition nécessaire nouvellement formulée pour qu'un groupe puisse être un sous-groupe d'un groupe résiduellement fini de présentation finie est aussi suffisante.

Conjecture (Théorème de plongement de Higman pour les groupes résiduellement finis). Tout groupe résiduellement fini qui a le problème du mot résoluble et qui est effectivement résiduellement fini se plonge dans un groupe résiduellement fini de présentation finie.

Ce chapitre se termine par une étude de la fonction de profondeur du groupe non effectivement résiduellement fini construit pour le théorème ci-dessus. Cette fonction mesure la vitesse à laquelle ce groupe peut être approximé par ses quotients finis : à un entier n, on associe la taille du plus petit quotient fini de ce groupe dans lequel tous ses éléments non-triviaux de longueur au plus n ont une image non-triviale. Le groupe construit plus haut est *très mal* approximé par ses quotients finis :

Théorème. *Il existe un groupe résiduellement fini avec le problème du mot résoluble dont la fonction de profondeur croît plus vite que toutes les fonctions récursives.*

Analyse calculable sur l'espace des groupes de type fini. Le dernier chapitre de cette thèse présente la théorie de l'analyse calculable sur l'espace des groupes marqués. Cette formule ne désigne en vérité rien d'autre que l'étude des problèmes de décisions posés à propos de groupes décrits par des algorithmes résolvant le problème du mot.

L'analyse calculable a été commencée par Turing, dans l'article même où il décrit ses machines, quand il propose d'étudier les nombres réels dont on peut algorithmiquement trouver des approximations arbitrairement bonnes. La définition des réels calculables permet de définir par la suite les fonctions définies sur $\mathbb{R}$ qui sont calculables : ce sont celles pour lesquelles il est possible, étant donnée une machine qui approxime un réel donné en entrée, de produire une machine qui produira, elle, des approximations de l'image de ce réel. Cette notion de calculabilité pour les fonctions réelles n'est en fait pas la seule à avoir été considérée, mais c'est la seule qui nous ait intéressés ici, puisque c'est elle qui permet de traduire directement des questions de calculabilité sur les groupes en termes de problèmes d'analyse calculable. Cette approche s'appelle l'analyse calculable markovienne, du nom du constructiviste russe Markov, à qui l'on doit, entre autres, une notion d'algorithme équivalente à celles de Church et de Turing, une construction d'un semi-groupe avec le problème du mot non résoluble, la preuve que le problème de l'homéomorphisme n'est pas résoluble pour les variétés de dimensions 4 ou plus (quand elles sont décrites par des triangulations), et le lemme qui porte son nom et qu'on utilisera avantageusement dans notre étude :

Lemme (Markov, [**76**]). Les fonctions calculables définies sur un espace métrique récursif effectivement complet ne peuvent pas avoir de discontinuités effectives.

Nous n'avons pas la place, dans cette introduction, pour définir tous les termes qui apparaissent dans l'énoncé du lemme de Markov. Notons seulement qu'en analyse calculable, on peut montrer dans plusieurs contextes assez généraux que *les fonctions calculables sont continues,* le lemme de Markov doit être vu comme un premier pas dans cette direction. Nous reviendrons tout de même dans cette introduction sur le problème de la continuité des fonctions calculables.

La théorie générale de l'étude des fonctions calculables sur les *espaces métriques calculables* et sur les *espaces polonais effectifs* repose sur l'utilisation de *numérotations,* et si l'on voulait définir ces concepts ici, il n'y aurait pas d'autre possibilité que de reproduire en son intégralité le chapitre 1 de préliminaires et la première moitié de notre chapitre 5.

On ne donne ici que les premières définitions concernant les numérotations :

Définition. Une numérotation d'un ensemble X est une fonction ν définie sur un sous-ensemble de $\mathbb{N}$ et dont l'image est une partie de X. On note cela par $\nu :\subseteq \mathbb{N} \to X$.

Définition. Un antécédent d'un point x de X pour ν est une *description de x pour ν*, ou encore un *nom de x pour ν*.

Définition. Si (X, ν) et (Y, μ) sont deux ensembles munis de numérotations, et si f est une fonction de X vers Y, on dit que *f est calculable par rapport à ν et μ* s'il existe une fonction récursive F qui étant donnée la description d'un point de x pour ν permet de calculer une description de $f(x)$ pour μ.

Ces trois définitions sont fondamentales dans l'étude de la calculabilité en mathématiques, mais on s'en sert la plupart du temps implicitement : ainsi, dire que l'on peut calculer l'abélianisé d'un groupe à partir d'une présentation finie revient à affirmer que la fonction qui à un groupe associe son abélianisé est calculable pour la numérotation associée aux présentations finies, mais il n'y a pas besoin d'expliciter cette seconde formulation pour comprendre la première.

Si l'on donne dans le chapitre 1 de cette thèse une introduction conséquente sur les numérotations, c'est notamment parce que les définitions que l'on y présente sont plus générales que celles habituellement considérées dans la littérature, et sont plus appropriées à l'étude des groupes que les définitions habituellement utilisées.

En particulier, on ne demande pas aux numérotations d'être surjectives, contrairement à ce que l'on peut trouver partout ailleurs dans la littérature. Grâce à nos définitions, l'étude des numérotations de l'ensemble des groupes marqués peut être vue comme l'étude des descriptions finies de groupes. Au contraire, l'idée selon laquelle les numérotations doivent être des fonctions surjectives est une idée héritée des constructivistes, qui considèrent que seuls les objets ayant des descriptions finies existent, mais qui a perduré alors même que les méthodes actuellement utilisées dans la théorie des énumérations ne sont plus du tout constructives.

L'espace des groupes marqués. Fixons un entier k non nul. Un *k-groupe marqué* est une paire constituée d'un groupe et d'un k-uplet d'éléments de ce groupe qui en forme une partie génératrice. Un morphisme de groupes marqués entre deux k-groupes marqués est un morphisme de groupe, au sens habituel, qui doit de plus envoyer l'uplet qui génère le premier groupe sur l'uplet qui génère le second groupe.

On définit ainsi la catégorie des groupes marqués. Remarquons ici que la plupart des résultats de cette thèse, et non seulement ceux du chapitre 5, concernent en fait plus l'étude des groupes marqués que celle des groupes.

Pour définir la topologie de l'espace des k-groupes marqués, il est utile de fixer un groupe libre de rang k muni d'une base. La donnée d'une famille génératrice de k éléments d'un groupe est alors équivalente à la donnée d'un morphisme surjectif du groupe libre choisi vers notre groupe. On démontre alors facilement que deux groupes marqués sont isomorphes, en tant que k-groupes marqués, si et seulement si ils sont définis par des morphismes dont les noyaux sont identiques. Cela permet ainsi d'identifier l'ensemble des classes d'isomorphismes de k-groupes marqués avec l'ensemble des sous-groupes distingués d'un groupe libre.

Or l'ensemble des parties d'un groupe libre est naturellement muni de la topologie produit, qui en fait un espace de Cantor, en choisissant une bijection entre le groupe libre et $\mathbb{N}$, on obtient une bijection explicite entre les parties du groupe libre et l'espace de Cantor $\{0,1\}^{\mathbb{N}}$. L'ensemble des classes d'isomorphismes de k-groupes marqués peut ainsi être identifié avec un sous-ensemble du Cantor, on a donc naturellement une topologie et une métrique sur les groupes marqués. L'ensemble des classes d'isomorphismes de k-groupes marqués muni de la métrique héritée du Cantor est noté $\mathcal{G}_k$. Un groupe marqué vu comme élément du Cantor est donc décrit par une suite infinie de zéros et de uns, que l'on appelle *l'écriture binaire* d'un groupe.

L'étude de l'analyse calculable sur l'espace des groupes marqués est l'étude de ce que l'on peut dire d'un groupe à partir d'un algorithme qui donne son écriture binaire.

La première partie de mon travail sur l'espace des groupes marqués, qui forme ici plus de la moitié du chapitre 5, a consisté à rattacher l'étude de l'analyse calculable sur les groupes au cadre général de l'analyse calculable.

La majorité des résultats importants que l'on montre sont des résultats négatifs : on montre qu'on ne sait pas montrer que les fonctions calculables définies sur l'espace des groupes marqués sont continues.

Théorème. *L'espace des groupes marqués n'a pas de suite dense calculable, le théorème de Ceitin ne peut donc pas lui être appliqué.*

Théorème. *L'espace des groupes marqués ne satisfait pas la condition d'axiome du choix effectif qui permettrait de lui appliquer le théorème de Moschovakis.*

Théorème. *L'espace des groupes marqués ne possède pas de "présentation récursive" au sens de Moschovakis.*

Les théorèmes de Ceitin et Moschovakis sont donnés au chapitre 5, ainsi que la définition d'une présentation récursive d'un espace polonais (qui n'a rien à voir avec la notion de présentation de groupe).

La conséquence des théorèmes cités ci-dessus est qu'aucune des méthodes actuellement connues qui visent à montrer que les fonctions calculables définies sur des espaces "normaux" sont continues ne peut être appliquée à l'espace des groupes marqués.

Conjecture. *Les fonctions calculables définies sur l'espace des groupes marqués sont continues.*

On sait, par le Lemme de Markov, que les fonctions définies sur l'espace des groupes marqués ne sont pas "effectivement discontinues", mais on ne sait pas montrer, ni qu'elles sont effectivement continues, ni même qu'elles sont continues.

On est confronté à un problème où le tiers exclu échoue pour les notions effectives. Ce genre de phénomène est courant et bien connu, car bien souvent il se trouve qu'une notion effective n'est pas la négation de la notion effective inverse. Détaillons cela en quelques phrases dans le langage du calcul propositionnel.

Si P est une proposition, on désigne par P^+ la notion effective associée. Ce concept est ambigu si l'on ne donne pas un cadre mathématique plus précis (en fixant un langage, des numérotations, etc), mais on se permet ici d'être imprécis.

Le plus souvent, on aura les implications suivantes :

$$P^+ \implies P;$$
$$(\neg P)^+ \implies \neg P;$$
$$(\neg P)^+ \not\Leftrightarrow \neg(P^+);$$
$$\neg(P^+) \not\Rightarrow \neg P;$$
$$\neg P \not\Rightarrow (\neg P)^+.$$

Finalement, alors que le tiers exclu tient pour les notions classiques (on a $P \vee \neg P$), il ne tient pas forcément pour les notions effectives, et on peut avoir $\neg(P^+) \wedge \neg((\neg P)^+)$.

Notons finalement que plusieurs résultats plus forts que la continuité sont connus dans le cadre des espaces polonais effectifs, on en cite ici deux, sans en définir les termes :

- Dans un espace polonais effectif, tout ensemble effectivement ouvert est un ouvert de Lacombe ;
- Dans un espace polonais effectif, les notions de calculabilité au sens de Markov et au sens de Borel sont équivalentes.

Ces notions sont définies dans le chapitre 5, on veut juste noter ici que chacune de ces affirmations peut être traduite en une conjecture dans l'espace des groupes marqués. (La notion de calculabilité au sens de Borel est une attribution posthume, dans laquelle Borel ne se serait pas reconnu, puisqu'il donna en 1912, dans l'article [**14**], une notion de calculabilité pour les fonctions réelles, notion bien sûr informelle puisque les fonctions récursives n'avaient pas encore été définies, mais dont on voit aujourd'hui facilement qu'elle correspond à la définition de Markov.)

Théorie descriptive effective. Une fois que le cadre général de l'analyse calculable sur l'espace des groupes marqués est introduit, on peut demander, à propos de différentes fonctions explicites, si elle sont calculables ou non. Les fonctions qui nous intéressent le plus sont en fait des fonctions indicatrices de différents ensembles de groupes : on veut pouvoir classifier les propriétés de groupes selon si elles sont, ou non, décidables à partir d'algorithmes du problème du mot.

On fait ainsi de la théorie descriptive des ensembles *effective*, sujet inventé et développé par Moschovakis dans le cadre des espaces polonais effectifs. On se trouve ici dans un cadre plus général que celui de Moschovakis, et on ne dispose donc pas de « théorèmes de hiérarchie » tels que ceux prouvés par Moschovakis sur les espaces polonais effectifs.

Ce qui nous intéresse le plus est de discuter la correspondance entre la hiérarchie de Borel et son analogue effectif, pour les trois premiers niveaux de ces hiérarchies. On attend la correspondance suivante (P désigne une propriété de groupes marqués) :

$$\text{P est ouverte et fermée} \longleftrightarrow \text{P est décidable}$$

$$\text{P est ouverte} \longleftrightarrow \text{P est semi-décidable}$$

$$\text{P est fermée} \longleftrightarrow \text{P est co-semi-décidable}$$

La flèche "P est ouverte et fermée $\leftarrow$ P est décidable " correspond à la conjecture qui dit que les fonctions calculables doivent être continues, et donc elle pourrait, peut-être, être remplacée par l'implication "P décidable implique P ouverte et fermée". Les autres flèches ne peuvent pas être remplacées par des implications, des contre-exemples classiques en analyse calculable peuvent le montrer, et leur sens est informel, on va maintenant détailler comment les interpréter.

De droite à gauche :

- Une propriété "naturelle" qui est semi-décidable est sûrement ouverte.
- Une propriété "naturelle" qui est co-semi-décidable est sûrement fermée.

Les exemples de propriétés semi-décidables non ouvertes sont des exemples pathologiques définis en termes de complexité de Kolmogorov.

De gauche à droite :

- Une propriété "naturelle" qui est ouverte est probablement semi-décidable.
- Une propriété "naturelle" qui est fermée est probablement co-semi-décidable.

Ces affirmations-ci sont justifiées par l'expérience, si l'on considère que sur 23 propriétés ouvertes ou fermées recensées dans cette thèse, seules deux sont susceptibles d'être des contre-exemples à cette règle, ce qui donne empiriquement une probabilité de $21/23 \simeq 0.913$ de vérifier la correspondance.

Voici un tableau compilant les résultats de théorie descriptive effective obtenus dans cette thèse :

Propriétés Clouvertes et décidables	**Propriétés Ouvertes et semi-décidables**
Être abélien ;	Être nilpotent ;
Être isomorphe à un groupe fini donné ;	Avoir la propriété (T) de Kazhdan ;
Avoir au plus n éléments, $n \geq 0$;	Avoir un centre non-trivial ;
Être nilpotent de classe k, $k \geq 1$;	Être parfait ;
Être isomorphe à un certain groupe marqué isolé.	Avoir de la torsion ;
	Être de rang k, $k \in \mathbb{N}^*$;
	Être virtuellement cyclique ;
	Être à croissance polynomiale ;
	Être polycyclique.

Propriétés fermées et co-semi-décidables	**Propriétés ni ouvertes ni fermés**
Être infini ;	Être résoluble ;
Être résoluble de classe k, $k \geq 2$;	Être moyennable ;
Être d'exposant fini ;	Être simple ;
Être un groupe limite de Sela ;	Être à croissance sous-exponentielle ;
Être ordonnable ;	Être de présentation finie ;
Être δ-hyperbolique, $\delta > 0$;	Être hyperbolique ;
Être à classes de conjugaisons infinies.	Être résiduellement fini.

Deux propriétés ne trouvent pas leur place dans ce tableau : l'adhérence des groupes finis est un ensemble fermé, pourtant peut-être pas co-semi-décidable, et l'ensemble des groupes isolés, qui sont les groupes qui, en tant que singletons dans l'espace des groupes marqués, sont ouverts, est un ensemble ouvert (bien entendu), mais pas forcément semi-décidable. On conjecture ainsi :

CONJECTURE. *L'adhérence de l'ensemble des groupes finis est un fermé qui n'est pas co-semi-décidable, et l'ensemble des groupes isolés est un ouvert qui n'est pas semi-décidable.*

Notons que la partie de cette conjecture qui concerne l'adhérence des groupes finis est liée à l'étude de la théorie universelle des groupes, on montre en effet qu'une preuve de cette conjecture impliquerait le célèbre théorème de Slobodskoi qui affirme que la théorie universelle des groupes finis est indécidable.

0.1. Traduction française de quelques termes anglais

Nous allons à présent donner la traduction de quelques termes mathématiques présents dans cette thèse. La traduction d'un terme mathématique anglais en un terme mathématique français consiste bien souvent en la traduction du mot sous-jacent. Malheureusement, il existe des exceptions à cette règle, ce qui justifie l'inclusion de cette section. Sont incluses quand c'est possible des références vers des articles en français où les termes apparaissent.

- Marked group : groupe marqué ([**22**]).
- Decidable, semi-decidable, co-semi-decidable : décidable, semi-décidable, co-semi-décidable.
- Marked quotient algorithm : algorithme pour les quotients marqués.
- Marked quotient algorithm with Word Problem as Input (WPI marked quotient algorithm) : algorithme pour les quotients marqués donnés par leur problème du mot.
- Relative marked quotient algorithm : algorithme pour les quotients marqués relatif à une classe de groupes.
- Group with Computable Finite Quotients (CFQ) : groupe dont les Quotients Finis sont Calculables (QFC).
- Effectively residually finite group : groupe effectivement résiduellement fini.

- Numbering : Numérotation.

  On notera qu'en 1970, au congrès international des mathématiciens de Nice, Ershov, dans une adresse en français ([**37**]), traduit *numbering* par « énumération ». Ce choix est justifié par le fait que sa définition est la suivante : une énumération d'un ensemble X est une fonction définie sur $\mathbb{N}$ et dont l'image est X. Dans cette thèse, la définition considérée pour un « numbering » ne demande pas que la fonction soit définie sur $\mathbb{N}$ tout entier. En particulier, une numérotation définie sur un ensemble qui n'est pas récursivement énumérable ne permettra pas forcément d'énumérer l'ensemble X qui est numéroté. On propose donc ici deux définitions françaises :
  - Une numérotation d'un ensemble X est une fonction définie sur un sous-ensemble de $\mathbb{N}$ allant vers X.
  - Une énumération d'un ensemble X est une numérotation de X vis-à-vis de laquelle X est récursivement énumérable (voir les préliminaires du chapitre 1).

  Notez aussi qu'on parle de « numérotation de Gödel » et pas d'« énumération de Gödel ». Le concept de numérotation employé ici n'est bien sûr qu'une généralisation du concept de Gödel (une numérotation de Gödel peut être définie comme une numérotation de l'ensemble des phrases d'un langage formel).
- Rogers semi-lattice : le demi-treillis de Rogers.
- Decision problem for groups : problème de décision à propos de groupes.
- Local decision problem : problème de décision local.
- Global decision problem : problème de décision global.
- Recursive enumeration algorithm for a group : algorithme d'énumération de l'élément neutre.
- Depth function/residually finite growth : fonction de croissance associée au caractère résiduellement fini d'un groupe.
- Busy beaver function : fonction du castor affairé.

  Traduction littérale communément utilisée, présente par exemple sur la page wikipedia correspondante.
- Clopen : clouvert.

  On remarque le mélange étrange entre anglais et français dans le terme clouvert, formé de la concaténation de "closed" et "ouvert". Mais le terme "fouvert" n'a pas été accepté par la communauté mathématique française.
- Markov computable function : fonction calculable au sens de Markov.

Introduction (English Introduction)

Introductory remarks. The research domain that lies at the intersection of computability theory and group theory is known as the theory of decision problems for groups. There are two main kinds of decision problems that arise in group theory: *local problems*, where questions about the elements of a fixed group are asked, and *global problems*, where the problems asked concern the properties of whole groups.

In both cases, decision problems take the form: is there an algorithm that, given a group (resp. an element of a group), can determine whether or not this group (resp. group element) satisfies any chosen property? Of course, this formulation does not precise how the group or group element should be described in order for a computer to manipulate it, this important problem will be much discussed in this thesis.

The history of the theory of decision problems for groups is singular, compared to other branches of effective mathematics, because it predates the invention of the theory of computability. Indeed, when Max Dehn, in 1911, formulated the *word problem*, the *conjugacy problem* and the *isomorphism problem*, any proper definition of the notion of a "computable function" was still at least twenty-four years away (Church, 1935), while the first proof of the unsolvability of the word problem for finitely presented groups, which of course relies on the mathematical definition of computable functions, was found by Novikov only in in 1955, and later by Boone in 1958.

Because of this, the theory of decision problems for groups was *not* developed as an "effective theory of groups", which would have consisted in, starting from the classical theory of groups, asking which parts of it are effective and which are not. On the contrary, results have remained much centered on the algorithmic possibilities offered by finite presentations, following Dehn's plan, that the solution of decision problems should be the basis of the building of a theory of finitely presented groups.

My approach can, a posteriori, be seen as the development of some parts of an effective theory of groups, and in particular in the obtention of results that would have been obtained at the very beginning of the theory of decision problems, had it been developed similarly to other branches of effective mathematics. I have in particular been inspired by the way in which the theory of computable analysis was built upon the classical theory of real analysis.

The results that appear in this thesis can thus be classified in two categories: those that are *below* the Boone-Novikov Theorem, and those that are *above* the Boone-Novikov Theorem.

The results that are *below* the Boone-Novikov Theorem concern mostly groups that need not be finitely presented. These results range from quite obvious (the Rice Theorem for recursively presented groups) to non-trivial, but, even in the non-trivial cases, the methods remain elementary. In a book on the theory of decision problems for groups, many of those results should naturally be presented *before* the Boone-Novikov Theorem.

On the other hand, the results that are *above* the Boone-Novikov Theorem make use of all the tools that were developed in the study of decision problems centered on finitely presented groups, and, in particular, of Higman's Embedding Theorem, which allows for short proofs of results that would have seemed very hard to prove prior to its finding.

Presented articles. The present manuscript gathers the results that were obtained in four articles, the first two of which are published:

Emmanuel Rauzy. Computability of finite quotients of finitely generated groups
Journal of Group Theory, 25(2):217-246, oct 2020 DOI: 10.1515/jgth-2020-0029

Emmanuel Rauzy. Obstruction to a Higman Embedding Theorem for residually finite groups with solvable word problem.
Journal of Group Theory, 24(3):445–452, oct 2020. DOI: 10.1515/jgth-2020-0030

Emmanuel Rauzy. Remarks and problems about algorithmic descriptions of groups
Preprint, 2021, arXiv:2111.01179

Emmanuel Rauzy. Computable analysis on the space of marked groups
Preprint, 2021, arXiv:2111.01190

The exposition of those works that is presented in this manuscript adds very little in terms of new mathematical content, and the first chapter which contains preliminary results is rather light (although the introduction to the theory of numberings is of importance). However, I have modified the internal organization of my articles, in order for them to form a more or less coherent whole, rather than being only an accumulation of results in different directions. Still, all chapters presented here can be read almost independently, with the exception of the short Chapter 4 which is a direct continuation of Chapter 3.

Development of the ideas presented here. The starting point of this thesis was the idea that we have a very poor understanding of what it means, for a group, to be finitely presented. Indeed, while many important problems that were raised in the 20th century and that served as motivations for the development of the theory of countable groups have been solved, many of these problems remain open in the realm of finitely presented groups. In particular, the positive solution to the Burnside problem, the proof of existence of groups of intermediate growth, proofs of existence of various "monsters", such as Tarski monsters and Dehn monsters, all provide groups that are not finitely presentable, and it seems that proving or disproving that such groups can exist and be finitely presented remains a far away prospect.

One of the most important results that concerns specifically finitely presented groups is the characterization, due to Higman, of their subgroups: Higman's Embedding Theorem states that a group is a finitely generated subgroup of a finitely presented group if and only if it is recursively presented. This beautiful result ties together distinct areas of mathematics, computability theory and group theory, or computability theory and group theory *and topology*, if we take into account the fact that finitely presented groups are exactly the fundamental groups of closed manifolds.

A good way to gain a better understanding of the way the property "being finitely presented" interacts with other group properties would be to obtain various "Higman-like theorems", results of the form: "the finitely generated subgroups of the finitely presented groups that satisfy the property P are exactly ...". For many properties that are inherited by subgroups, and that do not seem to interact with the property "being finitely presented", the most natural conjecture is often the most obvious one; it has the form:

Conjecture (Typical conjecture). *The finitely generated subgroups of the finitely presented groups that satisfy the property P are exactly the recursively presented groups that satisfy P.*

This conjecture is believed to be true for solvable groups, amenable groups, etc, it could also be true for torsion groups. The "Higman conjecture" associated to simple groups has the following form:

Conjecture (Higman–Boone). *The finitely generated subgroups of the finitely presented simple groups are exactly the groups that have solvable word problem.*

Some relative versions of Higman's Theorem have been proved: the Clapham-Valiev result for subgroups of finitely presented groups with solvable word problem, the Birget-Olshanskii-Rips-Sapir characterization of groups with word problem in NP, the Olshanskii-Sapir characterization of groups with solvable conjugacy problem as those that can be Frattini embedded inside a finitely presented group with solvable conjugacy problem (a Frattini embedding is an embedding that preserves conjugacy: non-conjugated elements in the subgroup cannot become conjugated in the overgroup), etc.

Note however that while all these theorems provide insight about finitely presented groups that satisfy additional algorithmic properties, they do not provide new results that intertwine algebraic properties of groups with algorithmic properties.

The only known example of an embedding theorem for groups that satisfy an algebraic property is a theorem of Baumslag that states that all finitely generated metabelian groups embed in finitely presented metabelian groups. However, the methods employed to prove this theorem are specific to metabelian groups and cannot be generalized.

Still, there is an algebraic property that has been known for a long time to interact positively with finite presentability, the property of being *residually finite*. Indeed, a theorem of McKinsey implies that all finitely presented residually finite groups have solvable word problem. The fundamental remark of McKinsey is that given a finitely presented group and a finite group described by its Cayley table, it is possible to determine whether or not the finite group is a quotient of the finitely presented group, by checking whether the finitely many defining relations of the finitely presented group hold in the finite group. It is this fact, together with the fact that it is possible to algorithmically produce a sequence that contains the Cayley tables of all finite groups, that implies that all finitely presented residually finite groups have solvable word problem. This in turn implies that all finitely generated subgroups of finitely presented residually finite groups have solvable word problem.

Following this remark, Dyson and Meskin have constructed recursively presented residually finite groups that have unsolvable word problem, testifying for the fact that *not all recursively presented residually finite groups embed into finitely presentable ones.*

McKinsey's algorithm lies at the core of the first two published articles that this thesis presents, which are contained in Chapter 3 and 4.

My researches around McKinsey's algorithm were focused in two main directions:

- The study of the infinitely presented groups whose finite quotients could be algorithmically enumerated, just as in finitely presented group;
- The study of the groups in which *enough* of their finite quotients could be algorithmically enumerated, enough to separate all non-trivial elements.

Groups which fall into the second category above are called "effectively residually finite". Subgroups of finitely presented residually finite groups must not only have solvable word problem, but they must also be effectively residually finite.

In Chapter 4 is constructed a residually finite group with solvable word problem that is not effectively residually finite, as was first constructed in [**105**]. This group shows that the condition "being effectively residually finite and having solvable word problem" is a necessary condition for a finitely generated group to embed in a finitely presented residually finite group that is indeed sharper than the sole condition of having solvable word problem. The following conjecture is proposed in Chapter 4:

Conjecture. *The subgroups of finitely presented residually finite groups are exactly the effectively residually finite groups that have solvable word problem.*

The study of groups whose finite quotients can be enumerated, that makes up the article [**108**], turned out to be very interesting for another reason: they provided a class of groups that generalizes that of finitely presented groups, but whose features felt closer to those of finitely presented groups than other known possible generalizations of finitely presented groups, such as recursively presented groups.

This remark has led me to introducing the general notion of a *marked quotient algorithm*, which provides an *algorithmic characterization of finitely presented groups*: the finitely presented groups are exactly those recursively presented groups whose recursively presented quotients can be recognized.

While this characterization does give some insight into the nature of finitely presented groups, it does not provide any help towards solving any of the motivational problems quoted at the beginning of this introduction, (like the finding of finitely presented torsion groups, of finitely presented groups of intermediate growth, and so on), and the main interest of this characterization of finitely presented groups is that it permits the study of various algorithmic generalizations of the notion of finite presentation.

The study of groups whose finite quotients can be enumerated, and the study of the algorithmic characterization of finitely presented groups that it inspired, both take place in the category of *marked groups*. A marked group is a pair constituted of a group and of a generating family, and most, if not all, of the finite descriptions of groups that are commonly used in order to study decision problems for groups give, in fact, descriptions of marked groups. Because of this, the study of decision problems for marked groups strictly contains the theory of decision problems for groups, at least as it was studied until now, since a group property can be seen as a peculiar instance of a property of marked groups, one that is invariant under group isomorphism.

The last result that appeared in the course of the study of groups whose finite quotients can be enumerated, and which has had much influence on the direction of my research, is the fact that *the isomorphism problem is unsolvable for finite groups described by recursive presentations.*

This result does not appear in this manuscript, at least as proven in my first article, thanks to a group whose finite quotients cannot be enumerated, because I quickly obtained a strong improvement of it: *recursive presentations never allow to distinguish between a group and a strict quotient of this group.*

This result leads to the Rice Theorem for recursively presented groups, that no non-trivial group property is decidable for recursively presented groups, and to an equivalent of the Rice-Shapiro Theorem, which characterizes the semi-decidable properties of recursively presented groups.

Those results are very easy to prove, and I was very surprised to see that they were not known. I think that, when studying the literature on decision problems for finitely presented groups that was written between the Markov-Post Theorem of 1947 up to the 1990s and the famous survey article of Miller, one might be led to adhere to the following idea: "comes the time to study decision problems for infinitely presented groups, recursive presentations will successfully play the central role that finite presentations had before".

This idea is vigorously rejected in this thesis, which, on the contrary, supports the idea that no single group description can take the role of finite presentations in the study of infinitely presented groups, and thus that decision problems must be studied with respect to each possible type of description of group that one encounters.

The algorithmic characterization of finitely presented groups can be used to shed some light on the reason why recursive presentations provide weak descriptions of groups. Indeed, finite presentations of groups, by providing a way to recognize the quotients of the groups they define, contain *global* information, and by that we mean: information on the location of a group in the lattice of marked groups. On the contrary, recursive presentations are not better than algorithms that, in a certain marked group, recognize the different ways to write the identity element, and thus they only provide *local information* about the group they define, that is information on the relations between the elements of this given group, but not on the relation between this group and other groups.

My third article ([**107**]) describes what a theory of decision problems for groups that is not aimed at studying finitely presented groups should look like, based on the three remarks described above:

- That finite presentations have an algorithmic equivalent;
- That decision problems should be stated and studied for marked groups;
- That recursive presentations are descriptions much too weak to be the basis of an interesting theory.

This article, while chronologically my third, is presented here first, in Chapter 2, as my actual first and second articles can be seen as proceeding from the point of view on decision problems that is described in Chapter 2, even though, as I have tried to explain in this introduction, those first and second articles motivated the point of view on decision problems described in Chapter 2.

One of the conclusions of the approach described there is that the most promising field of study in the theory of decision problems for groups is probably the study of decision problems for groups described by *finite presentations together with word problem algorithms.* (What this means is precisely explained in Chapter 1 thanks to the theory of numberings.) While, for this description, the isomorphism problem remains unsolvable, many problems that were unsolvable for finite presentations become solvable, such as the *triviality problem.*

In particular, the fact that with respect to this description, some groups can be algorithmically recognized, while others cannot, makes it possible to distinguish groups that are *easy to recognize* from groups that are *hard to recognize.* Finite presentations alone do not permit to make such a classification.

Computable analysis on the space of marked groups. As a first step towards the study of decision problems for groups described by finite presentations together with word problem algorithms, and as a good illustration of the possibilities given by the idea of studying systematically decision problems with respect to all possible descriptions of groups, I started investigating decision problems for groups described by word problem algorithms, my initial purpose was to include my results in a short section in my third article.

Another incentive for this project was the fact that, through Clapham's (or Valiev's) version of the Higman's Embedding Theorem, *decision problems for groups described by word problem algorithms* can be seen as being equivalent to *decision problems about subgroups of finitely presented groups with solvable word problem.* My work on the topological space of marked groups provides the first unified framework to build finitely presented groups with solvable word problem in which the subgroups with various properties cannot be recognized. For instance, we recover effortlessly the recent result that there is a group whose amenable subgroups are not recognizable, since, through my work, this result can be seen as a direct consequence of the trivial fact that the set of amenable groups is not closed, for the topology of the space of marked groups, inside of the set of LEF groups.

I quickly set the study of decision problems for groups described by word problem algorithms in the topological space of marked groups, as I had obtained the following lemma:

Lemma. Word problem algorithms do not permit to distinguish the elements of an effective sequence of marked groups from the limit of this sequence.

This of course led me to the belief that *computable functions defined on the space of marked groups should be continuous*, belief reinforced by the correspondence between the arithmetical hierarchy and the Borel hierarchy in the space of marked groups: all the decidable properties of groups that I encountered (being abelian, being isomorphic to some finite group, etc) were clopen properties, the semi-decidable properties were all open, and so on. The question of proving that all computable functions defined on the space of marked groups are continuous is known as the *continuity problem on the space of marked groups.* For quite a long time, I made no progress on this question.

Eventually, it became clear to me that my result about limits of effective sequences must have been well known in the case of real numbers, and so I decided to go looking for this result in the literature about computable analysis, hoping that I would find there ways of proving that functions defined on the space of marked groups are continuous.

I ran into quite a lot of trouble as I approached this topic.

The literature on computable analysis is not unified, and several non-equivalent definitions of a *computable function of a real variable* are commonly used.

A first problem that I encountered was that the most commonly used definition of a computable real valued function, the Kleene-Weihrauch definition, considers real numbers given by oracles, and following this definition, it is immediate to see that computable functions are continuous (this does not even use a reduction to the halting problem). An important part of the literature on computable analysis thus turned out to be useless regarding the continuity problem on the space of marked groups.

I finally found the definition of a computable function of a real variable that I was to focus on, and which, when applied to groups, does correspond to the study of decision problems from groups described by word problem algorithms: although it is the original definition used by Turing in his seminal 1937 paper, this definition of a computable function of a real variable is associated to Andrey Andreyevich Markov, who was responsible for rendering this definition popular. (Note that Andrey Andreyevich Markov is the son of the famous mathematician Andrey Andreyevich Markov, and, to avoid confusion, he is sometimes called Markov Jr. in articles translated from Russian originals. A. A. Markov, the father, is known for his groundbreaking work in the theory of stochastic processes, while A. A. Markov, the son, is known, among other things, for having proved that the homeomorphism problem is unsolvable for manifolds of dimension at least four.)

I thus began the study of Markovian computable analysis. However, I was confronted there to a new problem: most of the authors that have studied Markov computability, and that have laid the foundations of Markovian computable analysis, adhere to constructivist beliefs. I thus had to acquire some familiarity with the constructivist approach and language in order to understand, for instance, Aberth's book, or the work of Ceitin as presented in Kushner's account.

Still, I had found the general version of the lemma that I had been using on groups: it is known as Markov's Lemma, and its general statement is that computable functions defined on an effectively complete computable metric space are not "effectively discontinuous". Thanks to the Markovian definition of computable analysis, I was legitimated in naming the study of decision problems for groups described by word problem algorithms "Computable analysis on the space of marked groups" ([**106**]).

(As an aside, let me note that in the end, I think that I have benefited from being familiar with the constructivists' setting, as many phenomenon of failure of the excluded middle law naturally arise in computability: it is not the case that a function is either "effectively continuous" or "effectively discontinuous", and the continuity problem on the space of marked groups precisely asks how to go from knowing that functions are not effectively discontinuous to proving that they must be effectively continuous. The Brouwerian adage "mathematical statements should be positive" is also an endless source of good effective definitions.)

Once I had a clearer vision of the field of computable analysis, I was able to prove that none of the results of Mazur, Kreisel, Lacombe, Schoenfield, Ceitin or Moschovakis could be applied to the space of marked groups. Because of this, the space of marked groups is very interesting from the point of view of computable analysis, as it behaves very differently from other well studied Polish spaces. It is in fact the first natural example of a Polish space that is not an effective Polish space.

While I thought that the study of decision problems for groups described by word problem algorithms would be associated to new techniques, and would resemble neither the classical theory of decision problems for finitely presented groups, nor my previous work around McKinsey's algorithm, much to my surprise all these things came back to play roles, and sometimes very important ones:

- Finitely presented groups do have a central role in the study of computable analysis on marked groups, for instance both the Boone-Novikov and Boone-Rogers theorems can be interpreted in the space of marked groups as proving that it is impossible to decide whether or not one of the basic open sets is empty;
- Miller's construction of a finitely presented group for which all of its non-trivial quotients have unsolvable word problem, which is a powerful improvement on the Boone-Novikov theorem, and which relies on Higman's Theorem, was the crucial tool in proving that Moschovakis' Theorem fails on the space of marked groups;
- McKinsey's Theorem turned out to have a topological generalization, in terms of *finitely presented group in the adherence of an enumerable set of groups*, and the very important fact that the space of marked groups does not have a computable and dense sequence can be seen as proceeding from this version of McKinsey's algorithm;
- Two groups that I had constructed in my first and second articles also found a home in the study of the space of marked groups, one because it provides a Specker sequence of groups, that is a computable sequence of computable groups that converges to a non-computable group, the other one because it provides

a group which is the limit of a computable sequence of finite groups, but not of a computable sequence of its finite quotients.

A very important part of the work presented here on the study of computable analysis on the space of marked groups is in fact dedicated to the development of a correct working frame into which this study should take place, and I have in fact spent more time explaining the link between the study of decision problems from word problem algorithms and computable analysis in general than working specifically in the space of marked groups. Because of this, there is much work left to be done on the space of marked groups, in particular it seems to me of foremost importance to find examples of natural group properties that are open but not semi-decidable in the space of marked groups, proving that the effective hierarchy provided by the effective structure on the space of marked groups is more precise than the Borel hierarchy.

Contents. Chapter 1 contains some definitions and results that are needed in the course of the rest of this thesis. The definitions of numberings and of numbering types are of particular importance. Chapter 2 presents the results of [**107**], and sets the framework in which the rest of our work is developed.

In Chapter 3, the notion of a group whose finite quotients can be enumerated is studied, following [**108**]. In Chapter 4, we prove that there can be no Higman embedding theorem for residually finite groups with solvable word problem, using the property "being effectively residually finite". Those results first appeared in [**105**]. In Chapter 5, we describe the study of computable analysis on the space of marked groups, following the article [**106**].

In Chapter 6 we gather the open problems raised throughout our work.

CHAPTER 1

General preliminaries and vocabulary

1.1. Vocabulary about numberings

The theory of numberings gives a rigorous basis to the study of effective mathematics in general. However, it can be, and often is, used implicitly. We present here some results and definitions that are used explicitly only in Chapter 5, but that are in fact constantly present throughout this thesis.

In the following section, Section 1.2, we define marked groups and the numbering associated to word problem algorithms, and, in the subsequent section, Section 1.3, we explain in more details why the theory of numberings is not explicitly used in Chapters 2-4.

1.1.1. Numberings and numbering types. We now introduce numbered spaces and numbering types. For more details, see the chapter on numberings in Weihrauch's book [**128**]. The term "numbering type" is ours.

Definition 1.1.1. Let X be a set. A *numbering* of X is a function ν that maps a subset A of $\mathbb{N}$ to X. We denote this by: $\nu :\subseteq \mathbb{N} \to X$.

The set of numberings of X is denoted $\mathcal{N}_X$.

It is important for us not to impose on numberings to be surjective, contrary to what is customary, as this allows for a much more natural approach to the study of the different numberings of the (uncountable) set of marked groups.

The pair (X, ν) is a *numbered set.* The domain of ν is a subset of $\mathbb{N}$ denoted by $\mathrm{dom}(\nu)$.

The image $\nu(\mathrm{dom}(\nu))$ of ν is called the set of ν-computable points of X, and denoted $X_{c,\nu}$ or X_c when there is no ambiguity as to which numbering of X is considered. Given a point x in X, an integer n such that $\nu(n) = x$ is called a *ν-name,* or a *ν-description,* of x.

Definition 1.1.2. Let (X, ν) and (Y, μ) be numbered spaces. A function $f : X \to Y$ is called (ν, μ)-computable if there exists a partial recursive function $F :\subseteq \mathbb{N} \to \mathbb{N}$ such that for all n in the domain of ν, $f \circ \nu(n) = \mu \circ F(n)$. That is to say, there exists F which renders the following diagram commutative:

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ {\scriptstyle\nu}\uparrow & & {\scriptstyle\mu}\uparrow \\ \mathbb{N} & \xrightarrow{F} & \mathbb{N} \end{array}$$

Notice how this definition resembles that of a manifold, and in particular how the notion of differentiability is defined on a manifold using only differentiability in a Euclidian space.

Of course, whether a function f between numbered spaces (X, ν) and (Y, μ) is computable only depends on its behavior on the set of computable points of X. One easily checks the following:

Lemma 1.1.3. *If (X, ν), (Y, μ) and (Z, τ) are numbered sets, the composition of a (ν, μ)-computable function with a (μ, τ)-computable function is (ν, τ)-computable.*

The identity function $\mathrm{id}_{\mathbb{N}}$ on $\mathbb{N}$ defines its most natural numbering.

Definition 1.1.4. If (X, ν) is a numbered set, a *ν-computable sequence* is a $(\mathrm{id}_{\mathbb{N}}, \nu)$-computable function from $\mathbb{N}$ to X.

We consider a partial order on the numberings of a set X:

Definition 1.1.5. A numbering ν of a space X is *stronger* than a numbering μ of this same space if the identity on X is (ν, μ)-computable. We denote this $\nu \succeq \mu$. Those numberings are *equivalent* if $\nu \succeq \mu$ and $\mu \succeq \nu$ both hold. We denote this by $\nu \equiv \mu$.

The relation $\nu \succeq \mu$ exactly means that there is an algorithm that, given a ν-description of a point in X, produces a μ-description of it. The relation $\nu \succeq \mu$ can thus be interpreted as: a ν-description of a point x contains more information on this point than a μ-description of x.

Natural examples for groups are: a finite presentation of a group contains more information on this group than a recursive presentation, or a finite presentation together with a word problem algorithm contains more information than a sole finite presentations, etc.

The following lemma is a direct consequence of the fact that composition preserves computability of functions.

Lemma 1.1.6. *The relation $\succeq$ is transitive and reflexive. In particular, the relation $\equiv$ is an equivalence relation.*

Definition 1.1.7. The *numbering types* on X are the equivalence classes of $\equiv$.

If (X, ν) and (Y, μ) are numbered spaces, and if $f : X \to Y$ is a (ν, μ)-computable function between X and Y, then f will be computable with respect to any pair of numberings of X and Y which are $\equiv$-equivalent respectively to ν and μ.

Thus f can be considered computable with respect to the numbering types associated to ν and μ. Numbering types are in fact the objects that the theory of numberings aims at studying, rather than numberings themselves.

We denote $\mathcal{NT}_X$ the set of numbering types on X. Given a numbering, we denote $[\nu]$ its $\equiv$-equivalence class. We usually denote numbering types by capital greek letters, Λ, Δ, ...

The relation $\succeq$ defines an order on the set of numbering types, order whose description is an important part of the study of the different numberings of a set.

Definition 1.1.8. Let (X, ν) be a numbered set. Then the equivalence relation defined on $\mathrm{dom}(\nu)$ by $n \sim m \iff \nu(n) = \nu(m)$ is called the *numbering equivalence induced by* ν ([**38**]), and denoted η_ν.

Definition 1.1.9. A numbering is called *positive* when there is a recursively enumerable set $L \subseteq \mathbb{N} \times \mathbb{N}$ such that $\eta_\nu = L \cap \mathrm{dom}(\nu) \times \mathrm{dom}(\nu)$.

A numbering is called *negative* when there is a co-recursively enumerable set $L \subseteq \mathbb{N} \times \mathbb{N}$ such that $\eta_\nu = L \cap \mathrm{dom}(\nu) \times \mathrm{dom}(\nu)$.

It is called *decidable* when if it is both positive and negative.

Remark 1.1.10. Mind that in this last definition, one cannot define a decidable numbering as a numbering for which there exists a recursive set $L \subseteq \mathbb{N} \times \mathbb{N}$ such that $\eta_\nu = L \cap \mathrm{dom}(\nu) \times \mathrm{dom}(\nu)$. This last condition is strictly more restrictive than being decidable. A simple example where this fails is a numbering of a two points set $\{x, y\}$, defined on a pair of recursively enumerable but recursively inseparable sets A and B. Define ν on $A \cup B$ by $\nu(n) = x$ if $n \in A$ and $\nu(n) = y$ if $n \in B$. Then ν is decidable, but there cannot be a recursive set R as above. We do not give more details here. Recursively inseparable sets are defined and constructed in Lemma 5.3.11 of Chapter 5, as they are at the center of a construction of Miller that we use.

The following proposition is straightforward:

Proposition 1.1.11. *Suppose ν and μ are numberings of X. Suppose $\nu \succeq \mu$. Then, if μ is any of positive, negative or decidable, then so is ν.*

As a corollary, we obtain:

Corollary 1.1.12. *Being positive, negative or decidable are properties that are $\equiv$-invariant, and thus can be defined for numbering types.*

1.1.2. The lattice structure of $\mathcal{NT}_X$. We now introduce the lattice structure on $\mathcal{NT}_X$. Here, by lattice, we mean a partially ordered set that admits meet and join operations: we will thus show that any pair of numbering types in $\mathcal{NT}_X$ admits both a greatest lower bound and a least upper bound for the order $\succeq$.

The lattice operations of $\mathcal{NT}_X$ are the conjunction and the disjunction. Given two numberings ν and μ of a set X, we will define new numberings $\nu \wedge \mu$ and $\nu \vee \mu$ by saying respectively that a $\nu \wedge \mu$-name for a point x is a ν-name for x together with a μ-name for it, and that a $\nu \vee \mu$-name for a point y should be either a ν-name for it, or a μ-name for it. Those definitions are explained below.

We start by defining the maximal element of $\mathcal{NT}_X$.

Definition 1.1.13. Let $\Lambda_\emptyset$ denote the numbering type of the nowhere defined numbering, the only function $\nu_\emptyset :\subseteq \mathbb{N} \to \mathbb{N}$ with $\mathrm{dom}(\nu) = \emptyset$. (One thus has $\Lambda_\emptyset = \{\nu_\emptyset\}$.)

Proposition 1.1.14. *Any function $f : X \to Y$ is $(\Lambda_\emptyset, \Delta)$-computable, for any $\Delta \in \mathcal{NT}_Y$. And $\Lambda_\emptyset$ is maximal in $\mathcal{NT}_X$.*

Proof. This is true by vacuity: $f : X \to Y$ is $(\nu_\emptyset, \mu)$-computable (where $\mu \in \Delta$) if there is an algorithm that transforms a $\nu_\emptyset$-name of a point x into a μ-name of $f(x)$. But since no point has a $\nu_\emptyset$-name, this must be true. The second point immediately follows. □

We now define the "disjunction" of two numberings.

Definition 1.1.15. Let ν and μ be numberings of X. Define a numbering $\nu \vee \mu$ (the *disjunction* of ν and μ) by setting, for any natural number k, $\nu \vee \mu(2k) = \nu(k)$ and $\nu \vee \mu(2k+1) = \mu(k)$. The domain of $\nu \vee \mu$ is the set $\{2k, k \in \mathrm{dom}(\nu)\} \cup \{2k+1, k \in \mathrm{dom}(\mu)\}$.

Thus a $\nu \vee \mu$-name for a point x of X is either a ν-name for it, or a μ-name for it.

Proposition 1.1.16. *Let ν and μ be numberings of X. Then $\nu \succeq \nu \vee \mu$ and $\mu \succeq \nu \vee \mu$, and for any τ in $\mathcal{N}_X$, if $\nu \succeq \tau$ and $\mu \succeq \tau$, then $\nu \vee \mu \succeq \tau$.*

Proof. Given a ν-name n for a point x in X, by definition of $\nu \vee \mu$, one has $\nu \vee \mu(2n) = x$, and thus $2n$ is a $\nu \vee \mu$-name for x. This shows that $\nu \succeq \nu \vee \mu$. Similarly, one can prove that $\mu \succeq \nu \vee \mu$.

Suppose now that τ is any numbering of X such that $\nu \succeq \tau$ and $\mu \succeq \tau$. This means that there are recursive functions $F :\subseteq \mathbb{N} \to \mathbb{N}$ and $G :\subseteq \mathbb{N} \to \mathbb{N}$ such that $\forall n \in \mathrm{dom}(\nu), \nu(n) = \tau(F(n))$ and $\forall n \in \mathrm{dom}(\mu), \mu(n) = \tau(G(n))$. Let n be a number in $\mathrm{dom}(\nu \vee \mu)$. If n is even, $n = 2k$, then $\nu \vee \mu(n) = \nu(k) = \tau(F(k))$. If n is odd, $n = 2k+1$, then one has $\nu \vee \mu(n) = \mu(k) = \tau(G(k))$. The function ϕ defined by $\phi(2k) = F(k)$ and $\phi(2k+1) = G(k)$ is recursive, and thus $\nu \vee \mu \succeq \tau$. □

Proposition 1.1.17. *Let ν, μ and τ be numberings of X.*

- *If $\nu \equiv \mu$, then $\nu \vee \tau \equiv \mu \vee \tau$,*
- *$\nu \vee \mu \equiv \mu \vee \nu$,*
- *$\nu \vee \nu \equiv \nu$*
- *$\nu \vee \nu_\emptyset \equiv \nu$*

Proof. Straightforward, left to the reader. □

As a corollary to this proposition we can define the following:

Definition 1.1.18. Let Λ and Θ be numbering types of X. The numbering type $\Lambda \vee \Theta$ (read "or") is the defined as being the $\equiv$-class of $\nu \vee \mu$ for any ν in Λ and μ in Θ.

We now define the numbering obtained by giving as a name for a point in x both a ν-name and a μ-name for it, the "conjunction" of the numberings ν and μ.

We use a pairing function: for natural numbers, let $\langle k_1, k_2 \rangle$ designate the integer $2^{k_1}3^{k_2}$, and denote by val_2 and val_3 the 2-adic and 3-adic valuations.

Definition 1.1.19. Let ν and μ be numberings of X. Define a numbering $\nu \wedge \mu$ (the *conjunction* of ν and μ) by the following:

$$\mathrm{dom}(\nu \wedge \mu) = \{n \in \mathbb{N}, \mathrm{val}_2(n) \in \mathrm{dom}(\nu), \mathrm{val}_3(n) \in \mathrm{dom}(\mu), \nu(\mathrm{val}_2(n)) = \mu(\mathrm{val}_3(n))\},$$

$$\forall n \in \mathrm{dom}(\nu \wedge \mu),\ \nu \wedge \mu(n) = \nu(\mathrm{val}_2(n)).$$

As we have already said, a $\nu \wedge \mu$-name for a point x of X is constituted of both a ν-name and a μ-name for it. The arbitrary choice of an encoding of pairs used in the definition above is mitigated by the use of numbering types.

Proposition 1.1.20. *Let ν and μ be numberings of X. Then $\nu \wedge \mu \succeq \nu$ and $\nu \wedge \mu \succeq \mu$, and for any τ in $\mathcal{N}_X$, if $\tau \succeq \nu$ and $\tau \succeq \mu$, then $\tau \succeq \nu \wedge \mu$.*

Proof. We first show that $\nu \wedge \mu \succeq \nu$ and $\nu \wedge \mu \succeq \mu$. But given x in X and n in $\mathbb{N}$ such that $\nu \wedge \mu(n) = x$, by definition of $\nu \wedge \mu$ one must have $x = \nu(\mathrm{val}_2(n)) = \mu(\mathrm{val}_3(n))$, and thus the functions val_2 and val_3 are recursive witnesses respectively for $\nu \wedge \mu \succeq \nu$ and $\nu \wedge \mu \succeq \mu$.

Suppose now that $\tau \in \mathcal{N}_X$ is such that $\tau \succeq \nu$ and $\tau \succeq \mu$. This means that there are recursive functions $F :\subseteq \mathbb{N} \to \mathbb{N}$ and $G :\subseteq \mathbb{N} \to \mathbb{N}$ such that $\forall n \in \mathrm{dom}(\tau), \tau(n) = \nu(F(n))$ and $\forall n \in \mathrm{dom}(\tau), \tau(n) = \mu(G(n))$. Let n be a τ-name for a point x in X. Then $\langle F(n), G(n) \rangle$ is a $\nu \wedge \mu$-name for x, since $\nu(\mathrm{val}_2(\langle F(n), G(n) \rangle) = \nu(F(n)) = \tau(n)$ and $\mu(\mathrm{val}_3(\langle F(n), G(n) \rangle) = \mu(G(n)) = \tau(n)$. Thus $\tau \succeq \nu \wedge \mu$. □

The following proposition is straightforward but very useful.

PROPOSITION 1.1.21. *Let ν, μ and τ be numberings of X.*
- *If $\nu \equiv \mu$, then $\nu \wedge \tau \equiv \mu \wedge \tau$;*
- *$\nu \wedge \mu \equiv \mu \wedge \nu$;*
- *$\nu \wedge \nu \equiv \nu$;*
- *$\nu_\emptyset \wedge \nu \equiv \nu_\emptyset$.*

PROOF. Left to the reader. □

As a corollary to this proposition, the following definition is correct:

DEFINITION 1.1.22. Let Λ and Θ be numbering types of X. The numbering type $\Lambda \wedge \Theta$ (read "and") is defined as being the $\equiv$-class of $\nu \wedge \mu$ for any ν in Λ and μ in Θ.

The following proposition relates the conjunction and disjunction operations on $\mathcal{NT}_X$.

PROPOSITION 1.1.23. *Let ν, μ and τ be numberings of X.*
- $\nu \vee (\mu \wedge \tau) \equiv (\nu \vee \mu) \wedge (\nu \vee \tau)$
- $\nu \wedge (\mu \vee \tau) \equiv (\nu \wedge \mu) \vee (\nu \wedge \tau)$

PROOF. Sketch.
Let x be a point in X. A $\nu \vee (\mu \wedge \tau)$-name for x is either a ν-name of x, or a μ-name of it together with a τ-name. In the first case, the ν-name can be used to define both a $\nu \vee \mu$-name and a $\nu \vee \tau$-name for x, and thus a $(\nu \vee \mu) \wedge (\nu \vee \tau)$-name for x. In the second case, the μ-name constitutes a $\nu \vee \mu$-name of x, and the τ-name of x can be seen as a $\nu \vee \tau$- name, and thus one also obtains a $(\nu \vee \mu) \wedge (\nu \vee \tau)$-name for x.
The rest of the proof can be carried out similarly. □

THEOREM 1.1.24. *$(\mathcal{NT}_X, \succeq, \wedge, \vee, N_\emptyset)$ is a distributive lattice with a maximal element. It has no minimal element.*

PROOF. Everything was proved in the propositions of this subsection (SubSection 1.1.2), except the fact that $\mathcal{NT}_X$ has no minimal element, which is well known (see **[38]**). □

Note that if ν is a numbering of X, its image, the set of ν-computable points, is an invariant for the relation $\equiv$. We can thus define the set of points that are computable with respect to a numbering type Λ, which we denote $X_{c,\Lambda}$.
The function which associates to a numbering type its set of computable points is in fact a lattice morphism, between the lattice of numbering types on X and the lattice of countable subsets of X, ordered by reversed inclusion ($A \geq B$ if and only if $A \subseteq B$).
We denote by Φ this morphism:

$$\begin{array}{rccc} \Phi: & (\mathcal{NT}_X, \succeq, \wedge, \vee, \Lambda_\emptyset) & \longrightarrow & (\mathcal{P}_{\aleph_0}(X), \subseteq, \cap, \cup, \emptyset) \\ & \Delta & \longmapsto & X_{c,\Delta} \end{array}.$$

1.1.3. Other constructions of numberings.
There are several useful constructions that allow one to build numberings of complicated sets using numberings of simpler sets, beside the lattice operations.

DEFINITION 1.1.25. Given a numbered set (X, ν) and a subset Y of X, define *the restriction of ν to Y* to be the numbering $\nu_{|Y}$ defined by the following:

$$\mathrm{dom}(\nu_{|Y}) = \mathrm{dom}(\nu) \cap \nu^{-1}(Y),$$
$$\forall n \in \mathrm{dom}(\nu_{|Y}),\ \nu_{|Y}(n) = \nu(n).$$

It is easy to check the following proposition, which allows to define the restriction of numbering types:

PROPOSITION 1.1.26. *If Y is a subset of X and if $\nu \equiv \mu$, then $\nu_{|Y} \equiv \mu_{|Y}$.*

We define the product numbering. Denote again val_2 and val_3 the 2-adic and 3-adic valuations.

DEFINITION 1.1.27. If (X, ν) and (Y, ν) are numbered sets, the product of the numberings ν and μ is the numbering $\nu \times \mu$ of $X \times Y$ defined by the following:

$$\mathrm{dom}(\nu \times \mu) = \{n \in \mathbb{N},\ \mathrm{val}_2(n) \in \mathrm{dom}(\nu),\ \mathrm{val}_3(n) \in \mathrm{dom}(\mu)\}$$
$$\forall n \in \mathrm{dom}(\nu \times \mu),\ \nu \times \mu(n) = (\nu(\mathrm{val}_2(n)), \mu(\mathrm{val}_3(n)))$$

We do not prove the following easy proposition:

Proposition 1.1.28. *If $\nu_1 \equiv \nu_2$ and if $\mu_1 \equiv \mu_2$ then $\nu_1 \times \mu_1 \equiv \nu_2 \times \mu_2$.*

Finally, we give the definition of the numbering of a set of computable functions. Denote by ϕ_0, ϕ_1, ϕ_2... an effective enumeration of all recursive functions, as were defined first by Church and Turing. (We do not describe here how to obtain such an enumeration: numberings allow to translate any enumeration of the partial recursive functions to an enumeration of the computable functions between numbered sets, but those enumerations of partial recursive functions should not rest on the theory of numberings to be obtained. Effective enumerations of the partial recursive functions are constructed in all textbooks on computability, see for instance [**111**].)

Recall that associated to a numbering ν is an equivalence relation η_ν, in what follows we use it as a predicate: we denote $\eta_\nu(n, m)$ if n and m are equivalent for η_ν.

Definition 1.1.29. If (X, ν) and (Y, μ) are numbered sets, we define a numbering μ^ν of the set of functions that map ν-computable points of X to computable points of Y as follows:

$$\begin{aligned}\mathrm{dom}(\mu^\nu) = \{i \in \mathbb{N} |\ & \mathrm{dom}(\nu) \subseteq \mathrm{dom}(\phi_i), \\ & \forall n, m \in \mathrm{dom}(\nu), \eta_\nu(n, m) \implies \eta_\mu(\phi_i(n), \phi_i(m))\},\end{aligned}$$

$$\forall i \in \mathrm{dom}(\mu^\nu), \forall x \in X_{c,\nu}, \forall k \in \mathrm{dom}(\nu),\ (x = \nu(k)) \implies (\mu^\nu(i))(x) = \mu(\phi_i(k)).$$

The following commutative diagram renders this definition clearer:

$$\begin{tikzcd} X_{c,\nu} \arrow[r, "\mu^\nu(i)"] & Y_{c,\mu} \\ \mathbb{N} \arrow[u, "\nu"] \arrow[r, "\phi_i"] & \mathbb{N} \arrow[u, "\mu"'] \end{tikzcd}$$

Several examples follow from those constructions:

- The Baire space $\mathcal{N} = \mathbb{N}^\mathbb{N}$ is naturally equipped with the numbering $\mathrm{id}_\mathbb{N}^{\mathrm{id}_\mathbb{N}}$, which we usually denote $\nu_\mathcal{N}$.
- The Cantor space $\mathcal{C} = \{0, 1\}^\mathbb{N}$ admits a numbering induced by its natural embedding into $\mathcal{N}$, we denote it $\nu_\mathcal{C}$.

1.1.4. Recursively-enumerable and semi-decidable sets.

1.1.4.1. *Definitions.* Let (X, ν) be a numbered set.

Definition 1.1.30. A subset Y of X is called *ν-recursively enumerable* (ν-r.e.) if there is a recursively enumerable subset A of $\mathrm{dom}(\nu)$ such that $Y = \nu(A)$.

Definition 1.1.31. A set Y is *ν-semi-decidable* if there is a recursively enumerable subset A of $\mathbb{N}$ such that $A \cap \mathrm{dom}(\nu) = \nu^{-1}(Y)$. The set A is *ν-co-semi-decidable* if there is a co-recursively enumerable subset B of $\mathbb{N}$ such that $B \cap \mathrm{dom}(\nu) = \nu^{-1}(Y)$.

A set is *ν-decidable* if it is both ν-semi-decidable and ν-co-semi-decidable.

Of course, an equivalent formulation is that a set Y is ν-decidable if there exists a procedure that, given a ν-description of an element in X, decides whether or not it belongs to Y. It is ν-semi-decidable if there exists a procedure that stops exactly on the ν-descriptions of elements of Y, and ν-co-semi-decidable if there exists a procedure that stops exactly on the ν-descriptions of elements that do not belong to Y.

Remark 1.1.32. As Remark 1.1.10: a set Y can be ν-decidable without there being a recursive set A such that $A \cap \mathrm{dom}(\nu) = \nu^{-1}(Y)$. The example of Remark 1.1.10 works here as well: was defined there a numbering ν of a pair $\{x, y\}$, such that each of $\{x\}$ and $\{y\}$ is a ν-decidable subset, but there cannot be a recursive set A such that $A \cap \mathrm{dom}(\nu) = \nu^{-1}(\{x\})$.

1.1.4.2. *Precise classification of the undecidability of problems.* Consider a decision problem of the form "does the group G satisfy the property P?", asked with respect to a numbering of the set of finitely generated groups (or of the set of marked groups).

The status of the decidability of this problem can be classified in four categories:

- Decidable: there is an algorithm that always halt, and decides whether or not the group G satisfies P.
- Semi-decidable: there is an algorithm that will halt exactly when the group G satisfies P.
- Co-semi-decidable: there is an algorithm will halt exactly when the group G does not satisfies P.
- Completely undecidable: neither of the cases above.

Throughout this thesis, we will systematically try to classify the different problems we encounter between those four categories. In particular, we have often tried to pursuing our investigations beyond the information that a problem is *undecidable.* A problem in undecidable if it is either semi-decidable, co-semi-decidable or completely undecidable, and whenever facing an undecidable problem, one can ask for a precise classification in one of the categories written above.

Of course, the theory of higher computability allows, through the use of oracles, to classify "completely undecidable" problems according to their Turing degree, or in the arithmetical hierarchy, or in the hyperarithmetical hierarchy, and so on. The present study focuses only on the four categories written above.

Note that it is often the case that a problem is called "recursive" when it is decidable, "recursively enumerable" when it is semi-decidable, and "co-recursively enumerable" when it is co-semi-decidable. However, in the study of finite descriptions of objects that are more complex than natural numbers, this vocabulary is not appropriate. Indeed, the term "recursively enumerable" suggests that the positive solutions of a given problem can be listed. This is not always true for problems defined on sets that are not recursively enumerable.

A simple example is the following: there is an algorithm that, amongst residually finite groups, recognizes finite presentations of the ones that are non-abelian. (See Chapter 3). However, we show in Corollary 2.2.5 that there is no effective enumeration of all finite presentations that define non-abelian residually finite groups.

This problem thus should not be called "recursively enumerable", while it is semi-decidable. (Note however that for groups described by finite presentations, in the class of all finitely presented groups, the notions of semi-decidable set and of recursively enumerable set coincide. This comes from two fact: the finite presentation numbering is positive, and all finite presentations can be enumerated.)

The formal definitions of semi-decidable and recursively enumerable sets given above in the context of numbered sets clearly show why these notions differ.

1.2. Marked groups and word problem algorithms

1.2.1. Marked groups. A *k-marked group* is a finitely generated group together with a k-tuple of elements that generate it.

A *morphism of k-marked groups* from (G, S) to (H, S') is a group morphism between G and H that maps S to S' (respecting the order). Such a morphism is an isomorphism if the underlying group morphism is a group isomorphism, and of course marked groups are considered up to isomorphism. Marked groups together with morphisms of marked groups form a category.

We denote by $\mathcal{G}$ the set of isomorphism classes of marked groups.

It is in fact convenient, when studying k-marked groups, to fix a free group $\mathbb{F}_k$ of rank k, together with a basis S. A k-marking of a group G can then be seen as an epimorphism $\varphi : \mathbb{F}_k \to G$, the image of S by φ defines a marking with respect to the previous definition. Two k-marked groups are then isomorphic if they are defined by morphisms with identical kernels: the isomorphism classes of k-marked groups are in bijection with the normal subgroups of a rank k free group. This is nicely related to algorithmic descriptions of groups: the basis S of the free group $\mathbb{F}_k$ can be seen as a set of tape symbols which will be common to all algorithms that describe groups.

Notice that the relation $G \geq H$, defined for marked groups G and H by "there is a morphism of marked groups from G to H", defines an order on the set of marked groups, it is easy to see that it admits both meet and join operations. We describe now the meet and join operations of the lattice of marked groups. Notice that those operations correspond to the product and coproduct in the category of marked groups.

1.2.1.1. *Meet/Coproduct.* Let (G, S) and (H, S) be two k-marked groups, generated by the same family S. Consider two presentations $\pi_1 = \langle S|R_1\rangle$ and $\pi_2 = \langle S|R_2\rangle$ that define respectively (G, S) and (H, S). Define $(G, S) \wedge (H, S)$ to be the group given by the presentation $\langle S|R_1, R_2\rangle$.

In terms of morphisms from the free group, if N_1 is a normal subgroup of the free group $\mathbb{F}_k$ that defines (G, S) and N_2 defines (H, S), $(G, S) \wedge (H, S)$ is defined by the normal subgroup of $\mathbb{F}_k$ generated by $N_1 \cup N_2$.

The following follows directly from the construction above:

Proposition 1.2.1. *The marked group $(G, S) \wedge (H, S)$ is the greatest lower bound for the pair $\{(G, S), (H, S)\}$ with respect to the order of the set of marked groups.*

1.2.1.2. *Join/Product.* Let (G, S) and (H, S') be two k-marked groups, and denote $S = \{s_1, ..., s_k\}$ and $S' = \{s'_1, ..., s'_k\}$.

Denote by $(G, S) \vee (H, S')$ the subgroup of the group $G \times H$ generated by the elements $\{(s_1, s'_1), ..., (s_k, s'_k)\}$.

In terms of morphisms from the free group, if N_1 is a normal subgroup of the free group $\mathbb{F}_k$ that defines (G, S) and N_2 defines (H, S'), $(G, S) \vee (H, S')$ is defined by the normal subgroup $N_1 \cap N_2$.

The equivalence between those two definitions is straightforward: a product w of elements of the form (s_i, s'_i) defines the identity if and only if the product on each component defines the identity, and thus if and only if the word w defines and element of both the kernels N_1 and N_2.

Proposition 1.2.2. *The marked group $(G, S) \vee (H, S')$ is the least upper bound for the pair $\{(G, S), (H, S')\}$ with respect to the order of the set of marked groups.*

Proof. The proof is immediate from the characterization of $(G, S) \vee (H, S)$ that relies on a normal subgroup of the free group: a marked group (K, S'') defined by a normal subgroup N of $\mathbb{F}_k$ satisfies $(K, S'') \geq (G, S)$ and $(K, S'') \geq (H, S')$ if and only if both $N \subseteq N_1$ and $N \subseteq N_2$ hold, and this is the case if and only if $N \subseteq N_1 \cap N_2$ holds. □

The behavior of those operations is, by certain aspects, more complicated than the corresponding operations in the category of groups, the free and direct products.

Notice for instance the following problems:

(1) It follows directly from the definition of $(G, S) \wedge (H, S)$ that if both (G, S) and (H, S) are finitely presentable, then so is their meet. Whether the same holds for the join operation is unclear.
(2) It is obvious that if (G, S) and (H, S) have solvable word problem, the so does $(G, S) \vee (H, S)$. However, whether this holds also for the meet operation seems to be an open problem.

The set of k-marked groups equipped with this order is thus a lattice.

We call a group an *abstract group* when we want to emphasize the fact that it is not a marked group.

Note finally that each numbering of the set of marked groups gives rise to a numbering of the set of finitely generated groups, since given a numbering $\nu :\subseteq \mathbb{N} \to \mathcal{G}$, one can always compose it with the projection $\mathcal{G} \to \mathcal{G}/\sim$ to obtain a numbering of finitely generated groups. (Here $\sim$ denotes the *abstract group isomorphism* equivalence relation.) The converse need not be true in general, as it would suppose a canonical way of choosing a generating family of a given finitely generated group. However, we remark in Chapter 2 that all the commonly used numberings of $\mathcal{G}/\sim$ in fact come from numberings of $\mathcal{G}$.

1.2.2. Numbering of the elements of a marked group. In a marked group (G, S), it is customary to describe group elements by words on $S \cup S^{-1}$. This description is in fact canonical, in a sense that can be made precise using numberings.

We denote by $\Lambda_{(G,S)}$ the numbering type of (G, S) associated to the idea that elements be described by words on $S \cup S^{-1}$, we define it formally in the next definition. Denote by $(p_n)_{n\in\mathbb{N}}$ the sequence of prime numbers.

Definition 1.2.3. If (G, S) is a marked group, and $S = (s_0, s_2, ..., s_{k-1})$, we define the numbering $\nu_{(G,S)}$ on $\mathbb{N}$ as follows. For i between k and $2k-1$, denote by s_i the element s_{i-k}^{-1} of G. Given a natural number n, decompose it as a product of primes $n = p_0^{\alpha_0}...p_m^{\alpha_m}$. Then, for each i between 1 and m, denote $\widetilde{\alpha_i}$ the remainder in the Euclidian division of α_i by $2k$. We then put:

$$\nu_{(G,S)}(n) = s_{\widetilde{\alpha_1}} s_{\widetilde{\alpha_2}} ... s_{\widetilde{\alpha_m}} \in G.$$

The numbering type $\Lambda_{(G,S)}$ is the $\equiv$-equivalence class of $\nu_{(G,S)}$.

The arbitrary choices that are made in this definition are of course unimportant, as is shown by the following proposition:

Proposition 1.2.4. *The numbering type $\Lambda_{(G,S)}$ is the greatest numbering type, for the order $\succeq$, which satisfies the following conditions:*

- *All elements of G are $\Lambda_{(G,S)}$-computable;*
- *The group law and the inverse function on G are respectively $(\Lambda_{(G,S)} \times \Lambda_{(G,S)}, \Lambda_{(G,S)})$ and $(\Lambda_{(G,S)}, \Lambda_{(G,S)})$ computable.*

(In particular, any numbering type which satisfies these conditions can be compared to $\Lambda_{(G,S)}$ for the order $\succeq$.)

Note that the first condition of this proposition could be replaced equivalently by: "The elements of the generating tuple S are $\Lambda_{(G,S)}$-computable".

Proof. Suppose that ν is any numbering which is surjective and for which the group law and the inverse function of G are computable.

We show that $\nu_{(G,S)} \succeq \nu$, where $\nu_{(G,S)}$ is the numbering which was used to define the numbering type $\Lambda_{(G,S)}$. The generating set of G is denoted $S = (s_0, s_2, ..., s_{k-1})$. As ν should be surjective, there are numbers u_0, ...,u_{k-1} such that $\nu(u_i) = s_i$.

As the group law should be computable for ν, there is a recursive function F such that $\nu(F(i,j)) = \nu(i)\nu(j) \in G$, and a recursive function I that computes the inverse for ν.

Given an integer n, which we decompose as a product of primes, $n = p_0^{\alpha_0}...p_m^{\alpha_m}$, recall that $\nu_{(G,S)}(n) = s_{\widetilde{\alpha_1}}s_{\widetilde{\alpha_2}}...s_{\widetilde{\alpha_m}}$. Consider the function $H : \mathbb{N} \to \mathbb{N}$ defined as follows: map an integer α to its reminder modulo $2k$, which we denote $\widetilde{\alpha}$, then, if $\widetilde{\alpha} \leq k-1$, map it to $u_{\widetilde{\alpha}}$, otherwise, if $\widetilde{\alpha} \geq k$, map it to $I(u_{\widetilde{\alpha}-k})$.

Then $F(H(\alpha_1), F(H(\alpha_2), ...)))$ gives a ν-name for $\nu_{(G,S)}(n)$, and the procedure which produces this name from n is clearly recursive. $\square$

This proposition is in fact a simple application of a well known fact about numberings: if (X, Δ) is a numbered set, and if $\{f_1, ..., f_n\}$ are functions defined on cartesian powers of X to X, then there is a greatest numbering type Λ, that is below Δ ($\Delta \succeq \Lambda$), and which renders all functions f_i computable. See for instance **[128]**. The numbering type $\Lambda_{(G,S)}$ is obtained following this principle, using for Δ the equivalence class of the numbering ν_0 that describes only S: ν_0 is defined on $\{0, ..., k-1\}$ and it maps i to s_i.

Proposition 1.2.4 shows that describing group elements as products of the generators is the description that gives as much information as can be given on group elements if we want the group operations to be computable. The previous proposition thus has an easy corollary:

Corollary 1.2.5. *The numbering type $\Lambda_{(G,S)}$ is decidable if and only if there exists a numbering type Δ of G, which is surjective and decidable, and which makes the group operations of G computable.*

Proof. Suppose that Δ is as in the hypotheses of the corollary. Then, by Proposition 1.2.4, one should have $\Lambda_{(G,S)} \succeq \Delta$. Thus if Δ is decidable, then so should be $\Lambda_{(G,S)}$. $\square$

We can finally define solvability of the word problem:

Definition 1.2.6. A finitely generated group is said to have *solvable word problem* if $\Lambda_{(G,S)}$ is decidable.

In this case, what we call a *word problem algorithm* is the recursive function that witnesses for the fact that $\Lambda_{(G,S)}$ is decidable. We precise this in what follows.

1.2.3. Numbering associated to word problem algorithms. We give three definitions of the numbering type associated to word problem algorithms. What can be computed from such an algorithm is the subject of Chapter 5, but those notions are useful as early as Chapter 2.

1.2.3.1. *First definition of* Λ_{WP}. We will now define the numbering type associated to word problem algorithms. Our definition is based not only on the numbering type $\Lambda_{(G,S)}$, but also on the distinguished numbering $\nu_{(G,S)}$ which we have used to define the numbering type $\Lambda_{(G,S)}$.

The numbering $\nu_{(G,S)}$ is decidable if and only if there is a recursive function of two variables H such that $H(i,j) = 0$ if $\nu_{(G,S)}(i) = \nu_{(G,S)}(j)$ and $H(i,j) = 1$ otherwise. In this case, H is said to witness for the fact that $\nu_{(G,S)}$ is decidable.

Let ϕ_0, ϕ_1, ... be an effective enumeration of all partial recursive functions. We can consider that those functions depend on two variables using an encoding of pairs of natural numbers.

Define as follows a numbering ν_{WP}.

Pose $\nu_{WP}(n) = (G,S)$ if and only if n can be decomposed as $n = 2^k(2m+1)$, S is a family with k elements, $\nu_{(G,S)}$ is decidable, and ϕ_m is a recursive function that witnesses for the fact that $\nu_{(G,S)}$ is decidable.

To check that this is a correct definition, we must check that the marked group (G,S) is uniquely defined. This is to say: we must check that a word problem algorithm defines uniquely a marked group.

But this is straightforward: if, in the definition above, $n = 2^k(2m+1)$ codes for two marked groups (G,S) and (H,S'), first it must be that S and S' have the same cardinality k, and, secondly, that ϕ_m is a recursive function that witnesses for the decidability of both $\nu_{(G,S)}$ and $\nu_{(H,S')}$, but then (G,S) and (H,S') should satisfy exactly the same relations, and thus be isomorphic as marked groups.

We then define Λ_{WP} to be the $\equiv$-equivalence class of ν_{WP}.

1.2.3.2. *Cayley graph definition of* Λ_{WP}. Describing a marked group by a word problem algorithm is equivalent to describing it by its labelled Cayley graph. And thus when we investigate decision problems for groups described by word problem algorithms in Chapter 5, we are also studying "what can be deduced about a group, given its labeled Cayley graph". Of course, the graph should be suitably encoded into a finite amount of data. We detail this now.

Definition 1.2.7. If $(G, S = (s_1, ..., s_n))$ is a marked group, the labelled Cayley graph associated to it is the graph whose vertexes are elements of G, and whose (directed) edges are defined as follows: there is an edge with label s_i from the vertex $g_1 \in G$ to the vertex $g_2 \in G$ if and only if $g_1 s_i = g_2$.

We can then use the standard way to encode infinite graphs (of course, computable graphs) to define a numbering of Cayley graphs. This again uses a standard enumeration ϕ_0, ϕ_1, ϕ_2,... of partial computable functions.

An oriented edge-labelled graph is a quadruple $(V, E, C, c : E \to C)$, where: V is a set, the set of vertices, E is the set of oriented edges, i.e. a subset of $V \times V$, C is a set of colors, which here we suppose finite, and c is a function which defines the color of a given edge.

Such a graph $\Gamma = (V, E, C, c : E \to C)$ is called computable if it is isomorphic to a graph $\Gamma_1 = (V_1, E_1, C_1, c_1 : E_1 \to C_1)$, which satisfies additionally that: V_1 is a recursive subset of $\mathbb{N}$, E_1 is a recursive subset of $\mathbb{N} \times \mathbb{N}$, $C_1 = \{1, ..., k\}$ for some $k \in \mathbb{N}$, and $c_1 : E_1 \to C_1$ is a recursive function. In this case, Γ_1 is called a *computable model* of Γ.

Notice that each element of the tuple $(V_1, E_1, C_1, c_1 : E_1 \to C_1)$ is associated to some finite data that can be encoded: the characteristic function of V_1, which should be computable, the characteristic function of E_1, the natural number k such that $C_1 = \{1, ..., k\}$, and the code of the function c_1.

We can thus define a numbering ν_Γ of edge-labelled graphs by saying that a computable model $\Gamma_1 = (V_1, E_1, C_1, c_1 : E_1 \to C_1)$ of a graph Γ is encoded by a tuple (i, j, k, l), where: $\phi_i = \chi_{V_1}$, $\phi_j = \chi_{E_1}$, $C_1 = \{1, ..., k\}$, $\phi_l = c$.

This is a correct definition, because one easily checks that the tuple (i, j, k, l) defines a unique graph, and so there is no ambiguity in the definition given above.

We now have a numbering ν_Γ of edge-labelled graphs, we can restrict it to the set of Cayley graphs, and, because a labelled Cayley graph defines uniquely a marked group, we can consider that this new numbering is a numbering of the set of marked groups (we compose the numbering of graphs to the function that maps a labelled Cayley graph to the group it defines).

Note that, in the computable model of a Cayley graph, we can always suppose that there is a vertex at 0, and that it is associated to the identity element of the group whose graph it is.

This defines a numbering that we denote ν_{Cay}, which is associated to the idea "a marked group is described by algorithms that describe its Cayley graph".

We can now show:

Theorem 1.2.8. *The numbering ν_{Cay} is $\equiv$-equivalent to ν_{WP}.*

Proof. Sketch.

Given a ν_{Cay}-name for a marked group (G, S), i.e. given a Cayley graph Γ_1 for it, we can solve the word problem in (G, S) by following edges along a word: given a word $w = a_1 a_2 ... a_n$ on $S \cup S^{-1}$, starting from any vertex v_1 in Γ_1, we can find (by an exhaustive search) a sequence of vertices v_2, ..., v_{n+1} such that $v_{i+1} = v_i a_i$. We then solve the word problem by checking whether $v_1 = v_{n+1}$, i.e. by checking whether the word w defines a loop in the Cayley graph of G.

Conversely, given a word problem algorithm for (G, S), we can build a recursive model $\Gamma_1 = (V_1, E_1, C_1, c_1 : E_1 \to C_1)$ of the Cayley graph of (G, S) as follows:

- Consider an enumeration of all words on $S \cup S^{-1}$ following a given order, say by length and then lexicographically. We can then delete, using the word problem algorithm of G, any element that is redundant in this list. We obtain a list w_0, w_1, w_2, ... which contains a single word on $S \cup S^{-1}$ for each element of G.
- Define a numbering μ of G by saying that $\mu(i) = w_i$. We put $V_1 = \text{dom}(\mu)$.
- V_1 is $\mathbb{N}$ if G is infinite, it is $\{0, ..., \text{card}(G) - 1\}$ otherwise. A computable characteristic function for V_1 can be obtained as follows: the list w_0, w_1, ... can be enumerated, and thus, given some number i, if it was found that G contains more than $i + 1$ elements, i belongs to V_1. On the contrary, while the list w_0, w_1, w_2, ... is enumerated, we can search for an initial segment of it of length less than i, and that is stable by multiplication by any generator. If such a segment is found, G must be finite, and we know it has cardinality less than i. In this case, we can conclude that $i \notin V_1$.
- We define E_1 and c_1 by saying that (i, j) is an edge labelled by $s \in S$ if and only if $w_i s = w_j$. This can be effectively checked thanks to the word problem algorithm for (G, S), and thus we can produce the recursive functions that define E_1 and c_1.

□

1.2.3.3. *Computable group definition of Λ_{WP}.* Another point of view on Λ_{WP} follows the point of view of Malcev in **[72]** (see **[73]** for an English translation) and Rabin in **[104]**.

Definition 1.2.9. A countable group G is *computable* if there are recursive function $F : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ and $G : \mathbb{N} \to \mathbb{N}$ such that $(\mathbb{N}, F, G, 0)$ is a group that is isomorphic to $(G, \cdot, ^{-1}, e)$.

If G is a finitely generated group, an isomorphism $\Theta : (G, \cdot, ^{-1}, e) \to (\mathbb{N}, F, G, 0)$ can be described by giving the images of a generating family of G in $\mathbb{N}$.

We define a new numbering of marked groups, denoted ν_{MR}, (MR for Malcev-Rabin) as follows.

The description of a marked group (G,S) for ν_{MR} is an encoded quadruple (k,m,i,j), such that:

- There exists a group isomorphism $\Theta : (G,\cdot,^{-1} e) \to (\mathbb{N},F,G,0)$, where F and G are recursive function;
- k gives the cardinality of S;
- m can be decoded as a k-tuple, which give the images of the elements of S by Θ;
- i and j define the recursive functions F and G (i.e. $\phi_i = F$ and $\phi_j = G$).

As before, one can easily check that this definition is sound by checking that there is no ambiguity as to which isomorphism Θ is encoded by a number n.

The following theorem is well known.

Theorem 1.2.10. *The numberings ν_{WP} and ν_{MR} are $\equiv$-equivalent.*

Proof. The proof is essentially the same as Theorem 1.2.8. □

Corollary 1.2.11. *The finitely generated computable groups correspond exactly to the finitely generated groups which have solvable word problem.*

1.3. Numberings and algorithmic descriptions of groups

The theory of numberings is ubiquitously used in effective mathematics, albeit, most of the times, it is used implicitly. This does not mean that the effective mathematics that are written with no explicit mention of the underlying numberings are not written rigorously.

Indeed, just as it is commonly accepted that it is possible to affirm that something is computable without explicitly presenting a Turing machine that computes it, because when an effective process is described, it is usually very clear which steps should be taken to explicit a machine that computes it, it is customary to employ numberings implicitly when it is clear how the objects considered should be encoded. (And, of course, here is of prime importance the fact that numberings are considered up to equivalence, and, as we have seen several times by now, the arbitrary choices made when encoding objects are mitigated by the fact that the resulting numbering types are usually canonical objects, when the numberings were not.)

While most of the results contained in this thesis could have been written in the language of numbered sets, we have done so only when it was necessary, that is in Chapter 5, where we present general results of computable analysis that cannot be stated without using numberings.

The following example illustrates the fact that it is not always necessary to use numberings to formulate effective statements.

If Λ_{FP} is the numbering type associated to finite presentations, and Λ_{WP} the one associated to word problem algorithms, and if $\mathcal{C}$ is a set of finitely presented groups, the fact that groups in $\mathcal{C}$ have uniformly solvable word problem can be written as:

$$(\Lambda_{FP})_{|\mathcal{C}} \succeq (\Lambda_{WP})_{|\mathcal{C}}.$$

On the other hand, the sentence "groups in $\mathcal{C}$ have uniformly solvable word problem" is perfectly rigorous, and thus we do not have to explicit the relation between numbering types written above.

Chapters 3 and 4 of this manuscript are thus written without the vocabulary of numberings, both because it makes them easier to read, and because it allows to follow the original presentation of the articles that correspond to those chapters.

This is also the case of Chapter 2, although here the reasons for the lack of use of numberings are more dubious. Indeed, most of the results of Chapter 2 concern numberings, and can be expressed very naturally in terms of numberings. For instance, the main result of this chapter, which we have called "algorithmic characterization of finitely presented groups", states that the numbering associated to finite presentations is $\equiv$-equivalent to the conjunction of the numbering associated to recursive presentations and to the one associated to what we have called marked quotient algorithms.

However, there is one important aspect of the problems discussed in Chapter 2 that we are still unable to translate in terms of numberings, it is the concept of an "algorithmic description", that is, of a description of a group *in terms of algorithms* (like when a group is described by an algorithm solving the word problem, or by an algorithm giving a recursive presentation, and so on).

The possibility of defining algorithmic descriptions in terms of numberings is a research problem on which I have been working since I have encountered the theory of numberings.

The informal idea would be to define numberings that *cannot be defined without using an enumeration of all partial recursive functions. (*See for instance how the numbering associated to word problem algorithms defined above relies on such an enumeration.)

As things stand, I am unable to define this property properly, and so I have abandoned the idea of writing Chapter 2 in the vocabulary of numberings, as I would have been unable to use to this formalism throughout this chapter.

Note finally that the introduction on numberings that constitutes the present chapter should be enough to enable the reader to translate easily most of the statements that appear in Chapter 2 into statements about numberings, and by "most statements" I mean exactly "all statements, with the exception of the statements that concern algorithmic descriptions, and that I have myself been unable to express in terms of numberings".

1.4. Kolmogorov Complexity

In Chapter 2, we ask for an equivalent of the Kreisel-Lacombe-Schoenfield-Ceitin Theorem for different descriptions of groups. In Chapter 5, we pose this problem in particular for groups described by word problem algorithms, and give much more details. In particular, we explain there how this question is related to the Kolmogorov complexity, following the article [**58**].

The underlying problem is the following: the description of a group by a Turing machine provides an upper bound on the Kolmogorov complexity of this group. The task that one is faced with when trying to prove the Kreisel-Lacombe-Schoenfield-Ceitin is to prove that this information cannot be used to compute anything concrete about the group in question.

We will thus now briefly introduce the Kolmogorov complexity of strings and prove a property that we shall use in Section 5.2.5 (and only there), the existence of strings of maximal Kolmogorov complexity.

The following follows mostly Chapter 1 of [**118**].

Let Σ be a finite alphabet, and Ξ the set of words on Σ. Denote by $|w|$ the length of an element w of Ξ.

A partial computable function $D : A \to \Xi$, with $A \subseteq \Xi$, is called a *description mode*. (Note that since Turing machines manipulate symbols and words, the notion of a computable function defined on Ξ is defined just as easily as the notion of a computable function on $\mathbb{N}$.)

For a word w and a description mode D, *the complexity of w with respect to D* is the minimal length of a word v such that $D(v) = w$. This defines a function $C_D : \Xi \to \mathbb{N}$.

A description mode D_1 is said to be *not worse* than a description mode D_2 if there exists a constant $c \in \mathbb{N}$ such that for any word w in Ξ,

$$C_{D_1}(w) \leq C_{D_2}(w) + c.$$

The fundamental theorem of the Kolmogorov complexity is the following:

THEOREM 1.4.1 (Solomonoff–Kolmogorov). *There exists a description mode that is not worse than all description modes.*

PROOF. (Sketch) A description mode D_0 is defined as follows: on input a string that encodes a pair (D, v), where D is another decompressor and v is a string, it computes $D(v)$.

That is: D_0 is chosen to be a universal Turing machine.

The only detail that needs to be checked is that a string that encodes (D, v) can be chosen so that its length be of the form $|v| + c$, where c is a constant that depends on D but not on v. Kolmogorov's solution to this problem is to double the bits that code D: if w is a word that codes D, consider the word $\hat{w}$, obtained by doubling all letters of w, and ending it with a predefined sequence of two different letters, here we choose the sequence ab. For instance, if w is $aaba$, $\hat{w}$ is $aaaabbaaab$. Then the code for (w, v) is chosen to be the string $\hat{w}v$. The length of $\hat{w}v$ has the required dependence in v, and it is clear that each word w and v can successfully be retrieved from the word $\hat{w}v$.

Finally, consider any description mode D_1. Let w be a word, and v a description of w for D_1. Then the code for (D_1, v) is a description of w for D_0, and its length is of the form $c + |v|$, where c depends on D_1 but not on v. And thus D_0 is not worse than D_1. ☐

A description mode that is not worse than any other description mode is called *optimal*.

In what follows, we fix an optimal description mode D_0. The *Kolmogorov complexity of a word w* is the complexity of w with respect to D_0. Denote it $C(w)$, dropping the subscript D_0.

If D_1 is another optimal description mode, there must exist a constant k_{D_0,D_1} such that for any word w,

$$|C_{D_0}(w) - C_{D_1}(w)| \leq k_{D_0,D_1}.$$

Thus the Kolmogorov complexity is defined "up to a constant", and while the choice of an optimal description mode allows to discuss the Kolmogorov complexity of a single string, the actual purpose of the Kolmogorov complexity is to compare the complexity of sequences of strings.

PROPOSITION 1.4.2. *Applying any computable transformation to a sequence of strings can only diminish its asymptotic Kolmogorov complexity.*

PROOF. One easily proves this using the optimal description mode D_0 that was constructed above. $\square$

The simplest description mode is the identity function on Ξ. It is not an optimal description mode, since the sequences $(n)_{n\in\mathbb{N}}$ and $(n^2)_{n\in\mathbb{N}}$ have different asymptotic complexities with respect to the identity function, while they have the same Kolmogorov complexity by the proposition above.

The identity function can be used to show:

PROPOSITION 1.4.3. *There is a constant c such that for any string w, $C(w) \leq |w| + c$.*

PROOF. Apply the optimality of D_0: it is not worse then the identity on Ξ, and the complexity of a string with respect to the identity function is precisely its length. $\square$

We will use the following result in Section 5.2.5:

PROPOSITION 1.4.4. *There exists a sequence $(w_n)_{n\in\mathbb{N}}$ and a constant $k \in \mathbb{N}$ such that for all n, $|w_n| = n$ and $C(w_n) \geq n - k$.*

PROOF. Let s denote the cardinal of the alphabet Σ. Notice first that there are at most $s^n - 1 = 1 + s + s^2 + ... + s^{n-1}$ words of Kolmogorov complexity strictly less than n, since those words are all images of words of length strictly less than n. Thus for each n, there is a word of length n whose Kolmogorov complexity is at least n. $\square$

CHAPTER 2

Remarks and problems about algorithmic descriptions of groups

Introduction

The study of decision problems for groups began with Max Dehn who, in 1911 ([**28**]), formulated the three famous problems which are now associated with his name: the word problem, the conjugacy problem and the isomorphism problem.

His motivation in introducing those came from topology, in particular from the study of the fundamental group, which had been introduced not long before by Poincaré. Because of this, he defined those problems only for finitely presented groups; his 1911 article starts by stating the following (we quote Stillwell's translation, [**29**]):

> "The general discontinuous group is given by n generators and m relations between them:
> $$R_1(s_{i_1}, ...) = 1$$
> $$.........$$
> $$R_m(s_{i_m}, ...) = 1$$
> as first defined by Dyck ([**33, 34**]). The results of those works, however, relate essentially to finite groups. The general theory of groups defined in this way at present appears very undeveloped in the infinite case. Here there are above all three fundamental problems whose solution is very difficult and which will not be possible without a penetrating study of the subject."

Dehn then proceeds to defining his three problems.

This extract shows that Dehn introduced what would be known as computability theory (once it was invented) in the field of group theory at the express purpose of studying finitely presented groups, in order to build a theory of finitely presented groups.

If we consider this, in light of the fact that the finitely presentable groups are exactly the fundamental groups of closed manifolds, we can see that from its very beginning the theory of decision problems for groups lay at the intersection, not only of group theory and computability theory, but of group theory, computability theory *and* topology.

The intersection of these three domains has now been well studied, and it is very satisfying to think that there are many instances where a problem which was posed in one field was solved thanks to methods of another one.

Consider now the following question, which could be taken as the starting point of our investigation: *what would the theory of decision problems for groups look like, had it been invented by someone with no knowledge of topology?*

Actually, the first step in the making of an algorithmic theory of groups is not just a thought exercise that we propose here, it was actually written, independently by Maltsev in [**72**] (a translation is available in [**73**]) and by Rabin in [**104**]. They defined what is known as a computable group, and this notion can easily be seen to correspond to having solvable word problem with respect to a countably infinite generating set.

Although both of these papers prove some fundamental results, the working frame they propose is not entirely satisfactory: all the notions that those papers introduce were already known, or already expressible, in the language of the existing theory of decision problems for finitely presented groups. And, what's more, as the concept of "computable group" does not encompass all finitely presented groups, because of the Novikov-Boone Theorem, at the time of their writing, the set of groups that admit a finite description that can be manipulated by a computer was already known to exceed that of computable groups.

Thus upon reading those two articles, one might have the impression that, paradoxically, the intersection of group theory and computability theory is strictly contained in the intersection of group theory, computability theory and topology. In this article, we show that this does not have to be the case.

There is another approach to our motivating question which seems to be worth mentioning, and which consists in making analogies with the theory of computable analysis, which is the domain that lies at the intersection of computability theory and analysis. (In Chapter 5, we *apply* the theory of computable analysis to the space of marked groups. What we propose here is not to apply this theory, but to consider the ways in which it was developed, in order to see the differences with the way the theory of decision problems for groups was constructed.)

From this point of view, the introduction of groups with solvable word problem naturally corresponds to that of computable reals, decidable and semi-decidable properties of groups could have been defined by analogy with computable functions, etc.

This analogy also suggests that recursively presented groups would also have been introduced, as those are the equivalent of lower (or upper) semi-computable reals, which are well known objects in computable analysis ([**120**]). Note furthermore that those groups could have thus been defined with no reference to group presentations: a finitely generated group G together with a generating family S admits a recursive presentation on S if and only if there exists an algorithm that recognizes the words in the generators and their inverses that define the identity element of the group, thus solving "half" of the word problem. Throughout this chapter, we will usually favor the use of such algorithms over that of recursive presentations. They are called *r.e. algorithm*s, for *recursive enumeration algorithm*s. Recursively presented groups will be called r.p. groups for short. We use the notation $\mathcal{A}_{re}^{G;S}$ for a r.e. algorithm of the group G generated by S, and omit both superscripts G and S when possible. We discuss why we choose to use r.e. algorithms over recursive presentations in Section 2.2.

The parallel with computable analysis seems fruitful in several regards. For instance, in Section 2.2, we ask for an equivalent to one of the most important theorems of computable analysis, the Kreisel-Lacombe-Schoenfield-Ceitin Theorem.

Let us come back to our thought exercise, of describing a theory of decision problems for groups, had it been introduced with no topology in mind.

If we are to describe such a theory, one of our main concerns will be to define decision problems for groups that are not necessarily finitely presented. A key remark here is that we must distinguish between the so called "local" and "global" decision problems. This distinction is fundamental, so much so that Miller's well known 1992 survey article ([**83**]) starts with the following:

> "This is a survey of decision problems for groups, that is of algorithms for answering various questions about groups and their elements. The general objective of this area can be formulated as follows:
> Objective:
> To determine the existence and nature of algorithms which decide
> - local properties - whether or not elements of a group have certain properties or relationships;
> - global properties - whether or not groups as a whole possess certain properties or relationships.
>
> The groups in question are assumed to be given by finite presentations or in some other explicit manner."

Notice that, as opposed to Dehn, Miller allows descriptions of groups that are not finite presentations.

The reason why we introduce this distinction here is the following: the investigation of decision problems in regard to local properties of groups is not affected by whether the considered groups are finitely presented or not -a solution to the word problem in some group depends only on a choice of a generating set for this group, (of a marking of this group), but not of a presentation. We quote here Miller's definition of the word problem:

> "Word problem: Let G be a group given by a finite presentation. Does there exist an algorithm to determine of an arbitrary word w in the generators of G whether or not $w = I$?"

Notice how the assumption that G is given by a finite presentation can be dropped, the resulting definition stays valid in any countable group with a fixed generating family.

Let us now quote Miller's definition of the isomorphism problem:

> "Isomorphism problem: Does there exist an algorithm to determine of an arbitrary pair of finite presentations whether or not the groups they present are isomorphic?"

Here, if one were to take off the terms "finite presentations" from the formulation of this problem, and replace them by "a pair of groups", the problem stops being posed well, since of course an algorithm cannot take as input the abstract mathematical object that is a group.

Thus we must find an adequate replacement for "finite presentations" in this definition, which should also be more general than the notion of computable group, as we have already explained that this notion was too restrictive.

There is one last obvious notion one could use in the hope of finding a unique definition for decision problems about groups that need not be finitely presented: the notion of recursive presentation.

Of course, since a recursive presentation can be encoded in a Turing machine, and since Turing machines can run other Turing machines, recursive presentations can be used to define global decision problems.

However, we have the following proposition, which motivated the present investigation:

Proposition 2.0.1. *Let G and H be finitely generated, recursively presented groups, and suppose that H is a strict quotient of G. Then, the problem of deciding whether a given pair of recursive presentations, that both define either G or H, define the same group, is unsolvable.*

This proposition is stated and proved in Section 2.1.

This proposition prevents one from building an interesting theory of algorithmic problems for groups described by recursive presentations, and this, not because it shows that the isomorphism problem is unsolvable for groups described by recursive presentations, this is to be expected, but because it is unsolvable in any class of finitely generated groups that contains a group and a strict quotient of it.

An interesting theory of decision problems for groups strives on the fact that most decision problems are unsolvable in general, but become solvable in restricted classes of groups. The aim of the study of decision problems for finitely presented groups could be summed up by the following: "to quantify the complexity of classes of finitely presented groups thanks to solvability and unsolvability of different decision problems".

In that respect, the Adian-Rabin Theorem, which implies unsolvability of a wide range of problems in the class of all finitely presented groups, plays an important role in rendering this program possible.

But it is also fundamental for this program to make sense that there exist classes of groups for which decision problems be solvable, and we can quote here the great achievements that are the solutions to the isomorphism problems for polycyclic groups ([**114**]), for hyperbolic groups ([**26**]), etc. Proposition 2.0.1 is a much too powerful equivalent of the Adian-Rabin Theorem, and it kills the study of decision problems for groups described by recursive presentations.

This proposition (and its proof, see Section 2.1) might lead one to think that the problem lies in the fact that, as soon as groups are described by algorithms, some version of Rice's theorem, or some reduction of the halting problem, will be found, that prevents one from deducing anything from those descriptions.

The second result that we present here proves the contrary: our most important theorem gives a characterization of finitely presented groups in terms of solvability of two decision problems. For this purpose, we define what we call a marked quotient algorithm of a group:

Definition 2.0.2. For a group G generated by a finite set S, we call a *marked quotient algorithm* for G and S an algorithm that takes as input a r.e. algorithm $\mathcal{A}_{re}^{H;S'}$ of another group H over a generating family S', together with a function f that is onto from S to S', and stops if and only if f can be extended to a group homomorphism from G onto H.

We denote by $\mathcal{A}_Q^{G,S}$ a marked quotient algorithm for G over S, and omit writing G and S when possible.

Note that there already exist algorithms known as "quotient algorithms", those are algorithms that are able to decide whether or not a given group has a quotient in a certain family of groups. For instance, it was proven in [**19**] that no quotient algorithm exists for the class of all finite groups, but in [**18**] are given several infinite families of finite simple groups for which quotient algorithms do exist.

We can now state our main theorem:

Theorem 2.0.3 (Algorithmic characterization of finitely presented groups). *Any group that admits both a r.e. algorithm and a marked quotient algorithm is finitely presented.*

What's more, from any finite presentation of a group G, a pair of algorithms $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$ for this group can be effectively computed, and, conversely, from any pair $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$, a finite presentation of the group they define can be effectively retrieved.

This theorem is related to Higman's embedding theorem (it is much easier to prove) in the following way: Higman's Theorem shows that, when studying topology, the notion that comes from computability theory of a recursively presented group arises naturally -and even that recursively presented groups could be defined without needing the notions of Turing machines and of computable functions, only in terms of finitely generated subgroups of fundamental groups of closed manifolds. Our theorem does the opposite: it shows that, building a theory of algorithmic descriptions of groups, one could still arrive at the notion of "finitely presented group", solely in terms of algorithms, without even defining group presentations.

The purpose of the work presented in this chapter is to begin the systematical study of decision problems for groups described by algorithms, and that the concept of a "global algorithmic problem", which was intended by Dehn to be:

- the study of the properties that can be inferred on a group, given a finite presentation of it, assuming this given group belongs to some specified class of groups;

become:

- the study of the properties that can be inferred on a group, *given any algorithmic description of it*, assuming this given group belongs to some specified class of groups.

We now precise what we mean by an algorithmic description of a group.

The algorithmic descriptions that are easiest to think of arise from the study of local decision problems for groups: if a group G has solvable word problem, an algorithm $\mathcal{A}_{WP}$ that encodes a solution to the word problem with respect to a generating family of G defines G uniquely (in fact a marking of G, we come back to this fact later on). Similarly, the r.e. algorithm of a group defines it uniquely, and so does an algorithm associated to a solution to the conjugacy problem, etc.

This defines a wide range of algorithmic descriptions of groups, that all have a common problem: when trying to solve decision problems for groups based on these descriptions, the balance between the unsolvable and the solvable is overly in favor of the unsolvable. Witness of this are: the already mentioned Proposition 2.0.1, and the fact that the isomorphism problem is unsolvable for cyclic groups described by word problem algorithms (which also solve the conjugacy problem). This was proven by Jody Lockhart in [**66**], we recall it in Proposition 2.1.15.

What allows us to get out of this predicament is the marked quotient algorithm introduced for Theorem 2.0.3. The marked quotient algorithm of a group G is not attached to a local problem about G, but to the *global* decision problem associated with the group property "being a quotient of G", this problem being asked for groups described by r.e. algorithms.

It is this remark, that the solutions of global algorithmic problems can sometimes be taken as descriptions of groups, that will allow us to find a wide variety of algorithmic group descriptions, rich enough to recover all that can be done thanks to finite presentations, and to open up the possibility of finding descriptions that cannot be expressed purely in terms of presentations.

We can describe a hierarchy of decision problems for groups, associated to a hierarchy of algorithmic descriptions of groups, as follows.

The local decision problems are considered to be the problems of order 0. An algorithm that encodes the solution of a local problem in a group is a description of this group, which is also of order 0.

We can now consider order 1 decision problems, which are global decision problems asked about group descriptions of order 0, and similarly order 1 descriptions of groups, which are the descriptions of groups that arise as solutions of order 1 problems.

We can then define order 2 decision problems, and so on.

As was shown in Theorem 2.0.3, finite presentations of groups are descriptions of order 1. And thus the isomorphism problem for finitely presented groups is a decision problem of order 2. An algorithm that takes as input finite presentations of groups and stops only on those that define a certain group G constitutes a description of order 2 of G. Etc.

A very interesting consequence of the duality we introduce between decision problems and descriptions of groups is that, because, as we already mentioned, decision problems for groups are meant to be studied in their rapport with various classes of groups, algorithmic descriptions of groups attached to restricted classes of groups will naturally appear, as the solutions to these restricted problems.

And here we touch on what is perhaps the point where the results presented in this chapter lack the most: we are unable to produce a class of groups in which the *algorithmic* notion of "finite presentation with respect to this class", that is to say the description of a group G that is constituted of a solution to the problem "being a quotient of G", but *asked only for groups in this class*, together with a r.e. algorithm, does not in fact correspond to an already existing concept, already associated to finite presentations -for instance, finite presentations in varieties of groups (see Theorem 2.4.11).

It seems to us that the possibility of defining an actual algorithmic generalization of finite presentations would justify our going out of the usual frame of "the study of finitely presented groups", and it hinges on the finding of such a class of groups.

We call *relative marked quotient algorithms* the algorithms associated to the problem of recognizing the marked quotients of a group amongst groups that belong to a specified set, and discuss them in more details in Section 2.4.

The other important issue we encounter is that we are not able to give a formal definition for what is "an algorithmic description of a group", and the hierarchy that was described above is an informal object. See the remarks in Section 1.3 about this.

However, even if it were impossible to resolve, this lack of precise definitions is not a fatal issue: our definitions are only as "informal" as those of Miller, whose definition of local problems involved deciding whether elements of a group had "certain properties or relationships". If a definition of what is an "algorithmic description of a group" cannot be found, we are content with studying explicit examples of such descriptions, which abound, and to propose some new ones.

To propose new descriptions of groups, which would hopefully provide interesting theories of decision problems for groups, we rely on Theorem 2.0.3. Recall that it states that a finite presentation of a group can be seen as being a pair:

$$(\mathcal{A}_Q^G, \mathcal{A}_{re}^G)$$

where $\mathcal{A}_{re}^G$ is a r.e. algorithm for G, and $\mathcal{A}_Q^G$ is what we have called a marked quotient algorithm. There are two main directions that this theorem indicates to build interesting algorithmic descriptions of groups. The first one consists in weakening the marked quotient algorithm $\mathcal{A}_Q^G$, and replacing it with a marked quotient algorithm *relative to a class* $\mathcal{C}$. As we have already said, this is discussed in more details in Section 2.4, and we do not know yet whether this will prove fruitful.

The second path designated by Theorem 2.0.3 consists in *strengthening* the right hand side algorithm that constitutes the finite presentation, the r.e. algorithm. This is natural, because this algorithm provides very little information. For instance, semi-computable reals, which, as we already explained, are similar to r.p. groups, appear in other domains of mathematics mostly as encoding unsolvable problems (the halting problem, the computation of the Kolmogorov complexity of strings), and are not expected to be the appropriate basic objects for the defining of a theory of computable functions of the reals.

In that respect, it could be argued that the most natural algorithmic description of a group is not a single finite presentation for this group, but a finite presentation together with a word problem algorithm. In turn, could be studied algorithmic problems asked for groups described by finite presentations and conjugacy problem algorithms, and order problem algorithms, etc.

This approach is very promising: in fact, it was already successfully applied. In [**49**], Daniel Groves and Henry Wilton proved that, given a finite presentation of a group, together with a solution to the word problem for this group, it is possible to decide whether or not it is a limit group (see [**49**] for a definition of limit groups), and whether or not it is a free group. In [**48**], those two authors and Jason Fox Manning have showed that from the same description of a group, it is possible to tell whether or not it is the fundamental group of a geometric three manifold (basing their work on [**75**]). Note that the result of [**49**] that concerns free groups was also obtained in [**123**], where is studied the computability of Grushko decompositions for groups described by finite presentations together with word problem algorithms. We summarize those results in the following theorem:

Theorem 2.0.4 ([**49**],[**48**],[**123**]). *Given a group, described by a finite presentation together with a word problem algorithm, it is possible to decide whether or not this group is free, is a limit group, or is the fundamental group of a geometric three manifold.*

It is not only those results that are interesting, but also the methods of proofs involved. For instance, consider only the statement of this theorem that concerns free groups, it answers what may be the most basic problem about groups described by finite presentations together with word problem algorithms: is it possible to recognize free groups from this description?

The answer to this question given in [**49**] relies on the study of limit groups, and in fact on many elaborate results about limit groups (finite presentability and coherence of limit groups, Makanin's Theorem which establishes the decidability of the universal theory of free groups, etc).

We think it is fair to argue that the very interesting methods involved in the proof of Theorem 2.0.4 justify, *a posteriori*, that the question which is answered in this theorem was an interesting one.

There are some differences between our formalism and the one that is used in [**48**]. It is however not difficult to see that the proofs of [**48**] do imply Theorem 2.0.4. In Section 2.2, we discuss in more details the formalism of [**48**] and explain why we prefer ours, using our analogy with computable analysis: the computability notion of [**48**] resembles that of Banach-Mazur computability, which was supplanted, in computable analysis, by other notions of computability.

The fact that, for the description "finite presentation-word problem algorithm", there exist groups that are recognizable, and some that are not (by [**81**], this is discussed in Section 2.3), shows that the study of group recognition is more interesting when based on this description than on the one associated to finite presentations, since it is well known that the Adian-Rabin theorem implies that the problem "does this presentation define G?" is always unsolvable in the class of all finitely presented groups, even when G is the trivial group.

We talk more about group recognition in Section 2.3.

Overall, it seems that the study of decision problems for groups given by descriptions that are stronger than finite presentations should prove fruitful.

We now formulate our last remark about the question of what a theory of decision problems for groups, separated from topology, could look like: it seems to us that in such a theory, finitely presented groups still have to play a central role, not because they are the objects that such a theory aims at studying, but because they serve as the model of a class of groups, associated to a group description, in which a satisfying theory of decision problems can be built.

Decision problems for marked groups. Another point we want to make in this chapter is the following: the appropriate category to study decision problems for groups is the category of marked groups.

Our first argument in this direction is given to us by Dehn's definition of the isomorphism problem ([**29**]):

> "The Isomorphism Problem: Given two groups, one is to decide whether they are isomorphic or not (and further, whether a given correspondence between the generators of one group and elements of the other group is an isomorphism or not)."

Thus Dehn asked for both a solution to the usual isomorphism problem, *and* a solution to the "marked isomorphism problem". This last aspect seems to have been all but forgotten later on.

Note that all the algorithmic descriptions of groups we have encountered so far provide descriptions of marked groups. Finite and recursive presentations, word problem algorithms, conjugacy problem algorithms, etc, all define marked groups. The marked quotient algorithm which we introduced is also attached to a marked group, even more: it provides information about the location of a given marked group in the lattice of marked groups.

It does not seem natural, working with descriptions that define marked groups, to always discard the additional information provided by the "markings", and only consider problems about the underlying abstract groups.

More importantly, as long as no groups descriptions are used, that are attached to abstract groups and do not define marked groups, the study of decision problems for marked groups strictly contains the study of abstract decision problems. That is because a property of abstract groups is simply a property of marked groups that is *saturated* ([**23**]), that is to say which is invariant under abstract group isomorphism. Thus the study of decision problems for marked groups does not remove anything to the study of decision problems for abstract groups, and on the contrary, one can gain a better understanding of decision problems for abstract groups by having studied more generally decision problems for marked groups.

For instance it can be very helpful to be able to tell when a problem is undecidable, both for marked and abstract groups, or when it is decidable for marked groups, but undecidable for abstract groups, and that the undecidability arises precisely during the transition from the category of marked groups to the category of groups.

The following is an example of this kind.

The Adian-Rabin theorem implies unsolvability of the isomorphism problem for abstract groups, but it is in fact usually proved by relying on a single marked group: a sequence $(\pi_n)_{n\in\mathbb{N}}$ of finite presentations is constructed, such that the set of indices n for which π_n defines the same *marked group* as π_0 is recursively enumerable but not recursive. This can be done because the marked isomorphism problem is unsolvable for finitely presented groups. On the contrary, in Miller's proof from [**81**] (Theorem 26, Chapter IV) of the unsolvability of the isomorphism problem for finitely presented residually finite groups, a fundamental difference appears, because the marked isomorphism problem is solvable for such groups (by [**79**], see Section 2.3). In [**81**] is constructed a sequence $(\pi_n)_{n\in\mathbb{N}}$ of finite presentations of residually finite groups such that the set of indices n for which π_n defines some fixed *abstract group* G is recursively enumerable but not recursive, to achieve this, the presentations that do define G must be attached to infinitely many generating families of G, families whose average word length, with respect to some fixed generating family of G, grows faster than any recursive function.

We give one last example which shows, in our opinion, that the study of decision problems for marked groups was actually never very far. It again relies on the Adian-Rabin theorem.

When stating this theorem, many authors, to define a *Markov property*, state something along the lines of: "a Markov property is a group property which *is invariant under group isomorphism*, and such that...". See Rabin himself in [**103**], and for instance [**112**], [**6**], [**12**] and [**83**].

For instance, in Miller's article which we have already cited ([**83**]), one can find the following paragraph:

"Consider the problem of recognizing whether a finitely presented group has a certain property of interest. For example, can one determine from a presentation whether a group is finite? or abelian? It is natural to require that the property to be recognized is abstract in the sense that whether a group G enjoys the property is independent of the presentation of G."

Now notice that it does not make sense to require that *a group property* be invariant under isomorphism, by *definition* a group property is invariant under group isomorphism. And thus to make sense of what Miller calls a "property of finitely presented groups", we might have to consider that it is actually *a property of finite presentations*.

But it seems clear that properties of finite presentations that are completely detached from the groups they define are not really worth mentioning. Properties of marked groups are precisely the right intermediate between properties of finite presentations and properties of groups which allow to make sense of the requirement that a Markov property be a property of abstract groups: a property of abstract groups, as opposed to a property of marked groups.

We point out the following consequence of our introducing decision problems for marked groups: while the Adian-Rabin Theorem completely solves the problem of abstract group recognition for finitely presented groups (in the class of all finitely presented groups), and while the proof of this theorem can be adapted to obtain results about the recognition of marked groups, there is a gap in the results thus obtained about marked groups, which seemingly would require different techniques to be solved. We talk about this in Section 2.3.

By now, most of the content of this chapter was already presented. We end this introduction by giving summaries of the contents of our different sections.

In Section 2.1, we prove Proposition 2.0.1 as well as Theorem 2.0.3. We also propose three more undecidability results that are inspired by Proposition 2.0.1: one for groups described by what is known as co-recursive presentations, one for groups described by marked quotient algorithms, and finally, the already mentioned result of Lockhart in [**66**] about groups given by word problem algorithms.

In Section 2.2, we provide some additional discussion about possible critics to our proposed approach to decision problems for groups: the opportunity of considering decision problems that take "well chosen" recursive presentations as input, the defining of problems that are not defined on recursively enumerable sets of groups. We also discuss the possibility of establishing equivalents of the Kreisel-Lacombe-Schoenfield-Ceitin Theorem for algorithms that describe groups, and finally we talk about the notion of "recursive modulo the word problem" that comes from [**48**].

In Section 2.3, we study the consequences of our proposed approach to decision problems for groups, in the particular case of group recognizability: the problem of recognizing a fixed group. This is a natural step towards understanding the isomorphism problem. We explain there the advantage of finite presentations over other groups descriptions in terms of "uniform semi-recognizability".

We propose several problems there as well.

Our final section, Section 2.4, is dedicated to establishing some basic results about relative marked quotient algorithms. We remark that Theorem 2.0.3 can be applied in group varieties, define what we call "elementary marked quotients algorithms", (those are the relative marked quotients algorithms that could be defined thanks to finite presentations), and give a simple example of a non-elementary marked quotient algorithm: the finite quotient algorithm of the lamplighter group.

2.1. Recursive and finite presentations

2.1.1. Solving decision problems from recursive presentations. As opposed to when groups are described by finite presentations, in which case although the isomorphism problem is unsolvable in general, it can become solvable if we only consider groups that satisfy some constraints (either algebraic, or geometric, etc), if we wish to solve the isomorphism problem in some classes of groups given by arbitrary recursive presentations, the problem can almost never be solvable, even with dire restrictions on the considered groups.

Proposition 2.1.1. *There exists a recursively enumerable sequence of recursive presentations, defined on one generator, that define either the trivial group, or the group of order two, and such that the set of indices n for which π_n defines the trivial group is recursively enumerable but not recursive.*

Proof. Consider an effective enumeration of all Turing machines, M_0, M_1, M_2, M_3,... We define for each natural number n a presentation π_n. The presentation π_n has a single generator a, and starts with a single relation a^2. An effective enumeration of the relations of π_n is defined thanks to a run of the machine M_n. While this run lasts, no relators are added to π_n. If it stops, the relation $a = 1$ is added. Thus if M_n stops, π_n is of the form:

$$\pi_n = \langle a|a^2, a\rangle$$

If M_n runs forever, π_n is the finite presentation:

$$\pi_n = \langle a|a^2\rangle$$

(Note that thanks to this definition, the relations of the presentations π_n form a recursively enumerable set, and this uniformly in n, but not a uniformly recursive set, as we cannot know beforehand whether a is a relation. Replacing a by a^{2k+1} in π_n, where k is the number of steps needed for the machine M_n to stop, makes these uniformly recursive presentations.)

This is clearly a recursively enumerable set of presentations: they were defined thanks to a procedure that can be carried out effectively.

And of course, no algorithm can decide, given some n, whether π_n defines the trivial group, since it defines the trivial group if and only if the n-th Turing machine stops. □

Remark that the presentations written above are both finite... However, they are not given by the finite number of symbols that constitute the presentations, but by Turing machines, which, when run, will output the finitely many relations of the presentations. Thus there is no contradiction between our proof and the fact that the isomorphism problem is solvable for finite groups given by finite presentations. We discuss this in more details in Section 2.2, where we try to justify that this proposition was not obtained thanks to our using ambiguous definitions, but that on the contrary it indeed shows that the use of recursive presentations is impractical.

The previous proposition can be applied to both the study of abstract and of marked groups, because all the descriptions used define the same markings of the considered groups.

It is easy to see that the construction that appears in the proof of Proposition 2.1.1 can be applied, not only to the trivial group and the order two group, but to *any pair of recursively presented groups where one is a quotient of the other.* This yields the following proposition, which is almost the same as Proposition 2.0.1, which appeared in the introduction, but we precise it so that it be applicable to the study of marked groups as well as that of abstract groups.

Proposition 2.1.2. *Let G and H be finitely generated, recursively presented groups, and suppose that H is a strict quotient of G. Fix a marking of G, and consider the induced marking on H. Then, the problem of deciding whether a given pair of recursive presentations, that define either the chosen marking of G, or that of H, define the same group, is unsolvable.*

Proof. It is easy to adapt the proof of Proposition 2.1.1: to define π_n, enumerate the relations of G as long as the Turing machine M_n runs, if it stops, add the relations of H. □

It is not the case that any two groups are undistinguishable with respect to the recursive presentation description. It is easy to see that from a recursive presentation over one generator a, that defines either the order two group, or the order three groups, one can determine which one it is, because enumerating the consequences of the relations, one of a^3 or a^2 will eventually appear. More generally, it was shown by Kuznetsov ([**64**]) that the word problem is uniformly solvable for simple groups described by recursive presentations. And it is easy to see that the isomorphism problem is solvable for any finite set of finitely presented simple groups *described by recursive presentations,* but of course the class of simple groups does not contain a group and a strict quotient of it.

As a direct consequence of this last proposition, we get the following corollary, which may be seen as an equivalent of the well known Rice theorem, from computability theory:

Corollary 2.1.3. *No non-trivial group property can be determined for groups given by recursive presentations.*

Proof. It suffices to notice that for any non-trivial group property, (that is a property some group enjoy, while others lack), there must exist a pair of groups G and H, such that H is a quotient of G, and one satisfies the property, while the other does not. □

Corollary 2.1.4. *Any abstract or marked group property which is semi-decidable, in a class $\mathcal{C}$ of groups, for groups given by recursive presentations, is quotient-stable.*

Proof. This follows from the fact that, for the sequence of presentations constructed in the proof of Proposition 2.1.2, there is an algorithm that stops only on recursive presentations of the quotient group H. Thus there cannot be an algorithm that stops only on the recursive presentations of the group G. □

Note that group properties which are semi-decidable from recursive presentations do exist: being trivial, being finite, or, a non trivial example, having Kazhdan's property (T) ([**99**]).

We add here a last lemma that will allow us to prove an equivalent of the Rice-Shapiro Theorem for recursive presentations.

Lemma 2.1.5. *Let $\pi = \langle S | r_i,\, i \in \mathbb{N}\rangle$ be a recursive presentation, and denote by π_n the truncated presentation $\langle S | r_0, r_1, r_2, ..., r_n\rangle$. No algorithm that takes as input presentations in the set $\{\pi\} \cup \{\pi_n,\, n \in \mathbb{N}\}$ can stop exactly on the presentation π.*

Proof. Consider an effective enumeration of all Turing machines, M_0, M_1, M_2, M_3,... We define for each natural number k a presentation τ_k, defined thanks to a run of the k-th Turing machine. While this run last, add to τ_k the relations of π. If this run stops in t steps, stop adding any relations. Thus if M_k does not halt, τ_k is exactly π. If M_k halts in t steps, then only t relations were added to τ_k, which is thus equal to π_t.

It is then clear that no algorithm can recognize the presentations that define π, since such an algorithm would allow one to solve the halting problem. □

The following is an equivalent of the Rice-Shapiro Theorem, which characterizes the properties of r.e. sets that are semi-decidable.

THEOREM 2.1.6 (Rice-Shapiro for recursive presentations). *If P is property of marked groups that is semi-decidable from the recursive presentation description, then there exists a recursively enumerable sequence of finite presentations, such that a group satisfies P if and only if it is a marked quotient of a group defined by one of those presentations.*

PROOF. Let $\mathcal{A}_P$ be an algorithm that stops on recursive presentations for groups in P. Note that a finite presentation is a recursive presentation, and thus $\mathcal{A}_P$ can be applied to finite presentations. The set of all finite presentations is recursively enumerable, and thus the set of all finite presentations accepted by $\mathcal{A}_P$ is also recursively enumerable. Denote this set Π.

All that is left to show is that a group satisfies P if and only if it is a quotient of a marked group defined by a presentation in Π. Corollary 2.1.4 shows that being a quotient of a group in Π is a sufficient condition to satisfy P. Lemma 2.1.5 shows that it is also necessary, since it is easily seen to imply that a recursively presented group that satisfies P is always the quotient of a finitely presented group that satisfies P. □

The semi-decidable sets described in this theorem are necessarily open in the space of marked groups. The results of Chapter 5 show that obtaining a characterization of the sets that are semi-decidable for the word problem algorithm description is still a far off prospect.

2.1.2. The marked quotient algorithm and finitely presented groups. Recall that the notation $\mathcal{A}_Q^{G;S}$ designates a marked quotient algorithm for the group G over the generating set S, as defined in the introduction of this chapter.

Although we stated in Definition 2.0.2 that such an algorithm takes as input both a r.e. algorithm $\mathcal{A}_Q^{H;S'}$ for a groups H, and a function f from the generating family S of G onto that of H, we will in fact often suppose that both groups G and H are generated by S, this allows us to omit the function f and to consider that the input of a marked quotient algorithm is a single r.e. algorithm.

Notice first that any finitely presented group admits a marked quotient algorithm. Indeed, to know whether a group H is a marked quotient of a finitely presented group G, one only needs to check whether the finitely many relations of G hold in H. Given a finite set R of relations that define G over the generating set S, and the r.e. algorithm $A_{re}^{H;S}$, the marked quotient algorithm of G computes the boolean operation:

$$\mathop{\&}_{r\in R} A_{re}^{H;S}(r)$$

Notice however that finitely presented groups are not the only groups that admit marked quotient algorithms: a non-recursively presented simple group also has one -its only recursively presented quotient is the trivial group. However, Theorem 2.0.3 states that they are the only ones amongst recursively presented groups.

THEOREM (Algorithmic characterization of finitely presented groups; Theorem 2.0.3). *Any group that admits both a r.e. algorithm and a marked quotient algorithm is finitely presented.*

What's more, from any finite presentation of a marked group G, a pair of algorithms $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$ for it can be effectively computed, and, conversely, from any pair $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$, a finite presentation of the marked group they define can be effectively retrieved.

PROOF. The fact that a pair $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$ can be effectively obtained from a finite presentation of G was already explained: $\mathcal{A}_{re}^G$ lists the consequences of the relations of G, and $\mathcal{A}_Q^G$ tests whether the relations of G hold in an input group.

Consider a marked group G which admits a description $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$. Consider an enumeration of all Turing Machines M_0, M_1... For each natural number n, we define an algorithm $\mathcal{A}_{re}^n$. It is defined as enumerating the relations of a group G_n, rather than as recognizing relations. $\mathcal{A}_{re}^n$ does two actions in parallel: it simulates the machine M_n, all the while listing relations of G thanks to $\mathcal{A}_{re}^G$. If M_n stops, $\mathcal{A}_{re}^n$ stops using $\mathcal{A}_{re}^G$, and proceeds to listing all the consequences of the finitely many relations it has already obtained. On the contrary, if M_n does not stop, $\mathcal{A}_{re}^n$ outputs exactly the same list as $\mathcal{A}_{re}^G$.

Notice now that $\mathcal{A}_Q^G$ will accept every algorithm $\mathcal{A}_{re}^n$ corresponding to a non-halting Turing Machine, since it must accept the identity $\mathrm{id} : G \to G$.

But of course, it cannot accept only those, since it would otherwise allow one to solve the halting problem. Thus it will accept some $\mathcal{A}_{re}^n$ for a Machine M_n that stops in a finite amount of time. This implies that the relations of G are in fact all consequences of the first few relations that $\mathcal{A}_{re}^G$ is able to produce during a run of this machine M_n, and so that G is finitely presented.

Because the indices of halting Turing Machines can be listed, as well as the indices that define quotients of G, some integer n_0 that satisfies both those conditions can be effectively found, and from it, also a finite set of relations that define G over S.

Now notice that the construction described above is uniform, and allows one to obtain a procedure that takes as input a pair $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$ and outputs a finite presentation for G. $\square$

This theorem precisely means that every global algorithmic problem that was solved for finitely presented groups could have been solved with the two algorithms described in the theorem as initial data, and thus no intrinsic unsolvability lies in the use of algorithms to describe groups.

We now prove the very important (and well known) fact, that from any finite presentation of a group, any other presentation of the same group can be recognized, as is usually done by listing Tietze transformations ([**122**]), but using the point of view given by Theorem 2.0.3.

Proposition 2.1.7. *There is an algorithm that takes as input pairs of finite presentations, and stops exactly on pairs of presentations that define the same (abstract) group.*

Proof. From the point of view of algorithmic descriptions, only the Tietze transformations that introduce and delete generators are needed -those that change the marking of the considered group. Indeed, testing whether two finite presentations $(\mathcal{A}_{re}^G, \mathcal{A}_Q^G)$ and $(\mathcal{A}_{re}^H, \mathcal{A}_Q^H)$ over the same generating set S define the same marked group is done directly by the computation of $A_Q^G(A_{re}^H) \& A_Q^H(A_{re}^G)$.

All generating families of a group can be enumerated using only the r.e. algorithm, this is easy to see: choose arbitrary words in the generators, and then blindly search for an expression of the original generators in terms of products of those random words. This procedure terminates on all generating families of the group. When a finite set of elements is found to be a generating family, it means that one has access to expressions both of the new generators expressed as products of the old ones, and of the old generators expressed as products of the new ones. By the following lemma, Lemma 2.1.8, we can obtain a marked quotient algorithm with respect to any generating family we find, and thus recognize any finite presentation of a given group. $\square$

Lemma 2.1.8. *A finitely generated group which admits a marked quotient algorithm with respect to a marking admits one for all of its markings. Such algorithms can be obtained one from the other if one has access to expressions that describe the elements of each generating family in terms of products of elements of the other family.*

Proof. Let S and T be two finite generating sets of a group G (not necessarily of the same cardinality). We suppose that we have access only to A_Q^S, the marked quotient algorithm for G with respect to S. Fix for each s in S an expression $s = t_1^{\alpha_1}...t_k^{\alpha_k}$, with $\alpha_i \in \{-1, 1\}$ and $t_i \in T$, that gives s as a product of elements of T and of their inverses, and for each t in T an expression $t = s_1^{\beta_1} s_2^{\beta_2}...s_k^{\beta_k}$ that describes t in terms of the generators of S and their inverses.

Consider a group H generated by the same family T, given by its r.e. algorithm $A_{re}^{H;T}$, we want to determine whether (H, T) defines a marked quotient of (G, T).

Notice that if the identity on T defines a morphism ϕ of G onto H, the family S', defined, in H, by the same formulas as S in G, i.e. $s' = t_1^{\alpha_1}...t_k^{\alpha_k}$, should be the image of the family S by ϕ, and thus should be a generating family of H. We can therefore, using the algorithm $A_{re}^{H;T}$, look for an expression of the elements of T in terms of the elements of S' in H. If such an expression does not exist, our procedure will not stop, but H is not a quotient of G, thus this result is coherent. Otherwise we can use the formulas just found to obtain the r.e. algorithm for H with respect to S', $A_{re}^{H;S'}$. From it, we can ask whether the natural bijection $S \to S'$ defines a morphism, applying $A_Q^S(A_{re}^{H;S'})$. If this procedure does not end, H was not a quotient of G. If it terminates, in which case $S \to S'$ does define a morphism ψ, we still have to check whether ψ defines the identity on T. This is done using the expressions of the form $t = s_1^{\beta_1} s_2^{\beta_2}...s_k^{\beta_k}$, that define, in G, the elements of T in term of those of S, the corresponding expressions in H of elements of T in terms of elements of S', and the algorithm $A_{re}^{H;S'}$. One can check that the given conditions are necessary and sufficient for the identity of T to extend as a homomorphism. $\square$

Other examples in the same vein as this one can be found, where we prove results that are usually explained thanks to the manipulation of finite presentations, using only algorithms -without using Theorem 2.0.3. Easy examples are: computing the presentation of a free product, or the abelianization of a group. On the contrary we do not know how to obtain an enumeration of all finitely presented groups, described by pairs (A_{re}, A_Q), as is obtained by listing all possible finite presentations, without precisely listing those presentations and applying Theorem 2.0.3.

The manipulation of the algorithms that constitute a finite presentation, while possible, is often less convenient than the manipulation of actual presentations. For instance, we do not believe that the algorithmic description of finite presentations can ever be used to prove that a group is or is not finitely presented.

REMARK 2.1.9. The marked quotient algorithm, as defined, halts only on accepted inputs. Are there marked groups that could have an always halting marked quotient algorithm, which would answer "yes" or "no" depending on whether the input is a quotient of the considered group? Free groups, of course, can: the marked quotient algorithm for a free group over a basis accepts all inputs. This is, however, the only possible example, because if a non-free finitely presented group G admitted a always halting marked quotient algorithm with respect to a generating family S, the isomorphism problem would be solvable for the pair constituted of G and of the free group on S, *when described by recursive presentations.* By Proposition 2.1.2, this is impossible.

On the other hand, it is easy to remark that, given a finitely presented group G, one can obtain a modified marked quotient algorithm for G that always halts, by accepting groups described, not by their r.e. algorithm, but by their *word problem algorithm.* Instead of having to wait maybe infinitely long before knowing whether the finitely many relations of G hold in a group H, an answer will be found in a finite amount of time. This new type of algorithm is what we call a *marked quotient algorithm with Word Problem as Input*, which we will abbreviate as WPI marked quotient algorithm. Note, however, that such quotient algorithms do not characterize finitely presented groups amongst r.p. groups anymore. Indeed, Miller constructed (this construction appears in Section 5.3.3 of Chapter 5) a finitely presented group which has unsolvable word problem, such that no non-trivial quotient of it has solvable word problem. One can obtain such a group that is r.p., but not finitely presented, using exactly the same construction... except for Higman's embedding theorem. This group admits a marked quotient algorithm that takes as input word problem algorithms and answers whether or not the corresponding group is a quotient: this algorithm always answers no, except on the trivial group.

All this sums up why Theorem 2.0.3 uses the marked quotient algorithm which, by nature, does not always stop.

PROBLEM 2.1.10. Characterize the groups with solvable word problem that admit an WPI marked quotient algorithm: are these only the finitely presented groups with solvable word problem?

More variations of the quotient algorithm are studied in Section 2.4.

2.1.3. Three more unsolvability results.

2.1.3.1. *From the co-r.e. algorithm.* The interest in finitely presented groups, together with Higman's Theorem, makes being recursively presented a much more natural notion to introduce than the notion known as being co-recursively presented ([**74**]), which, contrary to what the names indicates, is not defined thanks to a presentation: a marked group is co-r.p. if there is an algorithm that recognizes the words in the generators of this group that define non-identity elements, and because this property of marked groups is saturated, it defines a group property. We call such algorithms co-r.e. algorithms, for *co-recursive enumeration algorithm,* and denote by $\mathcal{A}_{co-re}^{G,S}$ a co-r.e. algorithm for the group G generated by a family S.

From the algorithmic point of view, the notions of recursive and co-recursive presentations are more or less symmetrical. For instance, using our analogy with computable analysis, note that if r.p. groups correspond to lower semi-computable reals, co-r.p. groups will then correspond to upper semi-computable reals. (There is however nothing that would correspond to the bijection $x \mapsto -x$ that sends lower to upper semi-computable reals.)

This remark calls for the following proposition:

PROPOSITION 2.1.11. *Let G and H be finitely generated, co-recursively presented groups, and suppose that H is a strict quotient of G. Fix a marking of G, and consider the induced marking on H. Then, the problem of deciding whether a given pair of co-r.e. algorithm, that define either the chosen marking of G, or that of H, define the same group, is unsolvable.*

PROOF. As for Proposition 2.1.2, we prove this result for the trivial group and the order two group, and note that the same construction can be applied to an arbitrary pair.

To the n-th Turing machine M_n, associate the pair composed of one generator symbol a, and of an algorithm $\mathcal{A}_{co-re}^n$ defined as follow: to determine whether a word a^k, with k an integer, is a non-identity element, $\mathcal{A}_{co-re}^n$ starts by simulating the machine M_n. If it stops, $\mathcal{A}_{co-re}^n$ determines whether k is even or odd, and if k is odd, it indicates that a^k corresponds to a non-identity element.

If M_n does not stop, the algorithm $\mathcal{A}^n_{co-re}$ never recognizes any element -it is just an algorithm that runs forever, it thus defines the trivial group. On the other hand, if M_n stops, $\mathcal{A}^n_{co-re}$ recognizes exactly the non-identity elements of the group of order two. □

As before, this result implies a Rice Theorem for groups given by algorithms that recognize non-trivial elements:

Corollary 2.1.12. *No non-trivial group property can be determined for groups given by their co-r.e. algorithm.*

Corollary 2.1.13. *If P is a marked or abstract group property which is partially decidable for groups given by their co-r.e. algorithm, in a class $\mathcal{C}$ of groups, any extension of a group with P also has P.*

An example of such a property is: being non-trivial. We leave it to the reader to state and prove an equivalent of the Rice-Shapiro Theorem for co-recursively presented groups.

2.1.3.2. *From the marked quotient algorithm.* We will now prove another unsolvability result that is intended to show that, in Theorem 2.0.3, it is necessary to include both the marked quotient algorithm and the r.e. algorithm. Notice that the marked quotient algorithm of a finitely presented group defines it uniquely: if G and H have the same marked quotient algorithm A_Q and are r.p., it must be that $A_Q(A^G_{re})$ and $A_Q(A^H_{re})$ both are accepted, thus G and H are isomorphic (even as marked groups). However, this algorithm gives very little information:

Proposition 2.1.14. *No algorithm can solve the isomorphism problem for the trivial group and the order two group when they are described by their marked quotient algorithms.*

Proof. Again, to the n-th Turing machine M_n, associate the pair composed of one generator symbol a, and of an algorithm $\mathcal{A}^n_Q$ defined as follow: to determine whether an algorithm $\mathcal{A}^H_{re}$ defines a quotient of our group, try to determine if H is trivial, by waiting to see whether $\mathcal{A}^H_{re}$ accepts a. Of course, if at some point it is proven that H is the trivial group, stop and accept. All the while, start a run of the machine M_n. If it stops, accept also the order two group as a quotient: try to see whether $\mathcal{A}^H_{re}$ recognizes a^2 as a relation, and accept if it does.

It is clear that if M_n stops, this defines the order two group, while it defines the trivial group otherwise. □

Notice how we could not have made simultaneously this construction and that of Proposition 2.1.1: here the non-stopping Turing machines *must* correspond to the trivial group, whereas in Proposition 2.1.1, they had to correspond to the order two group. This proposition could be extended to any pair of groups as before.

2.1.3.3. *From the word problem algorithm .* The last result on this list concerns the word problem. We include it here because it is relevant in showing that the solutions to local algorithmic problems provide weak algorithmic descriptions.

Proposition 2.1.15. *No algorithm can solve the isomorphism problem for cyclic groups when they are described by their solution to the word-problem.*

Proof. Consider an enumeration of all Turing Machines M_0, M_1,... For each natural number, define an algorithm A^n_{WP} that works with a single generating symbol a as follows. Consider as input an element a^k, $k \in \mathbb{Z}$. If $k = 0$, then of course a^k is accepted as trivial. Otherwise A^n_{WP} simulates a $|k|$ steps run of M_n. If it does not stop during this run, then A^n_{WP} answers that a^k is a non-identity element. If on the contrary M_n stops in less than $|k|$ steps, say in p steps, we decide that A^n_{WP} should be the word problem algorithm for the group $\mathbb{Z}/p\mathbb{Z}$, and thus that it should answer that a^k is the identity if and only if p divides k.

It is easy to see that this definition is coherent, that is to say that A^n_{WP} will always answer as the word problem algorithm of the same group, whatever the input.

And A^n_{WP} defines $\mathbb{Z}$ if and only if M_n does not stop. □

The proof above is very similar to a proof of Lockhart in **[66]**. However, in Lockhart's article, the statement which is proved, thanks to this proof, is the following: "There is a r.e. class of recursive presentations with uniformly solvable word problem for which the properties of freeness and finiteness are unrecognizable". We will discuss the ambiguity of this statement, which is similar to that of Proposition 2.1.1, in SubSection 2.2.1.

Chapter 5 is dedicated to the study of decision problems for groups described by word problem algorithms. We show there that this corresponds to the study of computable analysis on the space of marked groups. The result of Proposition 2.1.15 can be seen as an application of a general statement due to Markov Jr, when applied to a converging sequence of cyclic groups.

2.2. Additional discussion

The use of algorithms to describe countable groups is just one amongst many means to obtain finite definitions of potentially infinite groups. We can quote here several approaches to describing groups that are commonly used:

- Finite presentations;
- Finite presentations in group varieties ([**61**]);
- Automaton to describe automatic groups ([**36**]);
- Automaton to describe groups of automaton ([**131**]);
- Finite sets of axioms to describe finitely axiomatizable groups ([**94**]);
- Finitely generated linear groups described by generating matrices ([**31**]);
- L-presentations ([**3**]);
- Etc.

Note that algorithmic descriptions of groups were already studied in [**11**], for groups that are not finitely generated. The results of this article justify our focus on finitely generated groups, by showing that almost no global problem is decidable for infinitely generated groups described by algorithms. Note that, in [**11**], some interesting problems are still raised in the setting of infinitely generated groups, by going past the problem of solvability of different decision problems, and asking more precisely for their location in the Kleene–Mostowski hierarchy.

In this section, we give some precisions about the point of view taken in the present article, and answer possible critics to it.

2.2.1. Recursive presentations.

2.2.1.1. *Ambiguity of the concept of recursive presentations.* We stated in the introduction that we consider that Proposition 2.0.1 prevents one from basing a theory of decision problems on recursive presentations, and we in fact choose to talk about r.e. algorithms rather than of recursive presentations, even though recursively presented groups are exactly those that admit a r.e. algorithm.

We now explain in more details those statements.

The first important point is the ambiguity that lies in the term "recursive presentation"[1], which appears clearly in the proof of Proposition 2.1.1, in which we prove that the trivial group and the order two group cannot be distinguished when described by recursive presentations. This proof relies on only two actual presentations, which are $\langle a|a^2, a\rangle$ and $\langle a|a^2\rangle$.

The ambiguity lies in the following: by definition, a recursive presentation is a presentation $\pi = \langle S|R\rangle$, where R is a recursively enumerable set of words. On the other hand, when we say that a group is "given by a recursive presentation", as input of an algorithm, we mean to say that is given a pair $(S, \mathcal{A}_{rel})$, where S is a set of generators symbols, and $\mathcal{A}_{rel}$ is an algorithm that enumerates the relations of the underlying presentation.

Thus in order to use a recursive presentation to obtain an algorithmic description of a group, a choice has to be made, we precisely take the step that goes from the abstract "there exists an algorithm that outputs the relations of π" to the statement: "$\mathcal{A}_{rel}$ is an algorithm that outputs the relations of π".

We can see that the proof of Proposition 2.1.1 precisely relied on making inefficient choices of algorithms that would output the relations of either of the presentations $\langle a|a^2, a\rangle$ and $\langle a|a^2\rangle$: instead of using the obvious algorithms associated to those presentations ("output a^2, then a, then stop", or "output a^2 and stop"), we gave infinitely many possible algorithms which defined $\langle a|a^2, a\rangle$ and $\langle a|a^2\rangle$, most of which were very inefficient: the algorithms that defined $\langle a|a^2, a\rangle$ produced first the relation $a^2 = 1$, then waited for a possibly very long time to output the relation $a = 1$, while the algorithms for $\langle a|a^2\rangle$ produced right away the relation $a^2 = 1$, but instead of stopping, continued running for ever.

The proof of Proposition 2.1.1 might seem to be "artificial", or to rely on an error of definition that could be easily avoided, precisely because it relies on poor choices of algorithms. But remark that those choices can be deemed as "poor" only because there are, in the case of the presentations $\langle a|a^2, a\rangle$ and $\langle a|a^2\rangle$, obvious better choices of algorithms that could be made, because those presentations are finite.

But in general, under the sole assumption that π is a recursive presentation, no canonical choice of an algorithm that outputs its relations can be made, whether it be regarding the order in which the relations should be produced, or regarding the time it takes for the algorithm to produce a new relation, etc. And thus there is no hope of excluding the "bad" algorithms that were used in the proof of Proposition 2.1.2 if we are to deal with generic recursively presented groups.

[1]We will not mention the fact that a proper name should be "recursively enumerable presentation". The well known trick that allows to replace a recursively enumerable set of relators by a recursive one is not harmless, for instance it requires the use a different marking of the considered group, or, worse, in [**46**] is defined a Dehn function for infinitely presented groups, which is not a group invariant anymore, and which can be used only in groups whose relations form an actually recursive set, and changing a r.e. set of relations into a recursive set of relations will change the behavior of this function...

Talking about the r.e. algorithm of a group allows us precisely to render explicit the choice of an algorithm that needs to be made, all the while making the only canonical choice that can be made: that the set of enumerated relations be the maximal one.

2.2.1.2. *Using restricted sets of recursive presentations algorithms.* After reading the previous paragraph, one might still be tempted to base a theory of decision problems for groups on recursive presentations, making use of the above described choice of an algorithm $\mathcal{A}_{rel}$ that gives the relations of a recursive presentation π: while, in general, no a priori properties can be expected of the algorithm $\mathcal{A}_{rel}$, it might be possible to make different assumptions on $\mathcal{A}_{rel}$ when working in different classes of groups. The most obvious example of this is that of finitely presented groups: in case π is a finite presentation, we can ask of $\mathcal{A}_{rel}$ that it halt after a finite computation. And we could expect that in different classes of recursively presented groups, some assumptions could be made about the algorithms $\mathcal{A}_{rel}$, in accordance to the nature of the groups in questions, and that this would prevent one from obtaining too strong undecidability results, such as those of Proposition 2.0.1.

We now explain why we do not recommend this approach.

Our argument is based on the following example. We already mentioned (in Proposition 2.1.15) the result of Lockhart from [**66**]: “There is a r.e. class of recursive presentations with uniformly solvable word problem for which the properties of freeness and finiteness are unrecognizable”. The groups in questions are finite and infinite cyclic groups.

Remark here that Proposition 2.1.2 shows that when a class of recursively presented groups contains a group and a strict quotient of it, then no sequence of “actual” recursive presentation of those groups can have uniformly solvable word problem. This is because the marked isomorphism problem is unsolvable for a pair composed of a group and of a quotient of it, when groups are “given by recursive presentations”, while it is solvable when those groups are described by word problem algorithms. And indeed what is actually proven in [**66**] is not a result that concerns a “r.e. class of recursive presentations”, but a r.e. sequence of *algorithms*, the choice of a way of enumerating the relations for the described presentations was explicitly made. This remark shows in particular that the ambiguity that lies in the use of the terms “groups given by recursive presentations” had already appeared in the literature before, and that we did not introduce it in order to obtain Proposition 2.0.1.

With this example in mind, consider the following: in any class $\mathcal{C}$ of groups with solvable word problem, if we allow ourselves to make choices about the algorithms that produce recursive presentations, then finding a sequence of algorithms that output recursive presentations which has “uniformly solvable word problem” is a trivial matter, as we can choose algorithms that output all the relations for the groups they define, increasingly with respect to some computable order.

Such algorithms exist because the considered groups have solvable word problem... but they are actually not different at all from word problem algorithms. In this context, the sentence “a set of recursive presentations with uniformly solvable word problem” is not different from the sentence “a set of word problem algorithms”, and it is clear that the latter should be preferred.

In general, it seems to us that one obtains more precise and more explicit conditions on algorithmic descriptions of groups by using algorithms that perform advanced tasks, than by considering recursive presentations described by algorithms that are assumed to satisfy additional properties.

2.2.2. The Kreisel-Lacombe-Schoenfield-Ceitin Theorem. Our analogy with computable analysis suggests to us the need to establish results that correspond to the Kreisel-Lacombe-Schoenfield-Ceitin Theorem, which is one of the most important theorems of computable analysis.

This theorem is stated precisely in its most general form in Chapter 5, see Section 5.2.6.

We give here a simple statement that follows from the Kreisel-Lacombe-Schoenfield-Ceitin Theorem applied to the real numbers.

THEOREM 2.2.1 (Kreisel-Lacombe-Schoenfield-Ceitin Theorem on $\mathbb{R}$). *One does not change the set of “computable functions defined on the computable reals” if one considers either of the following definitions:*

- *a function f is computable if there is a Turing machine that takes as input another Turing machine that, on input n, produces a rational approximation of a real x with error at most 2^{-n}, and produces as output a Turing machine which similarly gives approximations of $f(x)$;*
- *a function f is computable if there is a Turing Machine that, given access to an infinite tape (* an oracle*), on the n-th cell of which is written an approximation within 2^{-n} of a real x, will write on an output tape a sequence of approximations of $f(x)$.*

This theorem precisely shows that when we define a computable function of a computable variable described by a Turing machine, no information will be read off of the machine given as input, this means that a computable function of a computable real cannot be defined by defining a function that manipulates Turing machines independently of the underlying real numbers that are encoded by those machines.

To each algorithmic description of groups, we can associate an equivalent of the Kreisel-Lacombe-Schoenfield-Ceitin Theorem, which would state that what can be computed from Turing machines that encode the groups could also be computed with descriptions of groups given by oracles. And of course, such theorems should be relativized to different classes of groups. We do not expect that obtaining such theorems will be an easy task, as the Kreisel-Lacombe-Schoenfield-Ceitin Theorem itself is non trivial, and new difficulties seem to arise when trying to establish it for different algorithmic descriptions of groups.

In particular, in Chapter 5, we show how all the techniques currently available to prove equivalents of the Kreisel-Lacombe-Schoenfield-Ceitin Theorem fail in the case of the space of marked groups, and thus we are unable to prove that what can be deduced about a group from a word problem algorithm for this group is exactly what can be deduced given an oracle that solves the word problem.

We thus state the following problem:

Problem 2.2.2. Establish equivalents of the Kreisel-Lacombe-Schoenfield-Ceitin Theorem for different types of algorithmic descriptions of groups, such as recursive presentations, word problem algorithms, finite presentations together with word problem algorithms, etc. (Or prove that those theorems fail.)

2.2.3. Practical implementation. We stated in the introduction that the purpose of the study of decision problems for groups was to quantify the complexity of different classes of groups. This of course left out an obvious application of the study of decision problems for groups: actual implementation of the solutions to some decision problems.

We want to point out here that describing groups thanks to algorithmic descriptions does not hinder the practical implementation of solutions to decision problems.

A possible critic is that an algorithm which takes as input groups described by descriptions that are stronger than mere finite presentations will not be useful in practice without a way of effectively producing those stronger descriptions, when given only finite presentations. For instance, one might think that a result obtained for groups given by finite presentations together with word problem algorithms will not be useful in practice unless we are set in a class of groups with uniformly solvable word problem, but in this last case the study of groups given by descriptions that consist in finite presentations together with word problem algorithms does not add anything to the study of groups given by finite presentations.

This reproach is related to the central role that finite presentations play in the study of decision problems for groups, but it is not fundamental.

Of course, if some decision problem is implemented at the expressed purpose of launching a blind search for groups with certain properties, using random finite presentations, then surely the implementing of algorithms defined on non-recursively enumerable sets will be a problem. See 2.2.4 for more about this.

On the other hand, if an algorithm is meant to be used on known groups, to answer questions that might still be open about those groups (as in **[92]**, where an algorithm that detects Property (T) is used), then the use of algorithmic descriptions that differ from finite presentations is not a problem.

The step that goes from an abstract isomorphism class of groups to a first effective description of it cannot be deemed "computable" or not. Those descriptions appear through the work of mathematicians confronted to groups that stem from different areas of mathematics. In topology, the description that is most easily obtainable of the fundamental group of some manifold is often a finite presentation. But this is not always the case. The Thompson group F can be seen as a group of piecewise affine homeomorphisms, whose composition can readily be computed, thus giving a word problem algorithm for it, whereas, at first glance, there is no reason to believe that this group is finitely presented. This gives an example of a group for which one obtains much more naturally a solution to the word problem than a finite presentation. Thus there is no reason to believe that finite presentations are the only descriptions that can be used for actual implementation of decision problems. And in many cases, when a finitely presentable group is not yet described by a finite presentation, the most direct solution to finding a finite presentation for it uses the description of a normal form for its elements, which gives a solution to the word problem in this group. In this case, it is very natural to give to a program that should find properties about this group all the information we have on it.

2.2.4. The formalism of [48]. We finally address a remark that comes from **[48]**. In that article, the authors introduce a notion of "a property recursive modulo the word problem" which differs from the notion of "a property that can be recognized by an algorithm that takes as input both a finite presentation and a word problem algorithm for the group". This latter notion can be formalized thanks to the lattice structure on the set of numbering types, see Section 1.1.

Indeed, their definition can be rephrased as follows:

Definition 2.2.3 ([**48**], Definition 2.4). A class of finitely presented groups $\mathcal{C}$ is said to be recursive modulo the word problem if, whenever $\mathcal{D}$ is a set of presentations in which the word problem is uniformly solvable, there exist r.e. sets of groups $\mathcal{X}$ and $\mathcal{Y}$ such that $\mathcal{C} \cap \mathcal{D} \subseteq \mathcal{X}$, $\mathcal{D} \setminus \mathcal{C} \subseteq \mathcal{Y}$, $\mathcal{C} \cap \mathcal{D} \cap \mathcal{Y} = \emptyset$ and $(\mathcal{C} \setminus \mathcal{D}) \cap \mathcal{X} = \emptyset$.

Here, when we say that a set of finitely presented groups $\mathcal{C}$ is r.e., we of course mean that there exists an effective enumeration of finite presentations which contains at least a presentation for each group in $\mathcal{C}$.

A problem of this definition is the fact that the sets $\mathcal{X}$ and $\mathcal{Y}$ are to be r.e.. There exist families of finitely presented groups with uniformly solvable word problem that are not r.e., and even, that are not contained in any r.e. set of finitely presented groups with uniformly solvable word problem. This follows from the following results.

Lemma 2.2.4. *Let $\mathcal{C}$ be a set of finitely presented groups in which the word problem is uniformly solvable. If $\mathcal{C}$ is r.e., then there exists a recursive function which is a common asymptotic upper bound to the time complexity for the word problem for groups in $\mathcal{C}$.*

Proof. The argument is a diagonal argument: if G_1, G_2,... is an enumeration of the groups in $\mathcal{C}$, the function f which to n associates the longest time it took, in the first n groups of the enumeration, to determine whether or not a word of length at most n defines the identity, is recursive, and it bounds asymptotically the time complexity of the groups in $\mathcal{C}$.

One could also invoke the fact that the Higman-Neumann-Neumann Theorem which states that any countable group embeds in a two generated group is known to preserve solvability of the word problem, and thus for a class $\mathcal{C}$ of groups as in the statement of the lemma, there must exist a single finitely generated group with solvable word problem in which all the groups of $\mathcal{C}$ embed. The time complexity of the word problem in this group is an asymptotic upper bound for the time complexity of the word problem for any group that belongs to $\mathcal{C}$. ☐

Using this lemma with the work of Kharlampovich, Myasnikov and Sapir from [**60**], we obtain the following corollary:

Corollary 2.2.5. *The set of finitely presented residually finite groups is not r.e., and it cannot be contained in a r.e. set of finitely presented groups with uniformly solvable word problem.*

Proof. A well known result of McKinsey ([**79**]) shows that the word problem is uniform on all finitely presented residually finite groups. However, in [**60**], it is shown that for any recursive function f, there exists a finitely presented residually finite group whose word problem time complexity is more than f. The result then follows from Lemma 2.2.4. ☐

Now consider the consequences of this corollary on Definition 2.2.3: to prove that a set $\mathcal{C}$ is "recursive modulo the word problem", applying the definition with $\mathcal{D}$ being the set of residually finite groups, one needs to find r.e. sets $\mathcal{X}$ and $\mathcal{Y}$ that, in particular, must cover $\mathcal{D}$. But then the word problem cannot be uniformly solvable in $\mathcal{X} \cup \mathcal{Y}$, and thus one ends up working in sets where the word problem algorithm is not solvable to show that something is recursive modulo the word problem.

One could deal with this problem by imposing in Definition 2.2.3 that the set $\mathcal{D}$ be r.e.. We do not recommend such a definition either.

Remember our analogy with computable analysis: the notion of computability that we are interested in is an equivalent of what is known as Markov computability, (or computability "according to the Russian school", see [**102**]): a computable function defined on computable reals takes as input a Turing machine that defines a computable real and produces a Turing machine that defines the image of this number.

On the other hand, considering that a problem is solvable if and only if it is solvable on every r.e. set of groups (or r.e. set of finite presentations) is a definition similar to that of Banach-Mazur computability: a function is Banach-Mazur computable if and only if it maps every computable sequence of computable reals to a computable sequence of computable reals. This notion was first developed by Banach and Mazur between 1936 and 1939, but their results were not published at that time. The result of their work was written down later in [**78**], see [**54**] for a more modern point of view.

Banach-Mazur computability was shown to differ from Markov computability ([**55**]), and other notions were preferred to it (precisely because the example of [**55**] of a Banach-Mazur computable function that is not Markov computable can be considered "too" pathological). We likewise prefer Markov computability to Banach-Mazur computability for decision problems for groups.

While we expect most classes of groups in which many decision problems are solvable to be r.e. (all known sets of finitely presented groups on which the isomorphism problem is solvable are r.e.), we should not dismiss "by definition" the possibility of solving interesting problems on non r.e. sets of groups.

We stress the fact that defining a decision problem on a non-r.e. set is a common thing, the question "does there exist an algorithm that, for groups in $\mathcal{C}$, computes ...?" is valid even when the set $\mathcal{C}$ is not r.e.: the simple statement of computable analysis that the function $x \mapsto 3x$ is Markov computable asserts solvability of a problem defined on a non-r.e. set (as was remarked by Turing in 1937, [**124**]).

2.3. Recognizable groups

We now make a few remarks about the concept of group recognizability, that respond both our remark on the need of investigating decision problems for marked groups, and the one that concerns the need to investigate decision problems for groups described by stronger descriptions than finite presentations.

A group G is *recognizable* with respect to a certain type of description if there is an algorithm which takes as input a description of the selected type, and decides whether or not it is a description of G. This definition holds whether G is a marked group or an abstract group. We will thus talk about *abstract recognizability* and *marked recognizability.*

We call a group G (abstract or marked) *semi-recognizable* with respect to a type of description when there is an algorithm that stops exactly on the descriptions of this type that define G.

We will see that marked recognizability is not an isomorphism invariant: a group can be recognizable with respect to some marking, while not for others.

Let us first recall that the first result stated in this article, Proposition 2.1.2, implies the following:

Proposition 2.3.1. *No group, abstract or marked, is recognizable from the r.e. algorithm description.*

The trivial group is the only semi-recognizable group from the r.e. algorithm description (also either as a marked group, or as an abstract group).

The situation is quite different for finitely presented groups and for other strong descriptions. We discuss this now.

2.3.1. From finite presentations. The following is a well known result that follows from the use of Tietze transformations. A proof of it already appeared in this chapter (Proposition 2.1.7).

Proposition 2.3.2. *For groups described by finite presentations, every group, marked or abstract, is semi-recognizable. What's more, the method of recognition is uniform on all finite presentations.*

Remark 2.3.3. As we are trying to solve the isomorphism problem using descriptions that differ from finite presentations, one of the aspects that appears to be challenging and of foremost importance is to obtain group descriptions and classes of groups for which the result of this proposition holds.

Let us explain why.

In order to solve the isomorphism problem for a certain class $\mathcal{C}$ of groups using finite presentations, it suffices to be able to give a list of presentations with exactly one presentation for each isomorphism class of groups in $\mathcal{C}$ (this was first explained in detail in [**89**]). This very natural way to solve the isomorphism problem *does not work anymore* when dealing with descriptions of groups from which the groups are not uniformly semi-recognizable. For instance, it is easy to list a sequence of Word Problem Algorithms which contains exactly one copy of a description of every finitely generated abelian group, using the classification theorem for finitely generated abelian groups. This does not allow one to solve the isomorphism problem for abelian groups given by word problem algorithms (since this isomorphism problem is unsolvable, see Subsection 2.1.3.3). On the contrary, using this classification theorem to list a set of finite presentations which is in bijection with the isomorphism classes of finitely generated abelian groups *does* constitute a solution to the isomorphism problem for abelian groups given by finite presentations (and this, not even taking into consideration whether or not the proof of the theorem is effective). The former case is very unsatisfactory: we cannot ask for a better classification theorem than that of finitely generated abelian groups, and still it does not translate into a solution to the isomorphism problem for abelian groups given by their word problem algorithm.

The solution to the isomorphism problem in a class $\mathcal{C}$ of groups should constitute a step towards obtaining a classification theorem for groups in $\mathcal{C}$, and looking for ways to solve the isomorphism problem in already classified classes of groups is probably not something one would be inclined to spend much time on.

This emphasizes the need for finding descriptions of groups for which semi-recognizability results can be established. In particular, the algorithmic generalizations of finite presentations we discuss in Section 2.4 would satisfy this property.

We now return to discussing recognizability for finitely presented groups.

Some finitely presented groups admit markings which are recognizable. For instance, notice that given a finite presentation π over n generators, while it is well known that we cannot decide whether π defines a free group, we can easily decide whether it defines a free group of rank n: it suffices to check whether or not π has relations. This shows that free groups are recognizable with respect to any basis. Also, cyclic groups defined by presentations on one generator are recognizable, since any presentation on a single generator a can be simplified to a presentation of the form $\langle a|a^n\rangle$ for some integer n. However, we have the following proposition which shows that this is a rare situation.

Proposition 2.3.4. *Consider a finitely presented group G, marked by a generating family S. Add two generators symbols u and v to S, which correspond to the identity in G. Then G is not recognizable from finite presentations with respect to this new marking. And G is not abstractly recognizable.*

Proof. This proof follows from the construction given in [**83**] which is used to prove the Adian-Rabin theorem. The theorem cannot be applied directly, since "being isomorphic to G" is not always a Markov property, and because the Adian-Rabin theorem is stated only for abstract group properties.

We only use the fact that, in [**83**], given a finitely presented group H with unsolvable word problem, a family of finite presentations π_w, $w \in H$, is constructed, such that: π_w can be effectively constructed from w, and π_w defines the trivial group if and only if $w = 1$ in H. What's more, π_w is defined on two generators u and v.

It is then easy to see that for a given group G as in the statement of this proposition, the presentation for the free product of G and of the group defined by π_w defines a marking of G with respect to the enlarged generating family if and only if $w = 1$ in H, and that, by Grushko's theorem on the rank of a free product, G and this free product are abstractly isomorphic also if and only if $w = 1$. □

This result could surely be improved, as we conjecture that only the aforementioned groups (free groups, cyclic groups and the trivial group) admit markings which are recognizable from finite presentations. However, to obtain such results, one would have to encode undecidability of some sort with much less space to work with than the one given by two generators fully dedicated to encoding something uncomputable. The easiest open example may be to prove that, given a presentation on two generators a and b, one cannot decide whether or not it is a presentation of $\mathbb{Z}$ on the generating set $\{1; 0\}$.

Problem 2.3.5. Characterize the marked groups that are recognizable from the finite presentation description, and fill the "two additional generators" gap left by the proof of the Adian-Rabin Theorem.

There is, however, one case for which the situation is clear, that is for groups with unsolvable word problem:

Proposition 2.3.6. *Any marking of a finitely presented group with unsolvable word problem is unrecognizable from the finite presentation description.*

Proof. The proof is a variation on the solution to the word problem for simple groups ([**64**]). Consider a generating family S of G, and a finite presentation of G over S. To know whether a word w on S is the identity in G, add w to the relations of G. Then $w = 1$ in G if and only if G is isomorphic, as a marked group, to the group defined by this new presentation. □

Note that this proposition holds more generally for any group description for which "adding a relation" is a recursive operation. This remark allows us to comment on the link between recognizability as a marked group, and as an abstract group. Indeed, we ask:

Problem 2.3.7. Is there a natural example of a group G, recognizable as an abstract group, while some of its markings are not recognizable? (in some class $\mathcal{C}$ of groups, with respect to some type of description.)

This question is very much related to a question of Kharlampovich and Sapir (in [**61**]): can a variety of groups have solvable isomorphism problem (for groups given by presentations), while still containing a group with unsolvable word problem? In a class of groups where the isomorphism problem is solvable, any group is abstractly recognizable, thus by Proposition 2.3.6, a positive answer to that question would give a positive answer to Problem 2.3.7. It was also remarked in [**61**] that for a Hopfian group G with unsolvable word problem, the marked unrecognizability of Proposition 2.3.6 translates into unrecognizability as an abstract group.

2.3.2. From word problem algorithms. Section 5.5 of Chapter 5 is dedicated to the study of groups recognizable thanks to word problem algorithms. We include here some results that appear there, as we feel that they illustrate the fact that the systematic study of marked and abstract recognizability is the source of interesting problems.

Proposition 2.3.8. *Isolated groups are recognizable as marked groups from word problem algorithms.*

Conjecture 2.3.9. *Isolated groups are the only groups recognizable as marked groups from word problem algorithms.*

Isolated groups are defined in Section 5.5.

Proposition 2.3.10. *Finite groups are recognizable as abstract groups from word problem algorithms.*

Conjecture 2.3.11. *Finite groups are the only abstract groups recognizable from word problem algorithms.*

2.3.3. From finite presentations together with word problem algorithms. The next proposition gives further link between solvability of the word problem and marked recognizability.

Proposition 2.3.12. *Finitely presented groups with solvable word problem are recognizable, as marked groups, and with respect to any marking, when described by finite presentations together with word problem algorithms.*

Proof. The proof is straightforward, it suffices to check whether each group satisfies the relations of the other one using the word problem algorithm. □

The marked groups that are recognizable from the description "finite presentation-word problem algorithm" are thus completely classified. On the contrary, which groups are abstractly recognizable from this description is an interesting open problem.

Indeed, we have already mentioned in the introduction the results from [**49**] and [**48**], which provide examples of recognizable groups, including free groups and more generally limit groups.

On the other hand, Miller, in [**81**, Theorem 26, Chapter IV], has built a group that is unrecognizable in the class of finitely presented residually finite groups. As, by [**79**], finitely presented residually finite groups have uniformly solvable word problem, this group cannot, in the class of all finitely presented groups with solvable word problem, be recognizable from the description constituted of a finite presentation together with a word problem algorithm.

This shows that the problem of abstract group recognition from finite presentations together with word problem algorithms has a non-trivial answer. Miller's group has unsolvable conjugacy problem, and thus it bears some undecidability itself. We are not able to tell whether we should expect many groups to be recognizable from this description, or on the contrary that Miller's example be the norm, and recognizable groups be the exceptions.

Note that there are several issues one will encounter when trying to extend the methods of [**49**]. For instance, if one were to attempt to extend the results of this article to hyperbolic groups, one will be faced with the problem that it is not known whether the universal theory of every hyperbolic group is decidable, whereas the results from [**49**] relied on the fact that the universal theory of free groups is decidable.

We sum this up in the following problem, whose investigation we expect to be fruitful.

Problem 2.3.13. Describe the groups that are abstractly recognizable from the description that consists of a finite presentation together with a word problem algorithm.

2.4. Relative marked quotient algorithms

We've seen in Section 2.1 that giving a finite presentation of a marked group G is equivalent to giving a pair of algorithms, the r.e. algorithm for G, and a marked quotient algorithm that recognizes r.e. algorithms for marked quotients of G. But as algorithmic problems for groups are often set in restricted classes of groups, one in fact seldom needs the full strength of the marked quotient algorithm of a group. Because of this, we here introduce *relative marked quotient algorithms,* and we will see that, for some classes of groups, the relative marked quotient algorithms are not associated to finite presentations.

2.4.1. Definitions and Basic properties.

2.4.1.1. *Definitions.* We've seen already, in the first section of this chapter, that there are two natural marked quotient algorithms that finitely presented groups admit, the first one characterizes them amongst r.p. groups.

(1) The *marked quotient algorithm* A_Q^G of a marked group G, which takes as input a r.e. algorithm for a group, and stops if and only if the input is a marked quotient of G.
(2) The *WPI marked quotient algorithm* (for Word Problem as Input), which we note $A_{Q_{WP}}^G$, which takes as input a word problem algorithm for a group, and decides whether or not this group is a marked quotient of the starting group.

DEFINITION 2.4.1. Let $\mathcal{C}$ be a class of groups. We say that an algorithm $A^G_{Q,\mathcal{C}}$ is *a $\mathcal{C}$-marked quotient algorithm* for a marked group G if it takes as input r.e. algorithms for groups in $\mathcal{C}$ and stops exactly on marked quotients of G.

We say that an algorithm $A^G_{Q_{WP},\mathcal{C}}$ is *a WPI $\mathcal{C}$-marked quotient algorithm* if it takes as input word problem algorithms for groups in $\mathcal{C}$, and decides whether or not they define marked quotients of $\mathcal{C}$. If G admits such an algorithm, we say that *G has computable quotients in $\mathcal{C}$*

In Chapter 3, these notions are studied in more details in the case of $\mathcal{C}$ being the class of finite groups.

In particular, there is given an example of a residually finite group with computable finite quotients which is not recursively presented. This shows that the WPI marked quotient algorithm does not necessarily come from a marked quotient algorithm together with a recursive presentation.

The proof of Lemma 2.1.8, which asserted that having a marked quotient algorithm is independent of a given generating family, can easily be extended to relative quotient algorithms.

LEMMA 2.4.2. *If a group G admits a $\mathcal{C}$-marked quotient algorithm (or a WPI $\mathcal{C}$-marked quotient algorithm) with respect to some marking, then it admits $\mathcal{C}$-marked quotient algorithms (respectively, WPI $\mathcal{C}$-marked quotient algorithms) with respect to any marking.*

Note that in our definition, we supposed that the class $\mathcal{C}$ is a class of groups, i.e. a class of marked groups which is closed under abstract isomorphism. We could also consider marked quotient algorithms relative to sets of marked groups. In this case, the previous lemma may fail. We do not investigate marked quotient algorithms relative to sets of marked groups here.

2.4.1.2. *Basic Properties.* In what follows, $\mathcal{C}$ is a class of groups, and $\mathcal{QC}$ denotes either the set of groups which admit a $\mathcal{C}$-marked quotient algorithm, or the set of groups which admit a WPI $\mathcal{C}$-marked quotient algorithm.

The following propositions are straightforward.

PROPOSITION 2.4.3. *$\mathcal{QC}$ contains all finitely presented groups. More generally, it is stable by quotients by subgroups that are finitely generated as normal subgroups (i.e. adding finitely many relations to a group).*

PROPOSITION 2.4.4. *$\mathcal{QC}$ is stable by free and direct products.*

COROLLARY 2.4.5. *$\mathcal{QC}$ is stable by HNN-extensions or amalgamated products over finitely generated groups. More generally the fundamental group of a graph of groups with vertex groups in $\mathcal{QC}$ and finitely generated edge groups is again in $\mathcal{QC}$.*

For a class $\mathcal{C}$ of groups, call a group G *residually* $\mathcal{C}$ if every non-trivial element of G has a non-trivial image in a group of $\mathcal{C}$. Note $\mathcal{RC}$ the class of residually $\mathcal{C}$ groups. Any group G has a greatest quotient in $\mathcal{RC}$, namely the quotient of G by the intersection of all normal subgroups N of G for which G/N is in $\mathcal{C}$. This group is called the *residually $\mathcal{C}$ image* of G, and is noted $G_{\mathcal{C}}$ (as in [**35**]).

A residually $\mathcal{C}$ group is called *finitely presented as a residually $\mathcal{C}$ group* if it is the residually $\mathcal{C}$ image of a finitely presented group.

PROPOSITION 2.4.6. *A group G admits a $\mathcal{C}$-marked quotient algorithm if and only if the group $G_{\mathcal{C}}$ admits such an algorithm. Thus a group G which is finitely presented as a residually $\mathcal{C}$ group admits a $\mathcal{C}$-marked quotient algorithm.*

PROOF. This follows from the universal property of $G_{\mathcal{C}}$: any morphism from G to a group in $\mathcal{C}$ factors through $G_{\mathcal{C}}$. □

This proposition gives rise to many groups which have computable quotients in $\mathcal{C}$, sometimes for the sole reason that they do not have any quotients in $\mathcal{C}$. For instance, a simple group admits marked quotient algorithms in every class of groups to which it does not belong.

Note how this relate to the notion of *equationally noetherian groups* (introduced in [**7**], we quote the definition of [**50**]): a group G is called *equationally noetherian* if for every finitely generated group H, there exists a finitely presented group $\hat{H}$ and an epimorphism $\rho : \hat{H} \to H$ such that every morphism from $\hat{H}$ to G factors through ρ. It follows from that definition that for G an equationally noetherian group, every finitely generated group admits a $S(G)$-marked quotient algorithm, where $S(G)$ designates the set of subgroups of G.

2.4.2. Extending classical uses of marked quotient algorithms to relative algorithms. We quote three possible uses of marked quotient algorithms, by recalling instances of already existing proofs where finite presentations were used mostly for their marked quotient algorithm component, and discuss how to extend them to marked quotient algorithms relative to some classes of groups.

(1) The Tietze transformations algorithm ([**122**], see [**68**]);
(2) McKinsey's algorithm ([**79**]);
(3) Pickel's method (named in [**61**], from [**101**]).

2.4.2.1. *Tietze's Transformations.* This first point was already discussed in Sections 2.1 and 2.3. We noted there that finitely presented groups are uniformly semi-recognizable, and that this property is important because it allows us to avoid the embarrassing situation of having a completely understood and classified set of groups with unsolvable isomorphism problem. This result, usually explained by Tietze transformations, is immediately explained in terms of marked quotient algorithms: if two marked groups are each a quotient of the other, they are isomorphic.

To be able to use relative marked quotient algorithms to obtain semi-recognizability results, one needs to be able to find some groups in a class $\mathcal{C}$ that admit a $\mathcal{C}$-marked quotient algorithm. This is summed up in the following straightforward proposition:

Proposition 2.4.7. *Groups in a class $\mathcal{C}$ are uniformly semi-recognizable, either as marked groups, or abstractly, from the description that consists in a recursive presentation together with a $\mathcal{C}$-marked quotient algorithm.*

While, from the three results mentioned above, this proposition might not be the one that yields the most impressive results, it is in fact the most important one. We later discuss how it can apply to groups finitely presented inside varieties.

2.4.2.2. *McKinsey's Algorithm.* The second point is McKinsey's algorithm. The statement of McKinsey's theorem is the following (we only state it for groups, it was originally stated for *finitely reducible algebras*):

Theorem 2.4.8. *(McKinsey, [**79**]) A finitely presented residually finite group has solvable word problem.*

Recall that the proof of this theorem is as follows. All that we need to prove is that a finitely presented residually finite group is co-r.p.. Let G be a finitely presented residually finite group. Let w be a word on the generators of G. Enumerate all finite groups by listing all possible Cayley tables. For each obtained finite group, decide whether it is a quotient of G by checking whether the finitely many relations of G hold in it. Then, if it is indeed a quotient of G, and if the image of w in this quotient is a non-identity element, answer that $w \neq 1$ in G. If w is indeed a non-identity element of G, by definition of "being residually finite", this process terminates.

It is then easy to see that the most general formulation of this argument is the following:

Theorem 2.4.9. *Suppose that there exists a recursive enumeration by word problem algorithms of a class $\mathcal{C}$ of groups (and thus that $\mathcal{C}$ consists only of groups with solvable word problem). Then any residually $\mathcal{C}$ group with a $\mathcal{C}$-marked quotient algorithm is co-r.e., and thus any r.e. residually $\mathcal{C}$ group with a $\mathcal{C}$-marked quotient algorithm has solvable word problem.*

Furthermore, there exist uniform ways of producing the said co-r.e. and word problem algorithms, from, respectively, a $\mathcal{C}$-marked quotient algorithm or a $\mathcal{C}$-marked quotient algorithm with a recursive presentation.

Several versions of this theorem have already been used to prove that some classes of groups have solvable word problem. For instance, Sela, in [**116**], used the fact that a finitely presented residually H group, for H some hyperbolic group, must have solvable word problem.

Note also that one can replace residual finiteness in the statement of this theorem by conjugacy separability or subgroup separability *relative to a class $\mathcal{C}$*, and obtain theorems about the conjugacy and membership problems. For instance, subgroup separability with respect to amenable groups was introduced in [**43**] (but the class of amenable groups cannot be used to obtain a McKinsey algorithm). One should replace accordingly the enumeration of $\mathcal{C}$ by word problem algorithms by an enumeration by conjugacy problem algorithms, or by membership problem algorithms.

McKinsey's Algorithm is discussed at length in Chapter 3, the notions of conjugacy separability and of subgroup separability are properly defined there.

2.4.2.3. *Pickel's Method.* The last point corresponds to what is known as Pickel's method. In [**101**], Pickel showed that only finitely many nilpotent groups can have the same set of finite quotients. From this (and the fact that nilpotent groups are finitely presented), it is easy to deduce that every finitely generated nilpotent group is abstractly recognizable from other nilpotent groups. Indeed, consider a nilpotent group G, and the list G_1, ..., G_l of nilpotent groups with the same set of finite quotients as G. Given a finite presentation of a group H, use the semi-recognition property (Tietze transformations) to try and prove that H appears in the list G, G_1, ..., G_l; all the while, list all finite quotients of H and try to prove that some finite group is a quotient of G and not of H, or of H and not of G. This process always terminates and allows one to decide whether or not a given presentation defines G.

It seems that Pickel's method can be used only with the marked *finite* quotient algorithm. Indeed, not only does it rely on the solution to the isomorphism problem for finite groups, but also on the fact that from the WPI finite marked quotient algorithm of a group G, which tells us whether a given finite *marked group* is a quotient of G, one can obtain an algorithm which decides whether an *abstract* finite group is a quotient of G. This second fact uses both the isomorphism problem for finite groups and the fact that any abstract finite group admits finitely many markings of any given arity. The best directly available generalization of Pickel's method is thus:

Proposition 2.4.10. *There is an effective procedure which, given two WPI finite marked quotient algorithms for groups G and H, terminates if and only if their set of finite quotients differ.*

Note that, since Pickel's article, many results of the same kind were obtained for various classes of groups. A class $\mathcal{C}$ of residually finite groups is said to be *profinitely rigid* if any two different groups of $\mathcal{C}$ have different sets of finite quotients. Note that a marked residually finite group is always uniquely defined by its marked finite quotients, and thus the notion of profinite rigidity naturally appears as one moves from the category of marked groups to that of abstract groups. See **[109]** for a survey on profinite rigidity.

To be able to use notions of rigidity with classes of quotients that are possibly infinite, in order to obtain results akin to the one given above for nilpotent groups, one would have to define "abstract quotient algorithms". For instance, one could consider an algorithm that takes as input a finite presentation and decides whether or not it defines an abstract quotient of a given group G. However, such algorithms will never exist when set in unrestricted classes of groups. For instance, being able to decide whether or not a finite presentation defines an abstract quotient of a rank two free group is equivalent to deciding whether the rank of this group is less or equal to two, which is impossible by the Adian-Rabin construction (see the proof of Proposition 2.3.4). But this does not mean that such algorithms cannot exist in restricted classes of infinite groups, the study of such algorithms could be interesting for residually free groups, residually hyperbolic groups, etc.

2.4.3. Group varieties. There in fact already exists a wide range of classes of groups that satisfy the generalized Tietze transformations criterion: groups that are finitely presented *inside* group varieties. A group variety is a class of groups defined by a set of *laws*, and a law is a universal sentence of the form:

$$\forall x_1, ..., x_n,\ W = 1$$

for W a word on the letters $\{x_1, ..., x_n\} \cup \{x_1^{-1}, ..., x_n^{-1}\}$. A group is said *finitely presented inside a variety* $\mathcal{V}$ if it admits a presentation with finitely many relations apart from the infinitely many relations that constitute the laws of the variety $\mathcal{V}$. This notion in fact corresponds to being finitely presented as a residually $\mathcal{V}$ group, because group varieties are stable under taking subgroups and forming unrestricted direct products, and thus the class of residually $\mathcal{V}$ groups corresponds to $\mathcal{V}$. It follows that a group which admits a finite presentation inside a variety admits a marked quotient algorithm inside this variety. For a variety defined by a recursively enumerable set of laws, this is in fact an equivalence.

Theorem 2.4.11. *Let $\mathcal{V}$ be a group variety defined by a r.e. set of laws. A recursively presented group H of $\mathcal{V}$ admits a $\mathcal{V}$-marked quotient algorithm if and only if it is finitely presented as a group in the variety $\mathcal{V}$.*

What's more, from a pair $(A_{re}^H, A_{Q,\mathcal{V}}^H)$, a finite presentation in the variety $\mathcal{V}$ of H can be effectively obtained.

Proof. The proof is identical to that of Theorem 2.0.3, except that one needs to add the laws that defines $\mathcal{V}$ when building the algorithm $\mathcal{A}_{re}^n$. □

The residually $\mathcal{V}$ image $G_{\mathcal{V}}$ of a group G is the group obtained by imposing on G the laws of $\mathcal{V}$. The previous theorem thus characterizes groups with a $\mathcal{V}$ marked quotient algorithm amongst r.p. groups.

Corollary 2.4.12. *A r.p. group G admits a $\mathcal{V}$-marked quotient algorithm if and only if $G_{\mathcal{V}}$ is finitely presented as a group in $\mathcal{V}$.*

This shows that the notion of having a $\mathcal{V}$ marked quotient algorithm is already well known for group varieties.

2.4.4. Elementary marked quotient algorithms. The interest of relative quotient algorithms lies mostly in situations where the marked quotient algorithms are *not* given by finite presentations, and the case of an algorithm that, to decide whether a group H given by an algorithm A_{re}^H is a quotient of a group G, checks whether H satisfies a finite set of relations *which is independent of H* is considered to be the trivial case.

Definition 2.4.13. A marked quotient algorithm $\mathcal{A}_Q$ for a marked group H is called *elementary* if there exists a finite set $\mathcal{R}$ of relations, such that $\mathcal{A}_Q$ recognizes the set of marked groups that satisfy the relations in $\mathcal{R}$.

The previous theorem about group varieties -as well as its particular instance in the variety of all groups- thus proves that only elementary marked quotient algorithms exist, relative to group varieties.

A first example of a marked quotient algorithm that is not elementary is the finite quotient algorithm (in fact, the torsion quotient algorithm) of the Lamplighter group.

Proposition 2.4.14. *The lamplighter group has a non-elementary finite marked quotient algorithm.*

Proof. The lamplighter group L has a finite marked quotient algorithm. Indeed, it admits the following presentation:

$$\langle a, \varepsilon \,|\, \varepsilon^2,\ \left[\varepsilon, a^{-n}\varepsilon a^n\right], n \in \mathbb{Z}\rangle$$

To see whether a finite group F generated by two elements a_1 and ε_1 is a quotient of it, find a multiple N of the order of a_1 using the r.e. algorithm of F. Then, notice that F is a quotient of L if and only if it is a quotient of the group obtained from L by adding the relation a^N. But the quotient $L/\langle\langle a^N\rangle\rangle$ is in fact the finite wreath product $\mathbb{Z}/N\mathbb{Z} \wr \mathbb{Z}/2\mathbb{Z}$, which admits the finite presentation:

$$\langle a, \varepsilon \,|\, \varepsilon^2,\ a^N,\ \left[\varepsilon, a^{-n}\varepsilon a^n\right], 0 \leq n \leq N\rangle$$

It can be effectively checked whether F is a quotient of this finite group, and thus of L.

Because the lamplighter group is residually finite, to prove that its finite quotient algorithm is not elementary, it suffices to prove that it is not finitely presented as a residually finite group. Indeed, it is easy to check that if a finite number of relations r_1, ..., r_p characterized the finite quotients of L that are generated by two elements a_1 and ε_1, the group defined by the presentation $\langle a_1, \varepsilon_1 | r_1, ..., r_p\rangle$ would have L as its residually finite image.

We thus now prove that L is not finitely presented as a residually finite group.

Suppose that we have a group G, given by a presentation $\langle a_1, \varepsilon_1 | r_1, ..., r_p\rangle$, and a morphism $\phi : G \to L$, defined by $\phi(a_1) = a$ and $\phi(\varepsilon_1) = \varepsilon$, which satisfies that any morphism h from G to a finite group F factors through ϕ. The corresponding diagram is as follows:

$$\begin{tikzcd} G \arrow[r, "\phi"] \arrow[dr, "h_0"'] & L \arrow[d, dashed, "h_1"] \\ & F \end{tikzcd}$$

Since L is a quotient of G, L must satisfy the relations of G, those relations must thus be consequences of a finite number of the relations of L. In particular, there must be a natural number N such that the first N relations of L imply those of G. Consider the group H given by the presentation

$$\langle a, \varepsilon \,|\, \varepsilon^2,\ \left[\varepsilon, a^{-n}\varepsilon a^n\right], 0 \leq n \leq N\rangle.$$

The property of G, that all its finite quotients come from quotients of L, must be shared by H: any morphism h_0 from H to a finite group F defines a morphism h_1 from G to F, which by the property of G, factors through ϕ:

$$\begin{tikzcd} G \arrow[r, "\phi_1"] \arrow[rrd, "h_1"] & H \arrow[r, "\phi_2"] \arrow[rd, "h_0"] & L \arrow[d, dashed, "h_2"] \\ & & F \end{tikzcd}$$

To end the proof, we find a finite group which satisfies the relations of H, but not that of L. We define a subgroup of the group $\mathfrak{S}_{5N}$ of permutations on $\{1, ..., 5N\}$. Consider the element σ_0 of $\mathfrak{S}_{5N}$, defined by the following formula:

$$\sigma_0(i) = \begin{cases} i+2 & i \leq 5N-2 \\ i+2-5N & i \geq 5N-1 \end{cases}.$$

Let σ_1 be the product of the transpositions $(1, 2)$ and $(2N+4, 2N+5)$. It is then easy to see that the following relations hold between σ_0 and σ_1:

$$\sigma_1^2 = id,$$
$$\left[\sigma_1, \sigma_0^{-n}\sigma_1\sigma_0^n\right] = id,\ 1 \leq n \leq N,$$
$$\left[\sigma_1, \sigma_0^{-N-1}\sigma_1\sigma_0^{N+1}\right] \neq id.$$

The subgroup of $\mathfrak{S}_{5N}$ generated by σ_0 and σ_1 is thus a finite quotient of the group H, but not of L. This contradicts the supposition that L be finitely presented as a residually finite group. ☐

In Section 3.4, we characterize wreath products of groups that have a WPI marked quotient algorithm relative to finite groups.

The fact that the lamplighter group admits a finite quotient algorithm also follows from a more general result, due to Mostowski ([**88**]):

PROPOSITION 2.4.15. *A group that is finitely presented in a variety $\mathcal{V}$ defined by finitely many laws has a finite marked quotient algorithm (and a WPI finite marked quotient algorithm).*

PROOF. From the r.e. algorithm of a finite group, arbitrarily good upper bounds on its cardinality can be found. Thus to decide whether a finite group satisfies a law, only finitely many relations need to be checked. □

We have already seen that any group which satisfies the hypotheses of this proposition admits an elementary $\mathcal{V}$-marked quotient algorithm. However, the marked finite quotient algorithm given by Mostowski's proposition is not elementary: depending on the size of the input group, the number of relations that this algorithms tests may vary, and the set of relations to test is not fixed *a priori*.

These examples of non-elementary finite quotient algorithms rely on the fact that the considered groups do not belong to the class of finite groups. This thus leaves the following important problem unanswered:

PROBLEM 2.4.16. Find a class $\mathcal{C}$ of r.p. groups, and r.p. groups in $\mathcal{C}$, that admit non-elementary $\mathcal{C}$-marked quotient algorithms.

This would constitute a genuine algorithmic generalization of the notion of finite presentation.

CHAPTER 3

Computability of the finite quotients of finitely presented groups

Introduction

In this chapter, we study algorithms that recognize finite quotients of groups that may not be finitely presented.

The main definition of the previous chapter was the notion of marked quotient algorithm, and we explained how this notion adapts easily to relative marked quotient algorithms.

Here, we will focus on a variation of the marked finite quotient algorithm that uses finite presentations instead of recursive presentations to describe finite groups. The original definition, which appears in the previous chapter, is as follows:

Definition 3.0.1. A marked group (G, S) has a *marked finite quotient algorithm* if there exists an algorithm that stops exactly on r.e. algorithms for finite marked groups that are marked quotients of (G, S).

A marked group (G, S) has a *WPI marked finite quotient algorithm* if there exists an algorithm that takes as input a word problem algorithm for a finite marked group and decides whether or not it is a marked quotient of (G, S).

Because describing finite groups by r.e. algorithms (i.e. by recursive presentations) is very unnatural, we will consider here the following definitions:

Definition 3.0.2. A finitely generated marked group (G, S) is said to have Computable Finite Quotients (CFQ) if there is an algorithm that, given a pair (F, f), where F is a finite group given by a finite presentation and f is a function from S to F, determines whether or not the function f extends to a group morphism, that is whether there exists a group homomorphism $\hat{f} : G \to F$ such that for any s in S, $f(s) = \hat{f}(s)$.

If there exists an algorithm that terminates when the function f extends to a group morphism, but does not terminate otherwise, we say that G has Recursively-enumerable Finite Quotients (ReFQ).

If there exists an algorithm that terminates when the function f does not extend to a group morphism, but does not terminate otherwise, we say that G has co-Recursively-enumerable Finite Quotients (co-ReFQ).

Of course, having CFQ is equivalent to having both ReFQ and co-ReFQ.

Those definitions are more natural than the ones that stem from the general setting of marked quotient algorithms. However, the definitions of CFQ and ReFQ are equivalent to definitions that have been obtained in the previous chapter. Indeed, in Section 3.1, we prove:

Proposition 3.0.3. *A marked group (G, S) has ReFQ if and only if it has a marked finite quotient algorithm. It has CFQ if and only if it has a WPI marked finite quotient algorithm.*

McKinsey's Algorithm for infinitely presented groups. We have already quoted McKinsey's algorithm for the word problem in Chapter 2, we must here describe it again.

Given a finitely presented residually finite group G generated by a family S, and a word w, it is possible to prove that $w \neq 1$ in G: enumerating all finite groups, then choosing which of those are marked quotients of (G, S) by checking whether the finitely many relations of a finite presentation of (G, S) hold in those finite groups, and finally checking whether or not the element w defines a non-identity element in those finite quotients provides a procedure that recognizes all non-identity elements of (G, S) when it is residually finite.

It was remarked early on ([**71**], [**88**]) that this algorithm could be used to prove solvability of the conjugacy problem in conjugacy separable groups, and of the generalized word problem (uniform membership problem) in subgroup separable groups. Those notions are defined in SubSection 3.1.2.

Those properties were intensely studied, both for finitely presented groups and infinitely presentable ones. However, in infinitely presentable groups, those properties have an impact on solvability of problems *only* for groups whose quotients can be enumerated. Our present study thus bridges gaps left in several articles: the author found articles where it is implied either that separability results are sufficient to prove solvability of decision problems inside infinitely presented groups (this is not the case if the considered groups do not have CFQ), or on the contrary where it is thought that enumerations of finite quotients can never be done for infinitely presented groups.

Our section of examples of infinitely presented groups with CFQ, where we discuss CFQ for wreath products of groups and L-presented groups (which include known groups acting on rooted trees), is key in this regard. See Section 3.4.

Main Results. In order to show that the properties CFQ, ReFQ and co-ReFQ are properties in their own right, we prove in this chapter that they are independent from the word problem for residually finite groups. Our main results are thus the following two theorems:

THEOREM 3.0.4. *There exists a finitely generated residually finite group with solvable word problem, but that has uncomputable finite quotients.*

THEOREM 3.0.5. *There exists a finitely generated residually finite group with computable finite quotients, that still has unsolvable word problem.*

The proof of Theorem 3.0.4 relies on a family of groups introduced by Dyson in [**35**], which are doubles of lamplighter groups. Notice that Theorem 3.0.4 strengthen some of Dyson's results from [**35**], since one can find in this article a finitely generated group with solvable word problem that has uncomputable finite quotients. However, this group is not residually finite, in fact the proof of the fact that it has uncomputable finite quotients relies on the fact that it is not residually finite.

Theorem 3.0.5 is an easy consequence of the work of Slobodskoi in [**119**], modulo translation in the right mathematical language. (Notice that Dyson's main objective in [**35**] was to make progress towards Slobodskoi's Theorem from [**119**], that *the universal theory of finite groups is undecidable.*) We obtain another proof of this theorem, also using Dyson's groups.

Plan. We proceed as follows.

In the first section, we give several equivalent definitions of CFQ groups, and we explain how the profinite topology on a group can be used to describe the various decision problems that can be solved using McKinsey-type algorithms. The second section quickly enumerates some easy properties: free or direct products of groups with computable finite quotients also have this property, etc. In the third section interactions with the depth function for residually finite groups are explained. The fourth section provides examples of infinitely presented groups with CFQ: wreath products of groups with CFQ and L-presented groups. In the final section, we prove Theorem 3.0.4 and Theorem 3.0.5.

3.1. CFQ groups

3.1.1. Equivalent definitions. We start by proving the following proposition, presented in the introduction of this chapter.

PROPOSITION 3.1.1. *A marked group (G, S) has ReFQ if and only if it has a marked finite quotient algorithm. It has CFQ if and only if it has a WPI marked finite quotient algorithm.*

PROOF. We start with the equivalence between having ReFQ and having a marked finite quotient.

One direction is obvious. Indeed, (G, S) having a marked finite quotient algorithm means that it is possible to recognize finite quotients of (G, S) described by r.e. algorithms. Since a finite presentation contains more information than a r.e. algorithm, it is thus possible to recognize marked finite quotients of (G, S) described by finite presentations, and thus (G, S) has ReFQ.

One the other hand, suppose that (G, S) has ReFQ. Given a r.e. algorithm for a marked finite group (F, S'), we want an algorithm that stops if (F, S') is a quotient of (G, S).

We search for a finite presentation of a finite group that defines a marked group (F_1, S_1), that is both a quotient of (G, S) and that surjects itself onto (F, S'). That is to say we search for a sequence of morphisms of the form:

$$(G, S) \to (F_1, S_1) \to (F, S').$$

An exhaustive search for a finite presentation for (F_1, S_1) will work, because (G, S) has ReFQ, and because it is possible to check whether $(F_1, S_1) \to (F, S')$ holds, when (F_1, S_1) is given by a finite presentation and (F, S') by a r.e. algorithm. This produces the desired procedure.

We now prove the equivalence between having CFQ and having a WPI finite marked quotient algorithm.

The equivalence between having CFQ and having a WPI marked finite quotient algorithm directly follows from the following remark: for finite groups, the finite presentation description is equivalent to the word problem algorithm description. Indeed, it is well known that finite groups have uniformly solvable word problem (given a finite presentation of a finite group, one can for instance enumerate all finite presentations Tietze-equivalent to the starting presentation, until a Cayley table presentation is found, this presentation immediately solves the word problem). Conversely, given a word problem algorithm for a finite group, we can directly build the Cayley table presentation of the group it defines, thus obtaining a finite presentation of this group. ☐

It is easy to see that the first equivalence that appears in the proof above relies on the fact that finite groups are all finitely presented, and that there exists a process that enumerates all finite presentations for finite groups. This property, while shared by other families of groups (hyperbolic groups, polycyclic groups), is quite strong, and thus the general setting of "marked quotients algorithms" described in the previous chapter would lose some of its applicability if we were to try and developing it using only finite presentations.

Just as a group with solvable word problem is a group in which words in the generators corresponding to the identity can be enumerated by an algorithm which respects a computable ordering on the set of words in the generators, or a recursively presented group is a group in which these words can be enumerated, but without any guarantee on the order of the enumeration, groups with CFQ, ReFQ or co-ReFQ can be equivalently characterized by enumeration of their finite quotients. Let us precise this.

G is still a group generated by S, of cardinal n.

Consider an effective enumeration (F_1, f_1), (F_2, f_2), (F_3, f_3),... of all n-marked finite groups, which satisfies $card(F_n) \leq card(F_{n+1})$. (This can be obtained by listing in order all possible Cayley tables, then listing all n-tuples from those tables and determining when a tuple defines a generating set). Define $\mathcal{A}_G \subseteq \mathbb{N}$ to be the set of indices k for which f_k defines a morphism from G to F_k. Then G has CFQ, ReFQ or co-ReFQ if and only if $\mathcal{A}_G$ is, respectively, a recursive set, a recursively enumerable set or a co-recursively enumerable set.

Of course, those definitions are independent of a choice of a generating family.

Proposition 3.1.2. *Having one of CFQ, ReFQ or co-ReFQ is independent of a choice of a generating family.*

We could invoke Proposition 3.1.1 together with the corresponding result for marked quotient algorithms for the first and second properties.

Proof. Let S and T be two finite generating sets of a group G (not necessarily of the same cardinal). Fix for each s in S an expression $s = t_1^{\alpha_1}...t_k^{\alpha_k}$, with $\alpha_i \in \{-1, 1\}$ and $t_i \in T$, that gives s as a product of elements of T or their inverses, and for each t in T an expression $t = s_1^{\beta_1} s_2^{\beta_2}...s_k^{\beta_k}$ that describes t in terms of the generators of S and their inverses. For a finite group F and a function f from S to F, define the function f' from T to F by $f'(t) = f(s_1)^{\beta_1}...f(s_k)^{\beta_k}$. The function f defines a homomorphism if, and only if, f' also defines a homomorphism φ', that satisfies $\varphi'(s) = f(s)$ for s in S. That last condition is an equality in F that can be tested using the expressions $s = t_1^{\alpha_1}...t_k^{\alpha_k}$, even before it is known whether or not f' extends. Using this, all three properties, CFQ, ReFQ, co-ReFQ can be seen to be independent of the chosen generating family of G. ☐

3.1.2. Variations on McKinsey's algorithm and the profinite topology. To generalize McKinsey's algorithm, we will search for conditions on a finitely generated group G with CFQ, that might allow to solve various algorithmic problems. We will see that such conditions can be expressed through the use of the profinite topology on G. In what follows, we will say that we use McKinsey's algorithm to mean that we enumerate all finite quotients of a group, checking some condition in each quotient, and stopping when a finite quotient is found that satisfies the required condition.

Fix a finitely generated group G with CFQ (or simply ReFQ).

We will search for conditions that allow, given two disjoint subsets A and B of G, to *distinguish A from B* using McKinsey's algorithm, that is to say to decide, given an element of G that belongs to $A \cup B$, whether it belongs to A or to B.

The first condition we need is that, given a morphism from G onto a finite group F, it be possible to completely determine the images of A and of B in F. We will thus need the following definition:

Definition 3.1.3. A subset A of G is said to be *determinable in finite quotients of* G if there exists an algorithm that, given a morphism ϕ from G onto a finite group F (F is, as always, given by a finite presentation), can compute the image $\phi(A)$, i.e. solve the membership problem for $\phi(A)$ in F.

A family $(A_i)_{i\in\mathbb{N}}$ of subsets of G is *uniformly determinable in finite quotients of* G if each A_i is determinable in finite quotients of G, and if the algorithm that determines A_i in finite quotients of G depends recursively of i.

Note, as an example, that a finitely generated subgroup H of a group G is determinable in the finite quotients of G, because the image of H in a quotient of G is the group generated by the images of the generators of H.

The property of being determinable in finite quotients is interesting in itself, however we will not give it much attention. We still remark the following.

It is easy to build a family $(A_i)_{i\in\mathbb{N}}$ of subsets of $\mathbb{Z}$, such that: it is uniformly recursively enumerable, but not uniformly determinable in finite quotients of $\mathbb{Z}$. But a stronger result will naturally appear in the course of our study as a byproduct of the proof of Theorem 3.0.4:

Proposition 3.1.4. *There exists a recursive subset of $\mathbb{Z}$ that is not determinable in finite quotients of $\mathbb{Z}$.*

Proof. See Lemma 3.5.5, disregarding the statement about the subset of $\mathbb{Z}$ being closed. ☐

On the other hand, it is easy to build a subset of $\mathbb{Z}$ that is determinable in its finite quotients, but not recursive.

Proposition 3.1.5. *There exists a subset of $\mathbb{Z}$ that is not r.e., but still is determinable in finite quotients of $\mathbb{Z}$.*

Proof. For a function h that grows faster than any recursive function, consider the enumeration $2h(1)$, $2h(2)+1$, $3h(3)$, $3h(4)+1$, $3h(5)+2$, $4h(6)$,... This defines a set that is not r.e., but whose image in any quotient $\mathbb{Z}/n\mathbb{Z}$ of $\mathbb{Z}$ is all of $\mathbb{Z}/n\mathbb{Z}$. ☐

Given two disjoint subsets A and B of G that are indeed determinable in finite quotients of G, the profinite topology of G can be used to decide whether McKinsey's algorithm can tell them apart.

The *profinite topology* on a group G was introduced in [**51**], it is the topology whose open basis consists of cosets of finite index normal subgroups of G. We denote $\mathcal{PT}(G)$ the profinite topology on G. A closed subset of G in $\mathcal{PT}(G)$ is called a *separable* set. We note $\bar{A}$ the closure of a set A in $\mathcal{PT}(G)$. The following easy facts render explicit the link between the profinite topology and McKinsey-type algorithms:

Fact 3.1.6. *A subset A of a group G is open in $\mathcal{PT}(G)$ if and only if for any element a of A, there is a morphism ϕ from G onto a finite group F such that $\phi^{-1}(\phi(a)) \subseteq A$, i.e. such that if an element of G has the same image as a in F, it also belongs to A.*

Fact 3.1.7. *The closure $\bar{B}$ of a subset B of G is the biggest set of elements that satisfy the following condition:*

For any morphism ϕ from G onto a finite group F, $\phi(\bar{B}) \subseteq \phi(B)$.

This shows that the closure of a set B in $\mathcal{PT}(G)$ is precisely the set of elements that cannot be distinguished from B using McKinsey's algorithm.

We can now give conditions that allow two disjoint subsets A and B of G to be distinguishable by McKinsey's algorithm.

Proposition 3.1.8. *Let A and B be disjoint subsets of G that are determinable in finite quotients of G. Then McKinsey's algorithm can be used to distinguish A from B if and only if the following two conditions hold:*

$$A \cap \bar{B} = \varnothing;$$
$$\bar{A} \cap B = \varnothing.$$

Proof. This is a simple consequence of Fact 3.1.7. ☐

McKinsey's original result on residually finite groups can thus be interpreted as an application of this proposition to singletons, and a residually finite group is precisely a group in which all the singletons are closed in the profinite topology.

Two other well studied families of groups fall in the range of Proposition 3.1.8: conjugacy separable groups, and LERF groups.

A conjugacy separable group is a group G in which all the conjugacy classes are separable. It is easy to see that the conjugacy classes of a finitely generated group G are always uniformly determinable in finite quotients of G (when a class C is given by any of its elements), thus Proposition 3.1.8 can be used to distinguish conjugacy classes.

A LERF group (for locally extended residually finite), or subgroup separable group, is a group G whose finitely generated subgroups are separable.

We then have the following proposition:

Proposition 3.1.9. *Let G be a finitely generated group with ReFQ.*

- *If G is residually finite, then it is co-r.p.. If it is r.p. and residually finite, it has solvable word problem.*
- *If G is conjugacy separable, then there exits an algorithm that decides when two of its elements are not conjugate. If it is r.p. and conjugacy separable, then it has solvable conjugacy problem.*
- *If G is LERF, there is an algorithm that, given a tuple $(x_1, ..., x_n, g)$ of elements of G, stops exactly when g does not belong to the subgroup of G generated by $(x_1, ..., x_n)$. If it is r.p. and LERF, it has solvable generalized word problem.*

Proof. All three points follow from Proposition 3.1.8, noticing that it provides a uniform algorithm, and using it respectively with:

- A and B being singletons,
- A and B being conjugacy classes,
- A being a finitely generated subgroup of G and B a singleton.

In each case, one needs to use the fact that those sets are uniformly determinable in finite quotients, which is straightforward. □

Because r.p. groups naturally have co-ReFQ (Proposition 3.2.1), the statements that concern r.p. group in the previous proposition could be formulated with CFQ instead of ReFQ, without loss of generality.

Proposition 3.1.9 follows in a very straightforward way from the definitions of residually finite, of conjugacy separable and of LERF groups, and it is surprising that the study of these properties was not followed by a systematic study of the properties CFQ and ReFQ.

3.1.3. Membership problem for finite index normal subgroups. In the article [**16**], Bou-Rabee and Seward use, in the course of a proof ([**16**], Proof of Theorem 2), the fact that, if a group G has solvable "membership problem for finite index normal subgroups", (or "generalized word problem for finite index normal subgroups") and solvable word problem, then it admits an algorithm that recognizes its finite quotients.

We will now show that for r.p. groups, having solvable membership problem for finite index normal subgroups is actually equivalent to having CFQ. This will allow us to give another point of view on groups with CFQ.

When formulating the membership problem for finite index normal subgroups, it is implicit that the normal subgroup is given by a finite generating family.

Indeed, the membership problem for finite index normal subgroups in a group G asks for an algorithm that, given a tuple $(x_1, ..., x_k, g)$ of elements of G, the first k elements of which generate a finite index normal subgroup of G, will decide whether g belongs to that subgroup. As opposed to that, when working with property CFQ, we describe non-ambiguously a finite index normal subgroup N of G by a pair (F, f), where F is a finite group and f a function from the generators of G to F, which extends to a group homomorphism, the kernel of which is precisely N.

Of course, given that second description, the problem "does g belong to N" is solved by computing the image of g in F to see whether it is the identity of F. Thus a group in which one can go from the description of a normal subgroup by generators to a description of this subgroup by a morphism necessarily has solvable membership problem for finite index normal subgroups. We will see that for r.p. groups this is also sufficient.

On the other hand, given a description by morphism of the subgroup N, that is a morphism $\varphi : G \to F$ with $ker\, \varphi = N$, one can always obtain a description of it by generators, as one can effectively carry out the well known proof of Schreier's lemma, which is often used to prove that a finite index subgroup of a finitely generated group is itself finitely generated. Indeed, if S is a generating family of G, for any x in F and s in S, a preimage $\hat{x}$ of x can be found in G, by exhaustive search, and a preimage of $x\varphi(s)$ can be found as well, call it $\hat{y}$. Schreier's lemma asserts that the elements of the form $\hat{x}s\hat{y}^{-1}$ generate N.

This allows us to prove the following (the backward implication is directly adapted from [**16**]):

Proposition 3.1.10. *Property ReFQ is equivalent to having co-semi-decidable membership problem for finite index normal subgroups, that is to having an algorithm that decides when an element is not in a given finite index normal subgroup, and does not terminate otherwise.*

Proof. Suppose first that G has ReFQ, and let N be a finite index normal subgroup of G generated by a family $x_1, ..., x_k$. Let finally g be an element of G, we want to decide whether g belongs to N. Enumerate the quotients (F, f) of G, and look for a finite quotient in which the image of g is non-trivial, while the images of $x_1, ..., x_n$ are all trivial. If g does not belong to N, such a quotient exists (the projection $G \to G/N$), and this algorithm will terminate.

Suppose now that G has co-semi-decidable membership problem for finite index normal subgroups. Write $\langle S|R\rangle$ a presentation of G. Let (F, f) be a finite group together with a function from S to F. As f does not necessarily define a morphism, we cannot yet apply Schreier's method. But if $\mathcal{F}_n$ is a free group with basis the n generators of G, f does define a morphism φ from $\mathcal{F}_n$ to F, and thus we can find a family $x_1, ..., x_k$ of elements of $\mathcal{F}_n$ that generate $ker(\varphi)$. F is given by the presentation: $\langle S|x_1, ..., x_k\rangle$. (But x_1, ..., x_k generate $ker\, \varphi$ as a group, and not only as a normal subgroup as would be guaranteed by any presentation of F on the generators S).

Now f extends to a morphism if and only if F satisfies all the relations of G, that is to say if and only if the relations $x_1, ..., x_k$ imply the relations that appear in R, that is to say if and only if $\langle S|R, x_1, ..., x_k\rangle$ is just another presentation of F. But this is a presentation of G/N, where N is the subgroup of G generated by $x_1, ..., x_k$. If f does not extend to a morphism, G/N is a strict quotient of F.

Thus we can do the following: enumerate the elements of G, g_1, $g_2, ...$ Then use the membership algorithm for N, (which, as we suppose, can only show something does not belong to N), to find elements that define different classes in G/N, that is: find g_{i_0} that does not belong to N, then g_{i_1} which is such that neither itself nor $g_{i_0}g_{i_1}^{-1}$ belong to N, and g_{i_2} such that g_{i_2}, $g_{i_0}g_{i_2}^{-1}$ and $g_{i_1}g_{i_2}^{-1}$ don't belong to N... If F is a quotient of G, this method will yield $card(F)$ elements, at which point the algorithm has proven that F is a quotient of G. Of course, if F is not a quotient of G, it will never stop. □

In a r.p. group, determining whether g belongs to the subgroup generated by $x_1, ..., x_k$ can always be done when g belongs to that group, thus having co-semi-decidable membership problem for finite index normal subgroups is equivalent to having solvable membership problem for finite index normal subgroups. Similarly, r.p. groups always have co-ReFQ (Proposition 3.2.1), thus for such a group ReFQ and CFQ are equivalent. This yields:

Corollary 3.1.11. *For recursively presented groups, having CFQ and having solvable membership problem for finite index normal subgroups are equivalent properties.*

We can use this to show that in a r.p. group with CFQ, from the description of a finite index normal subgroup N by a generating family $x_1, ..., x_k$, one can deduce a pair (F, f), where F is a finite group and f extends to a morphism φ of G to F with kernel N.

Launch two procedures, one is the same as that described in the proof above: enumerate elements of G that define different cosets of G/N. We get successively better lower bounds on $card(G/N)$: $card(G/N) \geq 1, 2, 3, ...$

The other procedure gives upper bounds on the size of G/N. Start from the enumeration of all marked finite groups (F_1, f_1), (F_2, f_2),... For each pair (F_i, f_i), test whether G/N is a quotient of the group F_i according to one of the finitely many left inverses of f_i. This can be done because G/N is given by a recursive presentation (as we add finitely many relations to a presentation of G which we suppose recursive), thus there is an algorithm that tests whether the finitely many relations of a finite group F are satisfied in G/N, and terminates when indeed they are. This procedure yields upper bounds on the cardinal of G/N.

At some point, the lower and upper bounds will agree, and we will know that the pair (F, f) that has $card(F) = card(G/N)$ defines an isomorphism $F \simeq G/N$, and thus the normal subgroup N is described by the pair (F, f).

3.2. Basic properties

We will now quickly establish some basic results about the three properties CFQ, ReFQ and co-ReFQ. Some propositions presented here are redundant with the general propositions about marked quotient algorithms from Chapter 2. However, note that property co-ReFQ does not have an equivalent in Chapter 2, and it is often the one for which the hereditarity properties are slightly harder to prove, and thus the redundancy with the previous chapter is unavoidable.

3.2.1. Recursively presented groups. It is easy to see that r.p. groups have co-ReFQ.

Proposition 3.2.1. *Any recursively presented group has co-ReFQ.*

Proof. Let G be a r.p. group generated by S of cardinal n. Let (F, f) be a n-marked finite group. For any relation r of G, write $r = s_1^{\alpha_1}...s_k^{\alpha_k}$ with $\alpha_i \in \{-1; 1\}$ and $s_i \in S$, we again test the equality $e = f(s_1)^{\alpha_1}...f(s_k)^{\alpha_k}$. Since we suppose G r.p., this can be carried out successively on all relations of G. If f does not extend to a morphism, a relation true in G but not in F will eventually be found. □

3.2.2. Hereditarity.

Proposition 3.2.2. *If G and H both have one of CFQ, ReFQ, co-ReFQ, then so does their free product $G * H$.*

Proof. Note that we have shown that those properties are independent of the generating family, thus we can show this using as a generating family of $G * H$ the union of a generating family of G and of one of H. The proof is then straightforward, as a function from that generating family to a finite group extends to a morphism of $G * H$ if and only if both restrictions to G and to H extend as morphisms. □

Proposition 3.2.3. *ReFQ is inherited by finite index subgroups.*

Proof. Let G be a group, and N a finite index subgroup of G. An enumeration of the finite quotients of G gives an enumeration of finite quotients of N, by restricting the homomorphisms to N. Not all quotients of N need arise this way, but add the following: whenever a pair (F, f) is found that defines a quotient of N, list all quotients of the finite group F, and when a quotient $F \xrightarrow{\pi} F_0$ is found, add $(F_0, \pi \circ f)$ to the list of quotients of N. We claim that all finite quotients of N arise this way.

Let M be a finite index normal subgroup of N. Then M is of finite index in G, it may not be normal in G, but it contains a normal subgroup M_0 which is both finite index and normal in G. Then $G \to G/M_0$ restricts to a morphism $N \to N/M_0$, and M/M_0 is a normal subgroup in N/M_0, and the quotient of N/M_0 by M/M_0 is, of course, N/M. □

Note that any countable group embeds in a two generated simple group, and that simple groups always have CFQ. This of course shows that CFQ is not inherited by subgroups. Note that the author doesn't know of a residually finite group with CFQ, with a subgroup without CFQ.

The following problem is related to the search of a Higman Embedding Theorem for residually finite groups, see Chapter 4.

Problem 3.2.4. Find a finitely presented residually finite group with a finitely generated subgroup that does not have CFQ.

The following proposition provides insight into what happens when computing the quotient of a group with CFQ.

Proposition 3.2.5. *Let G be a finitely generated group. Let H be a group obtained by adding finitely many relations and identities to G. If G has any of ReFQ, co-ReFQ or CFQ, then so does H.*

Proof. Let G and H be two finitely generated groups, with a morphism π from G onto H. Let (F, f) be a marked finite group. It is obvious that if f extends to a homomorphism φ from H onto F, then $f \circ \pi$ extends as well to a morphism ψ, which, in addition, satisfies $ker(\pi) \subseteq ker(\psi)$. On the other hand, if $f \circ \pi$ extends to a morphism ψ, and if $ker(\pi)$ is contained in $ker(\psi)$, then ψ factors through π and f will extend to a morphism. The diagram is the following:

$$\begin{tikzcd} G \arrow[r, "\pi"] \arrow[dr, "\psi"'] & H \arrow[d, "\phi"] \\ & F \end{tikzcd}$$

Thus if G has CFQ or ReFQ, we can reduce the finite quotient question of H to a question about subgroups inclusion.

If $ker(\pi)$ is finitely generated as a normal subgroup, then the question "is $ker(\pi)$ contained in $ker(\psi)$?" can be solved in finite time, as it is solved by computing $\psi(r)$ for each r in a generating family of $ker(\pi)$. If $ker(\pi)$ is generated by an identity -that is a set of relations of the form $v(x_1, ..., x_k)$, where v is a element of the free group on k generators, and $x_1, ..., x_k$ take all possible values of G^k- this question can also be answered, because to check whether an identity holds in a finite group, one only needs to check finitely many relations.

If G is a co-ReFQ group, we can determine whether $ker(\pi)$ is contained in $ker(\psi)$ even without knowing if ψ defines a morphism from G, and thus if (F, f) does not define a quotient of H, we will either prove that $f \circ \pi$ does not extend to a quotient of G, or that, even if it were to define a quotient, the inclusion of kernels would not hold. With this, all cases of Proposition 3.2.5 are covered. □

Since free groups obviously have CFQ, this proves again that finitely presented groups have CFQ, and the improvement due to Mostowski [**88**] which asserts that groups defined by finitely many relations and identities have CFQ. This implies for instance that all finitely generated metabelian groups, while not being necessarily finitely presented, have CFQ, because a result of Hall ([**52**]) implies that any finitely generated metabelian group can be presented with the metabelian identity $(\forall x \forall y \forall z \forall t\, [[x, y]\, [z, t]] = e)$ together with finitely many relations.

The following is a direct consequence of Proposition 3.2.5:

Corollary 3.2.6. *The fundamental group of a graph of groups with vertex groups that have CFQ (or ReFQ or co-ReFQ) and finitely generated edge groups also has CFQ (respectively ReFQ or co-ReFQ). This includes free products amalgamated over finitely generated subgroups and HNN extensions over finitely generated subgroups.*

A direct product of groups that have one of CFQ, ReFQ or co-ReFQ, again has that property.

3.3. Relation with the Depth Function for residually finite groups

In [**15**], Bou-Rabee introduced the *residual finiteness growth function*, or *depth function*, ρ_G, of a residually finite group G. We briefly recall its definition.

Fix a generating family S of G. Consider the function ρ_S, that to a natural number n associates the smallest number k such that, for any non-trivial element of length at most n in G, there exists a finite quotient of G of order at most k, such that the image of this element in that quotient is non-trivial. This is the depth function of G with respect to the generating family S.

For two functions f and g defined on natural numbers, note $f \succeq g$ to mean that there exists a constant C such that for any number n, $f(n) \geq Cg(Cn)$. It is easy to see that one defines an equivalence relation $\simeq$ by putting $f \simeq g$ if and only if $f \succeq g$ and $g \succeq f$.

If S and S' are two different generating families of a group G, the functions ρ_S and $\rho_{S'}$ will satisfy $\rho_S \simeq \rho_{S'}$ ([**15**]). Thus one can define uniquely the depth function ρ_G of the group G by considering the equivalence class for $\simeq$ of a function ρ_S, for some generating family S.

The interaction between the depth function of a group and it having CFQ makes it worth mentioning here, and in fact, an ancestor of the depth function can be found in McKinsey's original article, [**79**], where an upper bound for what would be a "depth function" for lattices (partially ordered sets) is computed. This interaction also appears in [**16**].

Consider a residually finite group G, that has CFQ, generated by a family S. We know that G is then co-r.e., because McKinsey's algorithm applies: list all quotients of G in order, and check wether an element has non-trivial image in one of those quotients. How long this will take is bounded by the depth function. In particular, if for an element w of length n of G, the algorithm has already tested all quotients of size at most $\rho_S(n)$, and not found a quotient in which w is non-trivial, then $w = e$.

PROPOSITION 3.3.1. *Let G be a finitely generated residually finite group with CFQ.*

If there exists a generating family S of G and a recursive function h that satisfies $\rho_S \leq h$, then G has solvable word problem.

If G has solvable word problem, then for any generating family S of G, the depth function of G with respect to S is recursive.

PROOF. The first claim follows from the sentence that immediately precedes this proposition. The second claim is straightforward. ☐

REMARK 3.3.2. The depth function ρ_G of G is defined up to the equivalence relation $\simeq$. In the statement of this last proposition appears the condition "there exists a recursive function h that satisfies $\rho_S \leq h$". If two functions f and g satisfy $f \simeq g$, and if there exists a recursive function h that satisfies $f \leq h$, then of course there exists a recursive function h' that satisfies $g \leq h'$, and thus it makes sense to say that there exists a recursive function that bounds the depth function ρ_G. Note however that it is possible for a recursive function f to be equivalent, for $\simeq$, to a non-recursive function g, (and even worse: the equivalence class for $\simeq$ of any recursive function contains a non-recursive function), and because of this, the term "recursive depth function", must be manipulated carefully. In what follows, we will say that a group G has *recursive depth function* if for any generating family S of the group G, the function ρ_S is recursive. Proposition 3.3.1 implies that for a group with CFQ, either all its relative depth functions ρ_S are recursive, or none of them are. From this it is natural to ask the following problem:

PROBLEM 3.3.3. Let G be a finitely generated residually finite group, and S and S' two generating families of G. Is it possible that exactly one of ρ_S and $\rho_{S'}$ be recursive?

The following follows immediately from Proposition 3.3.1:

COROLLARY 3.3.4. *For residually finite groups with CFQ, having solvable word problem is equivalent to having recursive depth function.*

Theorem 3.0.5 thus asserts the existence of a group with CFQ but non-recursive depth function. Note that result similar to this corollary exists, that relates the Dehn function of a finitely presented group and solvability of the word problem in it.

In [**60**], it was shown that for any recursive function f, there is a finitely presented residually finite group with depth function greater than f, and yet with word problem solvable in polynomial time. For such a group, McKinsey's algorithm is far from being optimal. The group constructed in order to prove Theorem 3.0.4 shows that, for non-finitely presented groups, the situation can be even more extreme: in it, the word problem is solvable, but not by McKinsey's algorithm.

It was explained in the first section of this chapter that conjugacy separability and subgroup separability, when confronted to the property of having computable finite quotients, play a role similar to residual finiteness, but with respect to the conjugacy problem and the generalized word problem. It is then natural to introduce functions similar to the depth function of residually finite groups, but that quantify conjugacy separability and subgroup separability. This was first done in [**65**] for conjugacy separability, and in [**30**] for subgroup separability. What was said in this section about recursiveness of the depth function translates easily to those two functions.

3.4. Examples of groups with CFQ

In this section, we give examples of infinitely presented groups that have CFQ. Those examples contain some commonly used infinitely presented groups.

3.4.1. Wreath products. The (restricted) wreath product of two groups G and H, noted $H \wr G$, is the semi direct product $H \ltimes \bigoplus_H G$, where H acts on $\bigoplus_H G$ by permuting the indices. The wreath product $H \wr G$ of two finitely generated groups is always finitely generated, however, by a theorem of Baumslag ([**5**]), it is finitely presented only if G is finitely presented and H is finite (excluding the case where G is the trivial group). If $\langle S_G|R_G\rangle$ and $\langle S_H|R_H\rangle$ are presentations respectively of G and H, with $S_G \cap S_H = \varnothing$, then a presentation of $H \wr G$ is given by the following:

$$\langle S_G, S_H|R_G, R_H, \left[hg_1h^{-1}, g_2\right], (g_1, g_2, h) \in S_G \times S_G \times (H \setminus \{1\})\rangle$$

(where by $h \in H \setminus \{1\}$ we actually mean "for each $h \in H \setminus \{1\}$, choosing a way of expressing h in terms of a product of the generators of S_H").

The following proposition gives necessary and sufficient conditions for a wreath product to have CFQ.

Proposition 3.4.1. *The wreath product $H \wr G$ of two finitely generated groups G and H has CFQ if and only if either H has CFQ and is infinite, or if H is finite and G has CFQ.*

Proof. Suppose that $H \wr G$ has CFQ. From the presentation of $H \wr G$ given above, it is easy to see that H can be obtained as a quotient of $H \wr G$ by adding only finitely many relations to it. Thus by Proposition 3.2.5, H has CFQ.

Suppose additionally that H is finite. A finite group F, marked by a function f, is a quotient of G if and only if the function g, defined as the identity on the generators of H and as f on those of G, which thus sends a generating family of $H \wr G$ to one of $H \wr F$, can be extended to a morphism. But because H is finite, $H \wr F$ is also finite, and thus it can be determined whether or not it is a quotient of $H \wr G$. This proves that G has CFQ.

For the converse, suppose first that H is finite and that G has CFQ. Because H is finite, a presentation for $H \wr G$ can be obtained by adding finitely many relations to a presentation of $G * H$, which has CFQ because both G and H have. Thus Proposition 3.2.5 applies again.

Suppose now that H is infinite and has CFQ. In a finite quotient of $H \wr G$, some non-trivial element of H must necessarily have a trivial image. But it is easy to see, from the presentation of $H \wr G$ given above, that, for any element $h \neq 1$ of H, the relation $h = 1$ together with the relations of $H \wr G$ will always also imply that $[g_1, g_2] = 1$, for any pair (g_1, g_2) of elements of G. This implies that any finite quotient of $H \wr G$ is in fact a finite quotient of $H \wr G_{ab}$, where G_{ab} denotes the abelianization $G/[G,G]$ of G. It thus suffices to prove that if H has CFQ and G is abelian, then the wreath product $H \wr G$ also has CFQ.

Consider a finite group F together with a function f that goes from a generating family of $H \wr G$ to F. To decide whether f can be extended to a morphism, one can first check whether the restriction f_H of f to a generating family of H extends as a morphism from H onto a subgroup F_H of F, using the fact that H has CFQ. If it does not, then f cannot be extended to a group morphism. If it does, one can compute a presentation of F_H. F is then a quotient of $H \wr G$ if and only if it is a quotient of $F_H \wr G$. But G is finitely presented, being abelian, and F_H is finite, thus this wreath product is finitely presented, and a finite presentation of it can be obtained from the presentation of F_H. This presentation can in turn be used to decide whether F is a quotient of $F_H \wr G$, and thus of $H \wr G$.

(Note that this last part of the proof is non-constructive: we showed that $H \wr G$ admits a finite quotient algorithm because $H \wr G_{ab}$ does, and that $H \wr G_{ab}$ had CFQ because G_{ab} is finitely presented, but we do not claim that a finite presentation for G_{ab} can be effectively found, and thus the existence of an algorithm that recognizes the quotients of $H \wr G$ remains abstract.) □

3.4.2. L-presented groups. L-presented groups were introduced in [**3**]. An L-presentation, or finite endomorphic presentation, is a quadruple $\langle S|R_1|\Phi|R_2\rangle$, where S is a finite set of generating symbols, R_1 and R_2 are finite sets of relations, i.e. of elements of the free group $\mathcal{F}_S$ defined over S, and Φ is a finite set of endomorphisms of $\mathcal{F}_S$. Such a presentation defines a presentation, in the usual sense of the term, by adding to $R_1 \cup R_2$, as further relations, all elements of $\mathcal{F}_S$ that can be obtained from R_2 by iterating the endomorphisms of Φ. (This process can be carried out effectively, an L-presentation is thus always a recursively enumerable presentation.)

It was proven in [**53**] that a coset enumeration process can be carried out in L-presented groups, and that the membership problem for finite index subgroups is solvable in L-presented groups. This, we have seen, directly implies that those groups have CFQ.

Proposition 3.4.2 ([**53**]). *L-presented groups have CFQ.*

Many groups of interest that are not finitely presented were proven to admit L-presentations, including groups of intermediate growth, and various groups acting on rooted trees, like the Gupta–Sidki group (see [**3**]).

Note that many of those groups were proven to be conjugacy separable ([**130**]) or subgroup separable ([**47, 42**]). The fact that they have CFQ thus completes for those groups the proof of solvability of the conjugacy problem, or of the generalized word problem. (Although in some cases, solutions to these problems are known, that do not rely on McKinsey's algorithm).

3.5. Main unsolvability results

In this section, we prove the two main theorems of this chapter.

Theorem 3.0.5 follows from the construction that Slobodskoi used in [**119**] to prove undecidability of the universal theory of finite groups. However, the exact result we use was implicit in [**119**], and it was first pointed out to hold by Bridson and Wilton in [**19**]. We first show how, thanks to this result, the proof of Theorem 3.0.5 is immediate.

We then detail Dyson's construction from [**35**], which, with [**80**], contained the first examples of r.p. residually finite groups without solvable word problem. We show that this construction provides us with another proof of Theorem 3.0.5, which has the advantage of relying on groups whose structure is very explicit. We finally use Dyson's construction to build a residually finite group with solvable word problem, and yet without CFQ, thus proving Theorem 3.0.4.

3.5.1. Slobodskoi's example for Theorem 3.0.5. Given a finitely generated group G, its finitary image G_f is its biggest residually finite quotient, that is to say the quotient of G by the intersection of all its finite index normal subgroups. This definition corresponds to the notion of *residually finite image* introduced in Section 2.4 in the previous chapter, however it seems that the finitary image is a more common name for this object.

To each finite presentation, one can associate a residually finite group, by taking the finitary image of the group defined by this presentation. If π is a finite presentation for a group G, and if G_f is the finitary image of G, we say that π is a finite presentation of G_f *as a residually finite group,* and that G_f is *finitely presentable as a residually finite group.*

In 1981, in [**119**], Slobodskoi proved that the universal theory of finite groups is undecidable. As a consequence of the results stated in his paper, groups that are finitely presented as residually finite groups do not have uniformly solvable word problem. But in fact, more can be deduced from the proof that appears in [**119**], as was made explicit by Bridson and Wilton in [**19**]:

Theorem 3.5.1 ([**19**], Theorem 2.1). *There exists a finitely presented group G in which there is no algorithm to decide which elements have trivial image in every finite quotient.*

Two other equivalent formulations of this theorem are the following ones:

Theorem. *There exists a finitely presented group G whose finitary image is not recursively presented.*

Theorem. *There exists a group H which is finitely presentable as a residually finite group, but which has unsolvable word problem.*

This theorem then directly implies Theorem 3.0.5.

Proof of Theorem 3.0.5. A group which is finitely presented as a residually finite group must have CFQ, since it has the same finite quotients as a finitely presented group. Thus if such a group has unsolvable word problem, it satisfies the criteria required by Theorem 3.0.5. A group, finitely presented as a residually finite group, with unsolvable word problem exists by Theorem 3.5.1. ☐

3.5.2. Dyson's Groups. These groups are amalgamated products of two lamplighter groups.

The lamplighter group L is the wreath product $\mathbb{Z}\wr\mathbb{Z}/2\mathbb{Z}$ of $\mathbb{Z}$ and of the order two group. It admits the following presentation:

$$\langle a,\varepsilon \,|\, \varepsilon^2,\ \left[\varepsilon, a^i\varepsilon a^{-i}\right], i\in\mathbb{Z}\rangle$$

The element $a^i\varepsilon a^{-i}$ of L corresponds to the element of $\bigoplus_{\mathbb{Z}}\mathbb{Z}/2\mathbb{Z}$ with only one non-zero coordinate in position $i\in\mathbb{Z}$. We call it u_i. Consider another copy $\hat{L}$ of the lamplighter group, together with an isomorphism from L to $\hat{L}$ we note $g \mapsto \hat{g}$. For each subset $\mathcal{A}$ of $\mathbb{Z}$, define $L(\mathcal{A})$ to be the amalgamated product of L and $\hat{L}$, with $u_i = a^i \varepsilon a^{-i}$ identified with $\hat{u}_i = \hat{a}^i \hat{\varepsilon} \hat{a}^{-i}$ for each i in $\mathcal{A}$. It has the following presentation:

$$\langle a, \hat{a}, \varepsilon, \hat{\varepsilon} |\, \varepsilon^2,\, \hat{\varepsilon}^2,\, \left[\varepsilon, a^i \varepsilon a^{-i}\right], \left[\hat{\varepsilon}, \hat{a}^i \hat{\varepsilon} \hat{a}^{-i}\right], i \in \mathbb{Z},\, a^j \varepsilon a^{-j} = \hat{a}^j \hat{\varepsilon} \hat{a}^{-j},\, j \in \mathcal{A} \rangle$$

Recall that the *double* of a group G over a subgroup H is the free product of two copies of G amalgamated along the identity over H, denote it $G *_H G$. Dyson's group associated to $\mathcal{A}$ is thus the double of the Lamplighter group L over the subgroup generated by elements of the form $a^i \varepsilon a^{-i}$, for i in $\mathcal{A}$. Call $H_{\mathcal{A}}$ this subgroup.

For n a non-zero natural number, denote $L(\mathcal{A})_n$ the group $\langle L(\mathcal{A}) \,|\, a^n, \hat{a}^n \rangle$.

Denote by $\mathcal{A} \, mod\, n$ the set $\{r \in \{0, ..., n-1\}, \exists a \in \mathcal{A}, a \equiv r \, mod \, n\}$.

LEMMA 3.5.2. $L(\mathcal{A})_n$ is finitely presented and residually finite. It is the amalgamated product of two copies of the finite wreath product $\mathbb{Z}/n\mathbb{Z} \wr \mathbb{Z}/2\mathbb{Z}$, and it admits the following presentation:

$$\begin{aligned}\langle a, \hat{a}, \varepsilon, \hat{\varepsilon} |\, a^n,\, \hat{a}^n,\, \varepsilon^2,\, \hat{\varepsilon}^2,\, \left[\varepsilon, a^i \varepsilon a^{-i}\right], \left[\hat{\varepsilon}, \hat{a}^i \hat{\varepsilon} \hat{a}^{-i}\right], 0 \leq i \leq n-1,&\\ a^j \varepsilon a^{-j} = \hat{a}^j \hat{\varepsilon} \hat{a}^{-j},\, j \in \mathcal{A} \, mod \, n \rangle&\end{aligned}$$

PROOF. The given presentation is obtained from the presentation of $L(\mathcal{A})$, adding relations a^n and $\hat{a}^n$, and simplifying the relations as can be done. It then follows from that presentation that $L(\mathcal{A})_n$ is an amalgamated product of two copies of the finite wreath product $\mathbb{Z}/n\mathbb{Z} \wr \mathbb{Z}/2\mathbb{Z}$. Finally it is well known that an amalgamated product of finite groups is residually finite (see [**69**]). □

The properties of the group $L(\mathcal{A})$ are stated using the profinite topology on $\mathbb{Z}$. A basis of open sets for $\mathcal{PT}(\mathbb{Z})$ consists in sets of the form $n + p\mathbb{Z}$, for n and p integers. Thus a subset $\mathcal{A}$ of $\mathbb{Z}$ is open in $\mathcal{PT}(\mathbb{Z})$ if and only if for every n in $\mathcal{A}$ there exists an integer p such that $n + p\mathbb{Z} \subseteq \mathcal{A}$.

We can now state the properties of the group $L(\mathcal{A})$ that are relevant to our present study.

PROPOSITION 3.5.3. *Let $\mathcal{A}$ be a subset of $\mathbb{Z}$.*

(1) *$L(\mathcal{A})$ is r.p., co-r.p. or has solvable word problem if and only if $\mathcal{A}$ is respectively r.e., co-r.e. or recursive.*
(2) *$L(\mathcal{A})$ is residually finite if and only if $\mathcal{A}$ is closed in $\mathcal{PT}(\mathbb{Z})$.*
(3) *$L(\mathcal{A})$ has CFQ if and only if the function which to n associates $\mathcal{A} \, mod \, n$ is recursive.*

The first two points of this proposition were proven in [**35**]. We still include proofs for those two statements. The proof given for (1) is exactly that of Dyson. We then give two proofs of (2), one shorter and more conceptual than that of [**35**], which was suggested to us by an anonymous referee, and another one which, although not as clear as the previous one, can readily be rendered effective. In Chapter 4, in the proof of Proposition 4.1.1, we refer ourselves to this second proof.

Notice finally that the condition which appears in (3) could be stated as: $\mathcal{A}$ is determinable in finite quotients of $\mathbb{Z}$ (see Definition 3.1.3).

PROOF. We prove all three points in order.

If $L(\mathcal{A})$ is r.p. or co-r.p., then clearly $\mathcal{A}$ is r.e. or co-r.e., as n belongs to $\mathcal{A}$ if and only if $u_n = \hat{u}_n$ in $L(\mathcal{A})$, which proves one direction of (1).

It is clear that if $\mathcal{A}$ is r.e., the presentation of $L(\mathcal{A})$ given above is a recursive presentation.

Suppose now that $\mathcal{A}$ is co-r.e.. We can enumerate the complement of $\mathcal{A}$, and thus enumerate elements of the form:

$$(*) \quad w = a^{\alpha_1} x_1 \hat{a}^{\beta_1} y_1 ... a^{\alpha_k} x_k \hat{a}^{\beta_k} y_k z$$

where α_i, $\beta_i \in \mathbb{Z}$, x_1, ..., x_k and y_1, ..., y_k are elements of the base groups of L and $\hat{L}$ that have null components corresponding to indices in $\mathcal{A}$, and z is any element in L or in $\hat{L}$. The elements written this way are exactly the elements in normal form for the amalgamated product $L(\mathcal{A})$. Recall that the normal form in an amalgamated product ([**68**]) necessitates two choices of transversals, here one for $L/(\bigoplus_{\mathcal{A}} \mathbb{Z}/2\mathbb{Z})$ and one for $\hat{L}/(\bigoplus_{\mathcal{A}} \mathbb{Z}/2\mathbb{Z})$, and that an element in normal form is a consecutive product of an element of one transversal, followed by one of the other, etc, terminated by any element of one of these groups. But elements $a^\alpha x$, with α in $\mathbb{Z}$ and x in the base group with null components on $\mathcal{A}$, indeed form the most natural transversal for $L/(\bigoplus_{\mathcal{A}} \mathbb{Z}/2\mathbb{Z})$. Thus any non-trivial element g of $L(\mathcal{A})$ is equal to exactly one element in this enumeration. Ideally, we would then enumerate words that give the identity in $L(\mathcal{A})$, and listing words that can be obtained concatenating a word in normal form to a word that defines the identity would give the desired enumeration. Since $L(\mathcal{A})$ is not supposed r.e., we cannot directly enumerate this set of trivial words, but we will over-approximate it by a r.e. set. For w as in $(*)$, note B_w the set of all indices that appear in elements x_i or y_i of the base groups (B_w can be empty). The over approximation consists of the words corresponding to the identity in $L(\mathbb{Z}\setminus B_w)$. Note that w is also in normal form in $L(\mathbb{Z}\setminus B_w)$, thus non-trivial there. Of course, B_w is finite, thus $\mathbb{Z}\setminus B_w$ is r.e., thus as we already remarked, we can enumerate words (in a, $\hat{a}$, ε, $\hat{\varepsilon}$) that correspond to the identity in $L(\mathbb{Z}\setminus B_w)$. Then for any such word w_1, the product ww_1 corresponds to a non-identity element of $L(\mathbb{Z}\setminus B_w)$, thus to a non-identity element in $L(\mathcal{A})$, since, as $\mathcal{A}\subseteq(\mathbb{Z}\setminus B_w)$, $L(\mathbb{Z}\setminus B_w)$ satisfies more relations than $L(\mathcal{A})$. Thus enumerating products ww_1 with $w_1=e$ in $L(\mathbb{Z}\setminus B_w)$ will only yield non-identity elements in $L(\mathcal{A})$. What's more, every element of the form ww_2, where w_2 is a word that is the identity in $L(\mathcal{A})$, will arise this way, again because $L(\mathbb{Z}\setminus B_w)$ satisfies more relations than $L(\mathcal{A})$.

Because the algorithm that enumerates relations in $L(\mathbb{Z}\setminus B_w)$ depends recursively on w, this process can be applied simultaneously to all words w in normal form, giving an enumeration of all words that correspond to non-identity elements of $L(\mathcal{A})$, thus proving that $L(\mathcal{A})$ is co-r.p..

Finally $\mathcal{A}$ is recursive if and only if it is both r.e. and co-r.e., if and only if $L(\mathcal{A})$ has solvable word problem.

This ends the proof of (1).

We first sketch a conceptually simple proof of (2).

Recall that the group $L(\mathcal{A})$ is the double of the lamplighter group L over the subgroup $H_{\mathcal{A}}$ generated by elements of the form $a^i\varepsilon a^{-i}$, for i in $\mathcal{A}$. The statement (2) is implied by the two following arguments:

- the double of a residually finite group G over a subgroup H is itself residually finite if and only if H is separable in G;
- the subgroup $H_{\mathcal{A}}$ is separable in L if and only if $\mathcal{A}$ is separable in $\mathbb{Z}$.

Those arguments are easy to check, and because a second proof for (2) follows, we do not detail them.

We now give another proof of (2), which can be easily rendered effective. We in fact prove slightly more than (2): for a subset $\mathcal{A}$ of $\mathbb{Z}$, the finitary image $L(\mathcal{A})_f$ of $L(\mathcal{A})$ is $L(\overline{\mathcal{A}})$, where $\overline{\mathcal{A}}$ denotes the closure of $\mathcal{A}$ in $\mathcal{PT}(\mathbb{Z})$. First we show that if n belongs to $\overline{\mathcal{A}}$, then in any finite quotient (F,f) of $L(\mathcal{A})$, the images of u_n and $\hat{u}_n$ are the same in F. Let (F,f) be some finite quotient of $L(\mathcal{A})$. Call p and p' the orders of $f(a)$ and $f(\hat{a})$ in F. Then, because n belongs to $\overline{\mathcal{A}}$, $n+pp'\mathbb{Z}$ must meet with $\mathcal{A}$, as it is a neighborhood of n. Thus we have k such that $n+pp'k\in\mathcal{A}$, that is, such that $u_{n+pp'k}=\hat{u}_{n+pp'k}$ in $L(\mathcal{A})$. Then, in F (we omit to write the homomorphism onto F):

$$\begin{aligned} u_n &= (a^p)^{p'k}u_n(a^p)^{-p'k} = a^{pp'k}u_na^{-pp'k}\\ &= u_{n+pp'k}\\ &= \hat{u}_{n+pp'k}\\ &= (\hat{a}^{p'})^{pk}\hat{u}_n(\hat{a}^{p'})^{-pk} = \hat{u}_n \end{aligned}$$

This shows that $L(\mathcal{A})_f$ is a quotient of $L(\overline{\mathcal{A}})$. It is then sufficient to see that $L(\overline{\mathcal{A}})$ is residually finite to see that $L(\mathcal{A})_f = L(\overline{\mathcal{A}})$.

We suppose that $\mathcal{A}$ is closed to omit the closure notation. Let w be a non-identity element of $L(\mathcal{A})$, and write it in normal form $w=a^{\alpha_1}x_1\hat{a}^{\beta_1}y_1...a^{\alpha_k}x_k\hat{a}^{\beta_k}y_kz$, as in the proof of (1).

Suppose first that the normal form is the trivial one: $w=z$ with z in L or in $\hat{L}$. Then w is non-trivial in the quotient of $L(\mathcal{A})$ obtained by identifying the two copies L and $\hat{L}$ of the lamplighter group (i.e. $\langle L(\mathcal{A})\,|\,a=\hat{a},\varepsilon=\hat{\varepsilon}\rangle$), which is just the lamplighter group itself, which is residually finite.

We can now suppose the normal form has several terms. Each x_i is an element of $\bigoplus_{\mathbb{Z}\setminus\mathcal{A}}\mathbb{Z}/2\mathbb{Z}$, that is to say a product $\prod u_{k_{i,j}}$ with $k_{i,j}\notin\mathcal{A}$. Because $\mathcal{A}$ is closed, for each such $k_{i,j}$ there is $p_{i,j}$ that satisfies $(k_{i,j}+p_{i,j}\mathbb{Z})\cap\mathcal{A}=\varnothing$. Similarly, for each $\hat{x}_i$, introduce integers $p'_{i,j}$, $j=1,2,....$ Call N the product $\prod p_{i,j}\prod p'_{i,j}$. (It is 1 if, for all i, $x_i=\hat{x}_i=e$.) We claim that w is non-trivial in $L(\mathcal{A})_N$. Indeed, N was chosen so that for any (i,j), $k_{i,j}$ (or its remainder modulo N) does not belong to $\mathcal{A}\,mod\,N$. This implies that w is also in normal form in $L(\mathcal{A})_N$ (by Lemma 3.5.2), and thus non-trivial there.

Again by Lemma 3.5.2, $L(\mathcal{A})_N$ is residually finite, so we've proven that $L(\mathcal{A})$ is residually residually finite, which of course is the same as residually finite.

Finally we prove (3).

Suppose $\mathcal{A}\,mod\,n$ depends recursively of n. Let (F,f) be a finite group together with a function f from $\{a,\,\hat{a},\,\varepsilon,\,\hat{\varepsilon}\}$ to F. To determine whether f defines a homomorphism, compute the orders of $f(a)$ and of $f(\hat{a})$, and let n be their product. If f extends to a morphism, this morphism factors through the projection $\pi: L(\mathcal{A})\to L(\mathcal{A})_n$. By Lemma 3.5.2, a finite presentation for $L(\mathcal{A})_n$ can be found from the computation of $\mathcal{A}\,mod\,n$. It can then be determined in finite time from this presentation whether f defines a homomorphism from $L(\mathcal{A})_n$ to F.

Suppose now that $L(\mathcal{A})$ has CFQ. Let n be a natural number. To compute $\mathcal{A}\,mod\,n$, consider all possible presentations for $L(\mathcal{A})_n$: for $B \subset \{0, ..., n-1\}$, define the presentation $\prod_B$:

$$\langle a, \hat{a}, \varepsilon, \hat{\varepsilon} |\, a^n,\, \hat{a}^n,\, \varepsilon^2,\, \hat{\varepsilon}^2,\, \left[\varepsilon, a^i \varepsilon a^{-i}\right], \left[\hat{\varepsilon}, \hat{a}^i \hat{\varepsilon} \hat{a}^{-i}\right], 0 \leq i \leq n-1,$$
$$a^j \varepsilon a^{-j} = \hat{a}^j \hat{\varepsilon} \hat{a}^{-j},\, j \in B\rangle$$

All these presentations define residually finite groups that, being finitely presented, have CFQ. $L(\mathcal{A})_n$ also has CFQ, because it is obtained from $L(\mathcal{A})$ by adding two relations, and thus we can enumerate the quotients of $L(\mathcal{A})_n$. Also enumerate the quotients of all groups given by the presentations $\prod_B$, for $B \subset \{0, ..., n-1\}$. Those 2^n lists are all different (because, as the presentations $\prod_B$ give residually finite groups, a list contains a finite group in which the images of $a^j \varepsilon a^{-j}$ and $\hat{a}^j \hat{\varepsilon} \hat{a}^{-j}$ differ if and only if j does not belong to B), and only one corresponds to the list of quotients of $L(\mathcal{A})_n$. It can be determined, in a finite number of steps, which of those lists corresponds to $L(\mathcal{A})_n$, and thus which presentation $\prod_B$ gives a presentation of $L(\mathcal{A})_n$, and then one can conclude that $B = \mathcal{A}\,mod\,n$. □

From Proposition 3.5.3, to prove Theorem 3.0.4, it suffices to build $\mathcal{A}$ with the following properties: $\mathcal{A}$ is closed in $\mathcal{PT}(\mathbb{Z})$, $\mathcal{A}$ is recursive, there is no algorithm that takes n as input and computes $\mathcal{A}\,mod\,n$. Similarly, to prove Theorem 3.0.5, it suffices to build $\mathcal{A}$ such that: $\mathcal{A}$ is closed in $\mathcal{PT}(\mathbb{Z})$, $\mathcal{A}$ is not recursive, but there is an algorithm that, given n as input, computes $\mathcal{A}\,mod\,n$.

3.5.3. Building subsets of $\mathbb{Z}$ with prescribed properties. We first give an alternative proof of Theorem 3.0.5, before completing the proof of Theorem 3.0.4.

LEMMA 3.5.4. *There exists a non-recursive subset $\mathcal{A}_1$ of $\mathbb{Z}$, closed in $\mathcal{PT}(\mathbb{Z})$, for which $\mathcal{A}_1\,mod\,n$ depends recursively of n.*

Note that without the closeness assumption, this result would be a lot easier: it is precisely the result of Proposition 3.1.5, whose proof is very short, and yields a set that can be neither r.e. nor co-r.e.. However, for a closed set $\mathcal{A}$, the computation of $\mathcal{A}\,mod\,n$ will yield an enumeration of the complement of $\mathcal{A}$: indeed, if a is not in $\mathcal{A}$, some open set $a + b\mathbb{Z}$ must not meet $\mathcal{A}$, and thus a is not in $\mathcal{A}\,mod\,b$. This proves that if $\mathcal{A}$ is closed, and if $\mathcal{A}\,mod\,n$ is computable, then $\mathcal{A}$ is co-r.e.. This is just the translation for Dyson's groups of: if G is residually finite, and has CFQ, then G is co-r.p..

PROOF. We construct a set $\mathcal{B}$, which will be the complement of the announced $\mathcal{A}_1$. Thus it has to be open, r.e. but not co-r.e., and for any a and b, the question "is $a + b\mathbb{Z}$ a subset of $\mathcal{B}$" has to be solvable in a finite number of steps. Indeed, a belongs to $\mathcal{A}_1\,mod\,b$ if and only if $a + b\mathbb{Z}$ meets $\mathcal{A}_1$, if and only if $a + b\mathbb{Z}$ is not a subset of the complement of $\mathcal{A}_1$.

Call p_n the n-th prime number. Define two sequences $(x_n)_{n \geq 0}$ and $(y_n)_{n \geq 1}$ by the following:

$$x_0 = 1$$
$$x_n = p_n x_{n-1}^2$$
$$y_n = x_{n-1}$$

These sequences have the following properties:

- for any n, $x_n | x_{n+1}$ and $y_n | y_{n+1}$.
- for any integer b, there is some (computable) n such that $b | x_n$ and $b | y_n$.
- p_k divides x_n if and only if $k \geq n$, and p_k divides y_n if and only if $k > n$.
- for integers k, k', n, n', with $k \leq n$ and $k' \leq n'$, $y_k + x_n\mathbb{Z}$ and $y_{k'} + x_{n'}\mathbb{Z}$ are disjoint if and only if $k \neq k'$, and otherwise one is a subset of the other.

All these are clear, the fourth point follows from the third, by remarking that elements of $y_k + x_n\mathbb{Z}$ are all multiples of p_0, p_1,... p_{k-1}, but none of them is a multiple of p_k.

Consider a recursive function f whose image is r.e. but not co-r.e.. Assume that for any n, $1 \leq f(n) \leq n$ (it is easy to see that such a function exists). We then define $\mathcal{B}$ as the union:

$$\mathcal{B} = \bigcup_{n \in \mathbb{N}^*} y_{f(n)} + x_n\mathbb{Z}$$

Since f is a recursive function, $\mathcal{B}$ is r.e.. It is not co-r.e., however, because y_m belongs to $\mathcal{B}$ if and only if m belongs to the image of f (this follows directly from the properties of the sequences $(x_n)_{n \geq 0}$ and $(y_n)_{n \geq 1}$).

$\mathcal{B}$ is an open set, because it is defined as an union of open sets.

All that is left to see is that we can decide, for a and b integers, whether or not $a + b\mathbb{Z}$ is a subset of $\mathcal{B}$. Suppose that $a < b$. If $a = 0$, then $0 \in a + b\mathbb{Z}$, but $0 \notin \mathcal{B}$, thus $a + b\mathbb{Z}$ is not a subset of $\mathcal{B}$. If a is non-zero, no element of

$a + b\mathbb{Z}$ is divisible by b. Thus, because of the second property of the sequences $(x_n)_{n\geq 0}$ and $(y_n)_{n\geq 1}$ quoted above, there exists N such that if $N \leq k \leq n$, then $a + b\mathbb{Z} \cap y_k + x_n\mathbb{Z} = \varnothing$. Thus $a + b\mathbb{Z}$ is a subset of $\mathcal{B}$ if and only if it is a subset of the set $\mathcal{B}_N$, defined by:

$$\mathcal{B}_N = \bigcup_{n\in\mathbb{N}^*,\, f(n)\leq N} y_{f(n)} + x_n\mathbb{Z}$$

Define a pseudo-inverse g of f by $g(m) = \inf\{n,\, f(n) = m\}$. Because we chose f such that for any n, $f(n) \leq n$, for any m, $g(m) \geq m$. If m is not in the image of f, put $g(m) = \infty$. The set $\mathcal{B}_N$ can be expressed as the disjoint union:

$$\mathcal{B}_N = \bigcup_{k\in Im(f),\, k\leq N} y_k + x_{g(k)}\mathbb{Z}$$

Because $x_k | x_{g(k)}$, $\mathcal{B}_N$ is contained in the set $\mathcal{C}_N$, defined by:

$$\mathcal{C}_N = \bigcup_{k\leq N} y_k + x_k\mathbb{Z}$$

It can be determined whether $a + b\mathbb{Z}$ is contained in $\mathcal{C}_N$, because the sequences $(x_n)_{n\geq 1}$ and $(y_n)_{n\geq 1}$ can be computed. If $a + b\mathbb{Z}$ is not contained in $\mathcal{C}_N$, then it is not contained in $\mathcal{B}_N$ either.

If it is contained in $\mathcal{C}_N$, $a + b\mathbb{Z}$ is contained in $\mathcal{B}_N$ if and only if, for each k, $a + b\mathbb{Z} \cap y_k + x_k\mathbb{Z}$ is contained in $\mathcal{B}_N$. But, because $\mathcal{B}_N$ and $\mathcal{C}_N$ are disjoint unions, $a + b\mathbb{Z} \cap y_k + x_k\mathbb{Z}$ is contained in $\mathcal{B}_N$ if and only if it is contained in $y_k + x_{g(k)}\mathbb{Z}$. (If k is not in $Im(f)$, $g(k) = \infty$, by convention $y_k + x_{g(k)}\mathbb{Z} = \{y_k\}$.) Now this question can be effectively answered. If $a + b\mathbb{Z} \cap y_k + x_k\mathbb{Z}$ is empty, there is nothing to do. Otherwise, it is of the form $t + \mathrm{lcm}(b,\, x_k)\mathbb{Z}$. Enumerate $f(1)$, $f(2)$,..., $f(\mathrm{lcm}(b, x_k))$. Either k is in that list, in which case $g(k)$ can be computed and the question "is $a + b\mathbb{Z} \cap y_k + x_k\mathbb{Z}$ contained in $y_k + x_{g(k)}\mathbb{Z}$" can be settled, or k does not appear in the enumeration, which shows that $g(k)$ is greater than $\mathrm{lcm}(b,\, x_k)$. It this last case, as $x_{g(k)}$ is greater than $g(k)$, $y_k + x_{g(k)}\mathbb{Z}$ cannot contain any set of the form $t + \mathrm{lcm}(b,\, x_k)\mathbb{Z}$. □

This Lemma allows us to give another proof of Theorem 3.0.5:

Proof of Theorem 3.0.5, alternate. It follows from Proposition 3.5.3 that the group $L(\mathcal{A}_1)$, where $\mathcal{A}_1$ is the set constructed in Lemma 3.5.4, satisfies the requirements of Theorem 3.0.5. □

This group has a depth function which cannot be smaller than a recursive function. We now prove the last lemma which ends the proof of Theorem 3.0.4:

Lemma 3.5.5. *There exists a recursive subset $\mathcal{A}_2$ of $\mathbb{Z}$, closed in $\mathcal{PT}(\mathbb{Z})$, for which $\mathcal{A}_2 \, mod \, n$ does not depend recursively of n.*

Proof. Call p_n the n-th prime number. Fix some effective enumeration M_1, M_2,... of all Turing machines. Consider the following process: start running simultaneously all those machines, as is done to show that the halting problem is r.e.. While running calculations on the n-th machine, at each new step in the computation, produce a new power of p_{2n}: p_{2n}, p_{2n}^2, p_{2n}^3... If the computation on this machine stops after k steps, end the list p_{2n}, p_{2n}^2, ..., p_{2n}^k already produced with p_{2n+1}^{k+1}.

Call $\mathcal{A}_2$ the set of all powers of prime numbers obtained this way. $\mathcal{A}_2$ is obviously r.e., as it was defined by an effective enumeration process. It is even recursive. Indeed, for a number x, if x is not the power of a prime, then x is not in $\mathcal{A}_2$. If it is the power of a prime of even index, say $x = p_{2n}^k$, then x belongs to $\mathcal{A}_2$ if and only if the n-th Turing machine does not stop in less than k calculations steps. This question can be effectively settled. Similarly, if x is the power of a prime of odd index, $x = p_{2n+1}^k$, then x belongs to $\mathcal{A}_2$ if and only if the n-th Turing machine stops in exactly k calculations steps, this also can be determined.

Of course, $\mathcal{A}_2 \, mod \, m$ does not depend recursively of m. Indeed, the question: "does 0 belong to $\mathcal{A}_2 \, mod \, p_{2n+1}$?" is, by construction, equivalent to "does the n-th Turing machine halt?".

Finally, we show that $\mathcal{A}_2$ is a closed set, which is equivalent to finding, for any x not in $\mathcal{A}_2$, a number y such that $x + y\mathbb{Z}$ does not meet $\mathcal{A}_2$. If x has several prime divisors, then $x + x\mathbb{Z}$ works, because any element of it has several prime divisors. If x is the power of a prime of even index, $x = p_{2n}^k$, and x is not in $\mathcal{A}_2$, it must be that the n-th Turing machine stops in strictly less than k steps. Thus the only elements in $\mathcal{A}_2$ that are multiples of p_{2n} will have a valuation in p_{2n} lower than k. Thus $x + x\mathbb{Z}$ will also work. The last case is if x is the power of a prime of odd index, $x = p_{2n+1}^k$. In this case, we claim that $x + p_{2n+1}x\mathbb{Z}$ does not meet $\mathcal{A}_2$. Indeed, x is the only power of p_{2n+1} contained in $x + p_{2n+1}x\mathbb{Z}$, all other elements of it have at least two different prime divisors. □

With this we end the proof of Theorem 3.0.4:

Proof of Theorem 3.0.4. By Proposition 3.5.3, the group $L(\mathcal{A}_2)$ defined thanks to the subset $\mathcal{A}_2$ given by Lemma 3.5.5 satisfies the requirements of Theorem 3.0.4. $\square$

CHAPTER 4

Obstruction to a Higman embedding theorem for residually finite groups with solvable word problem

Introduction

In the previous chapter, we have investigated generalizations of McKinsey's Algorithm (from [**79**]) to infinitely presented groups.

Our original motivation for studying McKinsey's Algorithm was its relation to the Higman Embedding Theorem for residually finite groups. Indeed, while all finitely presented residually finite groups have solvable word problem, several recursively presented residually finite groups are known, that fail to have solvable word problem: for instance one example was constructed by Meskin in [**80**], and one by Dyson in [**35**].

Because solvability of the word problem is inherited by subgroups, recursively presented residually finite groups without solvable word problem cannot embed inside finitely presented residually finite groups.

A question of paramount importance is the search for necessary and sufficient conditions for a residually finite group to embed in a finitely presented one.

In [**60**], Kharlampovich, Myasnikov and Sapir asked whether "unsolvability of the word problem is the only obstacle" to embed recursively presented residually finite groups into finitely presented residually finite groups.

We will prove here the following:

Theorem 4.0.1. *There exists a finitely generated residually finite group with solvable word problem, that does not embed in any finitely presented residually finite group.*

The condition of having computable finite quotients introduced in the previous chapter is not inherited by subgroups, and so it is not helpful to prove this theorem: a priori, nothing prevents a finitely presented residually finite group to have a subgroup that has solvable word problem but that fails to have CFQ (although we have failed to producing an example of this phenomenon).

Here, we focus on a weaker condition than CFQ, that has the advantage of being inherited by finitely generated subgroups.

Call a marked group G *effectively residually finite* if there is an algorithm that takes a word w on the generators of G as input, which satisfies that $w \neq e$ in G, and produces a morphism φ from G to a finite marked group F that satisfies $\varphi(w) \neq e$. (As in the definition of property CFQ, the finite group is described for instance by its Cayley table, and the morphism is given the images of the generators of G.)

This property satisfies obviously the following:

Fact 4.0.2. *A finitely generated subgroup of an effectively residually finite group is itself effectively residually finite.*

This result is completed by the following easy consequence of McKinsey's algorithm:

Fact 4.0.3. *A residually finite group with CFQ, and thus any finitely presented residually finite group, is effectively residually finite.*

Those two facts together yield that a finitely generated subgroup of a finitely presented residually finite group is effectively residually finite. And thus the main theorem that we prove here and which implies Theorem 4.0.1 is the following:

Theorem 4.0.4. *There exists a finitely generated residually finite group with solvable word problem, that is not effectively residually finite.*

The proof again relies on Dyson's groups, we find necessary and sufficient conditions on a subset $\mathcal{A}$ of $\mathbb{Z}$ for Dyson's group $L(\mathcal{A})$ to be effectively residually finite.

Of course, having CFQ and being effectively residually finite are related properties.

Indeed, notice that a residually finite group G generated by a family S has CFQ if and only if there exists an effective enumeration of *all* morphisms of marked groups from (G,S) onto finite groups. In this case, G is automatically co-r.p..

The marked group (G,S) is effectively residually finite and co-r.p. if and only if there exists some effective enumeration of morphisms onto finite groups:

$$\phi_1 : (G,S) \to (F_1,S_1),\ \phi_2 : (G,S) \to (F_2,S_2),\ \phi_3 : (G,S) \to (F_3,S_3),\ ...$$

which contains enough morphisms so that

$$\bigcap_{n\in\mathbb{N}^*} \ker(\phi_i) = \{e\}.$$

While those definitions are close, we prove here that the two defined notions do differ:

THEOREM 4.0.5. *There exists a residually finite group with solvable word problem that is effectively residually finite but that fails to have CFQ.*

In fact, Dyson's group $L(\mathcal{A}_2)$ associated to the subset of $\mathbb{Z}$ built in Lemma 3.5.5 of the previous chapter satisfies the hypotheses of this theorem. (This result thus shows that the group $L(\mathcal{A}_2)$ could not have been used to prove Theorem 4.0.4, and that we indeed need a new construction.)

Note that we will give in Chapter 5 an interpretation of the definition "being effectively residually finite" in terms of convergence in the space of marked groups: a group is co-r.p. and effectively residually finite if and only if it is the limit of an effective sequence of its finite quotients. This is detailed in Example 5.4.7.

In the final section of this chapter, we relate our results to the depth function of the constructed groups, and we show:

THEOREM 4.0.6. *There exists a residually finite group with solvable word problem whose depth function grows faster than any recursive function.*

This chapter is organized as follows: first, we give necessary and sufficient conditions for Dyson's groups to be effectively residually finite, and prove that the group $L(\mathcal{A}_2)$ introduced in the previous chapter indeed was effectively residually finite. In a second section, we build a subset $\mathcal{A}_3$ of $\mathbb{Z}$ that satisfies the properties required to prove Theorem 4.0.1. Our last section concerns the growth of the depth function.

4.1. Dyson's Groups and Effective residual finiteness

We use Dyson's groups with the same notations as those of the previous chapter: for $\mathcal{A} \subseteq \mathbb{Z}$, the group $L(\mathcal{A})$ is an amalgamated product of two lamplighter groups.

Recall that a set $\mathcal{A}$ is open in the topology $\mathcal{PT}(\mathbb{Z})$, the profinite topology on $\mathbb{Z}$, if and only if for any n in $\mathcal{A}$, there exists p such that $n + p\mathbb{Z} \subseteq \mathcal{A}$.

Call an open set $\mathcal{A}$ *effectively open* (in $\mathcal{PT}(\mathbb{Z})$) if there is an effective procedure that, given any number n in $\mathcal{A}$, computes some integer p such that $n + p\mathbb{Z}$ is contained in $\mathcal{A}$. A set is *effectively closed* if its complement is effectively open.

We can now state the properties of the group $L(\mathcal{A})$ that will allow us to prove Theorem 4.0.4.

PROPOSITION 4.1.1. *Let $\mathcal{A}$ be a subset of $\mathbb{Z}$.*

(1) *$L(\mathcal{A})$ is residually finite if and only if $\mathcal{A}$ is closed in $\mathcal{PT}(\mathbb{Z})$.*
(2) *$L(\mathcal{A})$ is effectively residually finite if and only if $\mathcal{A}$ is effectively closed in $\mathcal{PT}(\mathbb{Z})$.*

The first point already appears in the previous chapter, and is written here only as a reminder.

PROOF. Suppose first that $L(\mathcal{A})$ is effectively residually finite. Let x be a integer in the complement of $\mathcal{A}$, thus such that $u_x\hat{u}_x^{-1} \neq 1$ in $L(\mathcal{A})$. By our hypothesis, we can effectively find a morphism φ from $L(\mathcal{A})$ to a finite group F, with $\varphi(u_x\hat{u}_x^{-1}) \neq 1$ in F. Call N the product of the orders of the images of a and $\hat{a}$ in F. We then claim that $\mathcal{A}\cap(x + N\mathbb{Z}) = \varnothing$. Indeed, if it were not the case, there would exist an integer k such that $x + kN \in \mathcal{A}$, that is to say, such that $u_{x+kN} = \hat{u}_{x+kN}$ in $L(\mathcal{A})$. But then, this would imply:

$$\begin{aligned}\varphi(u_x) &= \varphi(a)^{kN}\varphi(u_x)\varphi(a)^{-kN} = \varphi(a^{kN}u_xa^{-kN})\\ &= \varphi(u_{x+kN}) = \varphi(\hat{u}_{x+kN})\end{aligned}$$

Because it can similarly be proved that $\varphi(\hat{u}_x) = \varphi(\hat{u}_{x+kN})$, this contradicts the assumption that $\varphi(u_x\hat{u}_x^{-1}) \neq 1$. Thus $\mathcal{A}$ does not meet $x + N\mathbb{Z}$.

The converse result (that if $\mathcal{A}$ is effectively closed, $L(\mathcal{A})$ is effectively residually finite), is a step by step effectivization of the proof given for Proposition 3.5.3 of the equivalent non-effective statement, and thus we do not carry it out here, the proof written out for Proposition 3.5.3 is presented in such a way that its effectiveness is obvious. □

From Proposition 4.1.1, to prove Theorem 4.0.4, it suffices to build $\mathcal{A}$ with the following properties: $\mathcal{A}$ is recursive, $\mathcal{A}$ is closed, but not effectively so.

We first check that the group without CFQ built in the previous chapter for Theorem 3.0.4 was in fact effectively residually finite.

PROPOSITION 4.1.2. *Dyson's group with solvable word problem but without CFQ built thanks to the subset $\mathcal{A}_2$ of $\mathbb{Z}$ built in Lemma 3.5.5 is effectively residually finite.*

PROOF. The proof of Lemma 3.5.5 is clearly effective: given a point x which is not in the constructed subset $\mathcal{A}_2$, a number y such that $x + y\mathbb{Z}$ does not meet $\mathcal{A}_2$ is explicitly given. Thus $\mathcal{A}_2$ is effectively closed, and $L(\mathcal{A}_2)$ is effectively residually finite. □

Theorem 4.0.5 directly follows from this proposition.

4.2. Construction in $\mathbb{Z}$

We first need some additional properties of the topology $\mathcal{PT}(\mathbb{Z})$.

The profinite topology on $\mathbb{Z}$ is in fact metrizable. This was proven for instance in [**67**], however this topology is called “Furstenberg's topology” there, following Furstenberg's article [**41**], which rediscovered the profinite topology on $\mathbb{Z}$.

A metric which generates $\mathcal{PT}(\mathbb{Z})$ is given by the following formula:

$$\|x\| = \frac{1}{sup\,\{n:\ 1|x, 2|x, 3|x, ..., n|x\}}$$
$$d(x,y) = \|x - y\|$$

The norm $\|x\|$ is thus the reciprocal of the greatest integer n with the property that 1, 2, ..., n all divide x.

Define a function θ on the natural numbers by $\theta(n) = \operatorname{lcm}\{1, 2, 3..., n\}$. $\theta(n)$ is the smallest non-zero natural number such that $\|\theta(n)\| \leq \frac{1}{n}$. The closed ball of radius $\frac{1}{n}$ and centered in x, which is the set $\{y \in \mathbb{Z}, d(x,y) \leq \frac{1}{n}\}$, is simply the set $x + \theta(n)\mathbb{Z}$. It is in fact also open. Call $\overline{B}(x,r)$ the closed ball centered in x and of radius r, and $B(x,r)$ the corresponding open ball (thought pay attention that the latter is not the interior of the former). The distance d is in fact ultrametric: for x, y and z integers, one has $d(x,z) \leq \max(d(x,y), d(y,z))$. This implies that each point of a ball can be taken as its center, and thus that if two balls intersect, one is contained in the other.

Of course, the distance d is effective: d is a recursive function. This implies that both the closed and open balls of $\mathcal{PT}(\mathbb{Z})$ are recursive sets.

LEMMA 4.2.1. *There exists a recursive subset $\mathcal{A}_3$ of $\mathbb{Z}$, closed in $\mathcal{PT}(\mathbb{Z})$, but not effectively so.*

PROOF. We construct a set $\mathcal{B}$, which will be the complement of the announced $\mathcal{A}_3$. Thus it has to be recursive and open but not effectively open.

Call p_n the n-th prime number. Define a sequence $(t_n)_{n\geq 0}$ by $t_n = p_1 ... p_n$. This sequence is defined so that p_k divides t_n if and only if $k \geq n$. Note also that t_n divides t_{n+1}.

Consider an effective enumeration of all Turing machines: M_1 is the first machine, M_2 is the second... We will build $\mathcal{B}$ as a disjoint union of open sets X_n, each X_n being a neighborhood of t_n defined thanks to a run of the n-th Turing machine M_n. If this machine does not halt, X_n is a closed ball centered at t_n of radius $\frac{1}{t_{n+1}}$. If it halts, it is a finite union of balls, one of which is centered at t_n, the radius of which depends of the number of steps needed for M_n to halt. Thanks to this, an information of the form “X_n contains a ball of radius r centered in t_n” will translate in “if M_n halts, it does so in less than N steps”, where N can be computed from r.

Initialize $X_n = \{t_n\}$. Call $r_n = \frac{1}{t_{n+1}}$ and $m = \theta(t_{n+1})$, so that the closed ball $\overline{B}(t_n, r_n)$ is the set $t_n + m\mathbb{Z}$.

Start a run of the machine M_n.

After each step of computation of M_n, note k the number of steps already done in the computation, and add, to X_n, $t_n + km$, $t_n - km$, as well as open balls centered in those numbers, that are contained in $\overline{B}(t_n, r_n)$, and do not contain t_n. That is, we replace X_n by:

$$X_n \cup B(t_n + km, \frac{1}{2}d(t_n, t_n + km)) \cup B(t_n - km, \frac{1}{2}d(t_n, t_n - km))$$

Because the distance d is ultrametric, both balls $B(t_n+km, \frac{1}{2}d(t_n, t_n+km))$ and $B(t_n-km, \frac{1}{2}d(t_n, t_n-km))$ are contained in $\overline{B}(t_n, r_n)$.

If, at some point, the machine M_n halts, X_n consists of t_n and of finitely many open balls centered at points $t_n \pm km$. By construction, a point from $X_n \setminus \{t_n\}$ is at distance at least $\inf_k \left\{\frac{1}{2}d(t_n, t_n \pm km)\right\}$ from t_n. This infimum can be computed, call it r. Then, compute the smallest natural number y such that $d(t_n, y) < r$, and call r' the distance $d(t_n, y)$. We then add to X_n the ball $B(u_n, r')$. This implies that X_n cannot contain any ball of center t_n and of radius strictly greater than r', because it does not contain y. In particular, any ball centered in t_n that contains one of the elements of the form $t_n \pm km$ that were added to X_n is not contained in X_n.

Of course, if the machine M_n does not halt, X_n will be the whole ball $\overline{B}(t_n, r_n)$.

This ends the definition of X_n, and $\mathcal{B}$ is defined as the union $\bigcup X_n$. This union is disjoint because, by the choice of the radius r_n, any element of X_n is divisible by p_n, but none is divisible by p_{n+1}. We now prove that $\mathcal{B}$ defined this way satisfies all three properties that appear in the statement of this Lemma.

$\mathcal{B}$ is clearly open, because each X_n is open, whether or not the machine M_n halts.

$\mathcal{B}$ is a recursive set. It is obviously recursively enumerable, because it was defined by an effective enumeration. To see that it is also co-r.e., let x be an integer, we want to decide whether x belongs to $\mathcal{B}$. By looking at the prime decomposition of x, one can find up to one n such that x might belong to X_n. Because X_n is always contained in $\overline{B}(t_n, r_n)$, if $d(t_n, x) > r_n$, x cannot be in X_n. Otherwise, it belongs to $\overline{B}(t_n, r_n) = t_n + m\mathbb{Z}$, and we can find k such that $x = t_n + km$. Then, if a run of M_n lasts more than k steps, automatically x will belong to X_n. On the other hand, if M_n stops in less than k steps, X_n can be determined explicitly as a finite union of open balls, and thus the question of whether x belongs to X_n can be settled. Because the problem "does M_n halt in more than k steps" is a computable one, in either case we will be able to determine whether or not x belongs to $\mathcal{B}$.

Finally, $\mathcal{B}$ is not effectively open. Suppose we have an algorithm that gives, for x in $\mathcal{B}$, an integer k such that $x + k\mathbb{Z}$ is contained in $\mathcal{B}$. Applying it to t_n, we can find a radius r such that $B(t_n, r)$ is contained in $\mathcal{B}$. We will show that this information implies a new information of the form: if the machine M_n halts, then it halts in less than N steps. This would of course allow one to solve the halting problem, thus such an algorithm does not exist.

Indeed, we have seen that if M_n halts in N steps, X_n cannot contain any ball centered in t_n that contains an element of the form $t_n \pm km$, with $k \leq N$. Turning this around, computing N such that $t_n + mN$ belongs to $B(t_n, r)$, (for instance $N = \theta(\lceil \frac{1}{r} \rceil + 1)$), the information "$B(t_n, r)$ is contained in $\mathcal{B}$" implies that either M_n does not halt, or it halts in less than N steps.

This ends the proof of Lemma 4.2.1. □

What we do not know yet is whether the condition of being recursively presented and effectively residually finite, which is necessary to be a subgroup of a finitely presented residually finite group, is also sufficient.

4.3. Non-recursive depth function

We add here a few remarks on the depth function of groups.

A finitely generated subgroup H of a finitely generated residually finite group G must have a depth function ρ_H that grows slower than that of G (see [**15**]). Because it is easy to see that a finitely presented residually finite group always has a recursive depth function, a subgroup of a finitely presented residually finite group must have its depth function bounded above by a recursive function. Note that in [**60**], it was shown that finitely presented residually finite groups can have arbitrarily large recursive depth function. It was also known that the depth function of a residually finite group could grow arbitrarily fast (see [**16**]).

The fact that a subgroup of a finitely presented residually finite group must have its depth function bounded above by a recursive function also follows from the following:

Proposition 4.3.1. *An effectively residually finite group with solvable word problem always has its depth function bounded above by a recursive function.*

Proof. Straightforward. □

As we have already checked that the group $L(\mathcal{A}_2)$ built to prove Theorem 3.0.4 in the previous chapter was effectively residually finite, this implies that this group has a depth function bounded above by a recursive function, while it does not have CFQ. We can show slightly better:

Corollary 4.3.2. *There exists a group with solvable word problem and recursive depth function, that does not have CFQ.*

PROOF. It is not clear whether the depth function of the group $L(\mathcal{A}_2)$ is recursive or not, but as remarked above we know that there exist recursive functions that grow faster than it. By taking the direct product of the group $L(\mathcal{A}_2)$ with a finitely presented group with recursive depth function greater that that of $L(\mathcal{A}_2)$, which exists by one of the main results of [**60**], one obtains a group which still has solvable word problem and uncomputable finite quotients, and whose depth function is recursive. $\square$

We finally show:

PROPOSITION 4.3.3. *Let $\mathcal{A}_3$ be the subset of $\mathbb{Z}$ given in Lemma 4.2.1. The depth function $\rho_{L(\mathcal{A}_3)}$ of $L(\mathcal{A}_3)$ cannot be smaller than a recursive function.*

PROOF. Suppose there exists a recursive function f such that $\rho_{L(\mathcal{A}_3)} \leq f$. Then, for each n which is not in $\mathcal{A}_3$, $u_n\hat{u}_n^{-1}$ is a non-identity element of $L(\mathcal{A}_3)$, and thus it has a non-trivial image in a finite quotient F of size at most $f(4n+2)$ (because u_n and $\hat{u}_n$ are of word length $2n+1$). Now the orders of a and $\hat{a}$ in F both divide $f(4n+2)!$, thus F is a quotient of the quotient of $L(\mathcal{A}_3)$ given by adding to it the two relations $a^{f(4n+2)!} = e$ and $\hat{a}^{f(4n+2)!} = e$ (i.e. $\langle L(\mathcal{A}_3) \mid a^{f(4n+2)!}, \hat{a}^{f(4n+2)!} \rangle$). Thus, in this group as well, $u_n\hat{u}_n^{-1}$ is a non-identity element, and we know that this implies that $n + f(4n+2)!\mathbb{Z}$ does not meet $\mathcal{A}_3$ (see the proof of Proposition 3.5.3). This shows that $\mathcal{A}_3$ is effectively closed, contradicting our hypothesis. $\square$

Theorem 4.0.6 follows directly.

CHAPTER 5

Computable analysis on the space of marked groups

Introduction

In this chapter, we describe the foundations of the theory of computable analysis on the space of marked groups. This corresponds to studying decision problems for groups and marked groups described by word problem algorithms, or, equivalently, to studying what can be algorithmically inferred on a group given its Cayley graph, or again, to study the Λ_{WP}-computable functions, we have already shown in the preliminaries that those formulations are equivalent (see Section 1.2).

Our study presents interesting aspects both from the point of view of group theory, and from the point of view of computable analysis, as it is remarkable that a space that presents as much undecidability as the space of marked groups should appear naturally, as opposed to being a space defined specifically to provide an example of a Polish space with a bad algorithmic behavior.

Computable analysis is a concept that goes back to Turing's seminal 1937 paper ([**124**]), its aim is to study which functions of a real variable can be deemed computable or not. It was later extended to computability on the Cantor and Baire spaces, to recursive metric spaces, which we define in Section 5.2, and even to computable topological spaces.

The topology of the space of marked groups is metrizable, and thus our present study will focus on the space of marked groups seen as a recursive metric space, not just as a computable topological space. We however do include results that are independent of a choice of a metric whenever possible.

The topology of the space of marked groups can be traced back to Chabauty in [**21**]. It has by now become a standard tool in the study of group properties, see for instance [**84**] and [**98**] where tools of descriptive set theory are used to obtain far reaching group theoretical results. We will still include a brief introduction to describe the topology and the metric of the space of marked groups, which we denote by $\mathcal{G}$ throughout.

Our main purpose here is to show that the study of decision problems for marked groups described by word problem algorithms corresponds exactly to the study of computable analysis on the space of marked groups, and to show that several methods and results of computable analysis can be directly applied to the space of marked groups to obtain group theoretic results.

The most useful result in this regard is Markov's Lemma for groups (Lemma 5.4.5), which we state now. Recall from the preliminaries that Λ_{WP} designates the numbering type associated to word problem algorithms.

Lemma 5.0.1 (Markov's Lemma for groups). *If $(G_n)_{n\in\mathbb{N}}$ is a Λ_{WP}-computable sequence of marked groups that converges to a marked group H with solvable word problem, with $G_n \neq H$ for each n, then no algorithm can tell H from the groups in $\{G_n, n \in \mathbb{N}\}$, when those marked groups are described by word problem algorithms.*

More precisely, $\{H\}$ is not a Λ_{WP}-semi-decidable subset of $\{H\} \cup \{G_n, n \in \mathbb{N}\}$.

This result can be used in conjunction with the Higman-Clapham-Valiev Theorem to build finitely presented groups with solvable word problem but with various undecidable problems. We show for instance that the existence of a finitely presented group with solvable word problem but unsolvable order problem derives from Markov's Lemma applied to the sequence $((\mathbb{Z}/n\mathbb{Z}, 1))_{n\in\mathbb{N}}$, which converges to $(\mathbb{Z}, 1)$ in the space of Marked groups.

More generally, we show that whenever a property P of group is effectively not open, i.e. when there is a Λ_{WP}-computable sequence of groups without P that converges to a Λ_{WP}-computable group with P, then there is a finitely presented group with solvable word problem in which no algorithm recognizes the finitely generated subgroups that have the property P.

Numberings of the space of marked groups. The main problem raised in Chapter 2 is the systematic study of the various possible algorithmic descriptions of groups.

The present chapter can be seen as participating in that study, by studying in particular the numberings of $\mathcal{G}$ that preserve its metric structure, i.e. those that make of $\mathcal{G}$ a *recursive metric space.* (In particular, the distance function on $\mathcal{G} \times \mathcal{G}$ cannot be computed from finite presentations.)

A precise definition of recursive metric spaces appears in Section 5.2. The first step to defining recursive metric spaces is to introduce the computable reals, and the standard numbering ν_C of $\mathbb{R}$. This numbering is defined as follows: a description of a real number x for ν_C is a machine that produces a sequence of rationals that converges to x at exponential speed.

A recursive metric space (X, d, ν) can then be defined as a metric space (X, d) together with a numbering $\nu :\subseteq \mathbb{N} \to X$ such that the distance function $d : X \times X \to \mathbb{R}$ is $(\nu \times \nu, \nu_C)$-computable.

The numbering ν_{WP} associated to the word problem description of marked groups allows to compute several distances defined on the space of marked groups, we quote here a consequence of Proposition 5.1.4:

Proposition 5.0.2. *There exists a distance d on $\mathcal{G}$ such that the space $(\mathcal{G}, d, \nu_{WP})$ is a recursive metric space.*

We conjecture that any "reasonable" numbering of $\mathcal{G}$ that makes of it a recursive metric space should be strongly related to ν_{WP}. We include in SubSection 5.3.2 some results related to this question.

A non-effectively separable Polish space. The study of computable analysis on the space of marked groups is very interesting from the point of view of computable analysis. Indeed, most of the theory of computable functions defined on metric spaces was developed in the setting of effectively Polish spaces, recursive metric spaces which are both effectively complete and effectively separable. A recursive metric space is *effectively complete* if there is an algorithm that computes the limit of effective Cauchy sequences, and it is *effectively separable* if it admits a computable and dense sequence. It is *effectively Polish* if it is both effectively complete and effectively separable. Those notions are properly defined in Section 5.2.

We will prove:

Theorem 5.0.3. *The space of marked groups is an effectively complete recursive metric space that is not effectively separable.*

A famous theorem of Ceitin states that the computable functions defined on the computable points of an effective Polish space are necessarily effectively continuous. This theorem thus cannot be applied to $\mathcal{G}$.

In [**85**], Moschovakis proved a generalization of Ceitin's Theorem, which gives, as of today, the weakest known conditions on a constructive metric space, that are sufficient in order for the functions defined on this space to be effectively continuous.

We however prove:

Theorem 5.0.4. *The hypotheses of Moschovakis' Effective Continuity Theorem are not satisfied by the space of marked groups.*

This theorem appears as Corollary 5.3.15 in the text, it is a consequence of the following result, which is our most important theorem:

Theorem 5.0.5 (Failure of an Effective Axiom of Choice for groups). *There is no algorithm that can, given two finite sets R_1 and R_2 of relations, produce a word problem algorithm for a group in which the relations of R_1 hold, while those of R_2 fail.*

This holds even under the assumption that the sets R_1 and R_2 given as input are always chosen so that there exists a group with solvable word problem that satisfies the relations of R_1, and in which the relations of R_2 fail.

Even though we are unable to prove that computable functions defined on the space of marked groups should be continuous, the recursive metric spaces where discontinuous computable functions exist that are known are very pathological objects (see SubSection 5.2.5.1 for an example), it would be very surprising if the space of marked groups was amongst them. Our main conjecture is thus the following.

Conjecture 5.0.6 (Main Conjecture). *Any Λ_{WP}-computable function defined on the computable points of the space of marked groups is continuous.*

This would have the following consequence on decision problems for groups described by word problem algorithms:

Conjecture 5.0.7. *A Λ_{WP}-decidable property of groups with solvable word problem should be clopen.*

The existence of a "natural" example of a Polish space which is not effectively Polish is remarkable. We can quote here a sentence of Moschovakis from [**87**, Chapter 3], who, after defining a recursive presentation of a Polish space (a concept closely related to effectively Polish spaces, see Definition 5.2.24 and Proposition 5.2.25), states:

"Not every Polish space admits a recursive presentation—but every interesting space certainly does."

We however prove in Theorem 5.3.4:

THEOREM 5.0.8. *The space $(\mathcal{G}, d)$ does not have a recursive presentation.*

Moschovakis' statement should be understood as the belief that the only examples of Polish spaces that do not admit recursive presentations will be artificially built counterexamples, while the Polish spaces that occur naturally in mathematics should always have those presentations. Thus the fact that there exists a naturally arising Polish space for which this fails is very interesting, and raises the challenge of finding new ways to prove continuity theorems, that do not rely on the effective separability of the ambient space.

The Borel and Kleene-Mostovski hierarchies. The study of decision problems for groups described by word problem algorithms defines a hierarchy of group properties, classifying properties as Λ_{WP}-decidable, Λ_{WP}-semi-decidable, Λ_{WP}-co-semi-decidable, or Λ_{WP}-completely undecidable. (We omit to write Λ_{WP} in what follows.) Those are the first levels of the arithmetical hierarchy for decision problems asked for groups described by word problem algorithms, and are the only levels we will be interested in in our present study.

This hierarchy resembles to the Borel hierarchy on $\mathcal{G}$, which was for instance studied in [**9**]. We expect the following correspondence between those hierarchies (here, P denotes a property of groups, or of marked groups):

$$\text{P is Clopen} \longleftrightarrow \text{P is Decidable}$$

$$\text{P is Open} \longleftrightarrow \text{P is Semi-Decidable}$$

$$\text{P is Closed} \longleftrightarrow \text{P is co-Semi-Decidable}$$

Our main conjecture asks whether the top arrow between clopen and decidable can be replaced by the implication: $\text{P is Decidable} \implies \text{P is Clopen}$. The reverse implication, $\text{P is Clopen} \implies \text{P is Decidable}$, is known to be true in the compact case, if we fix some bound on the number of generators of the marked groups we consider, but to fail in $\mathcal{G}$.

All other arrows are known to fail as actual implications.

However, they have the following informal meaning:

- A natural semi-decidable property in $\mathcal{G}$ is (*very much*) expected to be open, and a natural co-semi-decidable property in $\mathcal{G}$ is (*very much*) expected to be closed.

Semi-decidable properties that are not open are built using Kolmogorov complexity, those are sets one should not expect to run into when dealing with properties defined by algebraic or geometric constructions. (See SubSection 5.2.5 for an example.)

- A natural open property in $\mathcal{G}$ is (*a little*) expected to be semi-decidable, and a natural closed property in $\mathcal{G}$ is (*a little*) expected to be co-semi-decidable.

This second fact is rather of an empirical nature, and is justified by the results given in SubSection 5.4.6.9.

Results that establish that properties belong to the same level in the Borel and Kleene-Mostowski hierarchies are called *correspondence* results.

We are unable to give a natural example of an open property which fails to be semi-decidable, but we propose several candidates for which we conjecture that the topological classification does not capture the decidability status:

CONJECTURE 5.0.9. *The set of LEF groups is closed but not co-semi-decidable. The set of isolated groups is open but not semi-decidable.*

We detail this conjecture in Section 5.6, where we establish in particular that the part of this conjecture that concerns LEF groups would imply Slobodkoi's Theorem about the undecidability of the universal theory of finite groups.

However, we give in SubSection 5.4.6 a table that contains a wide range of group properties for which the correspondence between the Borel and Kleene-Mostowski hierarchy holds perfectly.

The isomorphism problem from word problem algorithms. After having studied the decidability status of various group properties, we start the study of the isomorphism problem for groups described by word problem algorithms. This study is naturally preceded by the study of marked and abstract recognizability, as explained in Chapter 2.

We emphasize the role of isolated groups, which are semi-recognizable, a property that is shared by all finitely presented groups with respect to the finite presentation description, but which seems very hard to find in other possible group descriptions.

We then discuss the relation between group recognizability from word problem algorithms and an order relation on the set of finitely generated groups introduced by Bartholdi and Erschler in [**4**], called the "preform" relation.

We use their results to establish the following theorem (Theorem 5.5.12):

THEOREM 5.0.10. *There exists an infinite set of groups with solvable word problem which are pairwise completely undistinguishable for Λ_{WP}.*

Here, two groups G and H are called Λ_{WP}-completely undistinguishable if neither $\{G\}$ nor $\{H\}$ is a Λ_{WP}-semi-decidable subset of $\{G, H\}$.

There remain however many open problems related to the recognizability of groups from word problem algorithms. In particular, we ask several questions regarding the relation between the topology of $\mathcal{G}$ and the topology that is seen looking only at groups with solvable word problem: can a marked group with solvable word problem be isolated from other marked groups with solvable word problem, while being the limit of a sequence of groups with unsolvable word problem? Any infinite group must preform a group that is non-isomorphic to it, but must an infinite group with solvable word problem preform another group with solvable word problem?

Markovian computable analysis. Note that the term "computable analysis", which we have used already several times in this introduction, is ambiguous, because several approaches to computing with real numbers (and to computing in metric spaces) were considered and developed independently. The only notion which interests us here is what is known as Markov computability, or as the Russian approach, see [**2**] for some historical remarks. The reason for this is that Markov computability is the only notion of computability which corresponds to what the theory of decision problems for groups aims at studying: what information can be computed about a group given a finite description of it, description which can be manipulated by a computer.

The Markovian definition of computable analysis was in fact the definition considered by Turing himself (and even by Borel as early as 1912...), and while it was studied in various countries by many mathematicians, an important body of work in Markovian computable analysis comes from Markov's school of mathematics, which is a school of constructive mathematics.

The tenants of Markov's school accept only a weak form of the excluded middle law, in the form of the following statement, known as Markov's Principle: "a Turing machine either halts or does not halt". They consider that objects *exist* only if their existence can be attested to by algorithmic means.

This fact renders a part of the literature that we refer to ([**1, 63, 20, 77, 76**]) less accessible that it would be, had it been written in a classical language. This motivated us in writing a rather detailed exposition of computable analysis on metric spaces, which appears in Section 5.2, and which should be very accessible.

For instance, we include a simple proof of a theorem that says that computable functions defined on effective Polish spaces must be continuous. This simple proof cannot be found in the constructivists' works, as in those works, the statement "the function f is continuous" has to be interpreted as "the computable function f is effectively continuous", and Ceitin's Theorem on the effective continuity of functions defined on effective Polish spaces is significantly more complicated than the corresponding non-effective continuity theorem.

Note also that in some settings, objects very similar to numberings are studied, called *notation systems,* see [**20, 63, 86**]*,* which use sets of word instead of natural numbers, the theory remains virtually unchanged.

Contents of this chapter. In Section 5.1, we describe the space of marked groups, and give a few results related to computability: impossibility of deciding whether or not a basic clopen set is empty, etc.

In Section 5.2, we describe the main results of computable analysis, giving proofs for some of them and references for the others. We quote in particular Markov's Lemma, Mazur's Continuity Theorem, the Ceitin Theorem, and Moschovakis' extension of this theorem.

In Section 5.3, we start investigating the space of marked groups as a computable metric space. We prove our most important theorems; which state that none of the continuity results given in the previous sections can be applied to the space of marked groups.

In Section 5.4, we apply Markov's Lemma to sequences of groups. We give a wide range of examples of open or closed group properties which are partially recognizable thanks to the word problem algorithm description.

In Section 5.5, we study the isomorphism problem for groups described by word problem algorithms, and the special problem of group recognizability.

In Section 5.6, we propose the sets of LEF groups and of isolated groups as sets for which the correspondence between the arithmetical hierarchy and the Borel hierarchy might fail, we motivate those conjectures.

In Section 5.7, we use Markov's Lemma in conjunction with various versions of Higman's Embedding Theorem, in order to give short and elegant proofs of some well known results, for instance of the existence of a group with solvable word problem but unsolvable order problem, etc.

5.1. The topological space of marked groups

For a natural number k, we denote by $\mathcal{G}_k$ the set of isomorphism classes of k-marked groups, and $\mathcal{G}$ the disjoint union of the $\mathcal{G}_k$.

For an abstract group G, we denote $[G]_k$ the set of all its markings in $\mathcal{G}_k$, and $[G]$ the set of all its markings in $\mathcal{G}$ (as in **[23]**). We also write $[G]_k$ or $[G]$ for a marked group G, it is the abstract isomorphism class of G in $\mathcal{G}_k$ or in $\mathcal{G}$.

5.1.1. Topology on $\mathcal{G}$. We define a topology on $\mathcal{G}$ by equipping each separate space $\mathcal{G}_k$ with a topology, the topology we then consider on $\mathcal{G}$ is the disjoint union topology of the $\mathcal{G}_k$. It is customary to embed each set $\mathcal{G}_k$ in the set $\mathcal{G}_{k+1}$ by identifying a marking $(G, (g_1, ..., g_k))$ with the same marking where the identity e_G of G is added as a last generator: the k-marking $(G, (g_1, ..., g_k))$ is identified with the $k+1$-marking $(G, (g_1, ..., g_k, e_G))$. We do not adhere to this convention, for reasons that appear clearly in Chapter 2 of this thesis: adding generators that define the identity to a marking can change whether or not a marked group is recognizable. It is thus detrimental in the study of decision problems for groups to identify a marking to the markings obtained by adding generators.

For each k, consider a finite set $\{s_1, ..., s_k\}$, choose arbitrarily an order on the set $\{s_1, ..., s_k\} \cup \{s_1^{-1}, ..., s_k^{-1}\}$, and enumerate lexicographically with respect to that order the elements of the free group $\mathbb{F}_k$ over S.

Denote by $i_k(n)$ the nth element obtained in this enumeration, i_k is thus a bijection between $\mathbb{N}$ and $\mathbb{F}_k$.

To a normal subgroup N of $\mathbb{F}_k$ we can associate its characteristic function $\chi_N : \mathbb{F}_k \to \{0, 1\}$, and composing it with the bijection i_k, we obtain an element of the Cantor space $\mathcal{C} = \{0, 1\}^{\mathbb{N}}$. This defines an embedding:

$$\Phi_k : \begin{cases} \mathcal{G}_k & \longrightarrow \{0;1\}^{\mathbb{N}} \\ N \lhd \mathbb{F}_k & \longmapsto \chi_N \circ i_k \end{cases}$$

of the space of k-marked groups into the Cantor space. We call the image $\Phi_k(G)$ of a k-marked group G the *binary expansion* of G.

With the Cantor set being equipped with its usual product topology, the topology we will study on $\mathcal{G}_k$ is precisely the topology induced by this embedding.

It is easy to see that $\Phi_k(\mathcal{G}_k)$ is a closed subset of $\{0;1\}^{\mathbb{N}}$ of empty interior. It is thus compact.

The product topology on $\{0;1\}^{\mathbb{N}}$ admits a basis which consists of clopen sets: given any finite set $A \subseteq \mathbb{N}$ and any function $f : A \to \{0;1\}$, define the set Ω_f by: $(u_n)_{n\in\mathbb{N}} \in \Omega_f \iff \forall n \in A; u_n = f(n)$. Sets of the form Ω_f are clopen and form a basis for the topology of $\{0;1\}^{\mathbb{N}}$.

Sets of the form $\mathcal{G}_k \cap \Omega_f$ thus define a basis for the topology of $\mathcal{G}_k$.

The set $\mathcal{G}_k \cap \Omega_f$ is defined as a set of marked groups that must satisfy some number of imposed relations, while on the contrary a fixed sets of elements must be different from the identity.

We fix the following notation. For m and m' natural numbers, and elements $r_1, ..., r_m;\ s_1, ..., s_{m'}$ of $\mathbb{F}_k$, we note $\Omega^k_{r_1,...,r_m;s_1,...,s_{m'}}$ the set of k-Marked Groups that satisfy the relations $r_1, ..., r_m$, while they do not satisfy $s_1, ..., s_{m'}$. We call $s_1, ..., s_{m'}$ *irrelations.*

The sets $\Omega^k_{r_1,...,r_m;s_1,...,s_{m'}}$ are called the *basic clopen sets.*

In what follows, we call the set $r_1, ..., r_m;\ s_1, ..., s_{m'}$ of relations and irrelations *coherent* if $\Omega^k_{r_1,...,r_m;s_1,...,s_{m'}}$ is not empty. The Boone-Novikov theorem which implies that there exists a finitely presented group with unsolvable word problem directly implies the following:

Theorem 5.1.1 (Boone-Novikov reformulated). *No algorithm can decide whether or not a given finite set of relations and irrelations is coherent. More precisely, there is an algorithm that stops exactly on incoherent sets of relations and irrelations, but no algorithm can stop exactly on coherent sets of relations and irrelations.*

We will call a set $r_1, ..., r_m;\ s_1, ..., s_{m'}$ of relations and irrelations *word problem coherent*, or *wp-coherent*, if the basic clopen set $\Omega^k_{r_1,...,r_m;s_1,...,s_{m'}}$ contains a group with solvable word problem. The remarkable fact that the notions of coherence and wp-coherence differ follows from a theorem of Miller from **[82]** which we expose in details in SubSection 5.3.3 (see Theorem 5.3.12).

The fact that coherence and wp-coherence differ can be equivalently formulated as: "groups with solvable word problem are not dense in $\mathcal{G}$". This is stated in Proposition 5.3.1.

This remark calls for the following theorem:

Theorem 5.1.2 (Boone-Rogers reformulated). *No algorithm can stop exactly on wp-coherent sets of relations and irrelations.*

Proof. This follows from the Boone-Rogers Theorem (**[13]**) which states that there is no uniform solution to the word problem on the set of finitely presented groups with solvable word problem. Indeed, it is easy to see that an effective way of recognizing wp-coherent sets of relations and irrelations would provide a uniform algorithm for the word problem on finitely presented groups with solvable word problem. □

The following theorem shows that wp-coherence is a property that is more complex than coherence.

THEOREM 5.1.3. *No algorithm can stop exactly on those sets of relations and irrelations which are not wp-coherent.*

The proof of this theorem relies on the construction of Miller mentioned above. We will use this construction to prove another important result, Theorem 5.3.10. Because of this, the proof of Theorem 5.1.3 is postponed until SubSection 5.3.3.

5.1.2. Different distances. The topology defined above on the space of marked groups is metrizable. We describe here two possible distances which generate this topology, and which we may most of the time use interchangeably.

As for the topology, the distance we use on the space of marked groups is defined on each $\mathcal{G}_k$ separately, and we can consider that groups marked by generating families that have different cardinalities are infinitely apart.

5.1.2.1. *Ultrametric distance.* For sequences $(u_n)_{n\in\mathbb{N}}$ and $(v_n)_{n\in\mathbb{N}}$ in $\{0;1\}^{\mathbb{N}}$ that are different, denote n_0 the least number for which $u_{n_0} \neq v_{n_0}$, and set $d((u_n)_{n\in\mathbb{N}},(v_n)_{n\in\mathbb{N}}) = 2^{-n_0}$. If the sequences $(u_n)_{n\in\mathbb{N}}$ and $(v_n)_{n\in\mathbb{N}}$ are equal, set $d((u_n)_{n\in\mathbb{N}},(v_n)_{n\in\mathbb{N}}) = 0$.

This defines an ultrametric distance on $\{0;1\}^{\mathbb{N}}$ which generates its topology.

5.1.2.2. *Cayley Graph distance.* Yet another way to define a distance on $\mathcal{G}_k$, that yields the same topology as the one seen above, is by using labeled Cayley graphs. We have seen in Section 1.2 that a labeled Cayley graph defines uniquely a marked group, and that a word problem algorithm can be seen as an algorithm that produces arbitrarily large (finite) portions of the labeled Cayley graph of a given group. We can define a new distance d_{Cay} as follow.

For two groups G and H, generated by the same family S, consider the respective Cayley graphs of G and H, Γ_G and Γ_H. The balls centered at the identity in Γ_G and Γ_H agree up to a certain radius, call r the radius for which the balls of radius r of Γ_G and Γ_H are identical, while their balls of radius $r+1$ differ. Then put $d_{\text{Cay}}((G,S),(H,S)) = 2^{-r}$. If Γ_G and Γ_H are identical, r is infinite, and we put $d_{Cay}((G,S),(H,S)) = 0$. It is easy to check that d_{Cay} is an ultrametric distance which induces the topology of the space of marked groups.

The distance d_{Cay} could be preferred to d, as it brings a visual dimension to proofs, however in most cases it is just less precise than d: the only difference between d and d_{Cay} is that, in the computation of d_{Cay}, the relations are considered "in packs", corresponding to their length as elements of the free groups, while they are considered each one by one when using d, the choice of an order of the free group being precisely what allows to give more or less "weight" to relations of the same length.

5.1.2.3. *The space $\mathcal{G}$ is a recursive metric space.* The following proposition is fundamental in the study of the space of marked groups as a recursive metric space. We give the definition of a computable real in Section 5.2, we still include here the proof of Proposition 5.1.4, as it is simple enough to be grasped without needing a precise understanding of computable reals, and on the contrary, it can motivate the introduction of the computable reals.

PROPOSITION 5.1.4. *The distances d is computable from the Cantor space to the set of computable reals. In particular, it is computable on $\mathcal{G}_k$, when seen as taking as input word problem algorithms, and having as images computable reals. Similarly, the distance d_{Cay} is computable on $\mathcal{G}_k$.*

PROOF. We sketch the proof for the distance d, the proposition is proven similarly for d_{Cay}.

It suffices to show that given two descriptions of computable sequence $(u_n)_{n\in\mathbb{N}}$ and $(v_n)_{n\in\mathbb{N}}$ in $\{0;1\}^{\mathbb{N}}$, and an integer N, it is possible to compute a rational approximation of $d((u_n)_{n\in\mathbb{N}},(v_n)_{n\in\mathbb{N}})$ within 2^{-N}. Given descriptions for $(u_n)_{n\in\mathbb{N}}$ and $(v_n)_{n\in\mathbb{N}}$, it is possible to enumerate their N first digits. If they are identical, one can conclude that $d((u_n)_{n\in\mathbb{N}},(v_n)_{n\in\mathbb{N}}) < 2^{-N}$. On the other hand, if their first N digit differ, we can compute the index of the first digit on which they differ, and thus compute the distance $d((u_n)_{n\in\mathbb{N}},(v_n)_{n\in\mathbb{N}})$ exactly. In any case one obtains an approximation of $d((u_n)_{n\in\mathbb{N}},(v_n)_{n\in\mathbb{N}})$ within 2^{-N}. □

5.1.3. Some references. The topology on the space of marked groups can be traced back to Chabauty in [**21**]. It was later used in different contexts, for instance to study torsion groups [**45**], limit groups [**23**] or to build groups with non-uniformly exponential growth [**91**]. Note the remarkable result of Christophe Champetier in [**22**], that there is no measurable function from $\mathcal{G}_k$ ($k \geq 2$) to $\mathbb{R}$ which is constant exactly on (abstract) isomorphism classes of groups. We will use results from this article that concern hyperbolic groups.

We will also see that the isolated groups of $\mathcal{G}_k$, that were characterized by Cornulier, Guyot and Pitch in [**27**], play an important role in our study, as they are recognizable from the word problem algorithm. We quote right away an important Lemma that appears in that same article, and which we use several times throughout this chapter:

LEMMA 5.1.5. *([**27**], Lemma 1) Consider two marked groups $G_1 \in \mathcal{G}_{m_1}$, $G_2 \in \mathcal{G}_{m_2}$. Suppose that the underlying abstract groups are (abstractly) isomorphic. Then there are clopen neighborhoods V_i, $i = 1, 2$ of G_i in $\mathcal{G}_{m_i}$ and a homeomorphism $\phi : V_1 \to V_2$ mapping G_1 to G_2 and preserving isomorphism classes, i.e. such that, for every $H \in V_1$, $\phi(H)$ is isomorphic to H (as abstract groups).*

Other articles related to the space of marked groups which we will use are: [**10, 8, 84, 9**].

5.2. Introduction on some results in Markovian computable analysis

5.2.1. Vocabulary remarks. Before starting with our introduction to computable analysis, we include here a paragraph which aims at explaining some choices of nomenclature made in this chapter. While none of our name choices are original, we sometimes follow authors whose vocabulary choices were not very influential.

Given a quantified statement ST that concerns the elements of a numbered set (X, ν), one can construct an effective statement that corresponds to ST by replacing all existential quantifiers by effective existential quantifiers, whose meaning is that the object whose existence is claimed can be constructed from the data that appears to the left of the effective existential quantifier in the quantified statement ST.

The obtained statement will usually be called *"effective ST"*, we thus say that (X, ν) satisfies ST *effectively,* or on the contrary that ST does not hold effectively in (X, ν).

Now an interesting aspect of this procedure, that allows one to associate effective statements to classical statements, is that it can turn classically equivalent notions into non equivalent effective notions. While this fact is the source of many interesting research problems and of many interesting results, it leads to many an ambiguity when it comes to naming properties.

Indeed, at times, the question of knowing which notion should be called "effectively P" is hard to decide, for instance when P is defined by a statement of the form: "x is said to have P if it satisfies one of the following equivalent conditions: ...", conditions which are not *effectively* equivalent.

Our definition of an *effectively continuous function* between metric spaces given in SubSection 5.2.3.2 corresponds to an effectivisation of the usual ϵ-δ definition (as in [**20**]), whereas in computable analysis, the more usual definition is an effectivisation of the characterization of continuous functions in terms of preimage of open sets. Those definitions are equivalent in effective Polish spaces, but it is unclear whether this remains true for functions defined on generic recursive metric spaces.

Note also that what we call an effective Polish space, or effectively Polish space, (which is an effectively complete and effectively separable recursive metric space, see SubSection 5.2.3.3 for a precise definition), is usually called a computable metric space, or an effective metric space, see for instance [**17, 121, 129**].

On the contrary, what we call a recursive metric space is not necessarily complete nor separable, this follows the definitions of Ceitin in [**20**], of Kushner in [**20**] and of Moschovakis in [**85**].

5.2.2. Numberings of $\mathcal{G}$ induced by numberings of the Cantor space. Recall that in Section 5.1 we defined an embedding Φ_k of the space $\mathcal{G}_k$ of k-marked groups into the Cantor space.

The natural numbering $\nu_{\mathcal{C}}$ of the Cantor space is obtained by seeing it as the set of functions from $\mathbb{N}$ to $\{0, 1\}$ (see Section 1.1 in the Preliminaries). The $\nu_{\mathcal{C}}$-computable elements are just called computable. We define two other numberings and numbering types on $\mathcal{C}$ by considering lower and upper semi-computable sequences:

DEFINITION 5.2.1. An element $(u_n)_{n\in\mathbb{N}}$ of $\mathcal{C}$ is lower (resp. upper) semi-computable if there exists a $(\mathrm{id}_{\mathbb{N}}, \nu_{\mathcal{C}})$-computable function $g : \mathbb{N} \to \mathcal{C}$ such that for each n, $g(n)$ is an increasing (resp. decreasing) computable sequence that converges to u_n.

This exactly means that the set $\{n \in \mathbb{N}; u_n = 0\}$ is r.e. (resp. co-r.e.).

This definition allows one to define two new numberings of $\mathcal{C}$, denoted $\nu_{\nearrow}$ and $\nu_{\searrow}$, associated to upper and lower semi-computable sequences, we leave out the details of the definitions of those numberings.

The following proposition explains why the numberings associated to word problem algorithms, recursive presentations and co-recursive presentations are very natural, from the point of view of computable analysis.

PROPOSITION 5.2.2. *The restriction of the numbering $\nu_{\mathcal{C}}$ to $\mathcal{G}_k$ is $\equiv$-equivalent to the restriction of the word problem numbering ν_{WP} to k-marked groups. The restriction of $\nu_{\searrow}$ to $\mathcal{G}_k$ is $\equiv$-equivalent to the numbering of $\mathcal{G}_k$ associated to recursive presentation, and the restriction of $\nu_{\nearrow}$ to $\mathcal{G}_k$ is $\equiv$-equivalent to the numbering of $\mathcal{G}_k$ associated to co-recursive presentation.*

Proof. Left to the reader. □

5.2.3. Effective Polish spaces. This introduction follows mostly Kushner ([**63**]), but it is hopefully more accessible, since the constructivist setting adds technical complications. Note that Subsection 5.2.4 follows closely Hertling ([**54**]), who studies Banach-Mazur computable functions.

5.2.3.1. *The computable reals.* A precise definition of the set $\mathbb{R}_c$ of the set of computable reals first appeared in Turing's famous 1937 article ([**124**]), but the numbering type of computable real numbers which is best suited to developing computable analysis was introduced one year later in the corrigendum [**125**].

We define a numbering $\nu_{\mathbb{Q}}$ of rationals. The numbering $\nu_{\mathbb{Q}}$ is defined on $\mathbb{N}$. Given a natural number n, decompose it as a product $n = 2^a 3^b 5^c n'$, with $\gcd(n', 30) = 1$. Put $\nu_{\mathbb{Q}}(n) = (-1)^a \frac{b}{c+1}$.

We now define the Cauchy numbering ν_C of $\mathbb{R}$. Fix an effective enumeration ϕ_0, ϕ_1, ϕ_2... of all partial recursive functions.

Definition 5.2.3. The Cauchy numbering of $\mathbb{R}$ is defined by the formulas:

$$\mathrm{dom}(\nu_C) = \{i \in \mathbb{N}, \exists x \in \mathbb{R}, \forall n \in \mathbb{N}, |\nu_{\mathbb{Q}}(\phi_i(n)) - x| < 2^{-n}\};$$

$$\forall i \in \mathrm{dom}(\nu_C),\ \nu_C(i) = \lim_{n\to\infty}(\nu_{\mathbb{Q}}(\phi_i(n))).$$

Thus the description of a real number x is a Turing machines that produces a sequence $(u_n)_{n\in\mathbb{N}}$ of rationals which converges to x with exponential convergence speed.

Definition 5.2.4. The set of ν_C-computable real numbers is denoted $\mathbb{R}_c$, and simply called the set of computable real numbers.

Several other definitions of the real numbers (decimal expansions, Dedekind cuts), when rendered effective, yield numbering types that define the same set of computable real numbers, but that are not $\equiv$-equivalent to the Cauchy numbering type -they contain strictly more information. See for instance [**90**].

A function $f : \mathbb{R}_c \to \mathbb{R}_c$ which is (ν_c, ν_c)-computable is simply called *computable.*

Proposition 5.2.5 (Rice, [**110**]). *Addition, multiplication and divisions are computable functions defined respectively on $\mathbb{R}_c \times \mathbb{R}_c$, $\mathbb{R}_c \times \mathbb{R}_c$ and $\mathbb{R}_c \times (\mathbb{R}_c \setminus \{0\})$.*

We include here a well known proposition of Turing which follows from Markov's Lemma, see Lemma 5.2.27.

Proposition 5.2.6 (Turing). *Equality is undecidable for computable reals. There is no algorithm that, given two computable reals x and y, chooses one of $x \leq y$ or $y < x$ which is true.*

5.2.3.2. *Recursive metric spaces.* We can now define what is a recursive metric space.

Definition 5.2.7. *A recursive metric space (RMS)* is a metric space (X, d) equipped with a numbering $\nu :\subseteq \mathbb{N} \to X$, such that the distance function $d : X \times X \to \mathbb{R}$ is $(\nu \times \nu, \nu_C)$-computable.

Note that for convenience we do not impose that the set of computable points be dense in X.

Proposition 5.1.4 thus implied that the space of marked groups is a recursive metric space thanks to the word problem algorithm numbering.

Example 5.2.8. The following spaces, equipped with their usual distances and numberings, are recursive metric spaces: $\mathbb{N}$, $\mathbb{R}_c$, the Cantor space $\{0,1\}^{\mathbb{N}}$, the Baire Space $\mathbb{N}^{\mathbb{N}}$.

Definition 5.2.9. A sequence $(u_n)_{n\in\mathbb{N}}$ of computable points in X is called *effectively convergent* if it converges to a point $y \in X$, and if there exists a recursive function $f : \mathbb{N} \to \mathbb{N}$ such that:

$$\forall (n, m) \in \mathbb{N}^2; n \geq f(m) \implies d(u_n, y) \leq 2^{-m}$$

Definition 5.2.10. A sequence $(u_n)_{n\in\mathbb{N}}$ of computable points in X is called effectively Cauchy if there exists a recursive function $f : \mathbb{N} \to \mathbb{N}$ such that:

$$\forall (p, q, m) \in \mathbb{N}^3; p, q \geq f(m) \implies d(u_p, u_q) \leq 2^{-m}$$

In both cases the function f is called a *regulator* for the sequence $(u_n)_{n\in\mathbb{N}}$.

There are different notions of "effective continuity", as we have discussed in SubSection 5.2.1, we give here that of Kushner ([**63**]).

Definition 5.2.11. A function f between computable metric spaces (X, ν) and (Y, μ) is called *effectively continuous* if given a ν-name n of a point x in X and the ν_C-name of a computable real number $\epsilon > 0$, it is possible to compute the ν_C-name of a number $\eta > 0$ such that

$$\forall y \in X;\ d(x, y) < \eta \implies d(f(x), f(y)) < \epsilon.$$

Note that a subtlety here is that the number η is allowed to depend not only on x and ϵ, but also on the given names for those points. A program that computes the ν_C-name for η given x and ϵ is said to *witness for the effective continuity of f*.

5.2.3.3. *Effective completeness and effective separability.* Since the space of marked groups is a Polish space, it is natural to ask whether it is an effective Polish space, that is, whether it is effectively complete and effectively separable.

Here, we define those two notions, and give some properties that follow from them. The importance of these notions lies in the facts that Ceitin's Theorem is set on effectively Polish spaces.

Definition 5.2.12. Let (X, d, ν) be a recursive metric space. An *algorithm of passage to the limit* (the name is from [**63**]) is an algorithm that takes as input a computable Cauchy sequence together with a regulator for it, and produces the ν-name of a point towards which this sequence converges.

A recursive metric space (X, d, ν) is *effectively complete* if it admits an algorithm of passage to the limit.

It is easy to see that any recursive metric space can be effectively completed into an effectively complete metric space.

Indeed, given a recursive metric space (X, d, ν), and denoting $(\overline{X}, d)$ the abstract completion of X, we naturally obtain a numbering $\overline{\nu}$ of $\overline{X}$, which makes of $(\overline{X}, d, \overline{\nu})$ a recursive metric space, by choosing, as $\overline{\nu}$-description of a point x of $\overline{X}$, the code of a Turing machine that produces a ν-computable Cauchy sequences of points of X, that admits $n \mapsto 2^n$ as a regulator.

This is just the construction of the completion of a metric space that uses Cauchy sequences, as rendered effective. See [**54**] or [**63**] for more details, where for example the following easy result is proved:

Proposition 5.2.13. *$(\mathbb{R}_c, d, \nu_C)$ is the effective completion of $(\mathbb{Q}, d, \nu_{\mathbb{Q}})$.*

We have the following easy proposition:

Proposition 5.2.14. *Let (X, d, ν) be an effectively complete metric space. A closed subset Y of X, together with the numbering induced by ν, is also an effectively complete metric space.*

Proof. It suffices to notice that algorithm of passage to the limit for X also works for Y. □

The following proposition can be found in [**63**]:

Proposition 5.2.15. *The Cantor space is effectively complete.*

Corollary 5.2.16. *The space of marked groups equipped with the word problem numbering is effectively complete.*

This last fact is straightforward and could have been proved directly. We now describe the effective notion associated to separability.

Definition 5.2.17. A recursive metric space (X, d, ν) is called *effectively separable* if there exists a computable sequence $(u_n)_{n \in \mathbb{N}}$ of points in X that is dense in X.

We can finally define effectively Polish spaces.

Definition 5.2.18. A recursive metric space (X, d, ν) which is both effectively complete and effectively separable is an *effective Polish space.*

There is not a single accepted term for this notion, which is sometimes referred to as a "recursive metric space". We explained in SubSection 5.2.1 why we talk instead of effective Polish space.

Example 5.2.19. The following spaces, equipped with their usual distances and numberings, are effective Polish spaces: $\mathbb{N}$, $\mathbb{R}_c$, the Cantor space $\{0, 1\}^{\mathbb{N}}$, the Baire Space $\mathbb{N}^{\mathbb{N}}$.

The following proposition, while obvious, shall be very useful in its group theoretical version.

Proposition 5.2.20. *Let (X, d, ν) be a recursive metric space, and Y be a ν-r.e. set in X. Then any computable point in the closure $\overline{Y}$ of Y is the effective limit of a computable sequence of points of Y.*

This proposition could have been phrased: any computable point in the closure of Y is automatically in its "effective closure".

Proof. This is straightforward: given a point x adherent to Y, we can define a computable sequence $(v_n)_{n\in\mathbb{N}}$ by: v_n is the first element, in a fixed enumeration of Y, which is proven to satisfy $d(x, v_n) < 2^{-n}$. □

This proposition has the immediate corollary:

Corollary 5.2.21. *In an effective Polish space with a dense and computable sequence $(u_n)_{n\in\mathbb{N}}$, the computable points are exactly the effective limits of effectively Cauchy sequences extracted from $(u_n)_{n\in\mathbb{N}}$.*

The following result shows one of the appeals of effective Polish spaces: the computable structure on an effective Polish space is entirely defined by the distance function between elements of its dense sequence.

Theorem 5.2.22. *An effective Polish space is computably isometric to the effective completion of any of its computable and dense sequence.*

Proof. Let (X, d, ν) be an effective Polish space, and $(u_n)_{n\in\mathbb{N}}$ any computable and dense sequence. Denote ϕ_0, ϕ_1, ϕ_2, ... an effective enumeration of all partial recursive functions.

The effective completion of $(u_n)_{n\in\mathbb{N}}$ defines another numbering of X, which we denote μ, and which is defined by the following: $\mu(i) = x$ if the subsequence $(u_{\phi_i(n)})_{n\in\mathbb{N}}$ extracted from $(u_n)_{n\in\mathbb{N}}$ thanks to the function ϕ_i converges to x with exponential speed: $d(u_{\phi_i(n)}, x) < 2^{-n}$ holds for all n.

Theorem 5.2.22 can then be formulated equivalently as: the numberings ν and μ are $\equiv$-equivalent, i.e. the identity on X is both (ν, μ)-computable and (μ, ν)-computable. This is what we show now.

By Corollary 5.2.21, the sets of ν and μ computable points of X are identical, denote X_c this set.

A μ-description of a point x in X_c is the description of a computable Cauchy sequence that converges to x, with $g : n \mapsto 2^{-n}$ being a regulator for this sequence. The algorithm of passage to the limit of (X, d, ν) can thus be applied to this description with g as regulator, and it yields precisely a ν-description of x. This shows that the identity on X is (μ, ν)-computable.

To show that it is also (ν, μ)-computable, one only has to notice that the procedure described in the proof of Proposition 5.2.20 is uniform, in that it allows, given a ν-description of a point x, to produce a computable sequence extracted from $(u_n)_{n\in\mathbb{N}}$ which converges to x with the desired speed. This is precisely a μ-description for x. □

The following corollary to Theorem 5.2.22 shows that, when a Polish space can be equipped with an effective Polish space structure, this structure is unique.

Corollary 5.2.23. *Given a Polish space (X, d) with a dense sequence $(u_n)_{n\in\mathbb{N}}$, there is at most one numbering type on (X, d) which makes of it an effective Polish space and of $(u_n)_{n\in\mathbb{N}}$ a computable and dense sequence.*

Proof. This follows directly from Theorem 5.2.22. □

Theorem 5.2.22 allows for a very simple definition of what is an effective Polish space, that relies only on the distance between the elements of a dense sequence. Weihrauch and Moschovakis both used such definitions. We give here a definition that mimics that of Moschovakis, see **[44]** for the complete definitions of Weihrauch and Moschovakis, and their differences. Note however that the definition that we give is weaker than the ones given by Weihrauch and Moschovakis -a space that admits a recursive presentation in this sense also admits one following the definitions of Weihrauch and Moschovakis.

Definition 5.2.24. A *recursive presentation of a Polish space* (X, d) is a dense sequence $(u_n)_{n\in\mathbb{N}}$ of points of X such that the function

$$\begin{aligned} \phi : \mathbb{N} \times \mathbb{N} &\to \mathbb{R}_c \\ (n, m) &\mapsto d(u_n, u_m) \end{aligned}$$

is $(\mathrm{id}_{\mathbb{N}} \times \mathrm{id}_{\mathbb{N}}, \nu_C)$-computable.

The term presentation is from **[87]**, and has no relation the notion of a presentation for a group. The following proposition renders explicit the link between recursive presentations and effective Polish spaces.

Proposition 5.2.25. *A Polish space (X, d) admits a recursive presentation if and only if it admits a numbering that makes of it a effectively Polish space.*

Proof. If ν is a numbering of X that makes of (X,d,ν) an effective Polish space, it means that there exists a ν-computable dense sequence $(u_n)_{n\in\mathbb{N}}$. The function ϕ defined by $\phi(n,m)=d(u_n,u_m)$ is then computable, and thus $(u_n)_{n\in\mathbb{N}}$ defines a recursive presentation of (X,d).

Conversely, if $(u_n)_{n\in\mathbb{N}}$ is a sequence which is dense in X, the condition that $(n,m)\mapsto d(u_n,u_m)$ be computable exactly asks that the function

$$\mu:\mathbb{N}\to X$$
$$n\mapsto u_n$$

defines a numbering of X which makes of (X,d,μ) a recursive metric space. Then, the effective completion of μ defines a numbering ν of X, for which (X,d,ν) is an effectively Polish space. ☐

We will prove the following result:

Theorem 5.2.26. *The space of marked group, associated to the distance d, does not have a recursive presentation.*

This theorem appears in Section 5.3. We will also prove that the space of marked groups does not contain any dense and computable sequences of groups described by word problem algorithms, but Theorem 5.2.26 is more general because we do not suppose a priori that a dense sequence should consist in groups described by word problem algorithms.

5.2.4. Markov's Lemma and abstract continuity. We will give here a proof of the fact that computable functions on an effective Polish space are continuous, starting with Markov's Lemma, which is both very useful and very simple to use, and which will remain our main tool in the space of marked groups, since the stronger theorems are not applicable there.

We fix a recursive metric space (X,d,ν) which we suppose effectively complete. Denote by $\mathcal{A}_{lim}$ an algorithm of passage to the limit for it.

Lemma 5.2.27 (Markov, [**76, 77**]). *Suppose that a ν-computable sequence $(u_n)_{n\in\mathbb{N}}$ effectively converges in X to a computable point x. Suppose additionally that for any n, $u_n\neq x$. Then there is a ν-computable sequence $(w_p)_{p\in\mathbb{N}}$ of $X^{\mathbb{N}}$ such that: for each p, $w_p\in\{u_n,n\in\mathbb{N}\}\cup\{x\}$, and the set $\{p,w_p=x\}\subseteq\mathbb{N}$ is co-r.e. but not r.e..*

Proof. Consider an enumeration of all Turing machines M_0, M_1, ... To the machine M_p, we associate an effective sequence $(x_n^p)_{n\in\mathbb{N}}$ of points in X. To define $(x_n^p)_{n\in\mathbb{N}}$, start a run of the machine M_p with no input. While it lasts, the sequence $(x_n^p)_{n\in\mathbb{N}}$ should be identical to the sequence $(u_n)_{n\in\mathbb{N}}$. If at some point, the machine M_p stops, the sequence $(x_n^p)_{n\in\mathbb{N}}$ should become constant.

To sum this definition up, $(x_n^p)_{n\in\mathbb{N}}$ is defined as follows:

While (M_p does not stop) enumerate $(u_n)_{n\in\mathbb{N}}$.

If (M_p stops in k computation steps), set $x_n^p=u_k$ for $n\geq k$.

Each sequence $(x_n^p)_{n\in\mathbb{N}}$ is Cauchy, and in fact it converges at least as fast as the original sequence $(u_n)_{n\in\mathbb{N}}$. Thus the algorithm of passage to the limit $\mathcal{A}_{lim}$ can be applied to any sequence $(x_n^p)_{n\in\mathbb{N}}$, using the regulator of convergence of $(u_n)_{n\in\mathbb{N}}$.

The sequence $(w_p)_{p\in\mathbb{N}}$ is the sequence obtained by using the algorithm of passage to the limit on each sequence $(x_n^p)_{n\in\mathbb{N}}$, for $p\in\mathbb{N}$.

If follows directly from our definitions that if the machine M_p is non-halting, the sequence $(x_n^p)_{n\in\mathbb{N}}$ is identical to $(u_n)_{n\in\mathbb{N}}$, and thus w_p, which is its limit, is equal to x. On the other hand, if M_p halts in k computation steps, we have $w_p=u_k$ and thus w_p is different from x. ☐

We leave it to the reader to establish Proposition 5.2.6 thanks to Markov's Lemma.

Corollary 5.2.28. *Let f be a computable function between recursive metric spaces X and Y, suppose that X is effectively complete, and let $(x_n)_{n\in\mathbb{N}}$ be an effective sequence that effectively converges to a point x in X. Then the sequence $(f(x_n))_{n\in\mathbb{N}}$ converges to $f(x)$.*

Proof. This proof is not effective, as we proceed by contradiction. Suppose that the sequence $(f(x_n))_{n\in\mathbb{N}}$ does not converge to $f(x)$. Then there must exist a subsequence $(x_{\phi(n)})_{n\in\mathbb{N}}$ of $(x_n)_{n\in\mathbb{N}}$ and a rational $r>0$ such that

$$\forall n\in\mathbb{N}, d(f(x_{\phi(n)}),f(x))>r.$$

The existence of such a sequence, which need not a priori be computable, implies that there must also exist such a sequence where, additionally, the function $\phi:\mathbb{N}\to\mathbb{N}$ is computable. This follows from Proposition 5.2.20, because the set of terms of the sequence $(x_n)_{n\in\mathbb{N}}$ for which $d(f(x_n),f(x))>r$ holds is a r.e. set.

Finally, the function f can be used to distinguish between the elements of the sequence $(x_{\phi(n)})_{n\in\mathbb{N}}$ and its limit x, as, given a computable point u in X, it is possible to chose one which is true between $d(f(u),f(x))>r$ and $d(f(u),f(x))<r$, if we know a priori that $d(f(u),f(x))$ is not equal to r. This contradicts Markov's Lemma, and thus $(f(x_n))_{n\in\mathbb{N}}$ must converge to $f(x)$. $\square$

Say that a function f is *effectively discontinuous* if there exists an effective sequence $(u_n)_{n\in\mathbb{N}}$ of points of X that effectively converges to a point y of X, while the sequence $(f(u_n))_{n\in\mathbb{N}}$ stays away from $f(y)$, i.e. while there is $r>0$ such that for all n, $d(f(x_n),f(y))>r$.

The previous corollary thus directly implies what is also often referred to as Markov's Lemma:

Corollary 5.2.29. *A computable function $f:X\to Y$ between an effectively complete recursive metric space X and a recursive metric space Y cannot have an effective discontinuity.*

We can use the previous corollary to prove that, under the additional assumption that the space X be an effective Polish space, any computable function $f:X\to Y$ is continuous -not necessarily effectively so. This corollary of Markov's Lemma was first proven by Mazur for functions defined on intervals of $\mathbb{R}_c$ (see [**78**]). Note that Mazur uses a notion of computability for functions defined on metric spaces that differs for the one we use throughout, and which is known as Banach-Mazur computability. However, the proof of his theorem is identical, when applied either to Markov computable functions, or to Banach-Mazur computable functions.

Corollary 5.2.30 (Mazur's Continuity Theorem). *Consider a computable function $f:X_c\to Y$ defined on the computable points of an effective Polish space X and with codomain a recursive metric space Y. Then f is continuous.*

Proof. Denote $(u_n)_{n\in\mathbb{N}}$ an effective and dense sequence of X.

Suppose that f is not continuous at a point x of X. This means that there exists a sequence $(x_n)_{n\in\mathbb{N}}$ and a real number $r>0$ such that $(x_n)_{n\in\mathbb{N}}$ converges to x, but for any $n\in\mathbb{N}$, $d(f(x_n),f(x))>r$. Any point of $(x_n)_{n\in\mathbb{N}}$ is the limit of an effective subsequence of $(u_n)_{n\in\mathbb{N}}$, by Proposition 5.2.20. And thus, by Corollary 5.2.28, for each point x_k of the sequence $(x_n)_{n\in\mathbb{N}}$, there must exist a point $u_{\phi(k)}$ in the sequence $(u_n)_{n\in\mathbb{N}}$, such that both inequalities $d(u_{\phi(k)},x_k)<2^{-k}$ and $d(f(u_{\phi(k)}),f(x))>r$ hold.

Thus there exists a subsequence $(u_{\phi(n)})_{n\in\mathbb{N}}$ of $(u_n)_{n\in\mathbb{N}}$, which converges to x and such that for any $n\in\mathbb{N}$, $d(f(u_{\phi(n)}),f(x))>r$. This subsequence is a priori not computable, but, by Proposition 5.2.20, the abstract fact that such a sequence exists automatically implies that there must also exist such a subsequence that is, in addition, both computable and effectively converging to x.

Finally, the existence of such a computable sequence is an obvious contradiction to Markov's Lemma, and thus f must be continuous. $\square$

5.2.5. Differences with the Borel hierarchy.

5.2.5.1. *Computable but discontinuous function.* We give here an example of a Markov computable function that is not continuous. This is done by considering a function defined on a peculiar domain. The fact that those exist is well known, we explain it here in terms of Kolmogorov complexity, following [**58**]. Some introductory remarks about Kolmogorov complexity appear in Chapter 1.

We set ourselves in the Cantor space $\{0;1\}^{\mathbb{N}}$, but this could be done in $\mathbb{R}_c$ as well.

Consider a sequence $(u_n)_{n\in\mathbb{N}}$ of finite strings of zeroes and ones, such that the length of u_n is n, and which has linear asymptotic Kolmogorov complexity: $K(u_n)\underset{n\to\infty}{\sim}n$. The existence of such a sequence was proven in Proposition 1.4.4. Consider now the sequence $v_n=0^n1u_n00000.....$ of elements of $\{0;1\}^{\mathbb{N}}$. This sequence also has linear asymptotic Kolmogorov complexity: a single Turing Machine can transform any element v_n into the corresponding u_n, this implies that asymptotically $K(u_n)\preceq K(v_n)$ (see Proposition 1.4.2). The other inequality is obvious as well.

We call $\mathcal{A}$ the subset of $\{0;1\}^{\mathbb{N}}$ consisting of the null sequence and of the set $\{v_n,n\in\mathbb{N}\}$.

Proposition 5.2.31. *The function $\delta_0:\mathcal{A}\to\{0;1\}$ which sends the null sequence to 1 and all other sequences to 0 is computable on $\mathcal{A}$. However, it is discontinuous.*

Proof. Because $K(v_n)\underset{n\to\infty}{\sim}n$, there must exist an integer $b\in\mathbb{Z}$ such that for all n, $K(v_n)>\frac{n}{2}+b$.

We now show how to compute δ_0. Let M_x be a Turing Machine that codes for an element x in $\mathcal{A}$. Denote by k the number of states of this machine. The element x has Kolmogorov complexity at most k, and thus either it is the null sequence, or, if it can be written v_n for some n, we have $k>\frac{n}{2}+b$, and thus $n<2(k-b)$.

But since the element v_n agrees with the null sequence only on its first n terms, this means that if x is not the null sequence, one of its first $2(k-b)$ digits must be a one. This can be easily checked, using the Turing Machine M_x until it has written the first $2(k-b)$ digits of x. ☐

The following question however remains open:

PROBLEM 5.2.32. Characterize those sequences $(u_n)_{n\in\mathbb{N}}$ converging to the null sequence in the Cantor space for which any Markov computable function defined on $\{u_n, n\in\mathbb{N}\}\cup\{0^\omega\}$ has to be continuous.

Note that peculiar instances of this problem naturally arise in our study: in Example 5.4.7, the group $L(\mathcal{A}_3)$ constructed in Chapter 4 which is not effectively residually finite is shown to satisfy the following property:

$L(\mathcal{A}_3)$ has solvable word problem, it is the limit in $\mathcal{G}$ of a sequence of its finite quotients, but any sequence of its finite quotients that converges to $L(\mathcal{A}_3)$ must be non computable.

Because of this, the question "can $L(\mathcal{A}_3)$ be distinguished from its finite quotients using word problem algorithms?" cannot be settled thanks to Markov's Lemma.

5.2.5.2. *A semi-decidable set that is not open.* The previous example was obtained by considering a function defined on a set with bad properties. On an effective Polish space, the decidable sets must be clopen, since their characteristic function must be continuous. By Markov's Lemma, the Markov semi-decidable sets cannot be "effectively not-open": if x is a point of a semi-decidable set X, and if $(u_n)_{n\in\mathbb{N}}$ is an effective sequence that effectively converges to x, then infinitely many elements of this sequence must belong to X. One might wonder whether this result can be strengthened to: "the ν-semi-decidable sets on an effective Polish space (X,d,ν) are open". An example of Friedberg (**[40]**) shows that this is not the case, we reproduce here the account of this result from **[58]**, which renders explicit the role of Kolmogorov complexity in the construction of this example.

This example is set in the Cantor space $\{0,1\}^{\mathbb{N}}$. For w an element of $\{0,1\}^*$, denote by $[w]$ the clopen set of all sequences that start by w.

THEOREM 5.2.33 (Friedberg, see **[58]**, Theorem 4.1). *On the Cantor space, the set*

$$A=\{0^\omega\}\cup\bigcup_{n:K(n)<\frac{\log(n)}{2}}[0^n1]$$

is semi-decidable but not open.

PROOF. A is not open, as it does not contain a neighborhood of 0^ω, because infinitely often in n one has $K(n)\geq\frac{\log(n)}{2}$.

We now show that A is semi-decidable.

There exists a program T that maps any element x of the Cantor space that is different from 0^ω to the number of zeroes that appear at the beginning of x.

We are now given a computable point x of $\{0,1\}^{\mathbb{N}}$.

The description of x by a Turing machine that produces it gives an upper bound K_0 on the Kolmogorov complexity of x. Noting l the length of the program T defined above, one has that either x is 0^ω, or, if x can be decomposed as $x=0^n1x'$, it must be that $K(n)\leq K(x)+l\leq K_0+l$.

Start enumerating x. If x starts with more than $2^{2(K_0+l)}$ zeros, then either it is the sequence 0^ω, or it can be written as $x=0^n1x'$, with $\frac{\log(n)}{2}>K_0+l$, and thus with $\frac{\log(n)}{2}>K(n)$. In any case, we know that x belongs to A, without having to compute n.

If x starts with less than $2^{2(K_0+l)}$ zeroes, a number n such that x can be rewritten as $x=0^n1x'$ can be effectively found. From this, to determine whether x belongs to A, one only needs to check whether $K(n)<\frac{\log(n)}{2}$ holds, which can be proven whenever it holds, as the Kolmogorov complexity is upper-semi-computable. ☐

Other examples of non-open but semi-decidable sets can be found in **[58]**. It is however clear that those examples are artificially built, and this justifies the heuristic which says that *a natural* semi-decidable property can be expected to be open.

Note finally that although we have just seen that a semi-decidable subset of an effectively Polish space does not have to be open, it must share the following property of open sets: it meets any computable and dense sequence.

PROPOSITION 5.2.34 (Moschovakis, **[85]**, Theorem 4). *Let $(X,d,\nu,(u_n)_{n\in\mathbb{N}})$ be an effectively Polish space. A non-empty ν-semi-decidable subset of X must intersect the dense sequence $(u_n)_{n\in\mathbb{N}}$.*

The proof of this result is very close to that of Mazur's Continuity Theorem, Corollary 5.2.30, we do not present it here. A consequence of this fact, pointed out in **[58]**, is the following:

COROLLARY 5.2.35 (Hoyrup, Rojas, [**58**], Section 4, Proposition 3). *In an effective Polish space $(X, d, \nu, (u_n)_{n\in\mathbb{N}})$, there is an algorithm that takes as input the code for a ν-semi-decidable set D and stops if and only if this set is non-empty.*

In case the input set is a non-empty semi-decidable set, this algorithm can produce the ν-name for a point in this set.

PROOF. Just enumerate the sequence $(u_n)_{n\in\mathbb{N}}$ in search of a point in D, by Proposition 5.2.34, D is non-empty if and only if it contains a point from $(u_n)_{n\in\mathbb{N}}$. $\square$

Note that the previous proposition contains an effective version of the Axiom of Choice for effectively Polish spaces: a possible formulation of AC is that for any set X, there exists a choice function that maps a non-empty subset of X to a point in this subset. The previous result implies that a computable choice function exists for semi-decidable subsets of an effective Polish space. We will give in Theorem 5.3.10 a strong negation of such an effective Axiom of Choice for the space of marked groups: there does not exist a computable function that, given a non-empty basic clopen set $\Omega^k_{r_1,\dots,r_m;s_1,\dots,s_{m'}}$, can produce a point in this set.

5.2.6. Kreisel, Lacombe and Schoenfield and Ceitin Theorems, Moschovakis' addendum.

5.2.6.1. *Theorems of Kreisel, Lacombe, Schoenfield, Ceitin.* The following theorem is probably the most important theorem in computable analysis. It was first proved by Kreisel, Lacombe and Schoenfield in 1957 in [**62**] in the case of functions defined on the Baire space $\mathbb{N}^{\mathbb{N}}$, and obtained independently by Ceitin in 1962 (an English version of this theorem can be found in [**20**]), in the more general setting of effective Polish spaces.

THEOREM 5.2.36 (Ceitin). *A computable function defined on the computable points of an effective Polish space (with images in any RMS) is effectively continuous.*

Moreover, for each pair constituted of an effective Polish space and any RMS, there is an algorithm that takes as input the description of a computable function defined between those spaces, and produces a program that will attest for the effective continuity of this function.

The original motivation of Kreisel, Lacombe and Schoenfield to prove this theorem was to prove that two notions of computability coincided: Markov computability, which is the one we have been working with up to now, and what is now called Borel computability (the term comes from [**2**]).

We will not define Borel computability in metric spaces in its most generic setting, which is that of *represented spaces,* a notion similar to that of a numbered space, except that the indexing of elements is done by elements of the Baire space $\mathbb{N}^{\mathbb{N}}$. (See for instance Weihrauch's book [**129**].)

It is however easy to define it on explicit spaces using oracle Turing machines. On the Cantor space, for instance, a function $f : \{0,1\}^\omega_c \to \{0,1\}^\omega_c$ is called *Borel computable* if there exists a Turing machine which, given an oracle Φ_x that produces a computable sequence x ($\Phi_x(n)$ is the n-th term of x), is able to compute the digits of the image of x by the function f.

A Borel computable function is automatically Markov computable, because a Turing machine that describes a sequence can be used to simulate an oracle, running the Turing machine produces the same answers as an oracle would. On the other hand, the description of a computable sequence by a machine could provide additional information on a sequence, information that would be read off of this machine.

The notion of Borel computability renders precise the idea that no "meta" information should be gleaned from the description of a sequence by a Turing machine. In [**58**], it was shown that the additional information about a sequence that is contained in the description of program that computes it, but not in an oracle, is precisely an upper bound on the Kolmogorov complexity of this sequence (this statement is made precise in [**58**]). This idea was already used in Section 5.2.5.

The following result is a consequence of the Effective Continuity Theorem.

THEOREM 5.2.37. *A function defined on the computable points of an effective Polish space is Markov computable if and only if it is Borel computable.*

Borel computability on the space of marked groups can similarly be defined in terms of oracle Turing machines: a function is Borel computable if it is computable given access to an oracle Turing Machine that solves the word problem in the input group. The previous result leads us to asking:

PROBLEM 5.2.38. Must a Markov computable function defined on $\mathcal{G}$ be Borel computable?

5.2.6.2. *Theorems of Moschovakis.* In 1964, Moschovakis gave a new proof of Ceitin's Theorem, at the same time providing the only known effective continuity result set in a more general context than that of effective Polish spaces.

In order to state it, we must first introduce some notions.

In what follows, (X, d, ν) denotes a recursive metric space.

DEFINITION 5.2.39. We say that (X, d, ν) satisfies *Moschovakis' condition (B)* if there is an algorithm that, given the code of a ν-semi-decidable set $A \subseteq X$, and an open ball $B(x, r)$, described by a ν-name for x and a ν_C-name for $r \in \mathbb{R}_c$, such that $A \cap B(x, r) \neq \emptyset$, will produce the ν-name of a point y in the intersection $A \cap B(x, r)$.

Note that in this definition, the algorithm is always given as input a pair of intersecting sets, it then produces a point in the intersection. This algorithm is not supposed to be able to determine whether or not two given sets intersect. This definition is an effective Axiom of Choice, similar to the one described in Proposition 5.2.34.

And an easy consequence of Proposition 5.2.34 is the following:

PROPOSITION 5.2.40. *An effectively Polish space satisfies Moschovakis' condition (B).*

We can now state Moschovakis' Theorem on the effective continuity of computable functions.

THEOREM 5.2.41 (Moschovakis, [**85**], Theorem 3). *A computable function defined on the computable points of an effectively complete RMS that satisfies Moschovakis' condition (B) and with codomain any RMS is effectively continuous.*

Moreover, for each pair constituted of such a space and of any RMS, there is an algorithm that takes as input the description of a computable function defined between those spaces, and produces a program that will attest for the effective continuity of this function.

We will prove in Corollary 5.3.15 that the hypotheses of this theorem fail for the space of marked groups, leaving open the conjecture which says that computable functions on $\mathcal{G}$ should be effectively continuous.

Another important theorem obtained by Moschovakis in [**85**], in the setting of effective Polish spaces, gives rise in $\mathcal{G}$ to a conjecture about semi-decidable open sets. Two more definitions are required to state this theorem.

DEFINITION 5.2.42. In a RMS (X, d, ν), a *Lacombe* set is a set of the form

$$\bigcup_{n \in \mathbb{N}} B(x_n, r_n),$$

where the sequence $(x_n)_{n\in\mathbb{N}}$ is a ν-computable sequence of points of X, and the sequence $(r_n)_{n\in\mathbb{N}}$ is a ν_C-computable sequence of computable reals.

There is no unanimously accepted definition for what should be an "effectively open set", we give here Moschovakis' definition.

DEFINITION 5.2.43. In a RMS (X, d, ν), an *effectively open* set is a semi-decidable set O such that there is an algorithm that, given the ν-name of a point x in O, produces the ν_C-name of a computable real r such that $B(x, r) \subseteq O$.

The following result of Moschovakis answered a question of Ceitin.

THEOREM 5.2.44 (Moschovakis, [**85**], Theorem 11). *In an effective Polish space, every effectively open set is a Lacombe set.*

What's more, there exists a primitive recursive function that transforms a description of a set as an effectively open set into a description for it as a Lacombe set.

We ask:

PROBLEM 5.2.45. Must every effectively open set in $\mathcal{G}$ be a Lacombe set?

We omit the proofs of the results quoted in this section, an account of those proofs can be found for instance in Chapter 9 of [**63**].

The fact that the proofs of the results that appear in this section need to be fundamentally different from the ones that appeared in Subsection 5.2.4 (Markov's Lemma and the non-effective continuity theorem) can be seen as a consequence of a theorem, due to Friedberg on $\mathbb{N}^{\mathbb{N}}$, and to Hertling for functions defined on $\mathbb{R}_c$, about Banach-Mazur computable functions.

We define Banach-Mazur computability now.

Definition 5.2.46. Let (X, ν) and (Y, μ) be two numbered sets. A function $f : X \to Y$ is (ν, μ)-*Banach-Mazur-computable* if it maps ν-computable sequences to μ-computable sequence.

Theorem 5.2.47 (Hertling, [**55**], 2005). *There exists a Banach-Mazur computable function defined on $\mathbb{R}_c$ that is not effectively continuous. In particular, this function is not Markov computable, and it does not admit a finite description in terms of a Turing machine.*

All the results that we proved in Subsection 5.2.4 in fact only relied on Banach-Mazur computability: each proof concluded by exhibiting a single computable sequence whose image would not be a computable sequence. See for instance [**54**]. Hertling's result shows that the proof of the effective continuity theorem cannot rely on the same methods.

While the results of Hertling and Friedberg show that the notion of Banach-Mazur computability does not coincide with the intuitive notion of "computability" for functions of a real variable, they show that the notion of Banach-Mazur computability has an important theoretical role in the study of Markov computable functions, allowing one to determine what hypotheses are needed to prove different theorems.

5.3. The space of Marked Groups as a recursive metric space

5.3.1. Non effective separability of $\mathcal{G}$ and recursive presentations of Polish spaces. The following fact was remarked in [**27**].

Proposition 5.3.1. *The Λ_{WP}-computable points of $(\mathcal{G}, d, \nu_{WP})$ are not dense in it.*

Proof. This follows directly from Theorem 5.3.12, which is due to Miller, and which we have already quoted in a previous section: there exists a coherent set of relations and irrelations that is not wp-coherent. ☐

This shows that the study of computability on the space of marked groups could be more precisely set in the closure of the set of groups with solvable word problem, the structure of an open set in $\mathcal{G}$ which contains no group with solvable word problem has no bearing on the present study.

Denote $\mathcal{G}_{WP}$ the closure of the set of (markings of) groups with solvable word problem in $\mathcal{G}$. By definition, the computable points of $\mathcal{G}$ are dense in $\mathcal{G}_{WP}$. However, we have the following theorem:

Theorem 5.3.2. *No sequence of marked groups can be both computable and dense in $\mathcal{G}_{WP}$.*

Proof. This is a simple application of Theorem 5.1.2, together with Corollary 5.2.35. Corollary 5.2.35 states that in an effectively Polish space, there is an algorithm that stops exactly on semi-decidable sets that are non-empty.

The basic clopen sets $\Omega_{r_i;s_j}$ are obviously semi-decidable in $\mathcal{G}_{WP}$, but a program that recognizes those basic clopen sets that are non-empty would allow one to recognize wp-coherent sets of relations and irrelations, contradicting Theorem 5.1.2. ☐

It is interesting to interpret this proof as a variation on McKinsey's algorithm for finitely presented residually finite groups. Notice that if X is a set of marked groups which is dense in $\mathcal{G}_{WP}$, then every finitely presented group with solvable word problem is "residually-X", and a proof similar to McKinsey's would then contradict the Boone and Rogers Theorem ([**13**]).

This proposition directly implies the following:

Corollary 5.3.3. *The recursive metric space $(\mathcal{G}_{WP}, d, \nu_{WP})$ is a Polish space that is effectively complete but not effectively separable, and thus it is not an effective Polish space.*

As we have already seen in the previous section, Mazur's Continuity Theorem and Ceitin's Effective Continuity Theorem both apply to effective Polish spaces. This corollary thus shows that those theorems cannot be directly applied to the space of marked groups.

We now prove a slightly more general result, which is not attached to a particular numbering type of $\mathcal{G}$.

Theorem 5.3.4. *The space of marked groups $(\mathcal{G}, d)$ does not have a recursive presentation in the sense of Definition 5.2.24.*

Proof. Recall that a recursive presentation of $(\mathcal{G}, d)$ would consist in a sequence $(u_n)_{n\in\mathbb{N}}$, dense in $\mathcal{G}$, and for which the distance between the n-th and m-th terms is computable.

If $\mathcal{G}$ admitted a recursive presentation, then so would $\mathcal{G}_k$ for any $k \geq 1$. Thus we only have to show that $\mathcal{G}_2$ does not have a recursive presentation as a Polish space.

Recall that we have defined an embedding $\Phi_2 : \mathcal{G}_2 \to \{0,1\}^\omega$ by fixing a computable order on the rank two free group. Call a set $r_1, ..., r_m;\ s_1, ..., s_{m'}$ of relations and irrelations *initial* if the $m + m'$ elements of the free group it contains are exactly the first $m + m'$ elements of this order.

We will prove that a recursive presentation of $\mathcal{G}_2$ would allow one to compute, given an integer n, the number of initial coherent sets of relations and irrelations that contain n relations.

We first show that this is sufficient to obtain a contradiction.

There are exactly 2^n possible initial sets of relations and irrelations that contain n relations. Since the incoherent sets of relations and irrelations form a r.e. set, if we had access to the number of initial coherent sets of relations and irrelations that contain n relations, we would be able to compute exactly those sets, by starting with the 2^n possible initial sets, and deleting incoherent ones until the number of coherent sets is attained.

But being able to compute the *initial* coherent sets of relations and irrelations in fact also allows one to compute all coherent sets of relations and irrelations, because a set $r_1, ..., r_m; s_1, ..., s_{m'}$, which is not initial, is coherent if and only if there is an initial and coherent set which contains the elements $r_1, ..., r_m$ as relations, and the elements $s_1, ..., s_{m'}$ as irrelations. Choosing n big enough, it would then suffice to construct all initial sets of relations and irrelations of length n to determine whether $r_1, ..., r_m; s_1, ..., s_{m'}$ is coherent. And we have seen that this is impossible by the Boone-Novikov theorem.

Suppose that $(u_n)_{n\in\mathbb{N}}$ defines a recursive presentation of $\mathcal{G}_2$, we show how to compute the number of initial coherent sets of relations and irrelations that contain n relations. Denote by $\lambda(n)$ this number. Again, because the incoherent sets of relations and irrelations form a r.e. set, λ is an upper semi-computable function: there exists a computable function $\lambda^>$ that, given n, produces a computable and decreasing sequence of integers that converges to $\lambda(n)$. What we show is that the existence of a recursive presentation of $\mathcal{G}_2$ implies that λ is also lower semi-computable, meaning that there exists a computable function $\lambda^<$ that, on input n, produces an increasing sequence of integers which converges to $\lambda(n)$.

Given i and n natural numbers, define x_i^n to be the maximal size of a subset of $\{u_0, u_1, ..., u_i\}$ of which any two elements are at least 2^{-n} apart.

We claim that $(i, n) \mapsto x_i^n$ is a computable function, and that, for any n, $i \mapsto x_i^n$ is an increasing function that converges to $\lambda(n)$.

Setting $\lambda^<(n) = (x_i^n)_{i\in\mathbb{N}}$ will then conclude the proof.

As the distance function d takes values only in $\{0\}\cup\{2^{-n}, n \in \mathbb{N}\}$, given the description of the distance $d(u_i, u_j)$ as a computable real, one can always effectively choose one of $d(u_i, u_j) < 3\times 2^{-n-2}$ and $d(u_i, u_j) > 3\times 2^{-n-2}$ which holds, and thus decide whether or not u_i and u_j are 2^{-n} apart. One can thus check every subset of $\{u_0, u_1, ..., u_i\}$ to find one of maximal size, all the elements of which are 2^{-n} apart. Thus $(i, n) \mapsto x_i^n$ is computable.

The function $(i, n) \mapsto x_i^n$ is increasing in i by definition.

Finally, we show that x_i^n goes to $\lambda(n)$ as i goes to infinity. Two points of $\mathcal{G}_2$ are at least 2^{-n} apart if and only if their binary expansion differ on one of their first n digits: those groups must be associated to different initial sets of relations and irrelations of length n. Because the sequence $(u_n)_{n\in\mathbb{N}}$ is supposed to be dense in $\mathcal{G}_2$, for any coherent initial set of relations and irrelations of length n, there should be a point of this sequence which satisfies those relations and irrelations.

And thus there must indeed exist a set of $\lambda(n)$ points in the dense sequence which are pairwise 2^{-n} apart, and this number is clearly maximal. □

5.3.2. Optimality of the numbering ν_{WP}. Note that Theorem 5.3.4, which states that the space of marked groups does not have a recursive presentation, can be seen, through Proposition 5.2.25, as answering the question: "which numberings of $\mathcal{G}$ make of $(\mathcal{G}, d)$ an effectively Polish space?". The answer to this question is that no numbering can make of $(\mathcal{G}, d)$ an effectively Polish space.

We will now include some results that ask which numberings of $\mathcal{G}$ can make of it a recursive metric space.

The precise question we want to ask is: what are the numberings that contain the least possible amount of information, while still making of $\mathcal{G}$ a RMS? And in particular, is ν_{WP} optimal in this sense?

A formal statement that would express the optimality of ν_{WP} could be: "any numbering ν that makes of $(\mathcal{G}, d, \nu)$ a recursive metric space satisfies $\nu \succeq \nu_{WP}$". And one could even ask for such a result, replacing the distance d by any distance that generates the topology of $\mathcal{G}$.

Mind however that this statement is false.

Indeed, given a marked group G with unsolvable word problem and isolated from groups with solvable word problem, as provided by Proposition 5.3.1, we can define a new numbering ν_G of $\mathcal{G}$, defined by $\nu_G(0) = G$ and, for $n > 0$, $\nu_G(n) = \nu_{WP}(n-1)$. Because G is isolated from groups with solvable word problem, there is a natural number n_G such that any group H that satisfies $d(G, H) < 2^{-n_G}$ also has unsolvable word problem. Thus it suffices to know the first n_G terms in the binary expansion of n_G to be able to compute the distance $d(G, H)$ from G to any marked group H with solvable word problem.

And thus the new numbering ν_G also makes of $(\mathcal{G}, d, \nu_G)$ a recursive metric space, in which G is a computable point, and one has $\nu_{WP} \succeq \nu_G$ and $\nu_G \not\succeq \nu_{WP}$.

Notice however that the numbering ν_G described above is not *saturated,* in the sense that the marked group G is ν_G-computable, but it is the only ν_G-computable point in the set $[G]$. We thus ask:

PROBLEM 5.3.5. Suppose that ν is a saturated numbering of $\mathcal{G}$, and that $(\mathcal{G}, d, \nu)$ is a RMS. Must one have $\nu \succeq \nu_{WP}$?

We however have a theorem that is close enough to the false statement given above.

THEOREM 5.3.6. *Suppose that $\hat{d}$ is any distance on $\mathcal{G}$ that generates the same topology as d, and that μ is a numbering of $\mathcal{G}$ such that $(\mathcal{G}, \hat{d}, \mu)$ is a RMS.*

Suppose furthermore that there is an algorithm that takes as input a μ-name for a k-marked group G and a ν_C-name for a radius $r > 0$, and produces a basic clopen set $\Omega^k_{R;S}$ (described by a pair of tuples of elements of the free group) such that:

$$G \in \Omega^k_{R;S};$$

$$\Omega^k_{R;S} \subseteq B_{\hat{d}}(G, r).$$

Then one has $\mu \succeq \nu_{WP}$.

Before proving this theorem, let us note that the hypotheses of this theorem can be explained as follows: we ask not only that the distance $\hat{d}$ define the usual topology of $\mathcal{G}$, but also that it define its usual *effective topology.* What we suppose for Theorem 5.3.6 is precisely that any effective open set of $(\mathcal{G}, \hat{d}, \mu)$ should also be effectively open for the topology defined by the basic clopen sets $\Omega^k_{R,S}$.

A *discriminating family* of a group G is a subset of G which does not contain the identity element of G, and which intersects any non-trivial normal subgroup of G. We will use Theorem 3.4 from [**27**], which is an analysis of Kuznetsov's method for solving the word problem in simple groups ([**64**]):

THEOREM 5.3.7 (Cornulier, Guyot, Pitsch, [**27**]). *A group has solvable word problem if and only if it is both recursively presentable and recursively discriminable.*

And this statement is uniform: there is an effective method that allows, given a recursive presentation and an algorithm that enumerates a discriminating family in a marked group G, to find a word problem algorithm for G.

We add the statement about the uniformity of this theorem, but it is easy to see that the proof given in [**27**] is uniform.

PROOF OF THEOREM 5.3.6. Consider a μ-computable marked group G. We show, given a μ-name for G, how to obtain a word problem algorithm for it.

Using the algorithm given by the hypotheses of the theorem, consecutively on each ball $B_{\hat{d}}(G, \frac{1}{n})$, we obtain a computable sequence $(\Omega_{R_n;S_n})_{n\in\mathbb{N}}$ of basic clopen subsets, such that for each n we have:

$$G \in \Omega_{R_n;S_n} \subseteq B_{\hat{d}}(G, \frac{1}{n}).$$

It follows that $\bigcap \Omega_{R_n;S_n} = \{G\}$, and thus that the union $\bigcup_{n\in\mathbb{N}} R_n$ defines a recursively enumerable set of relations that defines G, and that the set $\bigcup_{n\in\mathbb{N}} S_n$ defines a recursively enumerable discriminating family for G.

We can thus apply Theorem 5.3.7, which indicates that a word problem algorithm for G can be obtained from this data. □

REMARK 5.3.8. Any numbering that is stronger than ν_{WP} also makes of the space of marked groups a recursive metric space. An example of particular importance is the numbering associated to the description "finite presentation and word problem algorithm", which we denote $\nu_{WP} \wedge \nu_{FP}$, and whose study, as we stated in Chapter 2, is of foremost importance for group theory. Notice that $\{(\mathbb{Z}, 1)\}$ is a $\nu_{WP} \wedge \nu_{FP}$-decidable singleton. This shows that Markov's Lemma cannot be applied with respect to the numbering $\nu_{WP} \wedge \nu_{FP}$: the sequence $(\mathbb{Z}/n\mathbb{Z}, 1)$ is a $\nu_{WP} \wedge \nu_{FP}$-computable sequence that converges effectively to $(\mathbb{Z}, 1)$, but $(\mathbb{Z}, 1)$ can still be distinguished from finite groups with respect to $\nu_{WP} \wedge \nu_{FP}$.

The space of marked groups associated to the numbering $\nu_{WP} \wedge \nu_{FP}$ is thus a recursive metric space that is *not* effectively complete.

5.3.3. Two applications of a construction of Miller, failure of Moschovakis' (B) condition for the space of marked groups. In this section, we prove two important theorems that use variations on Miller's example of a finitely presented group that is isolated from groups with solvable word problem.

The following theorem was already stated, it is Theorem 5.1.3.

THEOREM 5.3.9. *No algorithm can stop exactly on those sets of relations and irrelations which are not wp-coherent.*

The following theorem is one of our most important results, we have already explained how it represents the negation of an effective Axiom of Choice for the space of marked groups.

THEOREM 5.3.10 (Failure of an Effective Axiom of Choice for groups). *There is no algorithm that, given a wp-coherent set of relations and irrelations, produces a word problem algorithm for a marked group that satisfies those relations and irrelations.*

The proofs for those results will be similar: they rely on Miller's constructions of a family of groups $L_{P,Q}$ indexed by two subsets P and Q of $\mathbb{N}$. For each of those theorems, we will find some conditions on the sets P and Q that are sufficient for the groups $L_{P,Q}$ to provide a proof of the Theorems 5.3.9 and 5.3.10, and then include a lemma to prove that such sets do exist.

We start by detailing Miller's construction.

5.3.3.1. *Miller's construction.* We detail the construction of Miller as it was exposed in [**83**]. This construction was first introduced in [**82**].

Step 1. Given two subsets P and Q of $\mathbb{N}$, we consider the group $L^1_{P,Q}$ given by the following presentation:

$$\langle e_0, e_1, e_2, ... | e_0 = e_i,\, i \in P,\, e_1 = e_j,\, j \in Q\rangle$$

For simplicity, we shall always assume that P contains 0 and Q contains 1.

Notice that $L^1_{P,Q}$ is recursively presented with respect to the family $(e_i)_{i\in\mathbb{N}}$ if and only if P and Q are r.e. sets, and that $L^1_{P,Q}$ has solvable word problem with respect to the family $(e_i)_{i\in\mathbb{N}}$ if and only if P and Q are recursive sets.

In what follows, the sets P and Q will always be recursively enumerable, and thus $L^1_{P,Q}$ is recursively presented.

Step 2. Embed the recursively presented group $L^1_{P,Q}$ in a finitely presented group $L^2_{P,Q}$ using some strengthening of Higman's Embedding Theorem. For our purpose, we need to know that:

- A finite presentation of $L^2_{P,Q}$ can be obtained from the recursive presentation of $L^1_{P,Q}$;
- If the group $L^1_{P,Q}$ has solvable word problem with respect to the family $(e_i)_{i\in\mathbb{N}}$, then the group $L^2_{P,Q}$ also has solvable word problem;
- The embedding of $L^1_{P,Q}$ into $L^2_{P,Q}$ is effective, i.e. there exists a recursive function that maps a natural number n to a way of expressing the element e_n as a product of the generators of $L^2_{P,Q}$.

Clapham's version of Higman's Embedding Theorem ([**25**]) satisfies the required conditions for this step of the construction. Clapham's Theorem is quoted precisely in Subsection 5.7.1.

Step 3. Embed the group $L^2_{P,Q}$ into a finitely presented group $L^3_{P,Q}$ with the following property: in any non-trivial quotient of $L^3_{P,Q}$, the image of the element $e_0e_1^{-1}$ is a non-identity element.

This is done as follows.

Consider a presentation $\langle x_1, ..., x_k | r_1, ...r_t\rangle$ for $L^2_{P,Q}$, denote w a word on $\{x_1, ..., x_k, x_1^{-1}, ..., x_k^{-1}\}$ that defines the element $e_0e_1^{-1}$ in $L^2_{P,Q}$. The group $L^3_{P,Q}$ is defined by adding to $L^2_{P,Q}$, in addition to the generators $x_1, ..., x_k$ that are still subject to the relations $r_1, ...r_t$, three new generators a, b and c, subject to the following relations:

(1) $a^{-1}ba = c^{-1}b^{-1}cbc$
(2) $a^{-2}b^{-1}aba^2 = c^{-2}b^{-1}cbc^2$
(3) $a^{-3}\,[w, b]\,a^3 = c^{-3}bc^3$
(4) $a^{-(3+i)}x_iba^{(3+i)} = c^{-(3+i)}bc^{(3+i)}$, $i = 1..k$

To use Miller's construction, we need to check the following points:

- If $w \neq 1$ in $L^2_{P,Q}$, then $L^2_{P,Q}$ is embedded in $L^3_{P,Q}$ via the natural map $x_i \mapsto x_i$.
- The presentation of $L^3_{P,Q}$ can be computed from the presentation of $L^2_{P,Q}$ together with the word w.
- If $L^2_{P,Q}$ has solvable word problem, then so does the group $L^3_{P,Q}$.
- The element $e_0e_1^{-1}$ has a non-trivial image in any non-trivial quotient of $L^3_{P,Q}$.

The second point is obvious. The last point is easily proven: remark that the third written relation, together with $w = 1$, implies the relation $b = 1$. This in turn implies that $c = 1$ thanks to the first relation, that $a = 1$ thanks to the second relation, and then that all x_i also define the identity element because of the relations of (4).

The first and third points are proven using the fact that the group $L^3_{P,Q}$ can be expressed as an amalgamated product.

Consider the free product $L^2_{P,Q} * \mathbb{F}_{a,b}$ of $L^2_{P,Q}$ with a free group generated by a and b, and the free group $\mathbb{F}_{b,c}$ generated by b and c. Then, provided that $w \neq 1$ in $L^2_{P,Q}$, the subgroup of $L^2_{P,Q} * \mathbb{F}_{a,b}$ generated by b and the elements that appear to the left hand side in the equations $(1) - (4)$ is a free group on $4 + k$ generators, which we denote A, and so is the subgroup B of $\mathbb{F}_{b,c}$ generated by b and the elements that appear to the right hand side in the equations $(1) - (4)$.

Thus the given presentation of $L^3_{P,Q}$ shows that it is defined as an amalgamated product of the form:

$$\begin{array}{ccc} (L^2_{P,Q} * \mathbb{F}_{a,b}) & * & \mathbb{F}_{b,c} \\ A & = & B \end{array}$$

This proves both the fact that $L^2_{P,Q}$ embeds in $L^3_{P,Q}$, and that the word problem is solvable in $L^3_{P,Q}$ as soon as it is in $L^2_{P,Q}$, since to solve the word problem in an amalgamated product such as $L^3_{P,Q}$, it suffices to be able to solve the membership problem for A in $L^2_{P,Q} * \mathbb{F}_{a,b}$ and for B in $\mathbb{F}_{b,c}$, we leave it to the reader to see that this can be done as soon as the word problem is solvable in $L^2_{P,Q}$.

Finally, we designate by $\Pi_{P,Q}$ the finite set of relations and irrelations that is composed of the relations of $L^3_{P,Q}$, and of a unique irrelation $w \neq 1$, where w is the word that defines the element $e_0 e_1^{-1}$ in $L^3_{P,Q}$.

Note that the set $\Pi_{P,Q}$ can be effectively produced from the codes for the r.e. sets P and Q.

This ends Miller's construction.

5.3.3.2. *First application: Miller's Theorem.* We include here a proof of Miller's Theorem.

A pair of disjoint subsets P and Q of $\mathbb{N}$ are said to be *recursively inseparable* if there cannot exist a recursive set H such that $P \subseteq H$ and $Q \subseteq H^c$, where H^c denotes the complement of H in $\mathbb{N}$.

We will need the following well known result:

Lemma 5.3.11. *There exists a pair (P, Q) of disjoints subsets of $\mathbb{N}$ that are recursively enumerable and recursively inseparable.*

Proof. Consider an effective enumeration ϕ_0, ϕ_1, ϕ_2,... of all recursive functions. Consider the set $P = \{n \in \mathbb{N},\ \phi_n(n) = 0\}$ and the set $Q = \{n \in \mathbb{N},\ \phi_n(n) = 1\}$. Those sets are obviously recursively enumerable. Suppose now that some recursive set H contains P but does not intersect H. Consider an index n_0 such that ϕ_{n_0} is a total function that computes the characteristic function of H.

If $\phi_{n_0}(n_0) = 0$, n_0 does not belong to H, but it belongs to P, this is not possible. But if $\phi_{n_0}(n_0) = 1$, then n_0 belongs to Q and to H, which is also impossible because H should not meet Q.

This is a contradiction, and thus the sets P and Q are indeed recursively inseparable. □

Theorem 5.3.12 (Miller, **[83]**). *Suppose that P and Q are disjoints subsets of $\mathbb{N}$ that are recursively enumerable and recursively inseparable. Then the set $\Pi_{P,Q}$ is coherent, but not wp-coherent.*

Proof. Suppose that a group K with solvable word problem satisfies the relations and irrelations of $\Pi_{P,Q}$.

Using the word problem algorithm for K, given an integer i in $\mathbb{N}$, we can solve the questions "is $e_0 = e_i$ in K", since, by the properties of Miller's construction, an expression of the element e_i in terms of the generators of $L^3_{P,Q}$, and thus of K, can be effectively found from i.

The set $\{i \in \mathbb{N},\ e_i = e_0\}$ is thus a recursive set that contains P. And it is disjoint from Q, because we have assumed that $e_0 \neq e_1$ in K.

This contradicts the fact that P and Q are recursively inseparable. □

5.3.3.3. *Proof of Theorem 5.3.9.* We first prove Theorem 5.3.9:

Theorem. *No algorithm can stop exactly on those sets of relations and irrelations which are not wp-coherent.*

Proof. Given r.e. disjoint sets, we apply Miller's construction to obtain the set $\Pi_{P,Q}$ of relations and irrelations.

By Theorem 5.3.12, if the sets P and Q are recursively enumerable, recursively inseparable sets, then $\Pi_{P,Q}$ is not wp-coherent.

On the contrary, if P and Q are both recursive sets, we have noted that $L^3_{P,Q}$ itself has solvable word problem, and thus $\Pi_{P,Q}$ is wp-coherent.

Because the set $\Pi_{P,Q}$ can be constructed from the codes for P and Q, an algorithm that stops exactly on those sets of relations and irrelations which are not wp-coherent would produce, through Miller's construction, an algorithm that, given a pair of r.e. sets P and Q that are either recursively inseparable or recursive, would stop if and only if those sets are recursively inseparable. We prove in the next lemma, Lemma 5.3.13, that such an algorithm cannot exit, this ends the proof of our theorem. □

Lemma 5.3.13. *There is no algorithm that, given the code for two recursively enumerable and disjoint subsets of $\mathbb{N}$, that are either recursive or recursively inseparable, stops only if they are recursively inseparable.*

Proof. Fix two recursively enumerable and recursively inseparable subsets P and Q of $\mathbb{N}$, that exist by Lemma 5.3.11.

Consider an effective enumeration M_0, M_1, M_2... of all Turing machines. For each natural number n, define a pair of recursively enumerable sets P_n and Q_n defined as follows:

To enumerate P_n, start a run of M_n.

While this run lasts, an enumeration of P gives the first elements of P_n. If M_n halts after k computation steps, stop the enumeration of P.

Thus if M_n halts, the set P_n is a finite set. On the contrary, if M_n does not stop, P_n is identical to P.

The set Q_n is defined similarly, replacing P by Q in its definition.

One then easily sees that the sets P_n and Q_n are uniformly recursively enumerable, and that P_n and Q_n are recursively inseparable if and only if M_n does not halt.

Since no algorithm can stop exactly on the indices of non-halting Turing machines, the lemma is proved. □

5.3.3.4. *Proof of Theorem 5.3.10.* We now prove Theorem 5.3.10:

Theorem (Failure of an Effective AC for groups). *There is no algorithm that, given a wp-coherent set of relations and irrelations, produces a word problem algorithm for a marked group that satisfies those relations and irrelations.*

Note that, just as Miller's Theorem (Theorem 5.3.12) was a strengthening of the Boone-Novikov Theorem on the existence of a finitely presented group with unsolvable word problem, this theorem strengthens the Boone-Rogers Theorem which states that the word problem does not have a uniform solution amongst finitely presented groups with solvable word problem.

Proof. We will build in Lemma 5.3.14 a pair of sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$ such that:

- The sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$ consist only of disjoint recursive sets;
- The sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$ are uniformly r.e., but not uniformly recursive;
- For any sequence $(H_n)_{n\in\mathbb{N}}$ of uniformly recursive sets, there must be some index n_0 such that either H_{n_0} does not contain P_{n_0}, or $H_{n_0}^c$ does not contain Q_{n_0}.

We apply Miller's construction to this sequence to obtain a computable sequence $(\Pi_{P_n,Q_n})_{n\in\mathbb{N}}$ of finite sets of relations and irrelations.

Suppose by contradiction that there is an algorithm $\mathcal{A}$ as in the theorem. For each natural number n, the set Π_{P_n,Q_n} is wp-coherent, because P_n and Q_n are recursive. Thus the algorithm $\mathcal{A}$ can be applied to Π_{P_n,Q_n}, to produce the word problem algorithm for a group that satisfies the relations and irrelations of Π_{P_n,Q_n}. Denote G_n the group defined by this algorithm.

For each n, the set $H_n = \{i \in \mathbb{N}, e_i = e_0 \text{ in } G_n\}$ is then a recursive set, and this in fact holds uniformly in n.

But of course, one has the inclusions $P_n \subseteq H_n$ and $Q_n \subseteq H_n^c$. This contradicts the properties of the sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$. □

Lemma 5.3.14. *There exists a pair of sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$ such that:*

- *The sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$ consist only of disjoint recursive sets;*
- *The sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$ are uniformly r.e., but not uniformly recursive;*
- *For any sequence $(H_n)_{n\in\mathbb{N}}$ of uniformly recursive sets, there must be a some index n_0 such that either H_{n_0} does not contain P_{n_0}, or $H_{n_0}^c$ does not contain Q_{n_0}.*

Proof. Fix a pair (P, Q) of recursively enumerable but recursively inseparable sets.

Denote by f a recursive function that enumerates P. Denote by $\hat{f}$ an increasing function extracted from f, defined as follows: $\hat{f}(0) = f(0)$, $\hat{f}(1) = f(\min\{k \in \mathbb{N}, f(k) > \hat{f}(0)\})$, $\hat{f}(2) = f(\min\{k \in \mathbb{N}, f(k) > \hat{f}(1)\})$, etc. It is clear that $\hat{f}$ thus defined is recursive. It is well defined because P is necessarily infinite.

Consider an effective enumeration M_0, M_1, M_2... of all Turing machines.

We define the set P_n thanks to a run of the machine M_n. While this run lasts, use the function $\hat{f}$ to enumerate elements of P in increasing order. If the machine M_n halts in k steps, the last element of P that was added to P_n is $\hat{f}(k-1)$. In this case, we chose that the set P_n should be the set

$$P_n = P \cap \{0, 1, ..., \hat{f}(k-1)\}.$$

This set can be enumerated using the function f, by keeping only the elements it produces that are below $\hat{f}(k-1)$.

Because the process described above is effective, it is clear that given an integer n, one can build a recursive enumeration of P_n, and thus the sequence $(P_n)_{n\in\mathbb{N}}$ is uniformly r.e..

If the machine M_n never stops, the set P_n is exactly the image of the increasing and recursive function $\hat{f}$, it is thus recursive. If the machine M_n stops, the set P_n is finite, it is then also recursive.

The sequence $(Q_n)_{n\in\mathbb{N}}$ is defined exactly as $(P_n)_{n\in\mathbb{N}}$, replacing the function f that enumerates P by a function g that enumerates Q. Thus the sequence $(Q_n)_{n\in\mathbb{N}}$ is also a sequence of recursive sets, that are uniformly recursively enumerable.

For each n, one has $P_n \subseteq P$ and $Q_n \subseteq Q$, and thus the sets P_n and Q_n are indeed disjoints. All that is left to show is that the sequences $(P_n)_{n\in\mathbb{N}}$ and $(Q_n)_{n\in\mathbb{N}}$ satisfy the last condition of the lemma: there cannot exist a sequence $(H_n)_{n\in\mathbb{N}}$ of uniformly recursive sets for which the following inclusions hold:

$$\forall n \in \mathbb{N},\, P_n \subseteq H_n \,\&\, Q_n \subseteq H_n^c$$

We proceed by contradiction, and suppose such a sequence exists.

Consider the set U of indices n for which $P \nsubseteq H_n$.

Claim: The halting problem is solvable on the set of Turing machines whose index is in U.

Given an integer n in U, we can find an integer y such that $y \in P$ but $y \notin H_n$. Since we have assumed that $P_n \subseteq H_n$, y is an element of P, but not of P_n. Denote by k an integer for which $f(k) = y$, this exists and can be computed, since f enumerates P.

By construction of P_n, the fact that $y \notin P_n$ indicates either that the Turing machine M_n does not halt, or that, if it does stop, it must be in strictly less than k steps. This information is sufficient to solve the halting problem in U. This proves the claim.

Denote by V the set of indices n for which $Q \nsubseteq H_n^c$. The definitions of the sets P_n and Q_n being symmetric, the halting problem is solvable on the set of Turing machines whose index is in V.

Because the sets U and V are easily seen to be recursively enumerable, the previous results imply that the halting problem is solvable on $U \cup V$. Indeed, given a point n of $U \cup V$, one can find at least one of U and V to which n belongs, and apply the method of resolution of the halting problem there.

Because the halting problem is solvable on $U \cup V$, its complement must be non-empty. But for any n in $(U \cup V)^c$, one has

$$P \subseteq H_n$$
$$Q \subseteq H_n^c$$

Since the set H_n is recursive, this contradicts the fact that P and Q are recursively inseparable. □

Theorem 5.3.10 allows us to prove that the space of marked groups does not satisfy Moschovakis' condition (B), and thus that Theorem 5.2.41 cannot be applied to the space of marked groups.

Corollary 5.3.15. *The space of marked group equipped with the numbering ν_{WP} does not satisfy Moschovakis' condition (B).*

Proof. A RMS (X, d, ν) satisfies Moschovakis' condition (B) if there exists an algorithm $\mathcal{A}$ that takes as input the description of a ν-semi-decidable set Y and the description of an open ball $B(x, r)$ in X, such that those set intersect, and produces a point that belongs to their intersection.

In $\mathcal{G}_k$, apply such an algorithm to a basic clopen subset $\Omega_{r_i;s_j}$ and to an open ball that contains all of $\mathcal{G}_k$ -any open ball of radius $r \geq 1$. This yields a program that takes as input a set of relations and irrelations that is wp-coherent, and produces the ν_{WP}-name of a point that belongs to it. The existence of such an algorithm was proven impossible in Theorem 5.3.10. □

5.4. First results for groups

We now apply the results of the previous sections to decision problems for groups described by word problem algorithms. The numbering ν_{WP} of $\mathcal{G}$ is most often used implicitly, a r.e. set of marked groups is thus a ν_{WP}-r.e. set of marked groups, and so on.

5.4.1. Positive results. In what follows, rather than studying the decidability of a single property of marked groups, we consider two disjoint properties P and Q and are interested in the following decision problem: given a word problem algorithm for a group that satisfies P or Q, can we decide which one of P and of Q it does satisfy? We will simply say "deciding between P and Q" in what follows. By setting $P = \neg Q$, we can go back to studying the decidability of a single property.

The first easy result concerns properties of marked groups that are far apart. Call two properties of marked groups P and Q *ε-apart* if

$$\inf(d(G,H), G \in P, H \in Q) > \varepsilon$$

for some strictly positive number ε.

Proposition 5.4.1. *If P and Q are properties of marked groups which are ε-apart in $\mathcal{G}_k$, then it is possible to decide between P and Q.*

Proof. The fact that P and Q are ε-apart means exactly that if the binary expansion of a marked group (G, S) agrees on its first $r = -\log_2(\varepsilon)$ terms with that of a group in P, it cannot be in Q. There are at most 2^r possible prefixes of length r to the binary expansion of a k-marked group, we can label each of those by P or Q, depending on whether there exists a group in P or Q which starts precisely by this prefix. (This is not the description of an effective process: there exists such a labelling. If no group in P or in Q starts with a given sequence, its label does not matter). Then the algorithm that recognizes P from Q, on input a marked group G, outputs the label of the sequence formed by the first r terms of the binary expansion of G. □

Of course, such an algorithm is "trivial", in the sense that it checks a number of relations that is independent from its input. Such algorithms are the least interesting ones, Conjecture 5.0.6 precisely asks whether any always halting algorithm defined on the set of marked groups should be trivial in this sense.

Corollary 5.4.2. *Any clopen property in $\mathcal{G}_k$ is decidable.*

Proof. This follows from compactness of $\mathcal{G}_k$, as a subset X of $\mathcal{G}_k$ is clopen if and only if its distance to its complement is strictly positive. □

Note that those results are set in $\mathcal{G}_k$, and not in $\mathcal{G}$. To extend Proposition 5.4.1 to $\mathcal{G}$, one has to be careful that the described labelling can be done in a manner that depends recursively on the number of generators.

5.4.2. Markov's Lemma for groups. In this subsection, we rephrase some results that are true of effectively complete recursive metric spaces to the special case of the space of marked groups.

Proposition 5.4.3. *If a computable sequence $(G_n)_{n\in\mathbb{N}}$ of marked groups effectively converges in $\mathcal{G}_k$, its limit has solvable word problem.*

Proof. This follows directly from the fact that $(\mathcal{G}_{WP}, d, \nu)$ is effectively complete (Corollary 5.2.16). □

This proposition admits a converse.

Proposition 5.4.4. *Suppose an effective sequence $(G_n)_{n\in\mathbb{N}}$ converges to a marked group H which has solvable word problem. Then a sequence $(G_{\psi(n)})_{n\in\mathbb{N}}$ can be extracted from $(G_n)_{n\in\mathbb{N}}$, such that: the extraction function ψ is recursive, and for all n, $d(G_{\psi(n)}, H) \leq 2^{-n}$.*

Proof. Define $\psi(n)$ to be the least integer p such that G_p and H agree on the first k terms of their binary expansions. The hypotheses of the proposition ensure this defines a recursive function. □

We can now state Markov's Lemma applied to the space of marked groups.

Lemma 5.4.5 (Markov's Lemma for groups). *If $(G_n)_{n\in\mathbb{N}}$ is an effective sequence of marked groups that effectively converges to a marked group H, with $G_n \neq H$ for each n, then there is an effective sequence of word problem algorithms $(A^p_{WP})_{p\in\mathbb{N}}$, which define marked groups that belong to the set $\{G_n; n \in \mathbb{N}\} \cup \{H\}$, such that A^p_{WP} defines H on a non-recursively enumerable set of indices.*

This result is a direct consequence of Markov's Lemma (Lemma 5.2.27), since the space of marked groups is an effectively complete recursive metric space. We add here a direct proof of it, because it is very simple and does not require the notion of an "algorithm of passage to the limit".

PROOF. Let f be a recursive modulus of convergence for $(G_n)_{n\in\mathbb{N}}$. Consider an effective enumeration M_0, M_1, M_2... of all Turing Machines. Fix some natural number l, we define a word problem algorithm A^l_{WP} as follows. To decide whether a word w defines the identity, start a run of M_l. If, after $|w|$ steps, it still has not stoped, answer $A^H_{WP}(w)$ (by Proposition 5.4.3, H has solvable word problem). If M_l stops in p steps, with $p<|w|$, find the first integer n such that $f(n)<2^{-p}$. In this case, we choose A^l_{WP} to define the n-th group of the converging sequence, G_n, thus it should answer $A^{G_n}_{WP}(w)$.

It is easy to see that this definitions is indeed coherent and that A^l_{WP} defines the group H if and only if the l-th Turing Machine does not halt. □

In what follows, we will often use Markov's Lemma for groups together with the following obvious proposition, which is a reformulation of Proposition 5.2.20.

PROPOSITION 5.4.6. *If a marked group G with solvable word problem is adherent to a r.e. set $\mathcal{C}$ of marked groups, then there is an effective sequence of marked groups in $\mathcal{C}$ that effectively converges to G.*

This proposition, while very easy, is often useful, it will be used with $\mathcal{C}$ being: the class of finite groups, of hyperbolic groups, of all the markings of a given abstract group G, etc. We now give an example which shows that this proposition can yield some non trivial results.

EXAMPLE 5.4.7. Recall from Chapter 4 that a marked group (G,S) is called *effectively residually finite* if there is an algorithm that, given a word w over $S\cup S^{-1}$ that defines a non-identity element g of G, produces a morphism onto a finite group in which the image of g is also a non identity element.

The following proposition gives an equivalent of this definition in terms of the topology of the space of marked groups.

PROPOSITION 5.4.8. *A marked group G is co-r.p. and effectively residually finite if and only if there exists a computable sequence of marked finite quotients of G that converges (not necessarily effectively) towards it.*

PROOF. Suppose first that $(F_n)_{n\in\mathbb{N}}$ is a computable sequence of marked finite quotients of G which converges to it.

G is co-r.p.: an enumeration of all words that define non-identity elements in some F_n gives an enumeration of the non-identity elements of G.

And given a word w that defines a non-identity element in G, a blind search for a group F_n in which w is non-trivial will always terminate, since such a group exists by assumption. Thus G is effectively residually finite.

Suppose now that G is co-r.p. and effectively residually finite. We build a computable sequence of quotients of G that converges towards it as follows. Enumerate words for all non-identity elements of G: w_1, w_2, w_3... For each word w_i, we can build a marked finite quotient F_{w_i} of G, in which the word w_i defines a non-trivial element. Now we define a sequence $(\hat{F}_n)_{n\in\mathbb{N}}$ by taking for $\hat{F}_n$ the product, in the category of marked groups, of the marked groups F_{w_1}, ..., F_{w_n}. That is to say that, if we denote by $S^i=(s^i_1,s^i_2,...,s^i_k)$ a generating set for F_{w_i}, the sequence $(\hat{F}_n)_{n\in\mathbb{N}}$ is defined by

$$\hat{F}_n \leq F_{w_1}\times ... \times F_{w_n}$$
$$\hat{F}_k = \langle (s^1_1,s^2_1,...,s^n_1),...,(s^1_k,s^2_k,...,s^n_k)\rangle$$

It is easy to see that the sequence $(\hat{F}_n)_{n\in\mathbb{N}}$ is also computable, and converges to G. □

In Chapter 4 was built a residually finite group $L(\mathcal{A}_3)$, with solvable word problem, which is not effectively residually finite.

It is interesting to note that, applied to this group, Proposition 5.4.6 produces a non-trivial result: the group $L(\mathcal{A}_3)$ is proven to be adherent to the set of finite groups thanks to a sequence of its quotients, which is necessarily non computable, however the proposition then shows that $L(\mathcal{A}_3)$ must also be the limit of an effectively converging sequence of finite groups. Furthermore, because it is always possible to detect when a finite group is *not* a quotient of a recursively presented group, there must even exist a sequence that effectively converges to $L(\mathcal{A}_3)$, which consists only of groups which are *not* quotients of $L(\mathcal{A}_3)$.

5.4.3. Cases that escape a topological characterization. Notice that our positive results about decidable properties, and our negative results, are not mutually exclusive. For P and Q properties of k-marked groups with no intersection, we have already seen that, in $\mathcal{G}_k$:

- If P and Q are ε-apart, there exists an algorithm that discriminates between them (Proposition 5.4.1);
- If there exists a sequence of groups in P that effectively converges to a group in Q, then no algorithm can distinguish groups in P from groups in Q (Markov's Lemma).

The negation of the first point says that there are two sequences of marked groups, $(G_n)_{n\in\mathbb{N}}$ and $(H_n)_{n\in\mathbb{N}}$, the first consisting of groups in P, the second one of groups in Q, and such that $d(G_n, H_n)$ goes to 0. This is much weaker than the second point.

If no such sequences exist, that are made only of groups with solvable word problem, then the properties P and Q must be ε-apart for groups with solvable word problem, (which is the same as saying that "P and having solvable word problem" is ε-apart from "Q and having solvable word problem"), and thus there exists an algorithm that tells P from Q. (Note that Conjecture 5.5.3 gives an instance of a property P for which we know that $\inf\{d(G,H); G\in P, H\in\neg P\} = 0$, but we do not know whether or not this is also true amongst groups with solvable word problem.)

We now suppose that P and Q are not ε-apart, even amongst groups with solvable word problem. Thus there exist sequences $(G_n)_{n\in\mathbb{N}}$ and $(H_n)_{n\in\mathbb{N}}$ as above, consisting only of marked groups with solvable word problem. By compactness, up to extracting subsequences, we can suppose that both sequences converge towards a marked group G. We then distinguish several cases, depending on:

- whether the sequences $(G_n)_{n\in\mathbb{N}}$ and $(H_n)_{n\in\mathbb{N}}$ are effective;
- whether the convergence of these sequences is effective;
- whether the limit group G has solvable word problem;
- whether the limit group G satisfies one of P or Q.

The case when all four hypotheses are satisfied is precisely the one where Markov's Lemma applies (Lemma 5.4.5). When the first three hypotheses hold, while the last one does not, we cannot conclude anything. This we prove now.

Example 5.4.9. We set ourselves in the space $\mathcal{G}_1$ of cyclic groups, but this could be done in any other space, replacing $\mathbb{Z}$ by any group which is the effective limit of an effective sequence. For each subset X of $\mathbb{Q}$, consider the property P_X defined by $P_X(G) \iff d(G,\mathbb{Z})\in X$. Consider a subset X of $\mathbb{Q}$, which does not contain 0, and call $Y = \mathbb{Q}\setminus(X\cup\{0\})$. We are interested in finding out whether we can decide between P_X and P_Y. None of X or Y contain 0, this guarantees that $\mathbb{Z}$ satisfies none of P_X or P_Y. This implies that given a group G in $P_X\cup P_Y$, the distance $d(G,\mathbb{Z})$ can be computed not only as a computable real, but as a rational, as we excluded the only problematic case $G=\mathbb{Z}$. From the computation of $l = d(G,\mathbb{Z})$, to tell whether $P_X(G)$ or $P_Y(G)$, it suffices to decide whether or not l belongs to X.

If X is a recursive set of rationals, this is possible, and thus one can effectively discriminate between P_X and P_Y.

If X is not recursive, and if we also suppose that X is a subset of $\{d(G,\mathbb{Z}),\, G\in\mathcal{G}_1\}$ (which is a recursive set), an algorithm that tells between P_X and P_Y would provide an algorithm for membership in X, which is impossible.

In either case, for effective sequences $(G_n)_{n\in\mathbb{N}}$ and $(H_n)_{n\in\mathbb{N}}$ to exist, that converge to $\mathbb{Z}$ and belong respectively to P_X and P_Y, one can for instance suppose that X contains the set $\{d(\mathbb{Z},\mathbb{Z}/2^n\mathbb{Z}), n>1\}$ and that Y contains $\{d(\mathbb{Z},\mathbb{Z}/3^n\mathbb{Z}), n>1\}$.

Overall, varying the set X, one obtains different situations where the first three hypotheses highlighted above are satisfied, but for which either P_X and P_Y can be effectively recognized one from the other, or not.

This situation is interesting, but in many cases it cannot arise: inquiring about the decidability of a single property, one will use the previously described setting with $P=\neg Q$, in which case the situation of Example 5.4.9 cannot arise.

The situation where only the last two of our highlighted hypotheses hold is the one for which the example of a discontinuous but Markov computable function given in SubSection 5.2.5.1 becomes relevant. The example given there shows that it is still possible in this case for P and Q to be distinguishable. However, the example given in SubSection 5.2.5.1 is defined on a sequence whose elements have a maximal Kolmogorov complexity, this is a strong negation of being computable, and we conjecture that in most cases, even when the sequences $(G_n)_{n\in\mathbb{N}}$ and $(H_n)_{n\in\mathbb{N}}$ are not computable, P and Q will not be distinguishable. But proving so would rely on obtaining a better understanding of the structure of the sets on which a discontinuous Markov computable function can exist. A starting point in the study of this problem is, again, the article of Hoyrup and Rojas, [**58**].

5.4.4. A Specker sequence of groups. A Specker sequence of computable reals is a computable and increasing sequence of computable reals which converges to a non-computable real. Specker was the first to exhibit such a sequence in 1949 in [**120**]. We more generally call *Specker sequence* a computable sequence in an effectively complete recursive metric space that converges to a non-computable point. The convergence speed of such a sequence is slower than any recursive function, it must be a Cauchy sequence that is not effectively Cauchy.

We could try and build a Specker sequence of groups by using any known construction of a Specker sequence in the Cantor space and translating it to the space of marked groups (for example using one of the Cantor spaces that are effectively embedded in $\mathcal{G}_k$, as are described in SubSection 5.4.5.3).

We will instead show that the group $L(\mathcal{A}_1)$ built in Chapter 3, that has unsolvable word problem but that still has CFQ, is naturally associated to a Specker sequence of groups.

Proposition 5.4.10. *There exists a residually finite group G which has unsolvable word problem, for which there exists a computable sequence $(F_n)_{n\in\mathbb{N}}$ of marked finite quotients of G which converges to it.*

Proof. The group $L(\mathcal{A}_1)$ has computable finite quotients, and thus it is effectively residually finite, and thus by Proposition 5.4.8, it is the limit of a computable sequence of its finite quotients. However, it has unsolvable word problem. □

5.4.5. Some effective Polish spaces inside of $\mathcal{G}$. We give here some examples of effective Polish spaces inside of $\mathcal{G}$. Any closed subset of $\mathcal{G}$ is effectively complete, and thus an effective Polish space that lives inside of $\mathcal{G}$ is just the closure of any computable sequence of marked groups.

5.4.5.1. *LEF groups.* LEF groups were first defined in [**127**], in terms of partial homomorphisms onto finite group. An equivalent definition appropriate to our setting is the following:

Definition 5.4.11. A group is called Locally Embeddable into Finite groups (LEF) if one (or, equivalently, any) of its marking is adherent in $\mathcal{G}$ to the set $\mathcal{F}$ of finite marked groups.

The set $\overline{\mathcal{F}}$ of (markings of) LEF groups is closed in $\mathcal{G}$, by definition, and, since $\mathcal{F}$ is ν_{WP}-r.e., it is also effectively separable. Thus $(\overline{\mathcal{F}}, d, \nu_{WP})$ is an effective Polish space.

5.4.5.2. *The closure of the set of Hyperbolic groups.* Let $\mathcal{H}$ denote the set of markings of word hyperbolic groups. We have:

Proposition 5.4.12. *The closure $\overline{\mathcal{H}}$ of $\mathcal{H}$ in $\mathcal{G}$ is an effective Polish space with respect to the numbering ν_{WP}.*

Proof. $\overline{\mathcal{H}}$ is closed by definition. Word problem algorithms for hyperbolic groups can be enumerated thanks to an enumeration of finite presentations for hyperbolic groups, the fact that hyperbolic groups have uniformly solvable word problem is due to Gromov, and the fact that the set of hyperbolic groups can be enumerated by finite presentations was shown by Papasoglu in [**100**]. □

$\overline{\mathcal{H}}$ was studied by Champetier in [**22**], where he showed for instance the following:

Proposition 5.4.13 (Champetier, [**22**], Corollary 5.10). *The closure of the set of non-elementary hyperbolic groups in $\mathcal{G}_k$ is a Cantor space.*

Note that some results of [**22**] were revisited and improved on in [**84**].

Since finite groups are hyperbolic, we have the inclusion $\overline{\mathcal{F}} \subseteq \overline{\mathcal{H}}$. Note that it is still an open question to prove that this inclusion is strict, this is equivalent to the problem of finding a non residually finite hyperbolic group. It is generally believed that the inclusion is indeed strict.

The continuity theorem (Corollary 5.2.30) can be applied on the effective Polish spaces $(\overline{\mathcal{F}}, d, \nu_{WP})$ and $(\overline{\mathcal{H}}, d, \nu_{wp})$, this yields the following proposition:

Proposition 5.4.14. *A property which is not clopen in $\overline{\mathcal{F}}$ or in $\overline{\mathcal{H}}$ cannot be decidable there.*

Moschovakis' Theorem on effectively open sets in effective Polish spaces can also be applied in $\overline{\mathcal{F}}$ and $\overline{\mathcal{H}}$, this yields:

Proposition 5.4.15. *If O is an effectively open subset of $\mathcal{G}$, then $O \cap \overline{\mathcal{F}}$ and $O \cap \overline{\mathcal{H}}$ are Lacombe sets.*

5.4.5.3. *Effective embeddings of Cantor sets in $\mathcal{G}_k$.* Let G_A, $A \subseteq \mathbb{N}$, be a family of groups indexed by subsets of $\mathbb{N}$.

Suppose that the function

$$\Theta : \begin{cases} \mathcal{P}(\mathbb{N}) \to \mathcal{G} \\ A \subseteq \mathbb{N} \mapsto G_A \end{cases}$$

is injective and continuous. Suppose furthermore that Θ maps recursive subsets of $\mathbb{N}$ to groups with solvable word problem, and that there is a procedure that allows, given an algorithm for the membership problem in a recursive set A, to obtain a word problem algorithm for the group G_A.

In this case, the function Θ defines *an effective embedding of the Cantor set inside $\mathcal{G}$.* Each such function is attached to a subset of $\mathcal{G}$ which is effectively Polish.

Effective embeddings of the Cantor set into $\mathcal{G}$ abound in the literature. We quote here a few examples.

- P. Hall's family of 3 soluble groups from [**52**];
- Bartholdi and Erschler's generalization of Hall's construction from [**4**], which we use to prove Theorem 5.5.12;
- Dyson's doubles of the lamplighter group from [**35**], which we used in Chapter 3 and Chapter 4;
- Amalgamated products of free groups that appear in [**24**];
- Miller's construction detailed in SubSection 5.3.3;
- Etc.

5.4.6. List of properties for which the Correspondence holds. We will now proceed to list a series of group properties for which the correspondence between the first level of the Kleene–Mostowski hierarchy and that of the Borel hierarchy holds perfectly: each quoted clopen property is ν_{WP}-decidable, (and, while any clopen set is decidable in $\mathcal{G}_k$, we do not need here to fix the number of generators, we are thus set in $\mathcal{G}$), each quoted open property is ν_{WP}-semi-decidable but not ν_{WP}-decidable, each closed property is ν_{WP}-co-semi-decidable and not ν_{WP}-decidable, and for properties that are neither closed nor open, no partial algorithm exist.

This list is presented as a table which appears in SubSection 5.4.6.9.

Most affirmations which appear in this table are obvious, easy applications of Markov's Lemma in each case provide the desired results. We now include references or proofs for the less obvious affirmations.

5.4.6.1. *Residually finite groups.* It is known that the set of residually finite groups is not closed, because the adherence of the set of finite groups, which is the set LEF groups, strictly contains the set of residually finite groups. The semi-direct product $\mathbb{Z} \ltimes \mathfrak{S}_\infty$ is the limit of the sequence of finite groups $\mathbb{Z}/n\mathbb{Z} \ltimes \mathfrak{S}_n$, as n goes to infinity; it is not residually finite since it contains an infinite simple group. Note that the described sequence effectively converges. (This example comes from [**127**]).

It is also well known that the set of residually finite groups is not open: non-abelian free groups are limits of non-residually finite groups. For instance, it follows from well known results on the Burnside problem that free Burnside groups of sufficiently large exponent are infinite groups that are not residually finite, and that, if S denotes a basis of a non-abelian free group, the sequence $(\mathbb{F}/\mathbb{F}^n, S)$, starting with $n \gg 1$, is effective and converges to $(\mathbb{F}, S)$ as n goes to infinity.

5.4.6.2. *Amenable groups.* Amenable groups do not form a closed set, since free groups are limits of finite groups. They do not form an open set either. An interesting example that proves this comes from [**4**]: there exists a sequence of markings of $\mathbb{F}_2 \wr \mathbb{Z}$ that converges to a marking of $\mathbb{Z}^2 \wr \mathbb{Z}$. (See Example 7.4 in [**4**].) This sequence is effective.

5.4.6.3. *Having sub-exponential growth.* The set of finitely generated groups with sub-exponential growth is neither closed nor open. This was proved by Grigorchuk in [**45**]: there is constructed a family of groups G_ω, $\omega \in \{0, 1, 2\}^{\mathbb{N}}$, for which it is explained when G_ω has intermediate or exponential growth, and for which it is proved that the convergence in the space of marked groups of a sequence of groups $(G_{\omega_n})_{n\in\mathbb{N}}$ coincides with the convergence in $\{0, 1, 2\}^{\mathbb{N}}$ (for the product topology) of the sequence $(\omega_n)_{n\in\mathbb{N}}$. These sequences are effective when the sequence $(\omega_n)_{n\in\mathbb{N}}$ itself is computable for the obvious numbering.

5.4.6.4. *Orderable groups.* A finitely generated group G is *orderable* if there exists a total order $\leq$ on G which is compatible with the group operation:

$$\forall x, y, a, b \in G,\ a \leq b \implies xay \leq xby.$$

The following well known characterization of orderable groups was already used in [**11**] in order to study the algorithmic complexity of the property "being orderable":

PROPOSITION 5.4.16. *A group G is orderable if and only if for any finite set $\{a_1, a_2, ..., a_n\}$ of non-identity elements of G, there are signs $\epsilon_1, ..., \epsilon_n \in \{-1, 1\}$, such that the sub-semi-group generated by $\{a_1^{\epsilon_1}, a_2^{\epsilon_2}, ..., a_n^{\epsilon_n}\}$ does not contain the identity of G.*

It is straightforward to notice that this characterization provides a way of recognizing word problem algorithms for non orderable groups, and that it shows that the set of orderable groups is closed.

PROPOSITION 5.4.17. *The set of orderable groups is ν_{WP}-co-semi-decidable and closed in $\mathcal{G}$.*

5.4.6.5. *Virtually cyclic groups; Virtually Nilpotent groups; Polycyclic groups.* For the three properties of being virtually cyclic, virtually nilpotent or polycyclic, we use the following lemma. Recall that $\nu_{r.p.}$ is the numbering of $\mathcal{G}$ associated to recursive presentations.

LEMMA 5.4.18. *Suppose that P is a ν_{WP}-semi-decidable (resp. $\nu_{r.p.}$-semi-decidable) subset of $\mathcal{G}$, and that Q is a $\nu_{r.p.}$-semi-decidable subset of $\mathcal{G}$.*

Then the set of groups that have a finitely generated normal subgroup in P, and such that the quotient by this normal subgroup is in Q, is ν_{WP}-semi-decidable (resp. $\nu_{r.p.}$-semi-decidable).

Note that in this statement, the normal subgroup should be finitely generated as a group, and not only finitely generated *as a normal subgroup*.

Proof. Given a word problem algorithm for a group G generated by a family S, we proceed as follows.

Enumerate all finite subsets of G.

There is an effective procedure that recognizes those finite subsets of G that generate a normal subgroup. Indeed, consider a finite set A in G, and the subgroup H it generates. The subgroup H is normal in G if and only if for each a in A and each s in S, the generating family of G, the elements $s^{-1}as$ and sas^{-1} both belong to H. An exhaustive search for ways of expressing $s^{-1}as$ and sas^{-1} as products of elements of A will terminate if indeed those elements belong to H.

For each finite subset A which generates a normal subgroup H of G, we can obtain a word problem algorithm (resp. a recursive presentation) for H thanks to the word problem algorithm for G (resp. thanks to a recursive presentation for G), and a recursive presentation for the quotient G/H, since an enumeration of the relations of G together with an enumeration of the elements of H yields a recursive presentation of G/H.

The hypotheses of the lemma then allow us to recognize when the group H is in P and the quotient G/H is in Q. □

This lemma can be used directly to show that the properties of being virtually nilpotent or virtually cyclic are ν_{WP}-semi-decidable.

Corollary 5.4.19. *The set of virtually cyclic groups is open in $\mathcal{G}$ and ν_{WP}-semi-decidable.*

Proof. Apply Lemma 5.4.18 with P being the set of cyclic groups and Q the set of finite groups, to prove that the set of virtually cyclic groups ν_{WP}-semi-decidable.

Analyzing the way the algorithm that stops on word-problem algorithms for virtually cyclic groups thus obtained works, one sees that the proof it produces of the fact that a group is virtually cyclic consists in finitely many relations. Those relations define an open set that contains only virtually cyclic groups. □

Corollary 5.4.20. *The set of virtually nilpotent groups is open in $\mathcal{G}$ and ν_{WP}-semi-decidable.*

Proof. Apply Lemma 5.4.18 with P being the set of nilpotent groups and Q the set of finite groups.

Again, this shows that the set of virtually nilpotent groups is ν_{WP}-semi-decidable, and analyzing the algorithm given by Lemma 5.4.18, we see that the set of virtually nilpotent groups is open in $\mathcal{G}$. □

Note that we have chosen those two properties because they both admit equivalent definitions in terms of asymptotic geometry: by a well-known theorem of Gromov, a group is virtually nilpotent if and only if it has polynomial growth, and a group is virtually cyclic if and only if it has zero or two *ends*, in the sense of Stallings.

It is thus remarkable that both of these properties can be recognized thanks to only finitely many relations.

Corollary 5.4.21. *The set of polycyclic groups is open in $\mathcal{G}$ and ν_{WP}-semi-decidable.*

Proof. (Sketch) Iterate Lemma 5.4.18 with Q being the set of cyclic groups, and P being the set polycyclic groups with a subnormal series of length n, to obtain the result.

We leave it to the reader to prove that this proof also gives the fact that the set of polycyclic groups is open in $\mathcal{G}$. □

5.4.6.6. *Groups with infinite conjugacy classes.* A group G has *infinite conjugacy classes* (ICC) if for each non identity element g of G the conjugacy class $\{xgx^{-1},\, x \in G\}$ of g is infinite.

Proposition 5.4.22. *The set of ICC groups is ν_{WP}-co-semi-decidable, and it is closed in $\mathcal{G}$.*

Proof. Given a word problem algorithm for a group G over a generating family S and an element g of G, it is possible to prove that the conjugacy class of g is finite: define a sequence of sets $(A_n)_{n\in\mathbb{N}}$ by

$$A_0 = \{g\},$$
$$A_{n+1} = \{s^{-1}xs;\, s \in S \cup S^{-1},\, x \in A_n\}.$$

The conjugacy class of g is finite if and only if there exists an integer n such that $A_n = A_{n+1}$. A blind search for such an integer will terminate if it exists.

This shows that the set of non-ICC groups is ν_{WP}-semi-decidable, one easily checks that this also shows that this set is open. □

5.4.6.7. *Hyperbolic groups.* Let δ be a positive real number. A marked group G is δ-hyperbolic if the triangles in G are δ-thin, that is to say if for any three elements g_1, g_2 and g_3 of G, the geodesic for the word metric that joins g_1 to g_2 stays in a δ-neighborhood of the geodesics that join respectively g_2 and g_3 and g_1 and g_3.

PROPOSITION 5.4.23. *Being δ-hyperbolic is a closed and ν_{WP}-co-semi-decidable property.*

PROOF. A group is not δ-hyperbolic if it admits a triangle that is not δ-thin. The fact that this triangle is not δ-thin can be seen in a sufficiently large ball of the Cayley graph of G, and thus any group that corresponds to G on this ball is also not δ-hyperbolic. It is easy to see that this condition can be effectively checked. □

Remark that being δ-hyperbolic is a marked group property, but not a group property, as can be seen from the fact that there exists a sequence of markings of $\mathbb{Z}$ converges to a marking of $\mathbb{Z}^2$, which is not hyperbolic.

A group is Gromov hyperbolic if any of its marking is δ-hyperbolic, for some δ that can depend on the marking. The set of Gromov hyperbolic groups is neither open nor closed in $\mathcal{G}$, but the previous proposition implies that it is a union of closed sets.

COROLLARY 5.4.24. *The set of hyperbolic groups is a F_σ subset of $\mathcal{G}$, and it is effectively not closed and effectively not open.*

5.4.6.8. *Sofic groups.* The set of sofic groups is known to be closed in $\mathcal{G}$. However, whether it is all of $\mathcal{G}$ or a strict subset of $\mathcal{G}$ is still an open problem. We ask:

PROBLEM 5.4.25. Is there an algorithm that recognizes word problem algorithms for non-sofic groups?

5.4.6.9. *Table of results.* The following table gathers our examples. Remark that for each subset of $\mathcal{G}$ that appears in this table, four properties of this set are expressed: whether or not it is open, whether or not it is closed, whether or not it is ν_{WP}-semi-decidable, and whether or not it is ν_{WP}-co-semi-decidable.

Clopen/decidable properties	**Open/semi-decidable properties**
Being abelian;	Being nilpotent;
Being isomorphic to a given finite group;	Kazhdan's Property (T) ([**117**, Theorem 6.7], [**99**]);
Having cardinality at most n, $n \in \mathbb{N}^*$;	Having a non-trivial center;
Being nilpotent of derived length $k > 0$;	Being perfect;
Being a certain marked isolated group (see Section 5.5).	Having torsion;
	Having rank at most k, $k \in \mathbb{N}^*$;
	Being virtually cyclic;
	Having polynomial growth;
	Being polycyclic.

Closed/co-semi-decidable properties	**Neither closed nor open properties**
Being infinite;	Being solvable;
Being k-solvable, for $k > 1$;	Being amenable;
Having a finite exponent;	Being simple;
Being a limit group (see SubSection 5.6.2);	Having sub-exponential growth;
Being orderable;	Being finitely presented;
Being δ-hyperbolic, $\delta > 0$;	Being hyperbolic;
Having Infinite Conjugacy Classes (ICC).	Being residually finite.

5.5. Recognizing groups from word problem algorithms

We will now include some results that concern the study of the isomorphism problem for groups described by word problem algorithms. We proceed step by step, studying first which marked groups are recognizable when described by word problem algorithms, then proceeding to ask which abstract groups are recognizable when described by word problem algorithms, and ending with the study of the isomorphism problem. As we will see, the finite sets of groups which have solvable isomorphism problem are characterized by the topology of $\mathcal{G}$.

5.5.1. Marked recognizability. The study of marked recognizability corresponds to the study of properties which are singletons in the space $\mathcal{G}_k$. A singleton is closed in $\mathcal{G}_k$, and, given two marked groups G and H, it is always possible to prove that they are different. This implies in particular that the isomorphism problem is always solvable for finite sets of marked groups, contrary to what happens for groups described by recursive presentations (see Proposition 2.0.1).

The singletons which are open in $\mathcal{G}_k$, which are its *isolated points*, were studied by Cornulier, Guyot and Pitsch in **[27]**. Recall that we mentioned in Lemma 5.1.5 a result of **[27]** that implies that if a group admits an isolated marking, all its markings are isolated, and thus that we can talk about an isolated group. Say that a group G admits *a finite discriminating family* if there is a finite set X of non-identity elements such that any non-trivial normal subgroup of G contains an element of X. A group which admits a finite discriminating family is called *finitely discriminable.*

Proposition 5.5.1. *(***[27]***, Proposition 2) A group is isolated if and only if it is finitely presented and finitely discriminable.*

If a basic clopen neighborhood $\Omega_{r_1,...,r_n;s_1,...,s_{n'}}$ of $\mathcal{G}_k$ is a singleton $\{G\}$, a presentation of G is given by $\langle S|r_1,...,r_n\rangle$, and the set $\{s_1,...,s_{n'}\}$ then defines a finite discriminating family for G. Note that in the vocabulary of B. H. Neumann in **[93]**, the triple $(S;r_1,...,r_n;s_1,...,s_{n'})$ is an *absolute presentation* for G. Isolated groups have solvable word problem, by Theorem 5.3.7.

We have already remarked that the basic clopen sets $\Omega_{r_1,...,r_n;s_1,...,s_{n'}}$ define decidable properties, we thus have:

Proposition 5.5.2. *Isolated groups are recognizable as marked groups.*

Obvious examples of isolated groups are the finite groups (for which the whole group forms a finite discriminating family), and the finitely presented simple groups (for which any non-trivial element is a discriminating family). More examples can be found in **[27]**. A nice example which is not finite and not simple is Thompson's group F: it is not abelian, but all its proper quotients are, thus any non-identity element of its derived subgroup forms a finite discriminating family.

Now we conjecture:

Conjecture 5.5.3. *The isolated groups are the only groups that are recognizable, as marked groups, from the word problem algorithm.*

This is an interesting conjecture because it raises problems of two different kinds. First of all, asking whether or not a decidable singleton must be clopen in $\mathcal{G}_{WP}$ is a peculiar instance of our Main Conjecture (Conjecture 5.0.6). On the other hand, it remains an open problem to know whether it is possible that a group with solvable word problem that is not isolated in $\mathcal{G}$ be isolated in $\mathcal{G}_{WP}$: this is an example of an instance where it is unclear how the topology of $\mathcal{G}$ changes when looking only at groups with solvable word problem.

5.5.2. Abstract Recognizability.

5.5.2.1. *First results.* For an abstract group G, recall that $[G]_k$ designates the set of all its markings in $\mathcal{G}_k$, and $[G]$ the set of all its markings in $\mathcal{G}$. The study of abstract recognizability in $\mathcal{G}_k$ is the study of the decidability of the properties that can be written $[G]_k$ for some group G. As before, we will first discuss where these properties lie in the Borel hierarchy, before trying to obtain decidability results.

The open and closed isomorphism classes of groups are completely described by the following:

Proposition 5.5.4. *Let G be a finitely generated group. The set $[G]_k$ is open if and only if G is isolated, and $[G]_k$ is closed if and only if it is finite.*

Proof. The first point states that $[G]_k$ is open if and only if all its points are open. Suppose that $[G]_k$ is open, while G is not isolated. It means that there exists a sequence of groups in $[G]_k$ that converges to some marking of G. But then, by Lemma 5.1.5, all markings of G must be adherent to $[G]_k$. This implies that $[G]_k$ should have no isolated points. This should also hold in some clopen neighborhood of a marking of G. But any perfect set (closed and without isolated points) in a Polish space is uncountable, by a well known result which is due to Cantor in the case of subsets of $\mathbb{R}$. This would imply that $[G]_k$ should be uncountable, a contradiction.

Now suppose that $[G]_k$ is infinite. By compactness of $\mathcal{G}_k$, it must have an accumulation point. If $[G]_k$ is closed, we obtain a sequence of markings of G that converges in $[G]_k$, and the same contradiction as above arises. ☐

For any k, finite groups have finitely many markings in $\mathcal{G}_k$. On the other hand, for G a group of rank k, generated by a family $(g_1,...,g_k)$, setting ourselves in $\mathcal{G}_{k+1}$, the set of generating families $(g_1,...,g_k,w)$, where w ranges over all elements of G, defines infinitely many markings of G, as soon as G is infinite. This yields:

Corollary 5.5.5. *$[G]_k$ is clopen for any k if and only if G is finite. And any finite group is abstractly recognizable.*

Finitely generated free groups of Hopfian varieties provide examples of groups with finitely many markings in some $\mathcal{G}_k$, indeed, a free group of rank k in a Hopfian variety has a single k-marking. Those include the free groups of varieties of nilpotent groups, or of the variety of metabelian groups. However, we do not know whether an isolated group can have finitely many markings in $\mathcal{G}_k$, such a group would be abstractly recognizable amongst k-marked groups.

Example 5.5.6. Limits in $\mathcal{G}_3$ of markings of the rank two free group F_2 consist in all HNN extensions of the form

$$\langle a, b, t | t^{-1}wt = w \rangle$$

where w is any element of the free group on a and b that is not a proper power (w can be trivial, in which case the defined group is free of rank three). This follows from the results of [**39**].

We finally introduce the "preform" relation.

By Proposition 5.1.5, if a marking of a group H is adherent to $[G]$, then all markings of H will be adherent to $[G]$. Define a binary relation $\overset{<}{\rightsquigarrow}$ on the set of finitely generated (abstract) groups by setting $G \overset{<}{\rightsquigarrow} H$ if and only if some marking of H is adherent to $[G]$ in $\mathcal{G}$. In this case, we say that G *preforms* H. This relation was introduced, named and studied by Laurent Bartholdi and Anna Erschler in [**4**], some of their results will be useful here. In particular, they show that the relation $\overset{<}{\rightsquigarrow}$ is a pre-order on the set of finitely generated groups, i.e. it is transitive and reflexive, but it is not an order, because non-isomorphic groups G and H can satisfy both $G \overset{<}{\rightsquigarrow} H$ and $H \overset{<}{\rightsquigarrow} G$.

We now investigate decidability issues, using the previous topological results.

All possible generating families of an isolated group can be effectively listed, and this can be done while keeping track of a finite discriminating family (expressing its elements in terms of products of elements of each generating family). And more generally, if some marking of a group G is isolated inside a class $\mathcal{C}$ of abstract groups, then one can define the concept of a "discriminating family with respect to $\mathcal{C}$", and such discriminating families can be transposed from a generating set to another.

The following proposition directly follows from these considerations:

Proposition 5.5.7. *Isolated groups are semi-recognizable as abstract groups, and more generally, if a marking of a group G is open in $\mathcal{C}$, for some class $\mathcal{C}$ of abstract groups, then $[G]$ is open in $\mathcal{C}$, and G is semi-recognizable as an abstract group in $\mathcal{C}$.*

By the description of open classes of isomorphism in $\mathcal{G}_k$, whether isolated groups are the only abstract groups semi-recognizable from word problem algorithms depends on a conjecture that we already proposed when talking about marked recognizability, Conjecture 5.5.3.

To make use of the statement about the closeness of isomorphism classes in $\mathcal{G}_k$, we must first obtain an effective version of this statement. Now, in $\mathcal{G}_k$, if G is a group with solvable word problem, all markings of G can be enumerated, and thus the class $[G]_k$ satisfies the hypothesis of Proposition 5.4.6: a group which is adherent to $[G]_k$ is the effective limit of a sequence of groups in $[G]_k$ if and only if it has solvable word problem. Thus all we have to do is to prove that some point adherent to $[G]_k$ has solvable word problem to conclude that G is not recognizable from the word problem algorithm.

Proposition 5.5.8. *Let G and H be groups with solvable word problem. If $G \overset{<}{\rightsquigarrow} H$, then no algorithm can tell H from G, in particular "being equal to H" cannot be semi-decidable.*

Proof. This is a direct application of Markov's Lemma, as we have stated that $[G]$ is a r.e. set, and thus Proposition 5.4.6 applies. ☐

This proposition can be used with any of the many examples of groups G and H that satisfy $G \overset{<}{\rightsquigarrow} H$ that can be found in [**4**], for instance it is showed there that the Grigorchuk-Aleshin group preforms a free group.

We believe that the following statement holds in general, but are not able to prove it:

Conjecture 5.5.9. *If G is an infinite group with solvable word problem of rank k, then G preforms a group H, non-isomorphic to G, with solvable word problem and of rank at most $k+1$.*

Some special instances of this question were studied in conjunction to the study of the elementary theory of free and hyperbolic groups. In [**116**], Sela proved, using a variation of McKinsey's result about finitely presented residually finite groups, the following (the slight difference with our statement is that in [**116**] is used, instead of $[G]$, the set of all markings of G *and* of its subgroups):

Proposition 5.5.10. *If G has solvable word problem, a finitely presented group, which is adherent to $[G]$, also has solvable word problem.*

Proof. A finitely presented group H, adherent to $[G]$, must be residually-G, have computable quotients in $[G]$ and a set of word problem algorithms for elements of $[G]$ can be enumerated, thus McKinsey's algorithm applies (see Chapter 3). □

Note that this result can be used in conjunction with another result of Sela ([**115**]), which states that all limit groups are finitely presented, to prove Conjecture 5.5.9 for limit groups: a limit group must always preform another group that has solvable word problem. The same techniques provide results for limits of hyperbolic groups ([**116**]), but this stays far from the degree of generality of Conjecture 5.5.9.

All this leaves the following unanswered:

Conjecture 5.5.11. *Finite groups are the only abstractly recognizable groups in $(\mathcal{G}, d, \nu_{WP})$.*

5.5.2.2. *A family of completely undistinguishable groups.* We will now use one of the main results of [**4**] to prove the following theorem, which shows how poorly suited the word problem numbering type is to solve the isomorphism problem for finitely generated groups.

Theorem 5.5.12. *There exists an infinite set $U = \{G_n, n \in \mathbb{N}\}$ of finitely generated groups with solvable word problem, such that for any pair $(G_i, G_j) \in U$, $i \neq j$, $[G_i]$ and $[G_j]$ are completely indistinguishable: being a marking of G_i is neither ν_{WP}-semi-decidable nor ν_{WP}-co-semi-decidable in $[G_i] \cup [G_j]$.*

The set U of groups contains infinitely many non-isomorphic groups, but any ball in a labeled Cayley graph of a group in U could in fact belong to any of the groups in U. This is a situation drastically opposed to what happens for isolated groups, for which a ball of large enough radius defines the group uniquely.

Proof. We use the proof of Proposition 5.1. of [**4**], which is an elaborate variation of a well known construction of Hall. This propositions concerns the relation $\overset{<}{\rightsquigarrow}$ as defined on the set of marked groups: say that a marked group G preforms a marked group H if the abstract groups defined by G preforms the abstract group defined by H.

We use the fact that is given a construction that associates to a pair of subsets C and X of $\mathbb{N}$ a marked group $H_{X,C}$, and that this construction satisfies the following: $H_{X,C}$ preforms $H_{Y,C'}$ if and only if $Y \subseteq X$, and the marked groups $H_{X,C}$ and $H_{Y,C'}$ are isomorphic as marked group if and only if $Y = X$ and $C = C'$.

To apply this construction in our setting, one can check that it additionally satisfies the two following features (this is straightforward):

(1) If X and C are recursive sets, $H_{X,C}$ has solvable word problem;
(2) The word problem in $H_{X,C}$ is at least as hard, in terms of time complexity, as the membership problem in C, modulo an additive term which corresponds to a reduction from one problem to the other.

Using those properties, fixing X and having C vary while being recursive, one obtains infinitely many marked groups which are all $\overset{<}{\rightsquigarrow}$-equivalent to each other.

To justify that those marked groups actually correspond to infinitely many different abstract groups, we cannot directly use the argument used in [**4**], which goes back to Hall, and which says: as C varies in $\mathcal{P}(\mathbb{N})$, one obtains uncountably many marked groups $H_{X,C}$, and since an abstract group only has countably many markings, those marked groups must in fact define uncountably many abstract groups. However, we can use the following argument, which can be seen as an effectivisation of Hall's argument: replacing the set C by recursive sets for which the membership problem has an arbitrary high time complexity, using property (2) given above, and the fact that the time complexity of the word problem in a group is an isomorphism invariant (up to a coarse equivalence relation, see for instance [**113**]), one must obtain infinitely many non isomorphic abstract groups. □

5.5.3. Isomorphism Problem. Of course, the isomorphism problem is solvable for finite groups described by word problem algorithms, this can easily be seen, for finite groups the word problem and finite presentation numberings are equivalent.

Our previous results about abstract recognizability translate directly into a characterization of which *finite* families of groups can have solvable isomorphism problem from the word problem description, since for a finite family of groups to have solvable isomorphism problem it is necessary and sufficient that each of its groups be abstractly recognizable.

Proposition 5.5.13. *A finite family $\mathfrak{D}$ of groups with solvable word problem has solvable isomorphism problem with respect to the numbering ν_{WP} exactly when no group in $\mathfrak{D}$ preforms another group in $\mathfrak{D}$.*

Proof. This follows from Proposition 5.5.7 and Proposition 5.5.8. □

In an infinite family of groups, to solve the isomorphism problem, once each group was proven abstractly recognizable, it is still left to show that this recognition can be made uniformly on all groups of the family.

In most families of groups which have solvable isomorphism problem when described by finite presentations, not only is the isomorphism problem unsolvable for the word problem algorithm description, but not all groups are recognizable. This is witnessed, for instance, by the pair $\{\mathbb{Z};\mathbb{Z}^2\}$ of abelian groups, and by the pair $\{\mathbb{F}_2;\mathbb{F}_3\}$ of hyperbolic groups.

Examples of infinite families of infinite groups with solvable isomorphism problem for ν_{WP} can be constructed using examples of isolated groups from [**27**], taking for instance a sequence $(G_n)_{n\in\mathbb{N}}$ of pairwise non-isomorphic isolated groups, for which absolute presentations, i.e. triples (generating set-defining relations-discriminating family) can be enumerated, since the algorithm for semi-recognition of an isolated group can be obtained from an absolute presentation.

Note that all examples that arise this way define families for which the isomorphism problem is solvable from finite presentations as well. This is unfortunate, but it should be clear by now that finite presentations are in general more powerful descriptions than word problem algorithms, when it comes to solving the isomorphism problem in different classes of groups.

5.6. Two candidates for failure of the correspondence

5.6.1. Isolated Groups. It does not seem possible to prove that a word problem algorithm belongs to an isolated group (even with a partial algorithm). From this, we conjecture that the set of isolated groups defines an open set which is not ν_{WP}-semi-decidable. However, we are unable to prove this undecidability result, which would require some techniques entirely independent of Markov's Lemma.

Conjecture 5.6.1. *The set of isolated group is open in $\mathcal{G}$ but not semi-decidable.*

Remark that the impossibility of partially recognizing isolated groups is also an open problem for groups described by finite presentations. While the Adian-Rabin theorem implies that no algorithm stops exactly on finite presentations of non-isolated groups, it fails to prove that no algorithm stops exactly on finite presentations of isolated groups. The problem of proving that the set of finite presentations of simple groups is not r.e. seems also to still be open. (The question appears for instance in [**89**].)

It is also unknown (the question appears in [**27**]) whether isolated points are dense in the set of groups with solvable word problem of $\mathcal{G}$. If they were, we would be able to tell that no sequence of word problem algorithms which contains each isolated group can be enumerated, by Proposition 5.3.2. We could still arrive to that conclusion if we knew that the word problem is not uniform on isolated groups, that is to say, since all isolated groups are finitely presented, if we knew that a solution to the word problem of an isolated group cannot be retrieved from a finite presentation for this group. For instance, it is well known that the word problem is uniform on the set of simple groups ([**64**]), however, Kuznetsov's argument fails if we add the trivial group to the set of simple groups. It would be very interesting to prove that the trivial group is unrecognizable from simple groups, from the finite presentation description, and this would prove both that the word problem is not uniform on all isolated groups, and that the set of finite presentations of simple groups is not r.e.. However, too few finitely presented infinite simple groups are known as of now to obtain such results.

5.6.2. LEF groups and the elementary theory of groups.

5.6.2.1. *Introduction on universal and existential theories of groups.* The space of marked groups was used by Champetier and Guirardel (in [**23**]) in order to study limit groups, which play an important role in the solution to Tarski's problem on the elementary theory of free groups. We include here a paragraph to emphasize the links with our present study, and we point out some differences. This will be the occasion to propose the set of LEF groups as another candidate for the failure of the correspondence between the Borel and arithmetical hierarchies.

We do not want to include many definitions, and refer [**23**] for precise definitions, and references. A formula is obtained with variables, logical connectors ($\wedge$ is "and", $\vee$ is "or", and $\neg$ is "not"), the equality symbol $=$, the group law $\cdot$, the identity element 1, and the group inverse $^{-1}$, and the two quantifiers $\forall$ and $\exists$. We use shortcuts where it is convenient (as the symbols $\neq$ or $\implies$), and always use implicitly all group axioms. A *sentence* is a formula with no free variables. A *universal sentence* is a sentence that uses only the universal quantifier, and an *existential sentence* uses only the existential quantifier.

For instance:

$$\forall x \forall y,\, x = y$$
$$\forall x \forall y \forall z,\, xy = yx \wedge yz = zy \wedge y \neq 1 \implies xz = zx$$
$$\exists x,\, x \neq 1 \wedge x^2 = 1$$

For a group G, let $T_\forall(G)$ denote the set of universal sentences that are true in G, and $T_\exists(G)$ the set of existential sentences that are true in G. For a class $\mathcal{C}$ of group we also write $T_\forall(\mathcal{C})$ and $T_\exists(\mathcal{C})$, meaning the set of universal (resp. existential) sentences that hold in *all* groups of $\mathcal{C}$.

In the space of marked groups, a universal sentence defines a closed set, and the correspondence with the arithmetical hierarchy holds, i.e., from a word problem algorithm, it is possible to prove that a group does not satisfy a given universal sentence. Similarly, an existential sentence defines an open set and the correspondence holds for such sets. We will not be interested here in formulas with alternating quantifiers.

Remark 5.6.2. Some universal sentences define sets that are open, and thus clopen, (as the sentence that define abelian groups), while other sentences define sets that are only closed, and not open (for instance the sentence that defines metabelian groups). It seems in fact very hard to determine whether or not a universal sentence defines a clopen set: indeed, notice that for sentences of the form

$$\forall x,\, x^n = 1,$$

we know that if $n \in \{1, 2, 3, 4, 6\}$, then the set defined by this sentence is clopen, by the negative solution to the Burnside problem for those exponents, while for $n \gg 1$, it is known that the set defined by the Burnside sentence is not open, as groups of large finite exponent can be constructed as limits of hyperbolic groups, which are not groups of finite exponent. (This is detailed in [**22**].) And it is an open problem to determine whether or not for $n = 5$ the above sentence defines a clopen set.

The following proposition of [**23**] is thus straightforward:

Proposition 5.6.3 ([**23**]; Proposition 5.2). *If a sequence of marked groups $(G_n)_{n\in\mathbb{N}}$ converges to a marked group G, then* $\limsup(T_\forall(G_n)) \subseteq T_\forall(G)$.

This proposition admits a converse, also due to Champetier and Guirardel, which strengthens the relation between the space of marked groups and the study of the elementary theory of groups. We reproduce its proof, because it is important in understanding the link between elementary theories of groups and the topology of $\mathcal{G}$.

Proposition 5.6.4 ([**23**]; Proposition 5.3). *Suppose that two groups G and H satisfy $T_\forall(H) \subseteq T_\forall(G)$. Then any marking of G is a limit of markings of subgroups of H.*

Proof. The proof in fact relies on the existential theories of the groups G and H, which satisfy the reversed inclusion: $T_\exists(G) \subseteq T_\exists(H)$. Fix a generating family S of G, and a radius r. Consider the set $\{w_1, ..., w_k\}$ of reduced words of length at most r on the alphabet $S \cup S^{-1}$. Consider the sets $J_1 = \{(i, j); w_i =_G w_j\}$ (where $=_G$ means that those words define identical elements of G) and $J_2 = \{(i, j); w_i \neq_G w_j\}$. Then G satisfies the existential formula:

$$\exists S, \bigwedge_{(i,j)\in J_1} w_i = w_j \wedge \bigwedge_{(i,j)\in J_2} w_i \neq w_j$$

By hypothesis, H must satisfy it as well, which means precisely that a subgroup of H must have the same ball of radius r as G. ☐

For a group H, denote $\mathcal{S}(H)$ the set of its subgroups.

Corollary 5.6.5. *Let G and H be finitely generated groups. The following are equivalent:*

- *A marking of G is adherent to the set $\mathcal{S}(H)$;*
- *All markings of G are adherent to the set $\mathcal{S}(H)$;*
- $T_\forall(H) \subseteq T_\forall(G)$.

We end this paragraph by using Markov's Lemma together with the results of Champetier and Guirardel.

Lemma 5.6.6 (Markov's Lemma for Elementary Theories). *Suppose that two groups G and H, with solvable word problem, satisfy $T_\forall(H) \subseteq T_\forall(G)$.*

Then $[G]$ is not ν_{WP}-semi-decidable inside the set

$$[G] \cup \bigcup_{K\in\mathcal{S}(H)} [K]\,.$$

Proof. This follows from Corollary 5.6.5, Proposition 5.4.6 (if H has solvable word problem, word problem algorithms for elements in $\mathcal{S}(H)$ can be enumerated), and from Markov's Lemma for groups (Lemma 5.4.5). ☐

5.6.2.2. *Limit groups and LEF groups.* We will use the following definition for limit groups (introduced in [**115**], see [**23**] for the equivalence): a group G is a *limit group* if some (or all) of its markings are adherent to the set of marked free groups. Note that if G is a subgroup of a group H, every universal sentence in G holds in H. This implies that all non-abelian free groups have the same universal theory, since each non-abelian free group is a subgroup of each other non-abelian free group.

Thus by Corollary 5.6.5, a group G is a limit group if and only if it satisfies $T_\forall(\mathbb{F}_2) \subseteq T_\forall(G)$, where $\mathbb{F}_2$ is the rank two free group. In fact, it is known that if a group G satisfies $T_\forall(\mathbb{F}_2) \subseteq T_\forall(G)$, then either it is abelian, and then it is free abelian, and $T_\forall(\mathbb{Z}) = T_\forall(G)$, or it has a free subgroup, which implies that $T_\forall(\mathbb{F}_2) = T_\forall(G)$.

The following proposition solves a decision problem for groups given by word problem algorithms, while relying heavily on the study of the elementary theory of groups.

Proposition 5.6.7. *Being a limit group is ν_{WP}-co-semi-decidable.*

Proof. A group G is a limit group if and only if it satisfies $T_\forall(F_2) \subseteq T_\forall(G)$. A theorem of Makanin ([**70**]) states that the universal theory of free groups is decidable, and thus that it is possible to enumerate all universal sentences that hold in free groups.

Since, given a word problem algorithm for a marked group (G, S), it is always possible to prove that a universal sentence is not satisfied in G, it is possible to detect groups that are not limit groups by testing in parallel all sentences of the universal theory of free groups. □

This result is a slight improvement of a result in [**49**], where the same is obtained, but making use of both a finite presentation and a word problem algorithm.

This result calls to our attention a second example of a natural property for which the correspondence between the arithmetical hierarchy and the Borel hierarchy might fail. Indeed, this last proof relies heavily on Manakin's theorem. While the universal theory of free groups is decidable, Slobodskoi proved in [**119**] that the universal theory of finite groups is unsolvable. From this we conjecture:

Conjecture 5.6.8. *The set of LEF groups is not ν_{WP}-co-semi-decidable.*

Denote by $\mathcal{F}$ the set of finite groups, recall that its adherence $\overline{\mathcal{F}}$ is the set of LEF groups. Note that, at first glance, Slobodskoi's Theorem does not seem to be the sole thing preventing us from applying the proof of Proposition 5.6.7 to LEF groups. Indeed, it relied on the fact that a group G is a limit group if and only if it satisfies $T_\forall(\mathbb{F}_2) \subseteq T_\forall(G)$, which in turn used the fact that the inclusion $T_\forall(\mathbb{F}_2) \subseteq T_\forall(G)$ is equivalent to the reverse inclusion $T_\exists(G) \subseteq T_\exists(\mathbb{F}_2)$ (see Proposition 5.6.4). This follows from the fact that the elementary theory of a single group is *complete*, i.e. every sentence or its negation is in it. But the theory of finite groups is of course not complete, as the existential theory $T_\exists(\mathcal{F})$ is empty, since the trivial group does not satisfy any existential sentence. However, the corresponding equivalence still holds.

Proposition 5.6.9. *A group G belongs to $\overline{\mathcal{F}}$ if and only if it satisfies $T_\forall(\mathcal{F}) \subseteq T_\forall(G)$.*

Proof. We use the fact that there exists a group K in $\overline{\mathcal{F}}$ that satisfies $T_\forall(K) = T_\forall(\mathcal{F})$. If a group G satisfies $T_\forall(K) = T_\forall(\mathcal{F}) \subseteq T_\forall(G)$, then it satisfies $T_\exists(K) \subseteq T_\exists(G)$, and by Corollary 5.6.5, G is a limit of subgroups of K. But K and all its finitely generated subgroups are in $\overline{\mathcal{F}}$, thus G must also be a limit of markings of finite groups.

The group K can be taken as the semi-direct product $\mathbb{Z} \ltimes \mathfrak{S}_\infty$, where $\mathfrak{S}_\infty$ denotes the group of finitely supported permutations of $\mathbb{Z}$, on which $\mathbb{Z}$ acts by translation. This group is the limit of the finite groups $\mathbb{Z}/n\mathbb{Z} \ltimes \mathfrak{S}_n$, as n goes to infinity ($\mathfrak{S}_n$ is the group of permutation over $\{1, ..., n\}$). Since K is in $\overline{\mathcal{F}}$, $T_\forall(\mathcal{F}) \subseteq T_\forall(K)$. However, because it contains a copy of every finite group, one also has the reversed inclusion. □

Thanks to this proposition, we have:

Proposition 5.6.10. *Conjecture 5.6.8 implies Slobodskoi's Theorem.*

Proof. Supposing that Slobodskoi's Theorem fails, one can reproduce the proof of Proposition 5.6.7, and prove that Conjecture 5.6.8 fails. □

Other conjectures can be obtained, that are similar to Conjecture 5.6.8: by a theorem of Kharlampovich ([**59**]), the universal theory of finite nilpotent groups is also undecidable, and it is also known that the universal theory of hyperbolic groups is undecidable (as proven by Osin in [**97**]).

Problem 5.6.11. Is the adherence $\bar{\mathcal{H}}$ of the set of hyperbolic groups ν_{WP}-co-semi-decidable? What of the adherence of the set of finite nilpotent groups?

5.6.2.3. *Properties not characterized by universal theories.* Note that not every decidable property for groups given by word problem algorithms can be solved by expressing the question that is to be solved as a problem about universal or existential theories, and applying techniques similar to the proof of Proposition 5.6.7.

We give here a simple example. Let H be the group $\mathbb{Z} * \mathbb{Z}^3$. It is a non-abelian limit group, thus H and the rank three free group $\mathbb{F}_3$ have the same universal theory. However, the property "being isomorphic to H" can be discerned from the property "being isomorphic to $\mathbb{F}_3$". Indeed, no sequence of markings of H can converge to a marking of $\mathbb{F}_3$, because "having rank at most three" is an open property in $\mathcal{G}$. On the other hand, suppose that a sequence of markings of $\mathbb{F}_3$ converges to a marking of H. By Lemma 5.1.5, this implies that the canonical marking of $\mathbb{Z} * \mathbb{Z}^3$ is a limit of 4-markings of $\mathbb{F}_3$. But this would imply that one can find a generating family (a, b, c, d) of $\mathbb{F}_3$, such that b, c and d commute. This is impossible, because abelian subgroups of free groups are cyclic, and thus this would imply that $\mathbb{F}_3$ is of rank two. By Proposition 5.5.13, the isomorphism problem is solvable for the pair $\{\mathbb{F}_3, H\}$, while those two groups have identical existential and universal theories.

5.7. Subgroups of finitely presented groups with solvable word problem

5.7.1. Higman-Clapham-Valiev Theorem for groups with solvable word problem. We now remark how Higman's Embedding Theorem gives further incentive for the study of algorithmic problems solved for groups described by word problem algorithms.

After Higman's proof of his famous Embedding Theorem ([**56**]), several papers proposed different improvements.

We note that it was remarked that the theorem is effective, meaning that it provides an algorithm that takes as input a recursively presented group, and outputs a finitely presented group, together with a finite family of elements of that group, that generate a group isomorphic to the first group.

We will be interested here in the version of Higman's Theorem that preserves solvability of the word problem (see [**126, 25**]). This theorem is known as the Higman-Clapham-Valiev Theorem.

Historical remarks about these results can be found in [**95**]. The following formulation of the Higman-Clapham-Valiev Theorem can also be found in [**13**].

THEOREM 5.7.1 (Higman-Clapham-Valiev, I). *There exists a procedure that, given a recursive presentation for a group G, produces a finite presentation for a group H, together with an embedding $G \hookrightarrow H$ described by the images of the generators of G, and such that if the word problem is solvable in G, then it is also solvable in H.*

One can also check that if one has access to a word problem algorithm for the group given as input to this procedure, one can obtain a word problem algorithm for the constructed finitely presented group. This yields:

THEOREM 5.7.2 (Higman-Clapham-Valiev, II). *There exists a procedure that, given a word problem algorithm for a finitely generated group, produces a finite presentation of a group in which it embeds, together with a word problem algorithm for this new group, and a set of elements that generate the first group.*

This proves that, in general, the description of a group by its word problem algorithm, or by a finite generating family inside a finitely presented group with solvable word problem, are equivalent (we leave it to the reader to render this statement precise: define a numbering of $\mathcal{G}$ associated to the idea "a group is described as a subgroup of a finitely presented group whose word problem algorithm is given", the Higman-Clapham-Valiev Theorem shows that this numbering is equivalent to ν_{WP}).

Thus the study of algorithmic problems that can be solved from the word problem description is identical to the study of decision problems about subgroups of finitely presented groups with solvable word problem.

The following theorem is a joint application of the Higman-Clapham-Valiev Theorem and of Markov's Lemma:

THEOREM 5.7.3. *Suppose that a ν_{WP}-computable sequence $(G_n)_{n\in\mathbb{N}}$ of k-marked groups effectively converges to a k-marked group H, and suppose that $H \notin \{G_n, n \in \mathbb{N}\}$. Then there exists a finitely presented group Γ, with solvable word problem, in which no algorithm can stop exactly on k-tuple of elements of Γ that define H.*

PROOF. By Markov's Lemma applied to groups, there exists a ν_{WP}-computable sequence $(L_n)_{n\in\mathbb{N}}$ of marked groups, such that for each p, $L_p \in \{G_n, n \in \mathbb{N}\} \cup \{H\}$, and the set $\{n \in \mathbb{N} | L_n = H\}$ is co-r.e. but not r.e.. The direct sum of those groups can be embedded in a finitely generated group which has solvable word problem (using the well known construction of Higman, Neumann and Neumann [**57**]), and in turn the Higman-Clapham-Valiev theorem can be applied to obtain a finitely presented group Γ which contains the sequence $(L_n)_{n\in\mathbb{N}}$. Moreover, it is easy to see that there exists an algorithm that, given a natural number n, produces a k-tuple of elements of Γ that generate L_n. (This comes from the fact that each of the described embeddings is effective.) This directly implies the claimed result. ☐

Note that when the conjugacy problem is uniformly solvable for the groups in $(G_k)_{k\in\mathbb{N}}$, we may want to apply the version of Higman's Theorem due to Alexander Olshanskii and Mark Sapir ([**95**], and [**96**] for non-finitely generated groups) that preserves its solvability.

5.7.2. Some examples. We now give some examples of possible applications of Theorem 5.7.3.

Proposition 5.7.4. *There exists a finitely presented group with solvable word problem, but unsolvable order problem.*

Proof. Apply Theorem 5.7.3 to a sequence of finite cyclic groups that converges to $\mathbb{Z}$. This yields a finitely presented group with solvable word problem in which one cannot decide whether a given element generates a subgroup isomorphic to $\mathbb{Z}$ or to a finite cyclic group. This is precisely a finitely presented group with solvable word problem, but unsolvable order problem. ☐

Proposition 5.7.5. *There exists a finitely presented group with solvable word problem, but unsolvable power problem.*

Proof. Apply Theorem 5.7.3 to the sequence of 2-markings of $\mathbb{Z}$ defined by the generating families $(1, k)$, $k \in \mathbb{N}^*$, which converges to (the only 2-marking of) $\mathbb{Z}^2$ when k goes to infinity (see [**23**]). This yields a finitely presented group with solvable word problem where, given a pair of commuting elements, one cannot decide whether they generate $\mathbb{Z}^2$, or if one of these elements is a power of the other: this is a group with unsolvable power problem. ☐

We can also use this theorem to strengthen a result of [**32**]:

Theorem 5.7.6. *There is a finitely presented group with solvable word problem in which the problem of deciding whether a given subgroup is amenable is neither semi-decidable nor co-semi-decidable.*

Proof. This is proven by using both a sequence of marked amenable groups which converges to a non-amenable groups and a sequence of non-amenable marked groups that converges to an amenable marked group. Such examples were given in SubSection 5.4.6. ☐

The "not semi-decidable" half of this result was obtained in [**32**].

Theorem 5.7.3 can more generally be applied to all the properties that appeared in SubSection 5.4.6 to produce results similar to this one.

We will stop multiplying examples, as all those results are well known, but it seems that explaining them in terms of convergence in the space of marked groups unifies several existing constructions.

CHAPTER 6

List of open problems

Most of the following problems appear in the text, but some only appear here.

6.1. From Chapter 2

PROBLEM 6.1.1. Formalize the hierarchy that intertwines algorithmic descriptions of groups and decision problems, as described in the introduction of Chapter 2. (This should probably rely on numberings.)

PROBLEM 6.1.2. Characterize the groups with solvable word problem that admit an WPI marked quotient algorithm: are these only the finitely presented groups with solvable word problem?

PROBLEM 6.1.3. Solve the marked recognizability problem for groups given by finite presentations, i.e. describe for which marked groups (G, S) is the problem "does the presentation π define (G, S)?" solvable in π .

The simplest case of this problem is the following:

PROBLEM 6.1.4. Can the marked group $(\mathbb{Z}, (0, 1))$ be recognized thanks to finite presentations?

PROBLEM 6.1.5. Describe the groups that are abstractly recognizable from the description that consists of a finite presentation together with a word problem algorithm.

PROBLEM 6.1.6. Is there a natural example of a group G, recognizable as an abstract group, while some of its markings are not recognizable? (in some class $\mathcal{C}$ of groups, with respect to some type of description.)

PROBLEM 6.1.7. Find a class $\mathcal{C}$ of groups and groups in $\mathcal{C}$ that admit a non-trivial marked quotient algorithm relative to $\mathcal{C}$.

6.2. From Chapter 3

PROBLEM 6.2.1. Find a residually finite group with CFQ with a finitely generated subgroup that does not have CFQ.

PROBLEM 6.2.2. Find a finitely presented residually finite group with a finitely generated subgroup that does not have CFQ.

PROBLEM 6.2.3. Let G be a finitely generated residually finite group, and S and S' two generating families of G. Is it possible that exactly one of the depth functions ρ_S and $\rho_{S'}$ be recursive?

PROBLEM 6.2.4. Prove that finitely generated linear groups have CFQ. (They are known to be effectively residually finite.) Do Betsvina-Brady groups have CFQ?

PROBLEM 6.2.5. (Ashot Minasyan) Find a conjugacy separable finitely generated group with solvable word problem but unsolvable conjugacy problem. Such a group cannot have CFQ.

6.3. From Chapter 4

PROBLEM 6.3.1. Prove a Higman Embedding Theorem for residually finite groups.

We conjecture that having solvable word problem and being effectively residually finite is a sufficient condition for a group to embed in a finitely presented residually finite group. A step towards solving this problem is the following:

PROBLEM 6.3.2. Give necessary and sufficient conditions on a subset $\mathcal{A}$ of $\mathbb{Z}$ for Dyson's groups $L(\mathcal{A})$ to embed in a finitely presented residually finite group.

6.4. From Chapter 5

Problem 6.4.1. Are the Λ_{WP}-computable functions defined on the set of marked groups with solvable word problem continuous?

Problem 6.4.2. Characterize those sequences $(u_n)_{n\in\mathbb{N}} \in \mathcal{C}^{\mathbb{N}}$ converging to the null sequence in the Cantor space for which any Markov computable function defined on $\{u_n, n \in \mathbb{N}\} \cup \{0^\omega\}$ has to be continuous.

Problem 6.4.3. Suppose that ν is a saturated numbering of $\mathcal{G}$, and that $(\mathcal{G}, d, \nu)$ is a RMS. Must one have $\nu \succeq \nu_{WP}$?

Problem 6.4.4. Is a non isolated group with solvable word problem not isolated from groups with solvable word problem?

Problem 6.4.5. Is it true that every $rank(n)$ infinite group preforms a $rank(n+1)$ group? Do free Burnside groups preform free Burnside groups of larger rank?

Problem 6.4.6. Does every infinite group with solvable word problem preforms another group with solvable word problem?

Problem 6.4.7. Is there an algorithm that recognizes word problem algorithms for non-sofic groups?

Problem 6.4.8. Show that no algorithm can, given a word problem algorithm, prove that the group it defines is not LEF.

Problem 6.4.9. Show that no algorithm recognizes word problem algorithms for isolated groups in $\mathcal{G}$.

Bibliography

[1] O. Aberth. *Computable Analysis*. McGraw-Hill International Book Company, 1980.

[2] J. Avigad and V. Brattka. Computability and analysis: the legacy of Alan Turing. In R. Downey, editor, *Turing's Legacy*, pages 1–47. Cambridge University Press, 2014.

[3] L. Bartholdi. Endomorphic presentations of branch groups. *Journal of Algebra*, 268(2):419–443, oct 2003.

[4] L. Bartholdi and A. Erschler. Ordering the space of finitely generated groups. *Annales de l'institut Fourier*, 65(5):2091–2144, 2015.

[5] G. Baumslag. Wreath products and finitely presented groups. *Mathematische Zeitschrift*, 75(1):22–28, dec 1961.

[6] G. Baumslag. *Topics in Combinatorial Group Theory*. Birkhäuser Basel, Sept. 1993.

[7] G. Baumslag, A. Myasnikov, and V. Remeslennikov. Algebraic geometry over groups I. Algebraic sets and ideal theory. *Journal of Algebra*, 219(1):16–79, sep 1999.

[8] M. G. Benli, R. Grigorchuk, and P. de la Harpe. Amenable groups without finitely presented amenable covers. *Bulletin of Mathematical Sciences*, 3(1):73–131, may 2013.

[9] M. G. Benli and B. Kaya. Descriptive complexity of subsets of the space of finitely generated groups. Sept. 2019.

[10] R. Bieri, Y. Cornulier, L. Guyot, and R. Strebel. Infinite presentability of groups and condensation. *Journal of the Institute of Mathematics of Jussieu*, 13(4):811–848, jan 2014.

[11] I. Bilanovic, J. Chubb, and S. Roven. Detecting properties from descriptions of groups. *Archive for Mathematical Logic*, 59(3-4):293–312, aug 2019.

[12] L. A. Bokut and G. P. Kukin. Unsolvable algorithmic problems for semigroups, groups, and rings. *Journal of Soviet Mathematics*, 45(1):871–911, apr 1989.

[13] W. W. Boone and H. J. Rogers. On a problem of J.H.C. Whitehead and a problem of Alonzo Church. *Mathematica Scandinavica*, 19:185, jun 1966.

[14] E. Borel. Le calcul des intégrales définies. *Journal de mathématiques pures et appliquées 6ème série*, 8:159–210, 1912.

[15] K. Bou-Rabee. Quantifying residual finiteness. *Journal of Algebra*, 323(3):729–737, feb 2010.

[16] K. Bou-Rabee and B. Seward. Arbitrarily large residual finiteness growth. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 2016(710), jan 2016.

[17] V. Brattka and G. Presser. Computability on subsets of metric spaces. *Theoretical Computer Science*, 305(1-3):43–76, aug 2003.

[18] M. R. Bridson, D. M. Evans, M. W. Liebeck, and D. Segal. Algorithms determining finite simple images of finitely presented groups. *Inventiones mathematicae*, 218(2):623–648, may 2019.

[19] M. R. Bridson and H. Wilton. The triviality problem for profinite completions. *Inventiones mathematicae*, 202(2):839–874, feb 2015.

[20] G. S. Ceitin. Algorithmic operators in constructive metric spaces. *Trudy Matematicheskogo Instituta Imeni V. A. Steklova*, pages 1–80, 1967.

[21] C. Chabauty. Limite d'ensembles et géométrie des nombres. *Bulletin de la Société mathématique de France*, 2:143–151, 1950.

[22] C. Champetier. L'espace des groupes de type fini. *Topology*, 39(4):657–680, jul 2000.

[23] C. Champetier and V. Guirardel. Limit groups as limits of free groups. *Israel Journal of Mathematics*, 146(1):1–75, dec 2005.

[24] H. K. Chong and D. T. Wise. An uncountable family of finitely generated residually finite groups. *Journal of Group Theory*, 25(2):207–216, sep 2021.

[25] C. R. J. Clapham. An embedding theorem for finitely generated groups. *Proceedings of the London Mathematical Society*, s3-17(3):419–430, jul 1967.

[26] F. Dahmani and V. Guirardel. The isomorphism problem for all hyperbolic groups. *Geometric and Functional Analysis*, 21(2):223–300, apr 2011.

[27] Y. de Cornulier, L. Guyot, and W. Pitsch. On the isolated points in the space of groups. *Journal of Algebra*, 307(1):254–277, jan 2007.

[28] M. Dehn. Über unendliche diskontinuierliche gruppen. *Mathematische Annalen*, 71(1):116–144, mar 1911.

[29] M. Dehn. On infinite discontinuous groups. In *Papers on Group Theory and Topology*, pages 133–178. Springer New York, 1987.

[30] J. Deré and M. Pengitore. Effective subgroup separability of finitely generated nilpotent groups. *Journal of Algebra*, 506:489–508, jul 2018.

[31] A. Detinko and D. Flannery. Linear groups and computation. *Expositiones Mathematicae*, 37(4):454–484, dec 2019.

[32] K. Duda and A. Ivanov. On decidability of amenability in computable groups. *Archive for Mathematical Logic*, mar 2022.

[33] W. Dyck. Gruppentheoretische studien. *Mathematische Annalen*, 20(1):1–44, mar 1882.

[34] W. Dyck. Gruppentheoretische studien. II. Ueber die zusammensetzung einer gruppe discreter operationen, über ihre primitivitat und transitivitat. *Mathematische Annalen*, 22(1):70–108, mar 1883.

[35] V. H. Dyson. A family of groups with nice word problems. *Journal of the Australian Mathematical Society*, 17(4):414–425, jun 1974.

[36] D. B. A. Epstein, M. S. Paterson, J. W. Cannon, D. F. Holt, S. V. Levy, and W. P. Thurston. *Word Processing in Groups*. A. K. Peters, Ltd., Nov. 1992.

[37] Y. L. Ershov. La théorie des énumérations. In *Actes du Congrès International des Mathématiciens*, pages P223–227, 1970.

[38] Y. L. Ershov. Theory of numberings. In *Handbook of Computability Theory*, pages 473–503. Elsevier, 1999.

[39] B. Fine, A. M. Gaglione, A. Myasnikov, G. Rosenberger, and D. Spellman. A classification of fully residually free groups of rank three or less. *Journal of Algebra*, 200(2):571–605, feb 1998.

[40] R. Friedberg. Un contre-exemple relatif aux fonctionnelles récursives. *Comptes rendus hebdomadaires des séances de l'Académie des Sciences (Paris)*, vol. 24, 1958.

[41] H. Furstenberg. On the infinitude of primes. *Amer. Math. Monthly*, 62(5):353, may 1955.

[42] A. Garrido. Abstract commensurability and the Gupta–Sidki group. *Groups, Geometry, and Dynamics*, 10(2):523–543, 2016.

[43] Y. Glasner and D. Kitroser. A note on LERF groups and generic group actions. Sept. 2014.

[44] V. Gregoriades, T. Kispéter, and A. Pauly. A comparison of concepts from computable analysis and effective descriptive set theory. *Mathematical Structures in Computer Science*, 27(8):1414–1436, jun 2016.

[45] R. I. Grigorchuk. Degrees of growth of finitely generated groups, and the theory of invariant means. *Mathematics of the USSR-Izvestiya*, 25(2):259–300, apr 1985.

[46] R. I. Grigorchuk and S. V. Ivanov. On Dehn functions of infinite presentations of groups. *Geometric and Functional Analysis*, 18(6):1841–1874, mar 2009.

[47] R. I. Grigorchuk and J. S. Wilson. A structural property concerning abstract commensurability of subgroups. *Journal of the London Mathematical Society*, 68(03):671–682, nov 2003.

[48] D. Groves, J. F. Manning, and H. Wilton. Recognizing geometric 3-manifold groups using the word problem. Oct. 2012.

[49] D. Groves and H. Wilton. Enumerating limit groups. *Groups, Geometry, and Dynamics*, pages 389–399, 2009.

[50] D. Groves and H. Wilton. The structure of limit groups over hyperbolic groups. *Israel Journal of Mathematics*, 226(1):119–176, may 2018.

[51] M. Hall. A topology for free groups and related groups. *The Annals of Mathematics*, 52(1):127, jul 1950.

[52] P. Hall. Finiteness conditions for soluble groups. *Proceedings of the London Mathematical Society*, s3-4(1):419–436, 1954.

[53] R. Hartung. Coset enumeration for certain infinitely presented groups. *International Journal of Algebra and Computation*, 21(08):1369–1380, dec 2011.

[54] P. Hertling. Banach-Mazur computable functions on metric spaces. In *Computability and Complexity in Analysis*, pages 69–81. Springer Berlin Heidelberg, 2001.

[55] P. Hertling. A Banach-Mazur computable but not Markov computable function on the computable real numbers. In *Automata, Languages and Programming*, volume 132, pages 962–972. Springer Berlin Heidelberg, mar 2002.

[56] G. Higman. Subgroups of finitely presented groups. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 262(1311):455–475, aug 1961.

[57] G. Higman, B. H. Neumann, and H. Neuman. Embedding theorems for groups. *Journal of the London Mathematical Society*, s1-24(4):247–254, oct 1949.

[58] M. Hoyrup and C. Rojas. On the information carried by programs about the objects they compute. *Theory of Computing Systems*, 61(4):1214–1236, dec 2016.

[59] O. G. Kharlampovich. Universal theory of the class of finite nilpotent groups is undecidable. *Mathematical Notes*, 33:254–263, 1983.

[60] O. G. Kharlampovich, A. Myasnikov, and M. V. Sapir. Algorithmically complex residually finite groups. *Bulletin of Mathematical Sciences*, 7(2):309–352, mar 2017.

[61] O. G. Kharlampovich and M. V. Sapir. Algorithmic problems in varieties. *International Journal of Algebra and Computation*, 05(04n05):379–602, aug 1995.

[62] G. Kreisel, D. Lacombe, and J. R. Shoenfield. Partial recursive functionals and effective operations. Constructivity in mathematics, Proceedings of the colloquium held at Amsterdam:pp. 290–297, 1957.

[63] B. A. Kushner. *Lectures on Constructive Mathematical Analysis*. American Mathematical Society, 1984.

[64] A. V. Kuznetsov. Algorithms as operations in algebraic systems. *Izv. Akad. Nauk SSSR Ser. Mat.*, 1958.

[65] S. Lawton, L. Louder, and D. McReynolds. Decision problems, complexity, traces, and representations. *Groups, Geometry, and Dynamics*, 11(1):165–188, 2017.

[66] J. Lockhart. Decision problems in classes of group presentations with uniformly solvable word problem. *Archiv der Mathematik*, 37(1):1–6, dec 1981.

[67] R. L. Lovas and I. Mezo. On an exotic topology of the integers. Aug. 2010.

[68] R. C. Lyndon and P. E. Schupp. *Combinatorial group theory*, volume 89. Springer-Verlag, Berlin, 1977.

[69] W. Magnus. Residually finite groups. *Bulletin of the American Mathematical Society*, 75(2):305–317, mar 1969.

[70] G. S. Makanin. Decidability of the universal and positive theories of a free group. *Mathematics of the USSR-Izvestiya*, 25(1):75–88, feb 1985.

[71] A. I. Maltsev. On homomorphisms onto finite groups. (Russian). *Uchen. Zap. Ivanovskogo Gos. Ped.Inst.*, 1958.

[72] A. I. Maltsev. Constructive algebras I. (Russian). *Uspehi Mat. Nauk*, 1961.

[73] A. I. Maltsev. *The metamathematics of algebraic systems, collected papers: 1936-1967*. North-Holland Pub. Co, Amsterdam, 1971.

[74] A. Mann. A note on recursively presented and co-recursively presented groups. *Bulletin of the London Mathematical Society*, 14(2):112–118, mar 1982.

[75] J. F. Manning. Algorithmic detection and description of hyperbolic structures on closed 3–manifolds with solvable word problem. *Geometry & Topology*, 6(1):1–26, jan 2002.

[76] A. A. Markov. On the continuity of constructive functions (Russian). *Uspekhi Matematicheskikh Nauk*, No. 3(61), 226–230, 1954.

[77] A. A. Markov. *On constructive functions*, chapter in Twelve papers on logic and differential equations, pages 163–195. American Mathematical Society Translations, 1963.

[78] S. Mazur. Computable analysis. 1963. http://eudml.org/doc/268535.

[79] J. C. C. McKinsey. The decision problem for some classes of sentences without quantifiers. *Journal of Symbolic Logic*, 8(3):61–76, sep 1943.

[80] S. Meskin. A finitely generated residually finite group with an unsolvable word problem. *Proceedings of the American Mathematical Society*, 43(1):8–8, jan 1974.

[81] C. F. Miller III. *On Group-Theoretic Decision Problems and Their Classification. (AM-68)*. Princeton University Press, Mar. 1972.

[82] C. F. Miller III. The word problem in quotients of a group. In *Aspects of effective algebra*, 1981.

[83] C. F. Miller III. Decision problems for groups — survey and reflections. In *Mathematical Sciences Research Institute Publications*, pages 1–59. Springer New York, 1992.

[84] A. Minasyan, D. Osin, and S. Witzel. Quasi-isometric diversity of marked groups. *Journal of Topology*, 14(2):488–503, apr 2021.

[85] Y. Moschovakis. Recursive metric spaces. *Fundamenta Mathematicae*, 55(3):215–238, 1964.

[86] Y. Moschovakis. Notation systems and recursive ordered fields. *Compositio Mathematica*, 17:40–71, 1966.

[87] Y. Moschovakis. *Descriptive set theory*. North-Holland, Amsterdam New York, 1980.

[88] A. W. Mostowski. On the decidability of some problems in special classes of groups. *Fundamenta Mathematicae*, 59(2):123–135, 1966.

[89] A. W. Mostowski. Uniform algorithms for deciding group-theoretic problems. In *Word Problems - Decision Problems and the Burnside Problem in Group Theory*, pages 525–551. Elsevier, 1973.

[90] A. W. Mostowski. On computable sequences. In *Studies in Logic and the Foundations of Mathematics*, pages 336–350. Elsevier, 1979.

[91] V. Nekrashevych. A group of non-uniform exponential growth locally isomorphic to $IMG(z^2 + i)$. *Transactions of the American Mathematical Society*, 362(01):389–398, jul 2009.

[92] T. Netzer and A. Thom. Kazhdan's property (T) via semidefinite optimization. *Experimental Mathematics*, 24(3):371–374, jun 2015.

[93] B. H. Neumann. The isomorphism problem for algebraically closed groups. In *Word Problems - Decision Problems and the Burnside Problem in Group Theory*, pages 553–562. Elsevier, 1973.

[94] A. Nies. Describing groups. *Bulletin of Symbolic Logic*, 13(3):305–339, sep 2007.

[95] A. Y. Olshanskii and M. V. Sapir. The conjugacy problem and Higman embeddings. *Memoirs of the American Mathematical Society*, 170(804):0–0, 2004.

[96] A. Y. Olshanskii and M. V. Sapir. Subgroups of finitely presented groups with solvable conjugacy problem. *International Journal of Algebra and Computation*, 15(05n06):1075–1084, oct 2005.

[97] D. Osin. On the universal theory of torsion and lacunary hyperbolic groups. *Groups Complexity Cryptology*, 1(2), jan 2009.

[98] D. Osin. A topological zero-one law and elementary equivalence of finitely generated groups. *Annals of Pure and Applied Logic*, 172(3):102915, mar 2021.

[99] N. Ozawa. Noncommutative real algebraic geometry of Kazhdan's Property (T). *Journal of the Institute of Mathematics of Jussieu*, 15(1):85–90, jul 2014.

[100] P. Papasoglu. An algorithm detecting hyperbolicity. In *Geometric and Computational Perspectives on Infinite Groups*, pages 193–200. American Mathematical Society, nov 1995.

[101] P. F. Pickel. Finitely generated nilpotent groups with isomorphic finite quotients. *Transactions of the American Mathematical Society*, 160:327–327, 1971.

[102] M. Porter, A. Day, and R. Downey. Notes on computable analysis. *Theory of Computing Systems*, 60(1):53–111, dec 2016.

[103] M. O. Rabin. Recursive unsolvability of group theoretic problems. *The Annals of Mathematics*, 67(1):172, jan 1958.

[104] M. O. Rabin. Computable algebra, general theory and theory of computable fields. *Transactions of the American Mathematical Society*, 95(2):341, may 1960.

[105] E. Rauzy. Obstruction to a Higman embedding theorem for residually finite groups with solvable word problem. *Journal of Group Theory*, 24(3):445–452, Oct. 2020.

[106] E. Rauzy. Computable analysis on the space of marked groups. 2021.

[107] E. Rauzy. Remarks and problems about algorithmic descriptions of groups. 2021.

[108] E. Rauzy. Computability of finite quotients of finitely generated groups. *Journal of Group Theory*, 25(2):pp. 217–246, Mar. 2022.

[109] A. W. Reid. Profinite rigidity. In *Proceedings of the International Congress of Mathematicians (ICM 2018)*. World Scientific, may 2019.

[110] H. G. Rice. Recursive real numbers. *Proceedings of the American Mathematical Society*, 5(5):784–784, may 1954.

[111] H. Rogers. *Theory of Recursive Functions and Effective Computability*. MIT Press, Apr. 1987.

[112] J. J. Rotman. *An Introduction to the Theory of Groups*. Springer New York, Aug. 1999.

[113] M. V. Sapir. Asymptotic invariants, complexity of groups and related problems. *Bulletin of Mathematical Sciences*, 1(2):277–364, mar 2011.

[114] D. Segal. Decidable properties of polycyclic groups. *Proceedings of the London Mathematical Society*, s3-61(3):497–528, nov 1990.

[115] Z. Sela. Diophantine geometry over groups I: Makanin-Razborov diagrams. *Publications mathématiques de l'IHÉS*, 93(1):31–106, sep 2001.

[116] Z. Sela. Diophantine geometry over groups VII: The elementary theory of a hyperbolic group. *Proceedings of the London Mathematical Society*, 99(1):217–273, feb 2009.

[117] Y. Shalom. Rigidity of commensurators and irreducible lattices. *Inventiones mathematicae*, 141(1):1–54, jul 2000.

[118] A. Shen, N. K. Vereshchagin, and V. A. Uspensky. *Kolmogorov Complexity and Algorithmic Randomness*. American Mathematical Society, Providence, 2017.

[119] A. M. Slobodskoi. Unsolvability of the universal theory of finite groups. *Algebra and Logic*, 20(2):139–156, mar 1981.

[120] E. Specker. Nicht konstruktiv beweisbare sätze der analysis. *Journal of Symbolic Logic*, 14(3):145–158, sep 1949.

[121] D. Spreen. Representations versus numberings: on the relationship of two computability notions. *Theoretical Computer Science*, 262(1-2):473–499, jul 2001.

[122] H. Tietze. Über die topologischen invarianten mehrdimensionaler mannigfaltigkeiten. *Monatshefte für Mathematik und Physik*, 19(1):1–118, dec 1908.

[123] N. W. M. Touikan. Detecting geometric splittings in finitely presented groups. *Transactions of the American Mathematical Society*, 370(8):5635–5704, mar 2018.

[124] A. M. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265, 1937.

[125] A. M. Turing. On computable numbers, with an application to the Entscheidungsproblem. A correction. *Proceedings of the London Mathematical Society*, s2-43(1):544–546, 1938.

[126] M. K. Valiev. On polynomial reducibility of word problem under embedding of recursively presented groups in finitely presented groups. In *Mathematical Foundations of Computer Science (4th Symposium, Mariánské Lázně, September 1–5, 1975)*, pages 432–438. Springer Berlin Heidelberg, 1975.

[127] A. M. Vershik and E. I. Gordon. Groups that are locally embeddable in the class of finite groups. *St. Petersburg Math.*, 1998.

[128] K. Weihrauch. *Computability.* Springer Berlin Heidelberg, 1987.

[129] K. Weihrauch. *Computable Analysis.* Springer Berlin Heidelberg, Sept. 2000.

[130] J. Wilson and P. Zalesskii. Conjugacy separability of certain torsion groups. *Archiv der Mathematik*, 68(6):441–449, jun 1997.

[131] A. Zuk. Automata groups. In *Topics in noncommutative geometry. Third Luis Santaló winter school – CIMPA research school on noncommutative geometry, Universidad de Buenos Aires, Buenos Aires, Argentina, July 26–August 6, 2010.*, pages 165–196. Providence, RI: American Mathematical Society (AMS), 2012.