Voir, savoir, faire : une étude de cas en logique modale

François Schwarzentruber

THÈSE

En vue de l'obtention du

Doctorat de l'Université de Toulouse

Délivrée par : Université Toulouse III Paul Sabatier (UPS)
Discipline ou spécialité : Informatique

Présentée et soutenue par :
François Schwarzentruber

Le mercredi 1er décembre 2010

Titre :
Seeing, Knowing, Doing : Case Studies in Modal Logic

Directeurs de thèse :

Olivier Gasquet
Professeur - Université Paul Sabatier

Rapporteurs :

Frank Wolter
Professeur - Université de Liverpool

Philippe Schnœbelen
Directeur de recherche - ENS Cachan

Examinateurs :

Nicholas Asher (Président)
Directeur de recherche - IRIT

Hans Van Ditmarsch
Chercheur - Université de Séville

Emiliano Lorini
Chargé de recherche - IRIT

John-Jules Meyer
Professeur - Université d'Utrecht

École doctorale :
Mathématiques Informatique Télécommunications (MITT)

Unité de recherche :
Institut de recherche en Informatique de Toulouse (IRIT)

Seeing, knowing, doing
Case studies in modal logic

François Schwarzentruber

Contents

Acknowledgements/Remerciements

Je tiens à remercier mes parents pour m'avoir soutenu pendant ces trois ans de travail, surtout au téléphone. Je tiens également à remercier profondément Suzanne Muller, ma professeur de piano de m'avoir apporté tant. Je remercie ma marraine, Monique Schwarzentruber, d'être venue me soutenir... à ma soutenance.

Je remercie ensuite – bien qu'il n'y est pas d'ordre – mon directeur de thèse, Olivier Gasquet, attentif et toujours à l'écoute. Il sait être disponible, très abordable et très accueillant. Vouvoiement et autres politesses stupides sont bannis pour laisser place à la franchise, l'honnêteté et le travail. On se montre tel qu'on est, on dit ce qu'on pense et c'est là l'essence même d'un travail agréable.

Je remercie tous les membres du jury d'avoir relu mon rapport de thèse et d'avoir assisté à ma soutenance à savoir Frank Wolter, Philippe Schnoebelen, John-Jules Meyer, Hans Van Ditmarsch, Nicholas Asher et Emiliano Lorini.

Personne avec qui j'ai travaillé

- Mon guide Emiliano Lorini qui, sans cesse, m'a fait avancer, m'a ouvert les yeux sur des problématiques intéressantes. Il a toujours une idée de modélisation intéressante comme les émotions et des concepts de la théorie des jeux. Par ailleurs, il m'a donné de très bons conseils à propos de STIT et Lineland/Flatland. Je le remercie pour son exigence.
- Mon guide Andreas Herzig. Son humeur est toujours positive et tournée vers l'avenir : penser aux projets, penser aux nouvelles idées, penser à la communication, penser aux articles futurs, etc. Il est très à l'écoute et très encourageant. Il m'a beaucoup incité à publier certains des travaux dans des conférences et à présenter le travail dans certains workshops dans le but de communiquer nos idées.
- Mon guide Philippe Balbiani. Grâce à lui, le projet Lineland/Flatland a pu voir le jour. Il m'a guidé pour établir la complétude de Lineland et a ouvert la voie pour d'autres recherches dans ce domaine. Il m'a aussi appris le panel des logiques de l'espace que je présente dans le chapitre 1 et comment placer

Lineland/Flatland par rapport à l'état de l'art.

D'autres gens

De l'équipe LiLAC

- Nadine Guiraud, attentionnée et à l'écoute. Pleine d'esprit et d'idées. Elle est à mes côtés et je sais que je peux compter sur elle. C'est la personne la plus formidable que je connaisse.
- Bilal Saïd, attentionné et à l'écoute. On a travaillé sur le logiciel LoTREC. On a aussi fabriqué un stylo électronique pour dessiner sur l'écran d'un ordinateur. Je le remercie profondément pour les relectures ;
- Marwa El Houri, souriante et attentionnée, que je remercie profondément pour ses relectures ;
- Guillaume Feuillade pour sa bonne humeur avec qui j'ai organisé un TER l'année 2010 avec trois étudiants au sujet de recherche de stratégies dans des systèmes hydrauliques ;
- Fahima Cheikh, souriante et joviale ;
- Mounira Kourjieh, attentionnée et à l'écoute.
- Pablo Seban, ami humaniste et pacifique ;
- Dídac Busquets, co-bureau agréable et discret ;
- Srdjan Vesic, souriant et sans-papier ;
- Emmanuel Navarro pour sa bonne humeur dûe à Python ;
- Nguyen Manh Hung de m'avoir aidé pour les dossier ATER ;
- Nicholas Asher pour ses discussions de logiciens et sa bonne humeur ;
- Jonathan Ben-Naim, fan de recherche d'informations ;
- Yannick Chevalier, fan de réécriture ;
- Laurent Perrussel et Jean-Marc Thévenin pour leur bonne humeur (même s'ils sont finalement loins...)
- Stergos Afantenos, Anaïs Cadilhac, Frédéric Moisan, Pierre Bisquert... des nouveaux !

Des anciens de LiLAC

- Nicolas Troquard pour d'énormes discussions sur STIT ;
- Guillaume Aucher le grand fan de Gödel-Escher-Bach ;
- Tiago De Lima, féru de méthode des tableaux pour PAL ;
- Sihem Belabbes et le dessin du dromadaire.

Des autres équipes

- Jérôme Lang pour m'avoir prêté un livre sur la théorie des jeux ;
- Didier Dubois, Hélène Fargier et Martin Cooper pour une discussion sur un problème de complexité (le fragment existentiel de la logique des réels) ;
- Florence Dupin de Saint-Cyr - Bannay pour sa bonne humeur un peu extrême mais tellement agréable ;
- Mathieu Serrurier pour quelques discussions et pour avoir donné des TPs avec lui ;
- Florence Boué pour sa bonne humeur (bonjour ! je viens pour avoir du papier !) ;
- Meriam Bayoudh pour sa bonne humeur ;
- Gilles Richard pour sa bonne humeur et sa classe de formules de la logique des propositions dont le cardinal de l'ensemble des modèles de chaque formule est égal à 6.

Et aussi...

- Martine Labruyère pour m'avoir guidé dans ce monde administratif et dans la bonne humeur ;
- Evelyne Terral et Isabelle Babouin pour les missions à l'étranger et aussi l'organisation de la venue du jury ;
- Brigitte Marchiset pour les nombreux dépôts SVN ;
- Jean-Pierre Baritaud pour l'organisation technique de la soutenance.

Les autres chercheurs

- Hans Van Dittmarch, pour sa bonne humeur, son violoncelle et ses remarques à propos de Lineland et Flatland ;
- Frank Wolter pour son accueil dans son laboratoire à Liverpool et aussi d'avoir accepté d'être dans le jury de ma thèse ;
- Dirk Walther, Daniel Pokrywczynski and Michel Ludwig pour leurs accueils au laboratoire à Liverpool ;
- Mikhail Rybakov pour les discussions à propos de la complexité du problème de satisfiabilité de $S5_2$ avec un nombre fini de propositions ;
- et Stanislav Kikot pour sa bonne humeur et son originalité.

Mes autre amis

Les toulousains

- Carole Pichereaux pour sa disponibilité et sa bonne humeur ;
- Karine Villeneuve pour sa bonne humeur et ses idées ;
- Stéphanie Flipo pour sa bonne humeur et ses idées ;
- Madeleine Weber pour son enthousiasme ;
- Jean-Paul Ibrahim et Céline Leredde pour leur bonne humeur (et la petite Mélina aussi) ;
- Marion Lebellego pour ses bonnes idées et son enthousiasme ;
- Brice Loustau, Léo Monsaingeon et Julien Lequeurre pour leur folie ;
- Erwan Hillion, Tiphaine Jézéquel pour leur folie aussi ;
- Mathieu Leroy-Lerêtre pour ses poissons ;
- Sucen Liu pour sa bonne humeur et ses poissons... tropicaux !
- Jean-Marie Crevat, Monique Frémeau et François Nirrengarten pour leur soutien moral ;
- Serge Krichevsky et tout son orchestre.

Les amis lointains (en distance)

- Mathieu Gentês, par téléphone le plus souvent ;
- Les gens du Master 2 : Sylvain Lemouzy, Aurore Miquel, Mikaël Mayorgas, assez fous ;
- Les fourmis de Lemouzy ;
- Romain Legendre pour des discussions gEeKs ;
- Les gens de l'ENS : Clément Dunand, Rayan, Madeleine ;
- Sarah Marchoud, toujours pleine d'idées ;
- Guillaume Grodwohl et Fabien Perrot-Fréchin, loins...

Des gens que j'ai rencontré sur ma route

- Julien Dutant que j'ai rencontré durant ESSLLI 2010 et qui m'a gentillement ouvert les yeux sur la logique épistémique telle qu'elle est étudiée par les philosophes. Cela m'a permis de rédiger la remarque 1.

Chapitre 1

Résumé de la thèse en français

1.1 Introduction

Dans le domaine des jeux vidéos par exemple, surtout des jeux de rôles, les personnages virtuels perçoivent un environnement, en tirent des connaissances puis effectuent des actions selon leur besoin. De même en robotique, un robot percoit son environnement à l'aide de capteurs/caméras, établit une base de connaissances et effectuent des mouvements etc. La description des comportements de ces agents virtuels et de leurs raisonnements peut s'effectuer à l'aide d'un langage logique.

Dans cette thèse, on se propose de modéliser les trois aspects "voir", "savoir" et "faire" et leurs interactions à l'aide de la logique modale. Dans une première partie, on modélise des agents dans un espace géométrique puis on définit une relation épistémique qui tient compte des positions et du regard des agents. Dans une seconde partie, on revisite la logique des actions "STIT" (see-to-it-that ou "faire en sorte que") qui permet de faire la différence entre les principes "de re" et "de dicto", contrairement à d'autres logiques modales des actions. Dans une troisième partie, on s'intéresse à modéliser quelques aspects de la théorie des jeux dans une variante de la logique "STIT" ainsi que des émotions contre-factuelles comme le regret.

Tout au long de cette thèse, on s'efforcera de s'intéresser aux aspects logiques comme les complétudes des axiomatisations et les complexités des problèmes de satisfiabilité d'une formule logique.

L'intégration des trois concepts "voir", "savoir" et "faire" dans une et une seule logique est évoquée en conclusion et reste une question ouverte.

Dans cette thèse, on se base sur la logique modale épistémique dont la syntaxe est définie de la manière suivante :

$$\varphi ::= \bot \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid K_a\varphi$$

où p est une proposition atomique et $a \in AGT$.

La construction $K_a\varphi$ se lit "l'agent a sait que φ est vraie". La sémantique, elle, est définie en terme de modèle de Kripke, c'est à dire de structure $\mathcal{M} = (W, R, V)$ où :

- W est un ensemble non vide de *mondes possibles* ;
- R attribue à chaque agent a une relation épistémique R_a qui est une relation d'équivalence sur W ;
- V est une valuation qui spécifie dans chaque monde quelles propositions sont vraies.

On peut voir une telle structure comme un graphe où W est l'ensemble des sommets du graphe, R regroupe les différents arcs étiquettés et V les étiquettes des sommets. On définit ensuite la condition de vérité $\mathcal{M}, w \models \varphi$ par induction structurelle sur φ de façon usuelle. En particulier pour l'opérateur épistémique, cela donne :

- $\mathcal{M}, w \models K_a\varphi$ si, et seulement si pour tout $v \in R_a(w)$, $\mathcal{M}, v \models \varphi$.

La donnée $(\mathcal{M}, w)$ s'appelle un modèle pointé. On dira qu'une formule φ est satisfiable si, et seulement si il existe un modèle pointé $\mathcal{M}, w$ tel que $\mathcal{M}, w \models \varphi$. On dira que φ est valide si, et seulement si pour tout modèle pointé $\mathcal{M}, w$ on a $\mathcal{M}, w \models \varphi$.

Les modèles pointés interviennent dans deux problèmes de décision :

- Le model-checking : en entrée on donne un modèle pointé $\mathcal{M}, w$ et une formule φ et en sortie "oui" si, et seulement si on a $\mathcal{M}, w \models \varphi$;
- Le problème de satisfiabilité : en entrée on donne une formule φ et en sortie "oui" si, et seulement si la formule φ est satisfiable (autrement dit, il existe un modèle pointé $\mathcal{M}, w$ tel que $\mathcal{M}, w \models \varphi$).

De manière assez naturelle, le model-checking est dans P. Concernant le problème de satisfiabilité en logique épistémique, Y. Moses et J. Y. Halpern [JYH96] ont démontré qu'il est respectivement NP-complet s'il y a un seul agent dans le système et PSPACE-complete s'il y a deux agents.

Dans cette thèse, on trouvera une réécriture plus concise de l'algorithme proposé dans le papier de Y. Moses et J. Y. Halpern qui décide le problème de satisfiabilité en logique épistémique : il s'agit d'un algorithme alternant qui travaille en temps polynomial (Figure 2.4).

1.2 Vers une nouvelle logique modale de l'espace

Notre but est de créer une logique modale épistémique à partir de ce que les agents savent sans communication. Autrement dit, il s'agit de construire une logique qui parle de ce que les agents savent à partir de ce qu'ils voient. Leurs perceptions dépendent de la géométrie de l'environnement des agents. Dans l'état de l'art, il existe déjà une multitude de logiques de l'espace. Citons :

- La théorie des réels du premier ordre (étudiée par Tarski) utilisée en mathématiques dont les symboles sont les nombres, le $+$, le $\times$, l'égalité, la comparaison des nombres $<$, les connecteurs booléens, les quantificateurs universels et existentiels. Cette logique permet de parler de la géométrie via un système de coordonnées qui bien souvent utilise les nombres réels.

- P. Balbiani et V. Goranko [BG02] ont inventé une logique modale basée sur des opérateurs modaux comme "toutes les droites qui passent par le point courant vérifient...", "tous les points de la droite courante vérifient..." etc.

- La logique modale S4 offre une interprétation topologique de la construction $\Box\varphi$: φ est vraie dans tout un voisinage autour du point courant.

- RCC-8 est une théorie du premier ordre fournissant des prédicats comme "deux régions se touchent à la frontière", "la première région est incluse dans la seconde" etc.

Toutes ces logiques sont loin de nos besoins : dans toutes ces logiques, la syntaxe parle de l'espace (le langage contient des opérateurs qui parlent de l'espace et/ou des variables qui parlent de régions ou de coordonnées) et la sémantique est définie en termes géométriques. On souhaite construire une logique épistémique sans opérateurs modaux de l'espace dans la syntaxe et où la géométrie n'intervient que dans la sémantique. En d'autres termes, nous voulons une logique épistémique où la sémantique d'une formule est donnée uniquement à partir de la localisation des agents dans l'espace. Nous avons donc décidé de créer notre propre approche.

1.3 Connaissance dans Lineland

Nous créons ici une logique épistémique comme la logique épistémique traditionnelle sauf que les mondes possibles sont ici des mondes géométriques où les agents occupent une position dans l'espace. Ici, on se place dans un cadre simple où l'espace est une ligne. On définit donc un monde de Lineland $w = (<, \vec{dir})$ par la donnée d'un ordre total strict sur les agents et une fonction $\vec{dir} : AGT \rightarrow$

$\{\texttt{Left}, \texttt{Right}\}$ qui dit pour chaque agent a si l'agent regarde à gauche ($\vec{dir}(a) = \texttt{Left}$) ou s'il regarde à droite ($\vec{dir}(a) = \texttt{Right}$).

Ensuite, nous définissons des relations épistémiques R_a pour chaque agent a sur l'ensemble de ces mondes de Lineland. On définit wR_au si, et seulement si l'agent a voit exactement la même chose dans le monde w et le monde v.

La syntaxe de la logique qu'on considère ici ressemble beaucoup à la logique épistémique classique sauf qu'il n'y a pas de propositions atomiques qui sont remplacées par des constantes qui parlent de la perception des agents :

$$\varphi ::= a \rhd b \mid \bot \mid \neg\varphi \mid \varphi \vee \varphi \mid K_a\varphi$$

où a, b sont des agents. La construction $a \rhd b$ se lit "l'agent a voit l'agent b" et la construction $K_a\varphi$ se lit "l'agent a sait que la formule φ est vraie".

La sémantique est alors naturelle : $w \models a \rhd b$ se définit directement avec les localisations des agents a et b qui sont données par $<$ et $\vec{dir}(a)$ et $w \models K_a\varphi$ se définit de façon usuelle avec la relation R_a. On obtient une unique structure de Kripke spécifique où les mondes sont exactement les mondes de Lineland et les relations sont définies ci-dessus.

Les résultats techniques concernent le model-checking et le problème de satisfiabilité qui sont tous les deux PSPACE-complet. On donne un algorithme alternant pour le model-checking à la figure 4.5 qui utilise un temps polynomial pour s'exécuter et on prouve la PSPACE-difficulté via une réduction au problème de satisfiabilité d'une formule booléenne quantifiée. Il n'est pas étonnant ici que le model-checking soit PSPACE-difficile et non pas dans P car l'entrée du problème n'est pas la donnée d'une structure de Kripke et d'une formule mais d'un seul monde de Lineland (ordre total $<$ et fonction $\vec{dir}$) et d'une formule.

On donne également une axiomatisation complète de cette théorie épistémique dans le cas où les agents sont disposés sur une ligne.

On aborde également l'implémentation d'un model-checker même si cette implémentation n'est pour l'heure pas efficace (au sens pratique).

1.4 Connaissance dans Flatland

Nous abordons ici la même approche mais dans le plan. Les agents ont chacun une position dans le plan. Désormais un monde de Flatland est défini par un couple $\langle pos, \vec{dir}\rangle$ où :

- $pos : AGT \to \mathbb{R}^2$;
- $\vec{dir} : AGT \to U$.

où U désigne le cercle unité.

On associe une position à chaque agent mais aussi une direction vers laquelle l'agent regarde. On conviendra qu'un agent perçoit un demi-plan ouvert dans la direction vers laquelle il regarde.

De la même manière, on définit des relations épistémiques sur ces mondes là : on dira que wR_au si, et seulement si l'agent a voit la même chose dans w et u. Et de la même manière nous avons défini une et une seule structure de Kripke spécifique.

Le language est toujours le même à savoir :

$$\varphi ::= a \rhd b \mid \bot \mid \neg\varphi \mid \varphi \vee \varphi \mid K_a\varphi$$

où a, b sont des agents.

Le traitement du model-checking et du problème de satisfiabilité d'une formule φ sont plus compliqués dans le cadre d'un plan que dans le cadre de la ligne. Nous n'avons pas réussi à trouver une structure de données concise permettant de représenter simplement et qualitativement un monde dans le plan. C'est pourquoi nous n'avons pas de résultat précis concernant la complexité théorique de ces deux problèmes. Nous savons seulement qu'ils sont décidables via une traduction dans la théorie des réels de Tarski.

Nous proposons deux ébauches. On étend le langage avec des annonces publiques mais cela ne pose aucun problème concernant la décidabilité. On propose aussi une sémantique plus faible où l'on ne tient plus compte des positions exactes des agents dans le plan mais uniquement de ce que les agents voient. Dans ce cadre, le model-checking et le problème de satisfiabilité sont dans PSPACE. Néanmoins, une implémentation efficace semble pour l'instant un problème difficile.

1.5 Vers la logique STIT

Notre but à présent est d'avoir un langage qui parle de ce que les agents font. Il y a dans la littérature plusieurs formalismes logiques qui semblent correspondre à notre attente même si ce n'est pas le cas :

- La logique PDL est une logique des actions (vues comme des programmes). C'est une logique modale qui offre des constructions modales comme $[\pi]\varphi$ signifiant "après l'exécution du programme π, il y a toujours φ qui est vraie. Le défaut principal de cette logique pour nous est de ne pas parler d'agents !

- La famille des logiques de coalitions (logique de coalition de Marc Pauly, Alternating-Time Logic etc.) sont des logiques de l'action qui offrent des constructions modales de la forme $\langle\langle J\rangle\rangle\varphi$ qui signifie "le groupe d'agents J

peut faire en sorte que φ soit vrai". Avec cette famille de logiques, on peut exprimer ce que les agents peuvent faire mais pas du tout ce que les agents font réellement. En particulier voici deux défauts :

- On ne peut pas mélanger l'une de ces logiques avec la logique épistémique pour exprimer facilement la différence entre de re (je peux faire une action pour que φ soit vraie et je sais laquelle) et de dicto (je sais que je peux faire une action pour que φ soit vraie mais je ne sais pas laquelle).
- Il est difficile de représenter des émotions contre-factuelles comme le regret à l'aide de ces logiques étant donné que de telles émotions dépendent des actions qu'on entreprend et non pas juste des actions qu'on peut entreprendre.

La famille des logiques STIT développé par Belnap etc. correspond à notre attente. Dans cette thèse, j'ai étudié la logique STIT suivante :

$$\varphi \ ::= \ \bot \ \mid \ p \ \mid \ (\varphi \vee \varphi) \ \mid \ \neg\varphi \ \mid \ [J]\varphi \ \mid \ X\varphi$$

où $[J]\varphi$ signifie "le groupe d'agents J fait en sorte que φ soit vraie" où $X\varphi$ se lit "φ est vraie à l'instant suivant".

Un modèle de la sémantique donné par Belnap de cette logique est une structure temporelle branchée (un arbre). Les nœuds de l'arbre sont appelés moments. Une histoire est une branche (un ensemble maximal linéairement ordonné de moments). On dira que deux histoires h_1 et h_2 se divisent au moment m si le moment m est le moment le plus dans le futur qui est commun aux deux histoires h_1 et h_2. On ajoute à chaque moment m d'une structure, une fonction de choix C qui pour chaque coalition J, partitionne l'ensemble des histoires passant par m en classes d'équivalence. Chaque classe d'équivalence représente un choix de la coalition J.

Bien sûr chaque fonction de choix C doit satisfaire des conditions techniques afin de bien modéliser la notion de choix :

- La propriété d'additivité : les histoires que peuvent choisir ensemble une coalition J est l'intersection des histoires que peut choisir chaque agent $a \in J$;
- Pas de choix entre des histoires non divisées : il est impossible à un certain moment m pour une coalition de choisir entre deux histoires qui ne se divisent pas en m. En d'autres termes, si deux histoires ne se divisent pas, elles correspondent toutes les deux aux même choix des agents ;

- Indépendance des agents : chaque agent a du système peut décider son propre choix et l'action jointe de la grande coalition AGT composée des actions choisis par chaque agent existe toujours.

Dans le but de simplicité et aussi afin de pouvoir combiner cette logique, on donne dans cette thèse une sémantique équivalente en termes de modèles de Kripke.

1.6 Problème de satisfiabilité et axiomatisation de fragments de STIT

Nous commençons par prouver que la logique STIT atemporelle est indécidable si le nombre d'agents est supérieur ou égal à 3 via une réduction à la logique modale produit $S5^n$. On prouve également que cette logique n'est pas axiomatisable avec un nombre fini de schémas d'axiomes et les seules règles de modus ponens et nécessitation.

C'est pourquoi nous nous intéressons à des fragments où l'on restreint les coalitions autorisées J dans le langage quand on écrit un opérateur $[J]$.

Par exemple P. Balbiani *et al.* avaient déjà montré que, si on ne s'autorise que des coalitions individuelles, c'est à dire des opérateurs $[\{a\}]$ où a est un agent, alors le problème de satisfiabilité est NEXPTIME-complet et que ce fragment est axiomatisable. Dans cette thèse, on étend ce résultat à un fragment atemporel plus grand : on commence par remarquer que si on autorise les opérateurs $[\{a\}]$ ainsi que l'opérateur pour la grande coalition $[AGT]$ alors le problème de satisfiabilité reste décidable et NEXPTIME-complet. Mieux, en fait, si on autorise toutes les coalitions présentes dans le treillis de la figure 7.3 alors le problème de satisfiabilité reste NEXPTIME-complet et la logique demeure axiomatisable. Ces résultats de décidabilité sont montrés à l'aide d'une filtration que l'on adapte un peu pour que les propriétés des modèles restent respectés. En fait, on donne également les propriétés du modèle fini pour plusieurs classes de modèles avec des bornes sur la taille de ces modèles. Par ailleurs, on montre que le problème de satisfiabilité de la logique où les coalitions sont imbriquées les unes dans les autres $J_1 \subseteq J_2 \subseteq J_3 ...$ est NP-complet si le nombre de coalitions est fixé et PSPACE-complet si le nombre de coalitions n'est pas connu à l'avance.

On remarque enfin que la logique temporelle avec l'opérateur X peut se plonger dans la logique atemporelle d'où on déduit également les complexités pour les problèmes de satisfiabilité. Le problème de satisfiabilité d'une formule du STIT individuel avec un seul agent et l'opérateur temporel X est PSPACE-complet et le problème de satisfiabilité d'une formule du STIT individuel avec au moins deux agents et l'opérateur temporel X est NEXPTIME-complet.

1.7 Un fragment STIT faible

On a étudié des fragments dans lesquels on a restreint les coalitions que l'on peut écrire dans une formule. À présent, on étudie un fragment où l'on interdit l'imbrication des opérateurs modaux, mais en autorisant toutes les coalitions possibles. Le fragment syntaxique est le suivant :

$\chi ::= \bot \mid p \mid \chi \wedge \chi \mid \neg\chi$ (formules propositionnelles)
$\psi ::= [J]\chi \mid \psi \wedge \psi$ (formules STIT)
$\varphi ::= \chi \mid \psi \mid \varphi \wedge \varphi \mid \neg\varphi \mid \langle\emptyset\rangle\psi$ (formules du langage))

Par exemple, la formule $[\{1\}][\{1,2\}]p$ n'est pas autorisée dans le langage.

On démontre un résultat de propriété du petit modèle à l'aide d'un argument de type "sélection de points" : en fait, toute formule satisfiable est satisfiable dans un modèle de taille polynomiale. De fait, on obtient que le problème de satisfiabilité d'une formule de ce fragment est NP-complet. On donne aussi un résultat d'axiomatisation : toutes les validités du fragment syntaxique sont démontrables.

1.8 Logique modale pour des jeux épistémiques

À présent, on donne un formalisme logique pour pouvoir raisonner à propos des jeux épistémiques. Il s'agit d'une logique modale et d'une sémantique en termes de modèles de Kripke, ainsi que d'une axiomatisation. On donne aussi des résultats de complexité.

Le langage de cette logique fournit plusieurs opérateurs modaux :

- Un opérateur d'action qui permet des constructions du type "la coalition C exécute l'action jointe δ_C" ;
- $\Box\varphi$: "dans tous les états possibles du jeu, φ est vraie" ;
- $K_i\varphi$: "l'agent i sait que φ est vraie" ;
- φ est vraie dans les mondes qui sont meilleurs non strictement pour l'agent i que le monde courant.

On se place dans un premier temps dans le cadre d'information complète. Le problème de satisfiabilité est NP-complet. On montre aussi comment exprimer les notions de meilleures réponses, d'équilibres de Nash et de rationalité à l'aide de ce langage logique. On retrouve des théorèmes connus de la théorie des jeux. On définit aussi la notion de stratégie strictement dominée et l'algorithme d'élimination des stratégies strictement dominées (IDSDS).

On fournit également une représentation plus concise de cet algorithme en ajoutant la notion d'une variante d'annonces publiques au langage.

On continue notre étude avec les jeux à information incomplète qu'on est aussi capable de représenter dans notre logique en affaiblissant notre logique à information complète : on relâche la contrainte qui liait $\Box$ et K_i. Dans ce cadre plus général, le problème de satisfiabilité est PSPACE-complet à moins qu'il n'y ait qu'un seul agent et qu'une seule action auquel cas il est NP-complet.

1.9 Emotions contre-factuelles

La logique STIT permet aisément de représenter la notion de responsabilité "le groupe d'agents J aurait pu éviter que χ soit vraie". Formellement, cela s'écrit de la façon suivante :

$$\mathtt{CHP}_J\chi \stackrel{\text{def}}{=} \chi \wedge \neg[AGT \setminus J]\chi.$$

c'est à dire χ est vraie et il est faux que les autres agents font en sorte que χ soit vraie (i.e. les autres permettent $\neg\chi$).

Pour représenter une émotion contre-factuelle on a besoin d'un opérateur épistémique. On étend donc le langage de STIT avec un opérateur épistémique.

Bien sûr, comme la logique STIT est indécidable, il est souhaitable d'utiliser un fragment syntaxique de cette logique si on veut l'utiliser en pratique dans un système qui représente des émotions contre-factuelles. On s'intéresse donc au langage suivant :

$\chi ::= \bot \mid p \mid \chi \wedge \chi \mid \neg\chi$ (formules propositionnelles)
$\psi ::= [J]\chi \mid \psi \wedge \psi$ (formules STIT)
$\varphi ::= \chi \mid \psi \mid \varphi \wedge \varphi \mid \neg\varphi \mid \langle\emptyset\rangle\psi \mid K_i\varphi$ (formules du langage)

Avec ce fragment (qui est une extension du fragment STIT vu précédemment), le problème de satisfiabilité d'une formule est PSPACE-complet. De plus, on donne une axiomatique : toute validité de ce fragment est démontrable.

À présent introduisons des atomes spéciaux $good_i$ pour tout agent $i \in AGT$. Ces atomes spéciaux désignent les mondes qui sont bons pour un agent.

On dira que χ est bon pour l'agent i si, et seulement si χ est vrai dans tous les mondes bons. Formellement :

$$\mathtt{GOOD}_i\chi \stackrel{\text{def}}{=} [\emptyset](good_i \rightarrow \chi).$$

On dira que χ est désirable pour l'agent i si, et seulement si i sait que χ est quelque chose de bon pour lui :

$$\mathtt{DES}_i\chi \stackrel{\text{def}}{=} K_i\mathtt{GOOD}_i\chi.$$

On peut ensuite définir dans notre langage quatre émotions contre-factuelles que sont le regret, la réjouissance, la déception, l'allégresse :

$$\mathtt{REGRET}_i\chi \stackrel{\text{def}}{=} \mathtt{DES}_i\neg\chi \wedge K_i\mathtt{CHP}_i\chi.$$

$$\texttt{REJOICE}_i\chi \stackrel{\text{def}}{=} \texttt{DES}_i\chi \wedge K_i\texttt{CHP}_i\chi.$$

$$\texttt{DISAPPOINTMENT}_i\chi \stackrel{\text{def}}{=} \texttt{DES}_i\neg\chi \wedge K_i\texttt{CHP}_{AGT\setminus\{i\}}\chi.$$

$$\texttt{ELATION}_i\chi \stackrel{\text{def}}{=} \texttt{DES}_i\chi \wedge K_i\texttt{CHP}_{AGT\setminus\{i\}}\chi.$$

Ces quatre émotions sont définies à partir de deux variables :

- selon que l'on désire χ ;
- selon que la responsabilité de χ vienne de i ou des autres agents.

1.10 Conclusion

Dans cette thèse, on a proposé de multiples formalismes en logique modale pour parler de la perception des agents dans un monde à une dimension ou deux dimensions. On a étudié une logique des actions, STIT, ainsi que les fragments. On a également décrit une logique modale permettant de modéliser certains concepts de la théorie des jeux. On a développé une logique permettant de décrire des émotions contre-factuelles.

Les perspectives sont à présent nombreuses :

- Inclure du dynamisme comme des opérations de mise à jour etc. : changement d'émotions, changement de connaissances sur l'état physique du monde.
- Etudier des variantes concernant la perception : d'autres géométries, d'autres types de perceptions, etc.
- Construire un cadre logique permettant de décrire toute la chaîne de représentation des connaissances d'un agent : représentation du monde physique et des actions des autres agents. On pourrait introduire des axiomes d'interactions entre perception et connaissance qu'une action est réalisée comme :

  $$a \rhd b \rightarrow ([\{b\}]\varphi \rightarrow K_a[\{b\}]\varphi)$$

  c'est à dire si un agent a voit un agent b et que b réalise une action telle que φ est vraie alors l'agent a sait que l'agent b réalise cette action.
- Trouver une façon d'implémenter efficacement les procédures de décision pour les logiques présentes dans cette thèse. Pour l'instant, raisonner directement avec la théorie des réels pour faire du raisonnement épistémique sur le monde est inefficace. Par exemple, le fragment STIT faible est NP-complet mais nous n'avons pas de bonne procédure de décision, procédure de décision qui aiderait grandement à la réalisation d'un système qui sait raisonner

sur les émotions contrefactuelles. L'implémentation de ces procédures de décisions pourraient être utilisée pour créer un jeu vidéo où les personnages virtuels sont dotés de raisonnement et de représentation des connaissances en terme d'état du monde virtuel et des actions des autres agents.

Chapter 2

Introduction

This thesis deals with knowledge reasoning using epistemic modal logic. Epistemic modal logic was already been studied in the domain of Artificial Intelligence [JYH96]. The contribution of this thesis is to investigate knowledge reasoning about two complementary issues: knowledge about the perception of what agents see in the world (Part I) and knowledge about actions and also about emotions (Parts II and III).

2.1 Our aim: reasoning about knowledge

In this section, we first present two examples in order to give an intuition about reasoning about knowledge, agents, actions, perceptions, etc. We then explain what is *automated* reasoning. Finally, we give some applications to illustrate how reasoning about knowledge can be useful in the real life.

2.1.1 Two examples

2.1.1.1 Muddy children

Let us begin with the famous example of the Muddy children [GO06], [Pla07], [FHMV95]. Let us consider three children Fahima, Marwa and Nadine settled as in the Figure 2.1. In particular, we suppose that their foreheads are dirty. Each of them do not know whether she is dirty or not but knows that the other are dirty.

Suppose that the following sentences are true for Fahima:

- Fahima does not know that she is dirty.
- Fahima knows that Marwa is dirty.

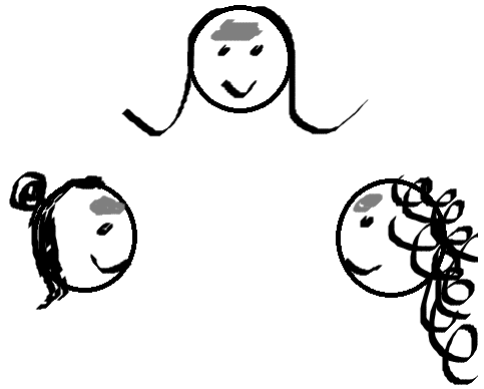


Figure 2.1: Muddy children with 3 children: Fahima, Marwa and Nadine.

- Fahima knows that Nadine is dirty.
- Fahima knows that Marwa knows that Nadine is dirty.
- Fahima knows that Marwa knows whether Fahima is dirty or not.
- Fahima knows that Nadine knows that Marwa is dirty.
- Fahima knows that Nadine knows whether Fahima is dirty or not.
- Fahima does not know if Marwa knows that Nadine knows that there is at least one of the children which is dirty, etc.

Now the father of Fahima, Marwa and Nadine comes and says: "at least one of you is dirty." We suppose that each child trusts the father and that each child knows that the others trust the father and so on. In fact, each of them learn this fact and thus Fahima knows the following sentence:

- Fahima knows that everyone knows that everyone knows that at least one of the children is dirty.

The father asks every child if she knows whether she is dirty or not. Every child answers the truth: actually they do not know. Thus, Fahima makes this reasoning:

- Suppose I am dirty. Marwa sees and knows that Nadine is dirty. Marwa would also know that I am dirty. Hence Marwa would know that two children are dirty. The same for Nadine: she would know that two children are dirty.

- Suppose I am clean. Marwa cannot imagine she is clean. Otherwise Nadine would have answered "I am dirty". In the same way, Nadine cannot imagine she is clean. Otherwise Marwa would have answered "I am dirty".
- Conclusion: Nadine and Marwa do the same reasoning. Everybody knows that there are at least two dirty children.

Then the father asks again to every child if she knows whether she is dirty or not. Every child answers the truth: actually they do not know. Thus, Fahima makes this reasoning:

- Suppose I am clean. In this case, Marwa would see that only Nadine is dirty. Furthermore, as Marwa actually knows that two of the children are dirty Marwa would have answered "I am dirty". But this was not the case.
- Conclusion: I know that I am dirty.

This example partly relies on what agents see. If Fahima did not see Marwa and if she did not know that Marwa sees her etc. she would not have been able to deduce that she was dirty. In Part I we propose a knowledge representation for problems dealing with perception.

2.1.1.2 Prisoner's dilemma

Let us consider two boys: Bilal and Pablo. Every day their father give them 5€ for pocket-money. Bilal and Pablo are usually wise but today, they are not: they have eaten together all the ice cream from the fridge! The father is not aware that all the ice cream has been eaten. Bilal and Pablo have two choices: admit the fault to the father or say nothing. But the two boys know how the father can react:

- if they both admit the fault, Bilal will have only 2€ for pocket-money and Pablo will have only 2€ for pocket-money;
- if they both say nothing, Bilal and Pablo will both have 5€ as usual;
- if Bilal admit the fault and Pablo say nothing, Bilal will have 3€ but Pablo will have nothing because he is not honest;
- if Pablo admit the fault and Bilal say nothing, Pablo will have 3€ but Bilal will have nothing because he is not honest.

The best choice (*Nash equilibrium*) is that they both admit the fault. Indeed, if Bilal admit the fault, he can win 2€ or 3€. But if Bilal say nothing, he may have no money from the father. This example partly relies on the knowledge about the actions and preferences of Bilal and Pablo. In Part II and III we will give different representations for problems dealing with actions and preferences.

2.1.1.3 Towards automated reasoning

We want a computer to be able to reason about knowledge (Fahima knows that her forehead is dirty), action (Bilal say nothing, Pablo admit the fault, etc.), preferences (Bilal prefer to win 2€ than nothing), emotions (Bilal regrets to have said nothing). We want also the computer to reason about mix of the previous ingredients: for instance, we want to design algorithms able to automatically reason about sentences like "Bilal knows that Pablo prefer that Fahima feel regret". In this sense, this thesis is part of the field of *artificial intelligence*.

In order to make a computer reason/compute, we need data structures to represent knowledge, action, preferences, emotions. In this sense, this thesis also deals about *knowledge representation*. Throughout this thesis, the most important data structures are Kripke structures that give a semantics to . Modal logic is suitable for many reasons:

- Modal logic is close to natural language. For instance, the fact "Fahima knows that Marwa's forehead is dirty" is represented by the *formula* $K_{Fahima}dirty_{Marwa}$. The symbol K_{Fahima} is called a *modal operator* and its meaning is "Fahima knows that". The symbol $dirty_{Marwa}$ is called an *atomic proposition* and represents the atomic fact "the forehead of Marwa is dirty". Thus the representation in terms of formulas is easy;

- Like the classical propositional logic, modal logics' satisfiability problems and model-checkings (see Definition 3 and 4) are often *decidable*: generally speaking, it means that we can use modal logics practically with a computer for automated reasoning.

- Modal logic is *expressive*. For instance, we can nest modal operators, that is to say we are able to reason about complex formulas like

  $$K_{Fahima}\neg K_{Nadine}dirty_{Marwa}$$

  ("Fahima knows that Nadine does not know that Marwa is dirty"). In particular, this kind of complex formulas cannot be easily expressed in classical propositional logic.

Generally speaking, the more a logic is expressive, the more it is difficult to reason with:

- classical proposition logic is "very" decidable (see Theorem 3) but not really expressive;

- Modal logics are often "quite" decidable (see Theorem 5 and "quite" expressive and suitable for artificial intelligence aspects;

- the *first order logic* is undecidable [Chu36], [Tur37] but "very" expressive, although not very suitable for artificial intelligence aspects (far from natural speaking)

2.1.2 Possible applications

2.1.2.1 Toy for kids

An Australian company [Ada09] is developing a teddy bear that can interact with young kids and propose activities. Modal logic may help this project to have a clean knowledge representation for the toy for preferences of the kid, knowledge of the environment of the kid. For instance:

- the teddy bear may know that the kid prefers listen to classical music than to jazz; (knowledge and preferences);
- the teddy bear may know that if the kid knows that there is a cat in the room and that the cat is looking at the teddy bear, the kid will cry. (knowledge, perception, action)
- the teddy bear may understand the notion of regret of an action of the kid and then sing a happy song to the kid in order to calm her.

2.1.2.2 Video Games

In the domain of video games, there are some specific kinds of games where the player are evolving in a virtual world (for instance the Middle-earth [TSBH95]). Such a game is often called *role playing game*. In the virtual world of such a role playing game, there are inhabitants, for instance weapon sellers, innkeepers, kings, dwarfs, warriors, etc. Those characters are artificial and are designed by the computer. In particular, the computer controls the behaviors of those different agents and the behavior of an agent is closely related to her knowledge about the world as well as agents actions, preferences, emotions etc. For instance:

- Dwarfs hates when elves wear hats (preferences);
- Dwarfs prefer to have axes than swords;
- The dwarf Bilal knows that the elf Pablo has sold his weapon (knowledge and action);
- The elf Pablo regrets to have sold his weapon but the dwarf Bilal does not know it (knowledge, action and emotion).

Thus, modal logics seem to provide a clean framework to represent knowledge of an artificial inhabitant of the virtual world of a role playing game and then to deduce a rational behavior of such an agent.

2.1.2.3 Modeling the world

Another more application of logician is to understand and to explain the world. As the physician Niels Bohr has modeled the atom with a planetary-model atom, modal logicians may model knowledge reasoning etc. with Kripke modal logic defined below.

2.2 Epistemic modal logic

Epistemic modal logic is a modal logic [BDRV02], [Che80], [HC72], [GO06], [Hin62] concerned by the notion of *knowledge*. Its name comes from the Greek word $\epsilon\pi\iota\sigma\tau\eta\mu\eta$ or "episteme" meaning knowledge.

2.2.1 Syntax

In this section we present the language of epistemic logic [JYH96]. The syntax is the raw symbols. Let us consider a countable infinite set ATM of atomic propositions and a finite non-empty set AGT of agents. The language is defined by the following rule:

$$\varphi \ ::= \ p \ \mid \ \bot \ \mid \ \neg\varphi \ \mid \ (\varphi \vee \varphi) \ \mid \ K_a\varphi$$

where p ranges over ATM and a ranges over ATM.

The intuitive meaning of $K_a\varphi$ is "agent a knows that φ is true". K_a is called a *modal operator*. As usual $\top =^{\text{def}} \neg\bot$, $(\varphi \wedge \psi) =^{\text{def}} \neg(\neg\varphi \vee \neg\psi)$, $\hat{K}_a\varphi =^{\text{def}} \neg K_a\neg\varphi$, $\varphi \rightarrow \psi =^{\text{def}} (\neg\varphi \vee \psi)$ and $(\varphi \leftrightarrow \psi) =^{\text{def}} ((\varphi \rightarrow \psi) \wedge (\psi \leftrightarrow \varphi))$. We follow the standard rules for omission of parentheses.

Example 1 *The formula $sun \wedge \neg K_a sun$ means "the sun is shining but the agent a does not know that the sun is shining".*

Now there are formulas that seem to be always false: $\bot$, $K_a sun \wedge \neg sun$, etc. There are formulas that seem to be always true: $\neg\bot$, $K_s un \vee \neg K_s un$, etc. There are formulas that can be true: $sun \wedge \neg K_a sun$, etc. In order to formally classify formulas that seem to be always false, formulas that can be true and formulas that seem to be always true, we define in the next section the *semantics* of the epistemic logic.

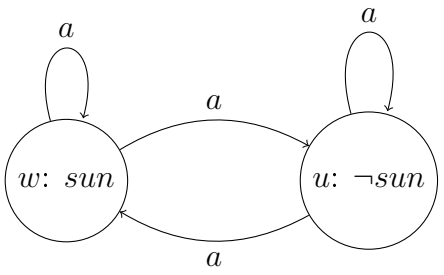


Figure 2.2: A model with possible worlds

2.2.2 Semantics

As we have seen, it seems that some formula are always true, some of them are surely false and some of them are sometimes true and sometimes false. In fact, the truth of a formula depends on a context: to say whether a formula is true or not we need a *model*.

Models are here made up of *possible worlds*. This kind of semantics have been introduced by Kripke in [Kri63]. The reader can find more informations in [Che80], [BDRV02] and [HC72]. The idea is that we model a real life situation by considering different *worlds*. One world is reported to be the *real world*. Then we introduce relations R_a for each agent a modeling the knowledge. Given two worlds w and v, wR_av means that agent a cannot distinguish world w from world v: they are both possible worlds for agent a. Finally we obtain a *graph*: vertices are worlds and edges is given by the relation R_a.

Example 2 *The Figure 2.2 shows a Kripke model for the formula* $sun \wedge \neg K_a sun$. *The world (or node)* w *stands for the real world. As agent* a *does not know that the sun is shining, she can imagine a world* u *where the sun is not shining. Such a world she can imagine is called a possible world.*

In what follows, we formally define the model:

Definition 1 (model)
A *model* is a tuple $\mathcal{M} = (W, R, V)$ where:

- W is a non-empty set of *worlds*
- $R : AGT \to W \times W$ is an *equivalence relation*;
- $V : ATM \to 2^W$.

A model $\mathcal{M}$ is sometimes called a *Kripke structure*, a *Kripke model* and so on. Given a model $\mathcal{M} = (W, R, V)$, the couple (W, R) is called a *frame* [BDRV02] [GKWZ03]. V is called *valuation*. The notion of frame is important in order to prove completness of an axiomatics [Sah75]. Elements of W are sometimes called nodes, states, possible worlds, possible states and so on. For all agents a, R_a is an equivalence relation. Indeed the relation "are indistinguishable from agent a" is supposed to be an equivalence relation.

In order to define how to evaluate modal formulas over a Kripke model, we give the following truth conditions:

Definition 2 (truth conditions)
We define $\mathcal{M}, w \models \varphi$ by induction on φ:

- $\mathcal{M}, w \models p$ iff $w \in V(p)$;
- $\mathcal{M}, w \not\models \bot$;
- $\mathcal{M}, w \models \varphi \vee \psi$ iff $\mathcal{M}, w \models \varphi$ or $\mathcal{M}, w \models \psi$;
- $\mathcal{M}, w \models K_a\varphi$ iff for all $u \in W$, wR_au implies $\mathcal{M}, u \models \varphi$.

The formula $K_a\varphi$ is true in w iff φ holds in all possible worlds for agent a. The more possible worlds an agent have the more she is ignorant. On the contrary, when a agent a knows everything, then the set of all possible worlds $R_a(w)$ is $\{w\}$: she only the real world as possible world. The operation of learning consists in deleting worlds in the model, for instance public announcements (see [Pla07] and Chapter 4, 5, 9).

We have supposed the relation R_a to be an equivalence relation. That is why those formulas are true in all worlds in all models:

- $K_a(\varphi \to \psi) \to (K_a\varphi \wedge K_a\psi)$ (K);
- $K_a\varphi \to \varphi$ (T);
- $K_a\varphi \to K_aK_a\varphi$ (positive introspection) (4);
- $\neg K_a\varphi \to K_a\neg K_a\varphi$ (negative introspection) (5)

This logic is called $\mathsf{S5}_n$ where n is the number of agents in AGT. Traditionally $\mathsf{S5}$ is the logic with only one operator associated with one equivalence relation and $\mathsf{S5}_n$ is called the *fusion* of $\mathsf{S5}$, $\mathsf{S5}, \ldots$ and $\mathsf{S5}$ (n times).

Remark 1 *Some authors think that requiring the relation to be an equivalence relation is too strong:*

- *Hintinkka rejects the the negative introspection (5) "unless you happen to be as sagacious as Socrates". [Hin62] He models epistemic reasoning with the logic S4 (see Subsubsection 3.2.2.2).*

- *Stalnaker [Sta06] studies the combination of the logic of belief $KD45$ (axiom 5 for beliefs does not yield to a contradiction), the logic of knowledge S4 plus the interactions $B_a\varphi \rightarrow K_aB_a\varphi$ (positive introspection of beliefs), $\neg B_a\varphi \rightarrow K_a\neg B_a\varphi$, $K_a\varphi \rightarrow B_a\varphi$ (knowledge implies belief), $B_a\varphi \rightarrow B_aK_a\varphi$ (strong belief) where B_a means "agent a believes that". In this system, he claims that we can prove $B_a\varphi \leftrightarrow \hat{K}_aK_a\varphi$ and that the operator K_a verifies the principles of the logic $S4.2$.*

- *Williamson rejects the positive introspection (4) in [Wil02].*

We decide in this thesis to focus on the system S5 for the sake of simplicity.

Now we can consider two classical decision problems linked to the Definition of truth conditions.

2.2.3 Two decision problems

A *decision problem* [Pap03] asks a question about a mathematic object (the input) that requires either a "yes" or a "no" answer (the output). In this subsection, we consider two classical decision problems: the model-checking and the satisfiability problem.

2.2.3.1 Model-checking

Definition 3 (model-checking)
The model-checking is the problem traditionally defined as follows:

- Input: a finite Kripke structure $\mathcal{M} = (W, R, V)$, a point $w \in W$ and a formula φ;

- Output: Yes iff $\mathcal{M}, w \models \varphi$.

In the model-checking, the input is made up of a graph $\mathcal{M}$ (a finite Kripke structure), a world w and a formula φ. The corresponding question about the input $\mathcal{M}$, w, φ is to know whether $\mathcal{M}, w \models \varphi$ or not. Model-checking has been widely studied in the litterature [BBF$^+$01] especially for temporal logics and their applications. Note that in Chapter 4 and 5, we will study specific kinds of model-checking where the input is a bit different. More specifically we do not give a Kripke structure as an input but another compact data structure representation of it.

2.2.3.2 Satisfiability problem

A formula φ is said to be *satisfiable* iff there exists a mode $\mathcal{M} = (W, R, V)$ and a world $w \in W$ such that $\mathcal{M}, w \models \varphi$.

Definition 4 (satisfiability problem)
The *satisfiability problem* is traditionally defined as follows:

- Input: a formula φ;
- Output: Yes iff the formula φ is satisfiable.

In the satisfiability problem, the input is a formula φ and the corresponding question about the input φ is to know whether φ is satisfiable or not.

In the same way a formula is said to be *valid* iff $\neg\varphi$ is not satisfiable. We can also be interested to the *validity problem* but we prefer to deal with the satisfiability problem. This choice is subjective: some people of LiLAC Team are focusing on satisfiability problems, model constructions and tableau methods etc. and are developing a satisfiability problems' solver for modal logic [Sai10]. Furthermore, the satisfiability problem is the "dual" of the validity problem in the sense that a formula φ is valid iff $\neg\varphi$ is not satisfiable. In fact, the main concern of this thesis is to study satisfiability problems in different contexts: epistemic modal logic with an original semantics dealing with geometry (part I), logics dealing with agents' actions (part II) and logics mixing knowledge and actions (part III). The word "satisfiability" is one of the most used word in this thesis!

2.3 Complexity classes

In this Section, we are interested in the notion of algorithm for solving decision problems and of complexity classes. Algorithms are effective, constructive, mechanical methods designed to solve decision problems. Running an algorithm requires time and space (memory) in order to compute the ouput of the decision problem.

The *Church-Turing thesis* [Tur37], [Chu36] states that the computation of an algorithm can be carried out a Turing machine. Usually, the following definitions are given in terms of Turing machine and the interested reader can find more on this in [Pap03], [CKS81]. Here for the sake of clarity, we decided to give some intuitions about the definition of determinism, non-determinism, alternation, P, NP etc. in terms of algorithms.

2.3.1 Algorithms

In this subsection, we present the notion of deterministic, non-deterministic and alternating algorithms. An algorithm is made up of affectations $x := v$, conditional **if** , **for** loops etc. An algorithm takes an input i and may succeed, fail or never halt. For a thorough introduction to the subject of algorithms see [Knu73], [AHU83] or [CLR92]. The notion of algorithm is informal and closer to the reality (programs in Java, Scheme etc.)

An algorithm is said to be *deterministic* iff there is no choice during the execution of the algorithm. If the machine is in a given state, then there is only one next state. We say that a deterministic algorithm decides a decision problem P iff for all input i the algorithm succeeds on the input i if the output of i in the problem P is "yes" and the algorithm fails on the input i if the output of i in the problem P is "no". We also say that P is *decidable*. Otherwise the problem is undecidable. Typically, algorithms written in Java, Scheme etc. are deterministic.

An algorithm is said to be *non-deterministic* iff there are existential choices during the execution of the algorithm. Sometimes, the algorithm can choose a value for a variable and can guess a whole Kripke structure. At some existential choice of a value in a fixed finite set $\mathbb{VALUES}$ for a variable x, the algorithm succeeds iff there is a value $v \in \mathbb{VALUES}$ such that the execution with $x = v$ will succeed. The algorithm succeeds iff there exists a successful execution. A non-deterministic algorithm decides a decision problem P iff for all input i the algorithm succeeds on the input i iff the output of i in the problem P is "yes". Note the asymmetry in the way of treating the "yes" and the "no" instances [Pap03][Section 2.7].

An algorithm is said to be *alternating* iff there are existential and universal choices and negating state during the execution of the algorithm. At some existential choice of a value in a fixed finite set $\mathbb{VALUES}$ for a variable x, the algorithm succeeds iff there is a value $v \in \mathbb{VALUES}$ such that the execution with $x = v$ will succeed. At some universal choice of a value in a fixed finite set $\mathbb{VALUES}$ for a variable x, the algorithm succeeds iff for all value $v \in \mathbb{VALUES}$ the execution with $x = v$ will succeed. At some negating state, the algorithm succeeds iff what remains to execute fails. An alternating algorithm decides a decision problem P iff for all input i the algorithm succeeds on the input i iff the output of i in the problem P is "yes".

Of course, all deterministic algorithms are non-deterministic and all non-deterministic algorithms are alternating. Several examples of algorithms are given in this thesis.

2.3.2 Complexity with time

We give direct definition of complexity class P, NP, AP, EXPTIME, NEXPTIME. For more details, see [Pap03].

A problem is in P iff there exists a deterministic algorithm running in polynomial time that can solve it. More precisely, there exists a polynomial P such that for any input i of size $|i|$ the algorithm runs in less than $P(|i|)$ steps.

A problem is in NP iff there exists a non-deterministic algorithm running in polynomial time that can solve it. More precisely, there exists a polynomial P such that for any input i of size $|i|$ all executions (due to existential choices) of the algorithm terminate in less than $P(|i|)$ steps.

A problem is in AP iff there exists a non-deterministic algorithm running in polynomial time that can solve it. More precisely, there exists a polynomial P such that for any input i of size $|i|$ all executions (due to existential and universal choices) of the algorithm terminate in less than $P(|i|)$ steps.

A problem is in EXPTIME iff there exists a deterministic algorithm running in exponential time that can solve it. In other words, there exists a polynomial P such that for any input i of size $|i|$ the algorithm runs in less than $2^{P(|i|)}$ steps.

A problem is in NEXPTIME iff there exists a non-deterministic algorithm running in exponential time that can solve it.

2.3.3 Complexity with space

A problem is in PSPACE iff there exists a deterministic algorithm requiring a polynomial amount of memory that solves it. More precisely, the algorithm decides the problem and there exists a polynomial P such that for any input i of size $|i|$ the execution of the algorithms requires less that $P(|i|)$ *bits* of memory.

A problem is in NPSPACE iff there exists a non-deterministic algorithm requiring a polynomial amount of memory that decides it. More precisely, there exists a polynomial P such that for any input i of size $|i|$ the algorithm requires less than $P(|i|)$ bits of memory. Even more precisely, there exists a polynomial P such that for any input i of size $|i|$, every execution (due to existential choices) requires less than $P(|i|)$ bits of memory. Notice that the definition of NPSPACE does not even require that the algorithm halts on all computations.

Actually, the two both notions are the same:

Theorem 1 *(Savitch's Theorem)[Pap03], [Sav70] PSPACE = NSPACE.*

The Savitch's Theorem says that for every problem such that there exists a non-deterministic algorithm using a polynomial amount of memory that can solve

it, there exists a deterministic algorithm using a polynomial amount[1] of memory that can solve it.

Actually, we have that [CKS81]:

Theorem 2 *PSPACE = NSPACE = AP.*

Note that the notion APSPACE also exists and we let the reader imagine the definition.

2.3.4 Hardness

Informally, a problem is NP-hard iff it encodes in itself the difficulty of all NP problems. More formally, a problem P is NP-hard iff for all problems Q in NP, there exists a translation tr:

- for all input i of Q, Q says "yes" to i iff P says "yes" to $tr(i)$;
- tr can be computed in polynomial time that is to say there exists an algorithm running in polynomial time with the following specification:
  - Input: i;
  - Output: $tr(i)$.

A problem is said to be NP-complete iff it is both NP-hard and in NP. Humans wonder whether "P = NP" or "P $\neq$ NP" [Coo06] and there is a possibility to get rich (1000000$!) if you get the good answer. In particular, if we prove that one satisfiability problem that are NP-complete presented in this thesis (as the satisfiability problem in Lineland without epistemic operators presented in Chapter 4 or the fragment of STIT presented in Chapter 8) is in P, we get rich.

More seriously, knowing whether P = NP or P $\neq$ NP has a tremendous consequences in real life. Suppose that P = NP. On the one hand, cryptographers would have to hurry up because their decision problems are often NP-complete hence in P so that their cryptosystems would be easily broken. On the other hand this may be a good news concerning many decision problems of logistics that are NP-complete thus in P. Unfortunately many computer scientists believes that P $\neq$ NP.

[1] may be a bit more!

2.4 Two standard problems

In order to prove that a problem belongs to a certain class of complexity, we can clearly exhibit an algorithm that can solve it. On the other hand, for proving hardness we need standard problems.

Theorem 3 *[Coo71] The satisfiability problem of the classical propositional logic (SAT) is NP-complete.*

Theorem 4 *[SM73] The satisfiability problem of the quantifier propositional logic (QSAT), that is to say, the logic whose language is defined by:*

$$\varphi \ ::= \ p \ \mid \ \varphi \wedge \varphi \ \mid \ \neg\varphi \ \mid \ \forall p\varphi$$

where p ranges over a countable infinite set ATM of atomic propositions is PSPACE-complete.

These two standard problems shall be used to prove hardness of the satisfiability problems in Lineland (Chapter 4) and Flatland (Chapter 5). The SAT problem will be also useful to prove hardness for the fragment of STIT in Chapter 8. The latter is also useful to prove complexity result for the satisfiability problem of the epistemic logic when the number of agents is greater that 2.

The Figure 2.3 sums up the relationship between the different complexity classes.

2.5 Reasoning in $\mathbf{S5}_n$

According to [JYH96], if the number of agents is equal to 1, the satisfiability problem is NP-complete. If the number of agents is greater that 2, the satisfiability problem is PSPACE-complete. Although it has been already proven in [JYH96], we are going to prove again that it is in PSPACE for many reasons. The first reason is to have a compact version of the algorithm (an alternating one) that can solve the satisfiability problem. The second reason is because we will adapt this algorithm in Chapter 10 for an extension of the epistemic logic.

Theorem 5 *The satisfiability problem of the epistemic logic is in PSPACE.*

Proof.

In this proof, we extend the notation $\mathcal{M}, w \models \varphi$ to sets of formulas: if Σ is a set of formulas, then $\mathcal{M}, w \models \Sigma$ stands for "for all formulas $\varphi \in \Sigma$, $\mathcal{M}, w \models \varphi$".

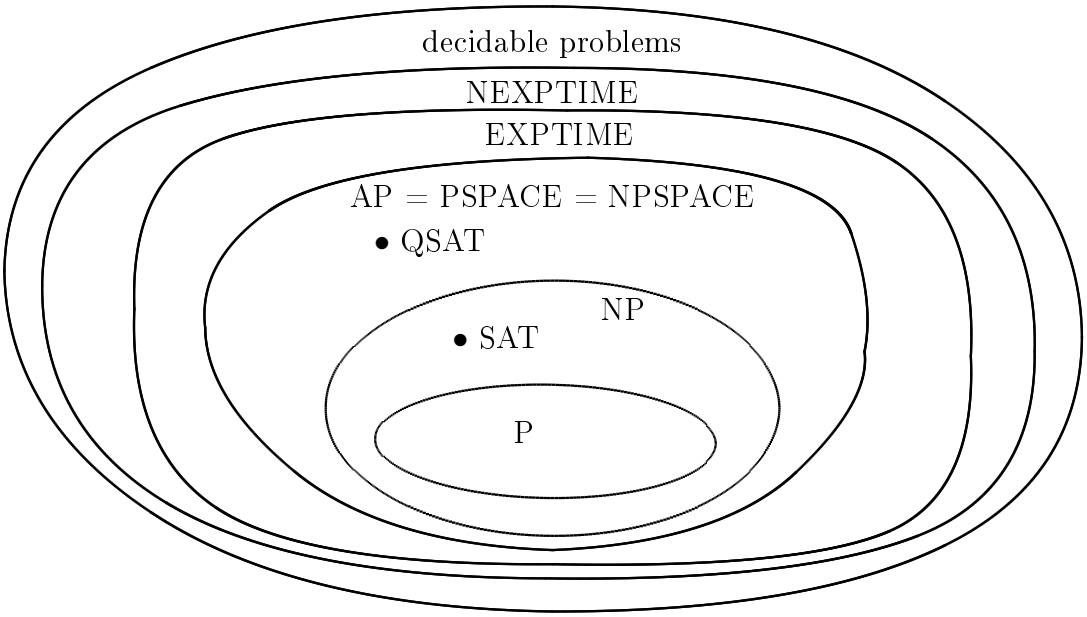


Figure 2.3: Complexity classes

Figure 2.4 shows an algorithm that can solve the satisfiability problem of $S5_2$ of a set of formulas Σ, that is to say the algorithm is supposed to succeed iff the set of formulas Σ is satisfiable. We give here an alternating procedure sat(..., 1). In the same way we can define the procedure sat(..., 2) by exchanging 1 and 2. For all Σ, the call sat(Σ, 1) terminates because at each sub-call there is at least one modal operator that disappears. It runs in polynomial time. We leave to the reader to check that if $n = 1$, the algorithm is non-deterministic so that it proves that the satisfiability problem of S5 is in NP. For all formulas φ, we define the set

$$CL(\varphi) = SF(\varphi) \cup \{\neg\psi \mid \psi \in SF(\varphi)\}.$$

$CL(\varphi)$ contains all the sub-formulas of φ and their negations. Let $l(\Sigma)$ be the number of operators in the formula of Σ that has the maximal number of operators. We prove by induction on $l(\Sigma)$ that iff all formulas of Σ begins with K_i or $\hat{K}_i$, then we have sat(Σ, i) succeeds iff Σ is satisfiable.

For all epistemic formulas we have equivalence between φ is satisfiable and $\hat{K}_1\varphi$ is satisfiable. In order to check the satisfiability of a single formula φ, we simply call the procedure sat($\{\hat{K}_1\varphi\}$, 1).

Basic case

If $l(\Sigma) = 1$, then for sure there is no recursive call. The correctness of the algorithm is proven like the inductive case.

function sat(Σ, 1)

$n :=$ the number of operator K_1 and $\hat{K}_1$ appearing in Σ

($\exists$) **choose** β a set of at most n subsets of $CL(\Sigma)$ such that there exists $S \in \beta$ such that $\Sigma \subseteq S$.

Check $K_1\psi$, $K_2\psi$, $\hat{K}_1\psi$ and Boolean coherence:

- for all $S \in \beta$, $\psi \in S$ xor $\neg\psi \in S$.
- for all $S, S' \in \beta$, $K_1\psi \in S$ iff $K_1\psi \in S'$;
- for all $S, S' \in \beta$, $\hat{K}_1\psi \in S$ iff $\hat{K}_1\psi \in S'$;
- for all $S \in \beta$, $K_1\psi \in S$ implies $\psi \in S$;
- for all $S \in \beta$, $K_2\psi \in S$ implies $\psi \in S$;
- for all $S \in \beta$, $\hat{K}_1\psi \in S$ iff there exists $S' \in \beta$ such that $\psi \in S'$;
- $\psi_1 \wedge \psi_2 \in S$ iff ($\psi_1 \in S$ and $\psi_2 \in S$);
- $\psi_1 \vee \psi_2 \in S$ iff ($\psi_1 \in S$ or $\psi_2 \in S$);

($\forall$) **choose** $S' \in \beta$

if there exists a formula of the form $\hat{K}_2\psi$ in S',

  call sat($\{K_2\theta \in S'\} \cup \{\hat{K}_2\theta \in S'\}$, 2)

Figure 2.4: Algorithm that can solve the satisfiability problem of $S5_n$ of a finite set of formulas Σ

Inductive case Let us consider a given set of formulas Σ such that all formulas of Σ begins with K_1 or $\hat{K}_1$.

($\Rightarrow$) We prove that sat(Σ, 1) succeeds implies that Σ is satisfiable.

If sat(Σ, 1) succeeds, we are to construct a "tree-like" model $\mathcal{M} = (W, R, V)$ such that there exists $w \in W$ such that $\mathcal{M}, w \models \Sigma$. sat($\Sigma$, 1) succeeds means that every call (possibility zero!) sat($\{K_2\theta \in S'\} \cup \{\hat{K}_2\theta \in S'\}$, 2) has succeeded. We have $l(\{K_2\theta \in S'\} \cup \{\hat{K}_2\theta \in S'\}) < l(\Sigma)$. So by induction for all $S' \in \beta$ there exist a model $\mathcal{M}^{S'} = (W^{S'}, R_1^{S'}, R_2^{S'}, V^{S'})$ and a world $w^{S'} \in W^{S'}$ such that $\mathcal{M}^{S'}, w^{S'} \models \{K_2\theta \in S'\} \cup \{\hat{K}_2\theta \in S'\}$.

The model $\mathcal{M}$ is obtained with the points S' of β and by gluing the pointed models $\mathcal{M}^{S'}, w^{S'}$ to S' for all S'. Broadly speaking, concerning the relations, β is a 1-equivalence class in $\mathcal{M}$. S' is in the 2-equivalence class of $w^{S'}$ inherited from $\mathcal{M}^{S'}$. For other points of one model $\mathcal{M}^{S'}$, relations are inherited

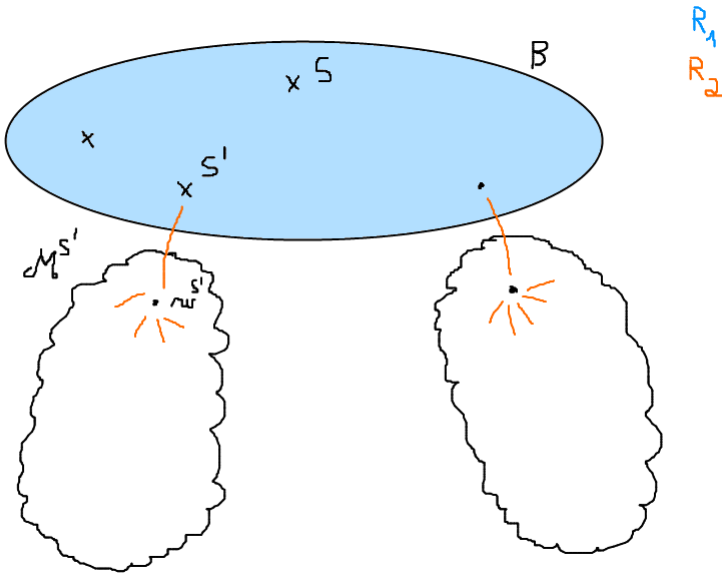


Figure 2.5: The model $\mathcal{M}$

directly those in $\mathcal{M}^{S'}$. Concerning the valuation, for all points coming from a model $\mathcal{M}^{S'}$, the valuation is inherited. For a point $S' \in \beta$, a proposition is true in S' iff it belongs to S'. Now let us explain the construction of $\mathcal{M} = (W, R, V)$ in more details:

- W is the union of β and all the worlds of all $\mathcal{M}^{S'}$ where S' ranges β;
- β is a 1-equivalence class. For all $S' \in \beta$, the 2-equivalence class of S' is the union of $\{S'\}$ and the 2-equivalence class of $w^{S'}$ in $\mathcal{M}^{S'}$. The 1-equivalence class of a point in a model $\mathcal{M}^{S'}$ is the same in $\mathcal{M}$ and $\mathcal{M}^{S'}$. The 2-equivalence class of a point not in the 2-equivalence class of a $w^{S'}$ in a model $\mathcal{M}^{S'}$ is the same in $\mathcal{M}$ and $\mathcal{M}^{S'}$.
- V is defined in the following way. For all propositions appearing in φ, for all worlds $S' \in \beta$, we have $p \in V(S')$ iff $p \in S$. Other propositions are false over β. For worlds from a model $\mathcal{M}^{S'}$, the valuation is inherited from the valuation of $\mathcal{M}^{S'}$.

The final result is a pointed model $(\mathcal{M}, S)$ satisfying for Σ. To prove it, we prove by induction on ψ that (Δ) for all $\psi \in CL(\varphi)$, for all $S' \in \beta$, $\mathcal{M}, S' \models \psi$ iff $\psi \in S'$.

(Propositions) It follows the Definition of V.

(Boolean cases) Left to the reader.

($K_1\psi$) The coherence test makes that true.

($K_2\psi$) Let $S' \in \beta$ be such that $K_2\psi \in S'$.

Lemma 1 *For all sub-formulas ϵ of $K_2\psi$, we have for all $v \in W^{S'}$, $\mathcal{M}, v \models \epsilon$ iff $\mathcal{M}^{S'}, v \models \epsilon$.*

PROOF.
Let us begin to prove the Lemma 1 by induction on ϵ.

(Propositions) ok.

(Boolean cases) ok.

($K_1\theta$) $\mathcal{M}, v \models K_1\theta$ iff for all $v' \in R_1(v)$, $\mathcal{M}, v' \models \theta$. By induction, it is equivalent to for all $v' \in R_1^{S'}(v)$, $\mathcal{M}^{S'}, v' \models \theta$. So it is equivalent to $\mathcal{M}^{S'}, v \models K_1\theta$.

($K_2\theta$) $\mathcal{M}, v \models K_2\theta$ iff for all $v' \in R_2(v)$, $\mathcal{M}, v' \models \theta$. It implies for all $v' \in R_2^{S'}(v)$, $\mathcal{M}, v' \models \theta$. By induction it implies for all $v' \in R_2^{S'}(v)$, $\mathcal{M}^{S'}, v' \models \theta$, that is to say $\mathcal{M}^{S'}, v \models K_2\theta$.

Reciprocally, suppose that $\mathcal{M}^{S'}, v \models K_2\theta$. We have $v' \in R_2^{\mathcal{M}^{S'}}(v)$, $\mathcal{M}^{S'}, v' \models \theta$. By induction (Lemma 1) it implies for all $v' \in R_2(v) \setminus \{S'\}$, $\mathcal{M}, v' \models \theta$. Moreover if $S' \in R_2(v)$ we also have to prove that $\mathcal{M}, S' \models \theta$. In this case, as $K_2\theta$ is a sub-formula of ϵ, we have either $K_2\theta \in S'$ or $K_2\theta \notin S'$. $K_2\theta \notin S'$ leads to $\mathcal{M}^{S'}, v \models \neg K_2\theta$, hence contradiction. So $K_2\theta \in S'$. It implies $\theta \in S'$. By induction (Δ) it means $\mathcal{M}, S' \models \theta$.

■

Now let us prove the equivalence $\mathcal{M}, S' \models K_2\psi$ iff $K_2\psi \in S'$.

$\boxed{\Leftarrow}$ Let $S' \in \beta$ be such that $K_2\psi \in S'$. Let us prove that $\mathcal{M}, S' \models K_2\psi$. As $K_2\psi \in S'$ we have that $\psi \in S'$ by test of coherence. So by induction (Δ) we have $\mathcal{M}, S' \models \psi$. Furthermore we have $\mathcal{M}^{S'}, w^{S'} \models K_2\psi$, that is to say for all $v \in R_2^{\mathcal{M}^{S'}}(w^{S'})$, $\mathcal{M}^{S'}, v \models \psi$. By Lemma 1, for all $v \in R_2^{\mathcal{M}^{S'}}(w^{S'})$ $\mathcal{M}, v \models \psi$. Finally, $\mathcal{M}, S' \models K_2\psi$.

$\boxed{\Rightarrow}$ Let $S' \in \beta$ be such that $K_2\psi \notin S'$, hence $\hat{K}_2\neg\psi \in S'$. We have $\mathcal{M}^{S'}, w^{S'} \models \hat{K}_2\neg\psi$. Thus there exists $v \in R_2(w^{S'})$ such that $\mathcal{M}^{S'}, v \models \neg\psi$ By Lemma 1 it is equivalent to $\mathcal{M}, v \models \neg\psi$. As $S'R_2w^{S'}$ and $w^{S'}R_2v$, we have $S'R_2v$. Hence $\mathcal{M}, S' \models \hat{K}_2\neg\psi$.

($\Leftarrow$) Σ is satisfiable implies that sat(Σ, 1) succeeds.

Suppose that Σ is satisfiable in a pointed model $\mathcal{M}, w$ where $\mathcal{M} = (W, R, V)$.

Consider the set of sub-formulas of Σ of the form $\hat{K}_1\psi$ true in w. For each such a sub-formula $\hat{K}_1\psi$ we consider the corresponding world in $u_\psi \in R_1(w)$ such that $\mathcal{M}, u_\psi \models \psi$.

Now we consider the execution of sat(φ, 1) such that β contains the set of sub-formulas true in w and the set of sub-formulas true in w_ψ for all ψ corresponding to a sub-formula $\hat{K}_1\psi$.

As $\mathcal{M}$ is a model and because we have created a set in β corresponding to u_ψ for all sub-formulas $\hat{K}_1\psi$ β is coherent.

Now let S' be a set in β such there exists $\hat{K}_2\psi \in S'$. S' corresponds either to w or a world u_ψ in the model $\mathcal{M}$. In any case, S' corresponds to a world in $\mathcal{M}$ which satisfies all formulas of S'. That is why we can find a world in $R_2(u)$ satisfying the formula $\{K_2\theta \in S'\} \cup \{\hat{K}_2\theta \in S'\}$. We have $l(\{K_2\theta \in S'\} \cup \{\hat{K}_2\theta \in S'\}) < l(\Sigma)$. By induction, the call sat($\{K_2\theta \in S'\} \cup \{\hat{K}_2\theta \in S'\}$, 2) succeeds. So sat($\varphi$, 1) succeeds.

■

2.6 The product logic $S5^n$

In this part, we are going to present the product logic $S5^n$. The reader is referred to [GKWZ03] for more details.

2.6.1 Syntax of $S5^n$

The language of $S5^n$ logic is built from a countably infinite set of atomic propositions ATM and modal symbols $\{1, ..., n\}$. The language $\mathcal{L}_{S5^n}$ of $S5^n$ is therefore defined by the following BNF:

$$\varphi \ ::= \ p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid \Box_i\varphi$$

where p ranges over ATM and i ranges over $\{1, ..., n\}$.

2.6.2 Semantics of $S5^n$

A Kripke model for the product logic $S5^n$ is a *Cartesian product*. More precisely:

Definition 5 ($S5^n$-model)
A $S5^n$-model is a tuple (X, R, V) where:

- $X = X_1 \times \cdots \times X_n$;

- R is a mapping associating to every $i \in \{1, ..., n\}$ the equivalence relation R_i defined by $R_i = \{\langle (x_1, \ldots, x_n), (y_1, \ldots, y_n)\rangle \in X^2 \mid \text{for all } j \neq i,\ x_j = y_j\}$;
- $V : ATM \to 2^X$.

The logic $S5^n$ and $\mathsf{S5}_n$ are different. The logic $\mathsf{S5}_n$ is the fusion of $\mathsf{S5}$, $\mathsf{S5}$ ... and $\mathsf{S5}$ (n times) and there are no *interaction* between the modal operators. On the contrary, in $S5^n$, models are cartesian product and in that sense there are interactions between the different modalities: for instance the formula $\Box_i\Box_j\varphi \leftrightarrow \Box_j\Box_i\varphi$ is valid for all i, j, φ.

2.6.3 Axiomatics for $S5^n$

Definition 6 (finitely axiomatizable)
[GKWZ03, Chapter 1] A logic L is finitely axiomatizable if there is a finite set Ax of formula schemes such that $\varphi \in L$ iff there is a sequence $(\varphi_1, \ldots, \varphi_k)$ of φ such that for $1 \leq i \leq k$, one of the following holds:

- either φ_i is a tautology of classical proposition logic or an instance of an axiom in Ax;
- either φ_i is obtained by necessitation from φ_j where $j < i$;
- or φ_i is obtained by *modus ponens* from φ_j and φ_k where $j, k < i$;
- $\varphi_k = \varphi$.

Theorem 6 *[GKWZ03, th. 8.2] If $n \geq 3$ then $S5^n$ is not finitely axiomatizable.*

Nevertheless $S5^n$ is axiomatizable if we weaken the definition of what an axiomatization is:

Theorem 7 *[Ven98] $S5^n$ is axiomatized by the following axiom schemas[2]:*

- *some axiom system of classical propositional logic;*
- *$S5(\Box_i)$;*
- *$\Box_i\Box_j\varphi \leftrightarrow \Box_j\Box_i\varphi$.*

and the following rules:

[2] We prefer give here a simpler axiomatics.

- *Modus Ponens rule:*
$$\frac{\vdash \varphi \qquad \vdash \varphi \to \psi}{\vdash \psi}$$

- *Necessitation rule:*
$$\frac{\vdash \varphi}{\vdash \Box_i \varphi}$$

- *Rectangle Rule:*
$$\frac{\vdash (p \wedge \tau(\neg\varphi \wedge p)) \to \varphi}{\vdash \varphi} \text{ where } p \text{ does not occur in } \varphi$$
where $\tau(\chi) = \Box_1 \dots \Box_n[(\bigwedge_{i\in\{1,\dots,n\}} \Diamond_1 \dots \Diamond_{i-1}\Diamond_{i+1} \dots \Diamond_n \chi) \to \chi]$.

Remark 2 *The rectangle rule is called* unorthodox rule *because the syntactic derivation requires an added constraint. Here: "where p does not occur in φ".*

2.6.4 Satisfiability problem for a $S5^n$-formula is undecidable

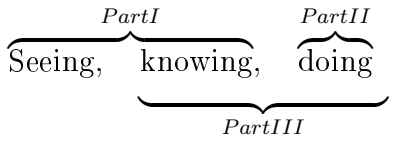
Theorem 8 *[GKWZ03, th. 8.6] If $n \geq 3$, the problem of satisfiability of a formula of $S5^n$ is undecidable.*

2.7 Contribution of this thesis

The contribution of this thesis is to give flavors to the standard epistemic modal logic introduced in Section 2.2 in two ways: first we will study the knowledge about what agents see, broadly speaking about perception of an agent in Part I. As the perception of an agent strongly deals with geometry, this part will first begin with a state of art about geometry and modal logic (Chapter 3). In this chapter we will see that it is difficult to give a meaning about a perception of an agent from what has already been studied in this area simply because there is no agent in those formalisms. We then propose a new version about epistemic modal logic and perception in a Lineland (Chapter 4), that is to say we suppose that the geometry is a line. We propose an algorithm for the model-checking and the satisfiability problem of the Lineland version of this epistemic modal logic. We also give a complete axiomatization. Of course, considering the world as a line is daring. That is why we propose another version of the epistemic modal logic where the version is Flatland [Abb84] (Chapter 5), that is to say, the world is a plane. We prove that the logic is decidable although we do not know the exact complexity. Finding an interesting axiomatization of the Flatland version of this logic is an open question. We also propose a weak version of Flatland epistemic logic.

In the Part II, we focus on the notion of action via the modal logic STIT ("see to it that"). This part begins with a state of art of existing logic of agency. In particular we see that the drawback of classical modal logic of agency like Coalition Logic CL [Pau02], Alternating-Time Logic ATL [AHK99] are not expressive enough to capture the notion of execution of actions. Indeed, they only capture the notion of capabilities of agents. In CL, ATL and so on, we can say that a group of agents has a strategy to ensure that a property φ is true but we cannot express that a group of agents actually performs a strategy to ensure a property φ. This kind of sentences can indeed be expressed in the logic STIT. In Chapter 6, we introduce the state of art about the logic STIT and we study the satisfiability problem and the axiomatizability of the group version of STIT. In order to have better results of decidability of the group version of STIT, we be interested in a weak fragment of group STIT in Chapter 8. Somehow, the Part II is a continuation of the thesis of [Tro07] containing the satisfiability problem of the individual version of STIT.

In the Part III, we are interested in applications of STIT: epistemic games in Chapter 9 and counterfactual emotions in Chapter 10. In this part, we give a "STIT-flavor" to the standard epistemic modal logic. In Chapter 9, we see how to represent an epistemic game via Kripke semantics and how to express Nash equilibrium and the algorithm of Iterated Deletion of Strictly Dominated Strategies in the language. In Chapter 10, we extend the standard epistemic modal logic with the fragment of STIT of Chapter 8. We then see how to represent counterfactual emotions in the language. More precisely, we see how to capture the notion of regret, rejoice, disappointment and elation. We also provide complexity results about the satisfiability problem both in Chapter 9 and 10.



Part I

Seeing, knowing

Chapter 3

Towards new "spatial" modal logics

In multi-agents applications, agents need to reason about what they see or not, and about what they know that other agents see or not. One may think of multi-players games for example, where the aim is to formalize that some agent, just by seeing where are her partners, *knows* that no enemy could sneak upon her from behind without being seen by the partners. We point out modal logics are often decidable (see Subsubsection 2.1.1.3) and that is why we would like the formalization to be in modal logic.

In order to formalize what a agent see, we need geometrical concepts. In other words, we are interested in spatial logics. But when we read the literature about temporal logics and spatial logics we may wonder why modal logics have been more famous in temporal logics than in spatial logics. In this chapter we try to answer to this question. In order to do this, we propose here a very modest state of art about temporal reasoning (Section 3.1) and then a modest state of art in formalization of geometrical concepts in logic (Section 3.2). Then we propose our crucial idea for providing a new modal logic for spatial reasoning (Section 3.3). Finally we compare our idea to existing works (Section 3.4).

3.1 Temporal logics

In Subsection 2.3.1, we saw two kinds of algorithms: those without choices (deterministic algorithms) and those with choices (non-deterministic ones and alternating ones). In the same manner, we can model the time in two ways. One can think of the time as linear or as branching.

3.1.1 Linear temporal logic

Linear temporal logic (LTL) [Pnu77] is a modal logic that expresses temporal concepts and considers the time as resolved (linear) and discret. The language of Linear temporal logic provides several modal constructions:

- $X\varphi$ means that φ holds in the "next" state;
- $F\varphi$ means that φ holds *eventually* (in the *future*);
- $G\varphi$ means that φ is *always* true;
- $\varphi U\psi$ means that φ holds until ψ is true.

The reader may be interested of other versions of "Linear temporal logic" (even if they are not called so in the literature) where there are only F and G operators [GKWZ03]. In what follows we give an overview for some of these logics:

- a modal logic where the discreteness is no more enforced: $S4.3$ is the modal logic of all models where the relation is a total order. It is also the logic where the domain is the set of real numbers $\mathbb{R}$ or the set of rational numbers $\mathbb{Q}$ and the relation is $\leq$;
- a modal logic where the discreteness is enforced: $S4.3 \oplus G(G(p \to Gp) \to p) \to (FGp \to Gp)$. Models (W, R, V) are such that W is the set of natural numbers and the relation R is $\leq$;
- a modal logic where the discreteness is no more enforced and the relation is strict: $K4.3$ is the modal logic of all models where the relation is a strict total order. It is NOT the logic of the class of models (W, R, V) where W is the set of real numbers or the set of rational numbers and the relation R is $<$;
- a modal logic where the discreteness is no more enforced but continuity is enforced: $K4.3 \oplus F\top \oplus GGp \to p$. Models (W, R, V) are such that W is the set of real numbers (or the set of rational numbers) and the relation R is $<$;
- a modal logic where the discreteness is enforced: $K4.3 \oplus F\top \oplus G(Gp \to p) \to (FGp \to Gp)$. Models (W, R, V) are such that W is the set of natural numbers and the relation R is $<$.

All those logics are different, that is to say they have different sets of satisfiable formulas. For all those logics, the satisfiability problem is NP-complete [Seg70], [Gol82], [ON80], [SC85].

3.1.2 Adding branching

Linear temporal logic can easily be extended to provide reasoning about branching time by adding two new modal operators:

- $E\varphi$: there is a branch in which φ holds;
- $A\varphi$: the formula φ holds in all branchs.

This logic is called CTL* (for "Computational Tree Logic Star") [EH86]. It embeds Linear Temporal Logic and it also embeds Computational Tree Logic (CTL) [CE82] [EH85] which a *syntactic fragment* where operators of branching and time are *fused*. Surprisingly CTL does not embed LTL.

3.1.3 Conclusion

LTL and CTL are modal logics expressive enough to capture different qualitative notions used especially in computer science:

- safety means that the system will never be in a "bad" state. For instance $AG\neg crash$ means that the system will never crash;
- liveness means that the system will eventually be in a good state whatever the branch. For instance $AF terminate$ may mean that the program will eventually terminate in all branchs;
- fairness means that a property will holds infinitely often. For instance $AGAF refresh$ may mean that the program will infinitely often refresh the screen.

Those notions are suitable to verify if a system matches with its specification. Furthermore you can note that LTL, CTL and CTL* have good complexity results both for model-checking and for the satisfiability problem.

3.2 Spatial reasoning

There are mainly two approaches to represent geometry: euclidean geometry and topology. Broadly speaking this section is an overview of the Chapters 4, 5, 6, 7 and 9 of the book [APHvB07].

3.2.1 Euclidean geometry

Euclidean geometry is the domain of geometry where we study the relations between points and lines in terms of orthogonality and parallelism.

3.2.1.1 Real number theory

A very powerful way to describe geometric facts consists in using coordinates of points. For instance to say that the point C is in the line (AB) you can write the property

$$\exists\lambda, ((x_C - x_A) = \lambda \times (x_B - x_A)) \wedge ((y_C - y_A) = \lambda \times (y_B - y_A))$$

where λ is a variable and $x_A, x_B, x_C, y_A, y_B, y_C$ are variables denoting the coordinates of the points A, B and C.

The lines (AB) and (AC) are *orthogonal* is represented by

$$(x_B - x_A) \times (x_C - x_A) + (y_B - y_A) \times (y_C - y_A) = 0.$$

Of course, this logic is *quantitative* and not only *qualitative*. For instance we can express that the distance between the points A and B is equal to 3 by

$$(x_A - x_B)^2 + (y_A - y_B)^2 = 3^2.$$

In this subsection we deal with the first order theory of real numbers. This theory was initially studied in [Tar51].

Here are other examples of formulas in this logic:

- $\forall x, x > 0 \rightarrow \exists y, x = y \times y$;
- $\forall x, \forall y, x < y \rightarrow \exists z, x < z \wedge z < y$.

Those formulas are interpreted over real numbers.

Syntax Let us introduce the syntax of the first order theory of real numbers. Let $\mathbb{VAR} = \{x, y, \ldots\}$ be a countable set of *variables*.

Definition 7 (expression)
An *expression* is defined by the following BNF:
$E ::= x \mid 0 \mid 1 \mid E + E \mid E \times E$
where $x \in \mathbb{VAR}$. We note $\mathbb{EXPR}$ the set of all expressions.

Definition 8 (language)
The *language* $\mathcal{L}_{\mathbb{R}}$ is defined by the following rule:
$\varphi ::= E = E \mid E > E \mid \top \mid \varphi \vee \varphi \mid \neg\varphi \mid \forall x\varphi$
where $x \in \mathbb{VAR}$ and $E \in \mathbb{EXPR}$.

Semantics Now we can give the semantics of those formulas. Broadly speaking, variables are interpreted as real numbers. Each symbol's ('+', '×', etc.) interpretation is natural. Nevertheless, we prefer to give it in details.

Definition 9 (interpretation)
An *interpretation* I is a map from V to $\mathbb{R}$.

Definition 10 (interpretation of expressions)
Given an interpretation $I : \mathbb{VAR} \to \mathbb{R}$, we extend I to a map $I^{expr} : \mathbb{EXPR} \to \mathbb{R}$ as follows:

- $I^{expr}(0) = 0$;
- $I^{expr}(1) = 1$;
- $I^{expr}(x) = I(x)$ for all $x \in \mathbb{VAR}$;
- $I^{expr}(E + \ E') = I^{expr}(E) + I^{expr}(E')$;
- $I^{expr}(E \times E') = I^{expr}(E) \times I^{expr}(E')$.

Definition 11 (interpretation of formulas)
Give an *interpretation* I and a formula $\varphi \in \mathcal{L}_{\mathbb{R}}$, we define $I \models \varphi$ by induction:

- $I \not\models \bot$;
- $I \models \varphi_1 \vee \varphi_2$ iff $I \models \varphi_1$ or $I \models \varphi_2$;
- $I \models \neg\varphi$ iff $I \not\models \varphi$;
- $I \models e = e'$ iff $I^{expr}(e) = I^{expr}(e')$;
- $I \models e > e'$ iff $I^{expr}(e) > I^{expr}(e')$;
- $I \models \forall x, \varphi$ iff for all $v \in \mathbb{R}$, $I[x := v] \models \varphi$.

where $I[x := v] : \mathbb{VAR} \to \mathbb{R}$ is defined as follows:

- $I[x := v](y) = I(y)$ for all $y \in \mathbb{VAR} \setminus \{x\}$;
- $I[x := v](x) = v$.

Decidability

Proposition 1 *[Wei93] [Tar51], [Eng83] The problem:*

- *input:* $\varphi \in \mathcal{L}_{\mathbb{R}}$*;*
- *output: yes if* φ *is satisfiable (i.e. there exists* I *such that* $I \models \varphi$*); no otherwise*

is decidable and in EXPSPACE.

An easy-to-understand algorithm is provided in [Eng83]. This algorithm relies basically on the idea of quantifier eliminations . Technical procedures are very close to the Theorem of Sturm ([Eng83], [Stu]). Given a polynomial $P(X)$, this theorem makes a bridge between:

- the number of distinct roots of P in the interval $]a, b[$. (Notice that the simple existence of such a root is expressed by a quantifier formula $\exists x, a < x \wedge x < b \wedge P(x) = 0$);
- $w(a) - w(b)$ where $w(a)$ is the number of changes of sign in $P(a), P_1(a), \ldots, P_r(a)$ and $w(b)$ is the number of changes of sign in $P(b), P_1(b), \ldots, P_r(b)$, where $P_1, \ldots, P_r$ are polynomials we can algorithmically compute from P. (That is to say we have deleted the quantification "$\exists x$")

The reader can also find an online version of the presentation of the algorithm presented like a game-book: `http://www.irit.fr/~Francois.Schwarzentruber/realqelim/index.html`.

The existential fragment We just point out a result in [Can88] and [Ren88].

Theorem 9 *The problem:*

- *input: a formula of the form* $\exists x_1, \exists x_2, \ldots, \exists x_n \varphi(x_1, \ldots, x_n)$ *where* $x_1, \ldots x_n$ *are variables and* $\varphi(x_1, \ldots, x_n)$ *is a boolean formulas where atomic predicates are of the form* $E_1 = E_2$ *or* $E_1 > E_2$ *where* E_1 *and* E_2 *are expressions over the variables* $x_1, \ldots x_n$*;*
- *output: yes iff the formula is true.*

is in PSPACE.

Implementations The theory of real numbers is successful in term of implementation. You can find a solver for it in `http://redlog.dolzmann.de/`.

3.2.1.2 Modal logic for euclidean spaces

Instead of using on the entire real number theory to model euclidean spaces, one may focus on a fragment of it: a modal logic [BG02]. The domain of a model is a set of points and lines of an euclidean space. The model is made up of four relations $\in$, $\ni$, $\|$, $\perp$ defined as follows:

- $x \in \Delta$ standing for "the point x belongs the line Δ";
- $\Delta \ni x$ standing for "the line Δ contains the point x";
- $\Delta \| \Delta'$ standing for "the line Δ is *parallel* to the line Δ'";
- $\Delta \perp \Delta'$ standing for "the line Δ is *orthogonal* to the line Δ'".

The syntax is pure modal logic. One distinguishs formulas for points and for lines and may have different modalities corresponding the relations described above. Here is an example of a formula that we can express in this logic:

$$A \wedge \langle\in\rangle(\Delta \wedge [\ni]q) \wedge [\in]\neg q$$

meaning "we consider a point in which A is true and this point belongs to a line on which Δ is true and q is true in all points of Δ. Furthermore, $\neg q$ is true in all lines passing by the point A."

Unfortunately the axiomatics is rather complicated: it requires an *unorthodox rule* (see Remark 2). The complexity of the satisfiability problem is high (NEXPTIME-complete) even if there is only $[\in]$ and $[||]$ in the language [BG02].

3.2.2 Topology

3.2.2.1 The mathematical notion of topology

One may have a pedagogical introduction in Topology and formal definitions in [GC97]. Topology has many applications. In mathematical analysis, it allows to define the notions of continuity of a function, limit of a function and therefore the notion of derivability and so on. It also allows to classify geometrical spaces: it does not care about distances between two points but it considers the general shape of the space. In topology, we describe a space given how the neighborhood of a point looks like. For me, topology should be called "the science of neighborhoods".

Formally:

Definition 12 (topological space)
Let E be a set. Let τ be a set of subsets of E. (E, τ) is called a *topological space* if:

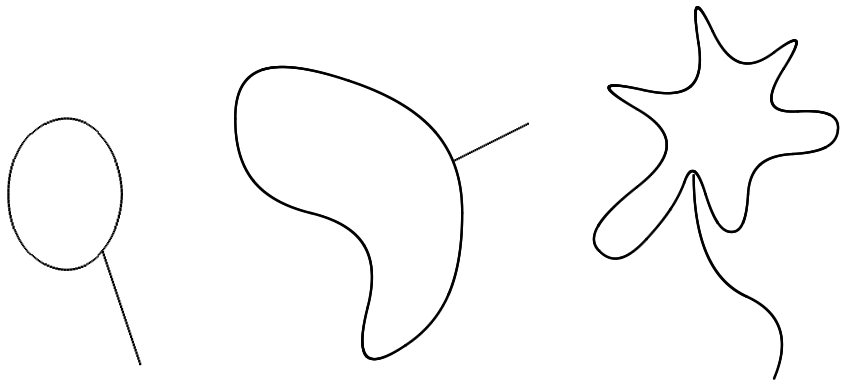


Figure 3.1: Three isomorphic topological spaces

- $\emptyset, E \in \tau$;
- for all set of indexes I, $(A_i)_{i \in I} \in \tau^I$, $\bigcup_{i \in I} A_i \in \tau$;
- for all finite set of indexes I, $(A_i)_{i \in I} \in \tau^I$, $\bigcap_{i \in I} A_i \in \tau$.

An element of τ is called an *open set* of E. Intuitively, given an open set $A \in \tau$, for each point $x \in A$, A must include a whole neighborhood of x. The set τ is often implicit and/or omitted and in that case we say that E is a topological space.

Example 3 *Let $\mathbb{R}$ be the set of all real numbers. The classical "topological space" $(\mathbb{R}, \tau)$ is made as follows: τ contains all the sets U that are unions of intervals of the form $]a, b[$ where $a < b$.*

With these settings, $A = [0, 1[$ is not an open set. Indeed, $0 \in A$ and for A to be an open it must contain a set of the form $]-\epsilon, \epsilon[$ for $\epsilon > 0$ and this is not the case.

We say that $f : E \to F$ is continuous iff for all opens O of F, the preimage $f^{-1}(O)$ is open.

As usual with "algebraic" structures we can introduce isomorphisms: $f : E \to F$ is an isomorphism iff f is bijective, f is continuous and f^{-1} is continuous. Two topological spaces E and F are isomorphic if and only if there exists an isomorphism from E to F. We are not going to give the formal definition of an isomorphism. Isomorphisms help us to classify the different topological spaces.

The Figure 3.1 shows three drawings of three isomorphic topological spaces. The Figure 3.2 shows you some topological spaces.

Isomorphisms has been wisely studied in the litterature. For instance here is a consequence of [L.E12] saying that the notion of topology captures the notion of *dimension*:

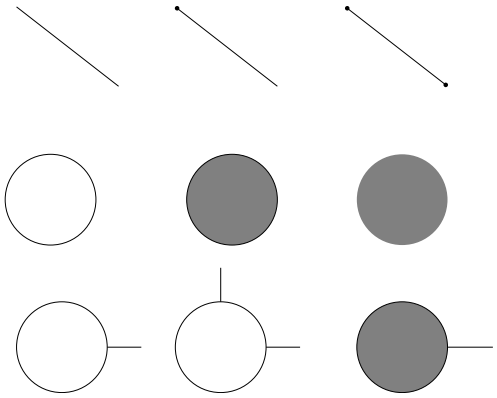


Figure 3.2: Some topological spaces

Theorem 10 *If $\mathbb{R}^n$ and $\mathbb{R}^m$ are isomorphic then $n = m$.*

In Chapter 4 and 5 we will define two different logics called Lineland (for $\mathbb{R}$) and Flatland (for $\mathbb{R}^2$). The two logics are different so the modal language we will introduce is expressive enough to capture the difference between one-dimensional and two-dimensional spaces.

3.2.2.2 Modal logic S4

This subsubsection is a summary of [vBB07]. Tarski [Tar38], [MT44] introduces a semantics in modal logic for topological spaces considered now as classical. Let ATM be a countable set of atomic propositions. Let us consider the standard modal language $\mathcal{L}$ given by the following rule:

$$\varphi \ ::= \ p \ \mid \ \bot \ \mid \ \neg\varphi \ \mid \ (\varphi \vee \varphi) \ \mid \ \Box\varphi$$

where $p \in ATM$.

Formulas are evaluated in a point of the space. The formula p means that the property represented by the atomic proposition p is true in the current point. The formula $\Box\varphi$ means that φ is true all around the current point. More precisely, there exists a neighborhood U containing the current point such that φ is true in all points of U.

More formally a model, called *topo-model* for this logic is a tuple $\mathcal{M} = (E, \tau, V)$ where:

- (E, τ) is a topological space;

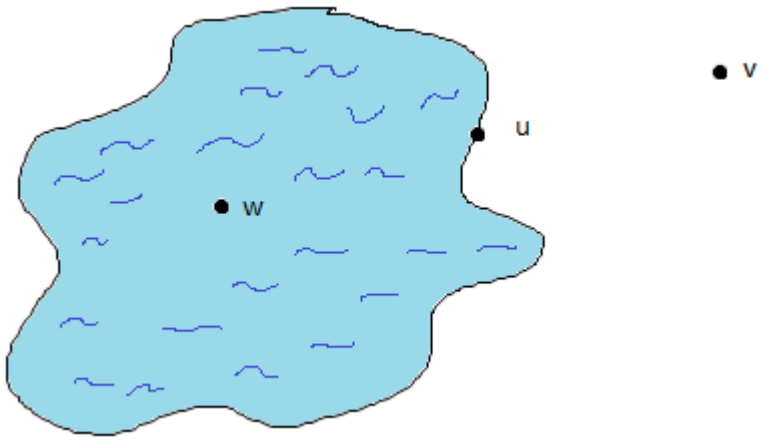


Figure 3.3: A topo-model $\mathcal{M}$

- V is a valuation that is to say a map from E to 2^{ATM}.

Definition 13 (truth conditions)
Truth of modal formulas is defined inductively as follows:

- $\mathcal{M}, x \models p$ iff $p \in V(x)$;
- $\mathcal{M}, x \models \Box\varphi$ iff there exists $U \in \tau$ such that $x \in U$ and for all $y \in U$, $\mathcal{M}, y \models \varphi$.

Example 4 *Let us consider the topo-model depicted by the Figure 3.3. The topological space is $\mathbb{R}^2$ and its usual topology. For all $x \in \mathbb{R}^2$, we have water $\in V(x)$ iff in x we have water.*
We have $\mathcal{M}, w \models \Box water$, $\mathcal{M}, u \models \neg\Box water \wedge \Diamond water$ and $\mathcal{M}, v \models \Box\neg water$.

We have the following results of axiomatization:

Theorem 11 *Let φ be a formula in the language $\mathcal{L}$. We have equivalence between:*

- *φ is provable in the system S4;*
- *φ is valid in a Kripke structure such that the relation is reflexive and transitive (Salqvist theorem, [BDRV02]);*
- *φ is valid over the class of topo-models [MT44];*

- *φ is valid over the class of topo-models $\mathcal{M} = (E, \tau, V)$ where (E, τ) is the Euclidian space $\mathbb{R}^n$, for any strictly positive integer n [MT44];*
- *φ is valid over the class of topo-models $\mathcal{M} = (E, \tau, V)$ where (E, τ) is the Cantor space [Min98].*

The proof of those results can be found in the reference written in the theorem or also in [vBB07]. One can notice that the *expressivity* of the language $\mathcal{L}$ is quite poor. In particular, it does not make the difference between the difference topological spaces $\mathbb{R}$, $\mathbb{R}^2$, $\mathbb{R}^3$, etc.

Theorem 12 *[BDRV02] The satisfiability problem of S4, that is to say, the following problem:*

- *input: a given formula φ in the language $\mathcal{L}$;*
- *output: yes iff there exists a topo-model $\mathcal{M} = (E, \tau, V)$ and a point $x \in E$ such that $\mathcal{M}, x \models \varphi$.*

is PSPACE-complete.

Recent works have increased the expressivity of S4 by the universal modality $[\forall]$ (in all worlds) or the modality $[\neq]$ (in all different worlds).

3.2.2.3 Qualitative relations: RCC − 8

RCC − 8 [RCC92] is a first order logic for spatial reasoning. Variables x, y, etc. are interpreted by region of a topological space. The logic is also made up of eight binary predicates in order to compare regions. We note $\mathbb{REL}_{\mathbf{RCC-8}}$ the set of those eight binary predicates. For instance if x and y are variables interpreting sets of points X and Y in a topological space, the predicate $EC \in \mathbb{REL}_{\mathbf{RCC-8}}$ (externally connected) such that the meaning of $EC(x, y)$ is that $X^\circ \cap Y^\circ = \emptyset$ and $X \cap Y \neq \emptyset$, that is to say X and Y are connected on their boundaries. The Figure 3.4 shows the interpretation of the eight relations of $\mathbb{REL}_{\mathbf{RCC-8}}$.

The syntax of the language of **RCC − 8** is defined by the following rule:

$$\varphi \ ::= \ R(x, y) \ \mid \ \neg\varphi \ \mid \ \varphi \wedge \varphi \ \mid \ \neg\varphi \ \mid \ \forall x \varphi$$

where x ranges over a set of variables and R over the set $\mathbb{REL}_{\mathbf{RCC-8}}$ of the eight relations of **RCC − 8**.

Generally speaking, the satisfiability problem in **RCC − 8** of a given first order formula is undecidable, more precisely not recursively enumerable. [LW06]

Nevertheless, it has been proved (with CSPSAT's formalism) that the following satisfiability problem:

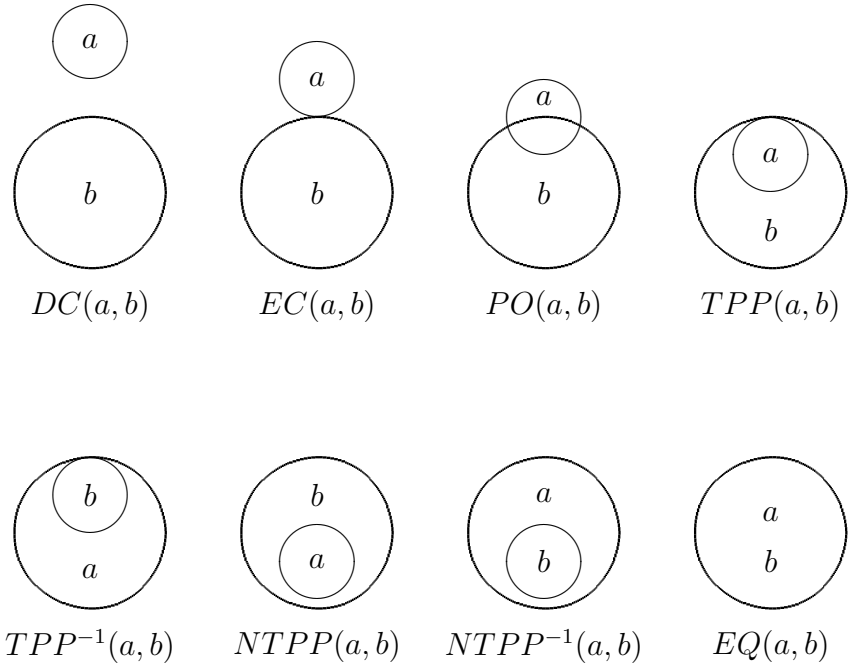


Figure 3.4: The eight $\mathbf{RCC-8}$-relations

- Input: a formula of the form $\varphi = \exists x_1, \exists x_2, \ldots \exists x_n, \bigwedge_{i,j \in \{1,\ldots n\}} \bigvee_{R \in C(i,j)} R(x_i, x_j)$ where n is a positive integer, $C(i,j)$ a subset of $\mathbb{REL}_{\mathbf{RCC-8}}$;
- Output: Yes iff the formula φ is consistent.

is NP-complete. [RN99].

3.3 Towards an epistemic spatial modal logic

Our approach is to create a modal logic where the syntax is the traditional epistemic modal logic whereas the semantics is spatial. In this section, we show why our approach is interesting.

3.3.1 Applications for spatial and epistemic reasoning

In robotics, agents are located in the world and may perceive the world with cameras. Those perception may infer some knowledge about the world. Hence we need a formal approach to model this knowledge.

In video games, for instance in platform games, artificial agents have artificial behavior. For instance an enemy e may attack the hero h if e knows that h is not

looking at e. This behavior is directly related to perception and knowledge.

3.3.2 In English: time is modal; spatial is not

In English, time is expressed easily with a conjugation or with the modal auxiliary "will" so that it is easy to translate some simple English sentences in temporal modal logic.

Example 5

It is cloudy.	$cloudy$
It will rain.	$Frain$
The weather was sunny.	$Psunny$ *where* P *is a past modal operator.*

We can also express in temporal modal logic some other subtleties like "could", "must", "would", "might" that can have a counterfactual meaning.

Example 6

It must rain tomorrow.	$AXrain$
It might be sunny tomorrow.	$EXsunny.$

On the contrary, spatial information generally needs adverb of place, etc. This linguistic argument supports that spatial reasoning is not adapted to be modeled in modal logic. If our aim is to have a spatial modal logic, it is preferable to build a modal logic using epistemic modalities and push the spatial reasoning aspect into the semantics.

3.3.3 Expressivity of temporal logic VS spatial logic

When we read the literature about temporal logics and spatial logics we may wonder why modal logics have been more famous in temporal logics than in spatial logics. We have seen in Section 3.1 that Linear Temporal Logic (LTL) [Pnu77] and Computational Tree Logic (CTL) [CE82] [EH85] are expressive enough for industrial needs: we can express safety (the system will never be in a "bad" state), liveness (the system will be in good state) and fairness (the property will holds infinitely often). Moreover the complexity of the satisfiability problem and the model-checking are often quite reasonable.

On the contrary we have also seen in Section 3.2 that spatial reasoning is not so successful because its high expressivity requires also high complexity for solving the satisfiability problem.

In our approach, the satisfiability problem is in many cases in PSPACE.

3.4 Comparisons between our approach and the literature

3.4.1 Classical epistemic logic VS Lineland/Flatland

In the classical epistemic modal logic $\mathsf{S5}_n$, wR_av stands for w and v are indistinguishable for the agent a. The epistemic logic $\mathsf{S5}_n$ has been combined with temporal modal operator [HV88]. In such temporal epistemic modal logic, the properties of *total recall* (*no forgetting*) and *no learning* are modeled as a constraint of the epistemic relation and the temporal relation. Kripke worlds are abstract: they are valuations.

On the contrary, in our approach, we would like to describe a situation directly by the graphical and natural representation of the system (position and direction of agents) and not with a Kripke structure. In fact, we also have *one* "canonical" Kripke structure made up of those graphical Kripke worlds that embed some geometrical informations: the position of agents and the direction where they look. (see Definition 14 and Definition 27) In the same way, the epistemic relation of an agent a is "built-in" and relies on geometry concepts: as depicted in Figure 3.5 two Kripke worlds are indistinguishable for agent a iff agent a sees the same thing in both worlds.

Other built-in logics in the literature There are other logics in the literature where the semantics is built-in, that is to say where Kripke's worlds are not abstract valuations and where the epistemic relations take into account the structure of those Kripke's worlds.

In [FHMV95] (Chapter 3), the author consider global states (you can think of them as possible worlds). A state is a tuple $(s_e, s_1, \ldots, s_n)$ where s_e is the state of the *environment* and s_i is the state of agent i for all agent i. Global states $s = (s_e, s_1, \ldots, s_n)$ and $s' = (s'_e, s'_1, \ldots, s'_n)$ are then said to be indistinguishable to agent i if agent i has the same state in both s and s', i.e., if $s_i = s'_i$. In other worlds, the epistemic relation is not arbitrary but directly built-in from the definition of states.

In [Jag09], the author develops a logic for rule-based agents. It is not the epistemic relation which is built-in but the belief change. A state s defines the belief of the agent: agent believes φ iff $\varphi \in V(s)$. Then the relation of belief change works as follows: T is a transition relation on states and sTu means the agent in state s can gain some belief and be in state t. This relation T is built-in with respect to a set of rules. In particular the relation T must satisfy the following statement: if a rule $\lambda_1 \ldots \lambda_n \to \lambda$ matches with the belief of the agent in state s then there must exist a state u such that sTu and $V(u) = V(s) \cup \{\lambda\}$.

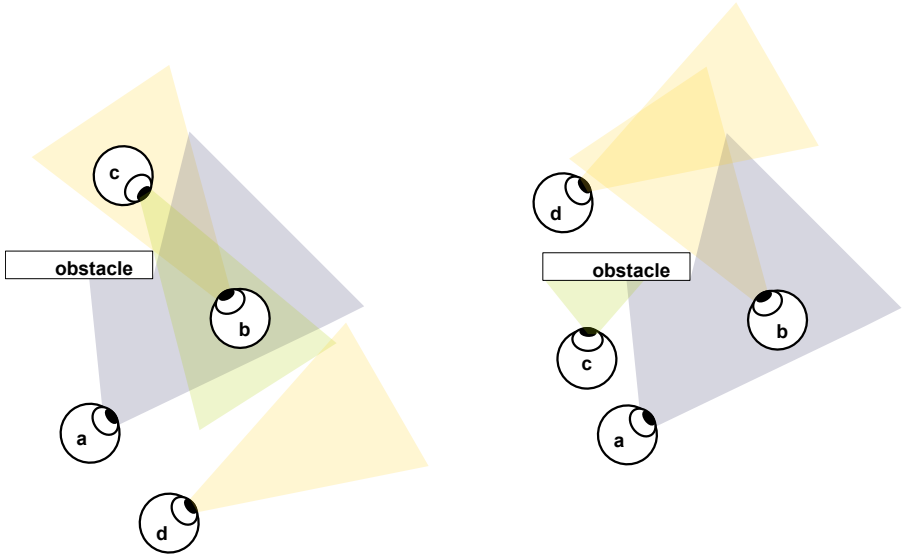


Figure 3.5: Two Kripke worlds that are undistisguishagle for agent a

In the same manner, in our approach, we provide a logic in which wR_au, that is to say, world u is possible for agent a in u, iff agent a sees the same thing both in w and u.

3.4.2 Spatial logic VS Lineland/Flatland

In the same way, in the next two chapters, we develop epistemic modal logic based on geometry. Our approach will be rather different concerning the syntax. In S4, $\mathbf{RCC-8}$, we have spatial operators like $\Box$ = "in the neighborhood", EC = "are externally connected", etc. whereas in the next two chapters the syntax relies on the standard epistemic modal logic and will provide the classical knowledge operator $K_a\varphi$ meaning "agent a knows that φ". This is motivated because we want to focus on epistemic reasoning.

Concerning the semantics, our approach is quite similar to the logic seen in this chapter. Indeed, in Chapter 4 and 5, we have also decided to encode the geometrical structure in the model.

Nevertheless, our approach is also different. In logics seen in this chapter, the domains in the semantics are geometrical entities:

- in S4 points x of a topo-model are points of the topological space;

- in the geometry seen in Subsection 3.2.1.2, the domain is the union of points and lines of the geometrical space;
- in $\mathbf{RCC-8}$, the domain is the set of regions of the space, etc.

Our approach rely on a Kripke model where a possible world is defined as the values of positions for all agents of the system. The epistemic relation will be defined directly from the possible worlds in term of what agents see.

3.4.3 Topological epistemic logic VS Lineland/Flatland

In the topological epistemic logic presented in [MP92], [PMS07] and [Hei06], the authors provide an epistemic modal logic based on the accuracy of the observation. Concerning the semantics, models are topo-models $\mathcal{M} = (E, \tau, V)$. The language made up of two operators $\Box$ (and its dual $\Diamond$) and K interpreted as follows, for all $x \in E$, $U \in \tau$ such that $x \in U$:

- $\mathcal{M}, (x, U) \models \Box\varphi$ iff $\mathcal{M}, (x, V) \models \varphi$ for all $V \in \tau$ such that $x \in V \subseteq U$;
- $\mathcal{M}, (x, U) \models \Diamond\varphi$ iff there exists $V \in \tau$ such that $x \in V \subseteq U$ and $\mathcal{M}, (x, V) \models \varphi$;
- $\mathcal{M}, (x, U) \models K\varphi$ iff $\mathcal{M}, (y, U) \models \varphi$ for all $y \in U$.

In (x, U), $x \in E$ represents the real world but the accuracy/precision of the observation is such that the agent only knows that the real world is in $U \in \tau$. The reading of the modal operators are:

- $\mathcal{M}, (x, U) \models \Diamond\varphi$: it is possible to have a better precision of the observation, that is to say, to have $V \subseteq U$ instead of U such that φ is true. In other worlds, the agent can make an *effort* to improve her precision such that φ is true.
- $\mathcal{M}, (x, U) \models K\varphi$: the agent knows that φ is true, that is to say, according to the current precision of the observation represented by U, φ is true in all possible worlds $y \in U$.

There are crucial differences between their approach and ours:

- For them, the geometry is used to represent the state of knowledge. For us, the geometry is devoted to give a position and a direction to agents. In this sense, their logic is abstract like $S5_n$.
- In their logic, there is only one agent.

Chapter 4 is devoted to the case where agents are points and are located on the line. In Chapter 5, agents are located on the plane.

Chapter 4

Knowledge in Lineland

4.1 Introduction

As we have seen in the previous Chapter, while many authors in Artificial Intelligence and Computer Science [FHMV95] developed *epistemic logic* and others have studied qualitative spatial reasoning [RN07] [CH01], fewer works concern their combination (but we can cite [PMS07] and [Hei06] which combine a spatial modal operator dealing with topology and an epistemic modal operator). For sure, one must then ask question about how knowledge is founded; in this Chapter, we choose to investigate the case where factual knowledge is based on what agents see. More precisely, we consider a framework where agents can see both other agents and where they are looking at. We do not provide operators in the language to deal with space but only an epistemic operator for each agent in the language.

Of course, our aim is to tackle concrete situations in the plane or in the space, but in this chapter we will focus on one dimension: agents are disposed along a line, looking right or left (see example of fig. 4.1). We will see that this simple case is already hard from the computational point of view (both model checking and satisfiability are PSPACE-complete). Interestingly, the obvious semantics induced by such situations can be axiomatized as shown in Section 4.4, thus providing a basis for a theory of knowledge about some qualitative geometry, which, we believe is the necessary condition for tackling the problems of model checking and the satisfiability problems in dimensions 2 and 3 and provide reasonable algorithms.

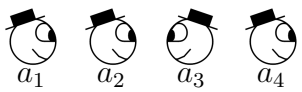


Figure 4.1: Example of a lineworld

This chapter is organized as follows: we present an epistemic language $\mathcal{L}_{PK}$ and its perception fragment $\mathcal{L}_P$ and their semantics in Section 4.2. Then we deal with the model-checking and satisfiability in Section 4.3. Finally we propose an axiomatization in Section 4.4. This chapter is an extension of [GS10] and [Sch09].

4.2 Lineland

4.2.1 Syntax

In this subsection, we introduce a language similar to the language of the standard language of epistemic logic $S5_n$ (see Subsection 2.2.1). Let AGT be a countable set of *agents* with typical members denoted a, $b, \ldots$. In this paper, the language $\mathcal{L}_{PK}$ of our epistemic theory is defined by the following rule:

$$\varphi \ ::= \ a \rhd b \ \mid \ \bot \ \mid \ \neg\varphi \ \mid \ (\varphi \vee \varphi) \ \mid \ K_a\varphi$$

where $a, b \in AGT$. The formula $a \rhd b$ is read "agent a sees agent b" and is called a perception literal. The formula $K_a\varphi$ is read "agent a knows that φ is true". As usual $\top =^{\text{def}} \neg\bot$, $(\varphi \wedge \psi) =^{\text{def}} \neg(\neg\varphi \vee \neg\psi)$, $\hat{K}_a\varphi =^{\text{def}} \neg K_a\neg\varphi$, $(\varphi \to \psi) =^{\text{def}} (\neg\varphi \vee \psi)$ and $(\varphi \leftrightarrow \psi) =^{\text{def}} ((\varphi \to \psi) \wedge (\psi \to \varphi))$. We follow the standard rules for omission of parentheses.

We will also be interested by the perception fragment $\mathcal{L}_P \subsetneq \mathcal{L}_{PK}$ without epistemic operators:

$$\varphi \ ::= \ a \rhd b \ \mid \ \bot \ \mid \ \neg\varphi \ \mid \ (\varphi \vee \varphi)$$

where $a, b \in AGT$. Formulas in $\mathcal{L}_P$ are called perception formulas.

4.2.2 Semantics

In this Subsection, we define *one* Kripke structure based on worlds and epistemic relations. Worlds are called here *lineworlds*. The geometry of Lineland is encoded directly inside such a world. A lineworld is the description of the arrangement of agents like in Figure 4.1. It is formally defined below in Definition 14. *Epistemic relations* between lineworlds rely on the perception of the agents (Definition 17).

Definition 14 (lineworld)
A *lineworld* w is a tuple $\langle <, \vec{dir} \rangle$ where:

- $<$ is a *strict total order* over AGT, that is to say:
  - $<$ is *irreflexive*: for all $a \in AGT$, $a \not< a$;

  - $<$ is *transitive*: for all $a, b, c \in AGT$, if $a < b$ and $b < c$ then $a < c$;
  - $<$ is *trichotomous*: for all $a, b \in AGT$, we have $a < b$, $b < a$ or $a = b$.

- $\vec{dir} : AGT \to \{\texttt{Left}, \texttt{Right}\}$.

The set of all lineworlds is noted W. Let us remark that if AGT is finite then the cardinality of W is equal to $card(AGT)! \times 2^{card(AGT)}$. Given a lineworld $w = \langle <, \vec{dir} \rangle$, the relation $<$ specifies how agents are ordered in the lineworld w from left to right. This relation is a *total* order because the shape of a world is a line. It is *strict* in order to prevent two agents to be at the same place (this is just a technical restriction). The mapping $\vec{dir}$ specifies whether an agent is looking left or right.

Example 7 *The Figure* 4.1 *represents the lineworld* $\langle <, \vec{dir} \rangle$ *defined by:*

- $a_1 < a_2 < a_3 < a_4$;
- $\vec{dir}(a_1) = \vec{dir}(a_2) = \vec{dir}(a_4) = \texttt{Right}$; $\vec{dir}(a_3) = \texttt{Left}$.

From the relation $<$ and the function $\vec{dir}$, we can define if an agent a sees another agent b.

Definition 15 (truth conditions)
We define $w \models \varphi$ by induction on φ:

- $w \models a \rhd b$ iff either ($\vec{dir}(a) = \texttt{Left}$ and $b < a$) or ($\vec{dir}(a) = \texttt{Right}$ and $a < b$);
- $w \not\models \bot$;
- $w \models \neg\varphi$ iff $w \not\models \varphi$;
- $w \models \varphi \vee \psi$ iff $w \models \varphi$ or $w \models \psi$.

The semantics of $w \models a \rhd b$ is intuitive: agent a sees agent b iff either b is on the left of a and a's direction is left or b is on the right of a and a's direction is right.

Example 8 *Let us reconsider the lineworld* w *depicted in Figure* 4.1. *We have* $w \models a_1 \rhd a_3$ *because* $\vec{dir}(a_1) = \texttt{Right}$ *and* $a_1 < a_3$. *Note that agents are* transparent*: here agent* a_2 *is transparent and agent* a_1 *sees beyond agent* a_2.

Now we define the notion of *mirror image* of a lineworld as depicted in Figure 4.2. It is useful for defining the epistemic relation.

Definition 16 (mirror image)
Let $w = \langle <, \vec{dir} \rangle, v = \langle <', \vec{dir}' \rangle \in W$. We write $w \approx v$ iff either $w = v$ or w is the *mirror image* of v, that is to say:

- for all $a, b \in AGT$, $a < b$ iff $b <' a$;
- and for all $a \in AGT$, $\vec{dir}(a) = \texttt{Left}$ iff $\vec{dir}'(a) = \texttt{Right}$.

Figure 4.2: A lineworld and its mirror image.

Let us note that $\approx$ is an equivalence relation on W such that each equivalence class is made up of exactly two lineworlds. Given $w \in W$ and $a \in AGT$, we note $V(a)_w = \{b \in AGT \mid w \models a \rhd b\}$. It is the set of all agents that agent a sees in the lineworld w. Now we define the epistemic relation R_a between worlds.

Definition 17 (epistemic relation)
Let $a \in AGT$. We define the *epistemic relation* R_a on the set of worlds W. For all $w = \langle <_w, \vec{dir}_w \rangle \in W$ and $v \in W$, we have wR_av iff there exists $u = \langle <_u, \vec{dir}_u \rangle$ such that:

- $u \approx v$;
- $V(a)_w = V(a)_u$;
- for all $b \in V(a)_w$, $\vec{dir}_w(b) = \vec{dir}_u(b)$;
- for all $b, c \in V(a)_w \cup \{a\}$, $b <_w c$ iff $b <_u c$.

Two worlds w and v are epistemically indistinguishable (wR_av) for agent a iff agent a sees exactly the same things in both worlds. Note that R_a is an equivalence relation on W. For all agents a and for all lineworlds w, $R_a(w)$ denotes the set of all lineworlds u such that wR_au. From now, the truth condition for $K_a\psi$ is standard: $K_a\psi$ is true iff ψ is true in all epistemically indistinguishable worlds for agent a.

Definition 18 (truth conditions)
We define $w \models K_a\psi$ iff for all $u \in R_a(w)$, $u \models \psi$.

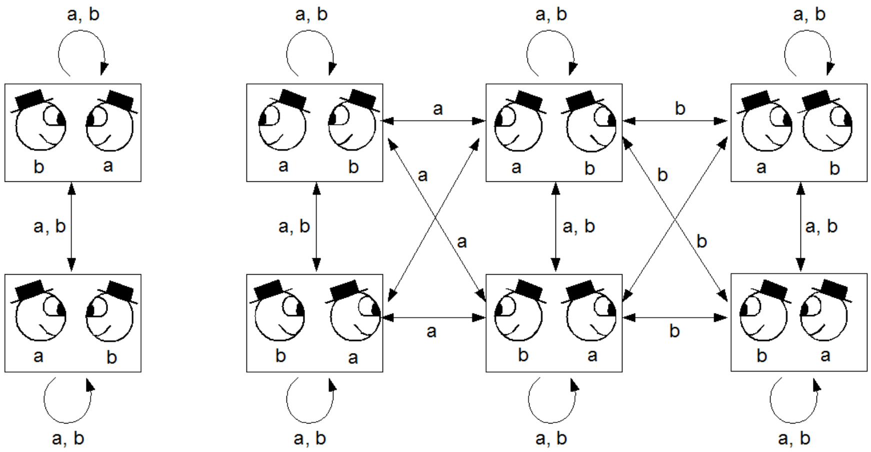


Figure 4.3: Kripke structure when $AGT = \{a, b\}$.

Example 9 *Consider the lineworld w depicted in Figure 4.1. We have $w \models K_{a_1} a_1 \rhd a_3$, $w \models \neg K_{a_2} a_1 \rhd a_3$ and $w \models K_{a_2} a_3 \rhd a_1$.*

The Figure 4.3 shows the Kripke structure when $AGT = \{a, b\}$. Nodes (rectangles) represent worlds, that is to say lineworlds where agents are settled in Lineland. Edges represent relations R_a and R_b.

4.2.3 Technical results

Now we give a characterisation of $<$ and $\vec{dir}$ in terms of truth conditions of perception formulas. We leave as an exercise to the reader to verify the following Proposition:

Proposition 2 *Let $w = \langle <, \vec{dir} \rangle \in W$. Let $a_0 \in AGT$. Suppose that $\vec{dir}(a_0) =$ `Right`. We have:*

- *For all $b \in AGT$, $b < a_0$ iff $b \neq a_0$ and $w \models \neg a_0 \rhd b$;*
- *For all $c \in AGT$, $a_0 < c$ iff $w \models a_0 \rhd c$;*
- *For all $b, c \in AGT$ such that $b \neq a_0$ and $c \neq a_0$,*

  $b < c$ iff $(b \neq c)$ and
  [either (1) $w \models \neg a_0 \rhd b, w \models \neg a_0 \rhd c$ and $w \models b \rhd a_0 \leftrightarrow b \rhd c$
  or (2) $w \models \neg a_0 \rhd b, w \models a_0 \rhd c$
  or (3) $w \models a_0 \rhd b, w \models a_0 \rhd c$ and $w \models b \rhd a_0 \leftrightarrow \neg b \rhd c$];

- *For all* $b \in AGT$,

$$\begin{aligned} \vec{dir}(b) &= \texttt{Right} \text{ if } w \models a_0 \rhd b \leftrightarrow \neg b \rhd a_0 \text{ or } b = a_0 \\ &= \texttt{Left} \text{ if } w \models a_0 \rhd b \leftrightarrow b \rhd a_0 \text{ and } b \neq a_0. \end{aligned}$$

The previous Proposition ensures that the order $<$ and the function $\vec{dir}$ are completely described by the truth conditions of the perception formulas involved in the Proposition. This leads to a characterization of $\approx$:

Theorem 13 *For all* $w, u \in W$, *we have equivalence between:*

1. $w \approx u$;
2. *For all* $\varphi \in \mathcal{L}_P$ *we have* $w \models \varphi$ *iff* $u \models \varphi$;
3. *For all* $\varphi \in \mathcal{L}_{PK}$ *we have* $w \models \varphi$ *iff* $u \models \varphi$.

Proof.
(i) $\Rightarrow$ (iii) By induction on φ.
(iii) $\Rightarrow$ (ii) Follows direclty from the fact that $\mathcal{L}_P \subseteq \mathcal{L}_{PK}$.
(ii) $\Rightarrow$ (i) Let us take a_0 such that $\vec{dir}_w(a_0) = \texttt{Right}$ (if such a a_0 does not exist simply take the mirror image of w instead of w). Now, if $\vec{dir}_u(a_0) = \texttt{Left}$, let v be the mirror image of u. Otherwise if $\vec{dir}_u(a_0) = \texttt{Right}$, let v be u.

Obviously, $\vec{dir}_v(a_0) = \texttt{Right}$. By applying Proposition 2 and (ii) we obtain $v = w$. Thus, $w \approx u$. ■

We shall say that a formula φ is *satisfiable* iff there exists a lineworld $w \in W$ such that $w \models \varphi$. A formula φ is said to be *valid* iff for all lineworlds $w \in W, w \models \varphi$.

Let $G \subseteq AGT$ such that G is finite. We introduce the notion of G-*describing conjunction*. Such a conjunction completely describes a situation concerning all agents in G.

Definition 19 (G-describing conjunction)
A G-describing conjunction is a maximal satisfiable conjunction of litterals of the form $b \rhd c$ or $\neg b \rhd c$ where $b, c \in G$.

Example 10 $\neg a \rhd a \wedge a \rhd b \wedge \neg b \rhd a \wedge \neg b \rhd b$ *is a* $\{a, b\}$*-describing conjunction.*

First, we prove that we can entirely describe a lineword with the truths of literals of the form $a \rhd b$, that is to say, the truth of an epistemic formula only depends on their truths.

Lemma 2 *Let φ be a formula of $\mathcal{L}_{PK}$. Let $G = \{b \in AGT \mid b \text{ occurs in } \varphi\}$. Let Φ be a G-describing conjunction. Let $w, w' \in W$ be such that $w \models \Phi$ and $w' \models \Phi$. We have $w \models \varphi$ iff $w' \models \varphi$.*

PROOF.

By induction on φ. Only the modal case is non-trivial that is to say when φ is of the form $\varphi = \hat{K}_a\psi$. Let $w, w' \in W$ be such that $w \models \Phi$ and $w' \models \Phi$. Suppose that $w \models \hat{K}_a\psi$ and let us prove that $w' \models \hat{K}_a\psi$. As $w \models \hat{K}_a\psi$, there exists $u = \langle <_u, \vec{dir}_u \rangle \in R_a(w)$ such that $u \models \psi$. Let $\zeta = \bigwedge_{a,b \in G | u \models a \rhd b} a \rhd b \wedge \bigwedge_{a,b \in G | u \not\models a \rhd b} \neg a \rhd b$. ζ is a G-describing conjunction such that $u \models \zeta$. Now we are going to prove that there exists $u' \in R_a(w')$ such that $u' \models \zeta$.

Let $V = \{a\} \cup \{b \in G \mid w' \models a \rhd b\}$. In other words, V contains the set of all agents that a sees plus a herself. Assume without loss of generality that $\vec{dir}_w(a) = \vec{dir}_{w'}(a) = \vec{dir}_u(a) = \texttt{Right}$. By applying Proposition 2 with $a_0 = a$, we obtain the following statements, referred below as (*):

- for all $b \in V$, $\vec{dir}_w(b) = \vec{dir}_{w'}(b)$;
- for all $b, c \in V$, $b <_w c$ iff $b <_{w'} c$.

Now, we define $u' = \langle <_{u'}, \vec{dir}_{u'} \rangle$ as follows. The relation $<_{u'}$ is defined by:

- the restriction on V of $<_{u'}$ is defined as the retriction on V of $<_{w'}$: for all $b, c \in V$, $b <_{u'} c$ iff $b <_{w'} c$;
- the restriction on $AGT \setminus V$ of $<_{u'}$ is defined as the retriction on $AGT \setminus V$ of $<_u$: for all $b, c \in AGT \setminus V$, $b <_{u'} c$ iff $b <_u c$;
- every agent in $AGT \setminus V$ is on the left of every agent in V: for all $b \in AGT \setminus V$ and for all $c \in V$, $b <_{u'} c$.

The relation $<_{u'}$ is a total order: it is clearly irreflexive, transitive and trichotomous. The function $\vec{dir}_{u'}$ is defined by:

- for all $b \in V$, $\vec{dir}_{u'}(b) = \vec{dir}_{w'}(b)$;
- for all $b \in AGT \setminus V$, $\vec{dir}_{u'}(b) = \vec{dir}_u(b)$.

In other words, all agents that agent a sees (in V) have the same position both in w' and u'. All agents that agent a does not see (in $AGT \setminus V$) have the same position both in u and u'.

By Definition of R_a we have $u' R_a w'$. Now let us prove that $u' \models \zeta$, that is to say we have to prove that for all $b, c \in G$ we have $u \models b \rhd c$ iff $u' \models b \rhd c$. As the semantics of $b \rhd c$ only depends on the positions of b and c, we simply check that:

1. for all $b, c \in G$, $b <_u c$ iff $b <_{u'} c$. Indeed:

   - If b, c are both in $G \setminus V$, then the result follows by Definition of $<_{u'}$.
   - If $b, c \in G \cap V$, $b <_{u'} c$ iff $b <_{w'} c$ iff $b <_w c$ (because of (*)) iff $b <_u c$ (because uR_aw)
   - If $b \in G \setminus V$ and $c \in G \cap V$, then by Definition of $<_{u'}$, we have always $b <_{u'} c$. We also have $b <_w a$, hence $b <_u a$ (by Definition of R_a) and $a <_w c$. Therefore, $a <_u c$. As $<_u$ is transitive, we have always $b <_u c$.
   - If $b \in G \cap V$ and $c \in G \setminus V$, then by Definition of $<_{u'}$, we do not have $b <_{u'} c$. And we do have $c <_u b$ so we do not have $b <_u c$.

2. for all $b \in G$, $\vec{dir}_u(b) = \vec{dir}_{u'}(b)$. Indeed:

   - If $b \in G \setminus V$, then by Definition of $\vec{dir}_{u'}$, we have $\vec{dir}_u(b) = \vec{dir}_{u'}(b)$.
   - If $b \in G \cap V$, then we have $\vec{dir}_{u'}(b) = \vec{dir}_{w'}(b) = \vec{dir}_w(b)(*) = \vec{dir}_u(b)$ (by Definition of R_a).

Finally, we have proved that $u' \models \zeta$. As $u' \models \zeta$, we have by induction $u' \models \psi$. Hence $w' \models \hat{K}_a\psi$. ■

4.2.4 Some valid formulas

As the relation R_a is an equivalence relation on W, all instances of axioms of $S5$ are valid. In particular $K_a\varphi \to \varphi$, $K_a\varphi \to K_aK_a\varphi$ and $\neg K_a\varphi \to K_a\neg K_a\varphi$.

Interestingly, the formulas below are valid too:

- $K_ab \rhd c \to a \rhd b$;
- If $b \neq c$, $a \rhd b \leftrightarrow K_ab \rhd c \vee K_a\neg b \rhd c$;
- $K_a(b \rhd c \vee d \rhd e) \to K_ab \rhd c \vee K_ad \rhd e$;
- $K_ab \rhd a \to K_aK_b \ldots K_aK_b(b \rhd a \wedge a \rhd b)$;
- $K_aK_bc \rhd d \wedge K_bK_ac \rhd d \to K_aK_bK_a \ldots K_bc \rhd d$.

4.3 Model checking and satisfiability

In this Section, we are interested in the model checking and the satisfiability problems.

Definition 20 (model checking in lineland)
We call *model checking in lineland* the following problem:

- Input: a formula φ, a lineworld w (where only agents occurring in φ are taken into account);
- Output: Yes if we have $w \models \varphi$. No, otherwise.

Definition 21 (satisfiability problem in lineland)
We call *satisfiability problem in lineland* the following problem:

- Input: a formula φ;
- Output: Yes if there exists a lineworld $w \in W$ such that $w \models \varphi$. No, otherwise.

4.3.1 Perception fragment

Remark 3 *As for the standard propositional logic, the model checking problem of a given formula from $\mathcal{L}_P$ in a given lineworld is easily proved to be in P.*

Let us now consider the satisfiability problem of a given formula from $\mathcal{L}_P$.

Theorem 14 *The satisfiability problem of a formula in the perception fragment $\mathcal{L}_P$ is NP-complete. If we restrict the language to a fixed finite number of agents then it is in P.*

PROOF.
In order to show that the satisfiability problem of a given formula from $\mathcal{L}_P$ is NP-hard, we shall reduce SAT to it. Let $p_1, p_2, \ldots$ be a non-repeating enumeration of a countable set of Boolean variables. Let $a_\infty \in AGT$ and $a_1, a_2, \ldots$ be a non-repeating enumeration of $AGT \setminus \{a_\infty\}$. For all Boolean formulas $\psi(p_1, \ldots p_n)$, let $\psi' = \psi(a_1 \rhd a_\infty, \ldots, a_n \rhd a_\infty)$ be a corresponding formula in $\mathcal{L}_P$. We claim that ψ is satisfiable iff ψ' is satisfiable.

$\boxed{\Leftarrow}$ Suppose that ψ' is satisfiable. Thus, there exists a lineworld w such that $w \models \psi'$. We simply extract the valuation ν from the lineworld w as follows: $\nu(p_i) = 1$ iff $w \models a_i \rhd a_\infty$. The reader is asked to show by induction on ψ that $\nu(\psi(p_1, \ldots, p_n)) = 1$ iff $w \models \psi(a_1 \rhd a_\infty, \ldots, a_n \rhd a_\infty)$.

$\boxed{\Rightarrow}$ Suppose that ψ is satisfiable. Hence, there exists a valuation ν such that $\nu(\psi) = 1$. We define a lineworld $w = \langle <, \vec{dir} \rangle$ from the valuation ν by:

- $a_1 < a_2 < \cdots < a_\infty$;
- For all positive integers i, $\vec{dir}(a_i) = \texttt{Right}$ iff $\nu(p_i) = 1$;

procedure $sat(\varphi(a_1, \ldots, a_n))$
  choose a strict total order $<$ on $\{a_1, \ldots, a_n\}$;
  choose a mapping $\vec{dir} : \{a_1, \ldots, a_n\} \to \{\texttt{Left}, \texttt{Right}\}$;
  if $(<, \vec{dir}) \models \varphi$ **then**
    accept
  else
    reject
  endIf
endProcedure

Figure 4.4: Algorithm to decide satisfiability in $\mathcal{L}_P$.

- $\vec{dir}(a_\infty) = \texttt{Right}$.

As the reader is asked to show by induction on ψ, $w \models \psi(a_1 \rhd a_\infty, \ldots, a_n \rhd a_\infty)$ iff $\nu(\psi(p_1, \ldots, p_n)) = 1$.

Moreover, the formula ψ' can be computed in logarithmic space. Hence, SAT is reducible in logarithmic space to the satisfiability problem in $\mathcal{L}_P$. Thus, the satisfiability problem in $\mathcal{L}_P$ is NP-hard. It is in NP since the procedure *sat* of Figure 4.4 provides a non-deterministic decision procedure solving it in polynomial time.

Of course, if we restrict the language to a fixed finite number of agents, then the procedure *sat* can be easily transformed into a deterministic procedure solving the satisfiability problem in $\mathcal{L}_P$ in polynomial time.

■

4.3.2 Perception and knowledge

Theorem 15 *The model checking of a formula in the epistemic language $\mathcal{L}_{PK}$ in a given lineworld is in PSPACE.*

Proof.

Since APTIME = PSPACE [CKS81], it suffices to prove that the model checking problem is in APTIME. The alternating procedures *istrue* and $isfalse$ of the Figure 4.5 take as input a lineworld w and a formula φ from $\mathcal{L}_{PK}$. The call $istrue(w, \varphi)$ stops with a reject iff $w \not\models \varphi$ and the call $isfalse(w, \varphi)$ stops with a reject iff $w \models \varphi$.

Their executions depend primarily on φ. Each case is either existential or universal. For exemple, for *istrue*, the case $\varphi_1 \vee \varphi_2$ is existential. It is an accepting case iff for some $i \in \{1, 2\}$, the case φ_i is accepting for *istrue*. Thus it corresponds to the fact that $\varphi_1 \vee \varphi_2$ is true at w iff for some $i \in \{1, 2\}$, φ_i is true at w. As

```
procedure istrue(w, φ)
    match (φ)
        ⊤: accept ;
        a ▷ b:
            if (a < b and dir⃗(a) = Right)
            or (b < a and dir⃗(a) = Left)
            then
                accept
            else
                reject
            endIf
        ψ1 ∨ ψ2:
            choose (∃)i ∈ {1, 2};
            call istrue(w, φi);
        ¬ψ: call isfalse(w, ψ);
        Kaψ:
            choose (∀)u ∈ W;
            if u ∈ Ra(w) then
                call istrue(u, ψ)
            else
                accept
            endIf
    endMatch
endProcedure
```

```
procedure isfalse(w, φ)
    match (φ)
        ⊤: reject ;
        a ▷ b:
            if (a < b and dir⃗(a) = Right)
            or (b < a and dir⃗(a) = Left)
            then
                reject
            else
                accept
            endIf
        ψ1 ∨ ψ2:
            choose (∀)i ∈ {1, 2};
            call isfalse(w, φi);
        ¬ψ: call istrue(w, ψ);
        Kaψ:
            choose (∃)u ∈ W;
            if u ∈ Ra(w) then
                call isfalse(u, ψ)
            else
                reject
            endIf
    endMatch
endProcedure
```

Figure 4.5: Algorithm for model checking.

well, for $isfalse$, the case $\varphi_1 \vee \varphi_2$ is universal. It is an accepting case iff for each $i \in \{1,2\}$, the case φ_i is accepting for $isfalse$. Thus it corresponds to the fact that $\varphi_1 \vee \varphi_2$ is false at w iff for each $i \in \{1,2\}$, φ_i is false at w.

In the call $istrue(w,\varphi)$ and the call $isfalse(w,\varphi)$, the input w is a lineworld where we only take into account agents occuring in φ. It is the same for the world $u \in W$ chosen in the cases $K_a\psi$. Remark that this non-deterministic choice can be done in linear time in the number of agents and that checking if $u \in R_a(w)$ can be done in quadratic time in the number of agents. Hence, this algorithm works in polynomial time. ■

Remark 4 *Satisfiability problem in lineland is also in PSPACE. Indeed, in order to check if a formula φ is satisfiable, we non-deterministically choose a lineworld w where we only take into account agents occuring in φ and then we call $istrue(w,\varphi)$.*

Theorem 16 *The model checking of a formula in the epistemic language $\mathcal{L}_{PK}$ in a given lineworld is PSPACE-hard. The satisfiability problem of a formula in the epistemic language $\mathcal{L}_{PK}$ is also PSPACE-hard.*

Proof.

The most fundamental complete decision problem for PSPACE is QSAT [Pap03]: given Boolean quantifiers $Q_1,\ldots,Q_n$, pairwise distinct Boolean variables $p_1,\ldots,p_n$ and a Boolean formula $\psi(p_1,\ldots p_n)$, determine whether $Q_np_n\ldots Q_1p_1\psi$ holds.

We shall reduce QSAT to the Lineland model checking problem. More precisely, given Boolean quantifiers $Q_1,\ldots,Q_n$, pairwise distinct Boolean variables $p_1,\ldots,p_n$ and a Boolean formula $\psi(p_1,\ldots,p_n)$, we shall construct a finite lineworld $w_n = (<_n, \vec{dir}_n)$ and a formula φ_n in $\mathcal{L}_{PK}$ such that $Q_np_n\ldots Q_1p_1\psi$ holds iff $w_n \models \varphi_n$. Suppose $a_1,b_1,a_2,b_2,\ldots$ is a non-repeating enumeration of AGT. Let $Q_np_n\ldots Q_1p_1\psi(p_1,\ldots,p_n)$ be an instance of QSAT.

First, we associate to $Q_np_n\ldots Q_1p_1\psi(p_1,\ldots p_n)$ a finite lineworld $w_n = (<_n, \vec{dir}_n)$ such that:

- $a_{n+1} <_n a_1, b_1, \ldots, a_n, b_n$;
- $\vec{dir}_n(a_{n+1}) = \texttt{Left}$.

Secondly, we associate to $Q_np_n\ldots Q_1p_1\psi(p_1,\ldots p_n)$ the formulas $\varphi_0,\varphi_1,\ldots,\varphi_n$ in $\mathcal{L}_{PK}$ as follows:

- $\varphi_0 = \psi(b_1 \rhd a_1,\ldots,b_n \rhd a_n)$;
- and for all positive integers i, if $i \leq n$ then if $Q_i = \forall$ then $\varphi_i = K_{a_{i+1}}(put_{a_i} \rightarrow \varphi_{i-1})$ else $\varphi_i = \hat{K}_{a_{i+1}}(put_{a_i} \wedge \varphi_{i-1})$

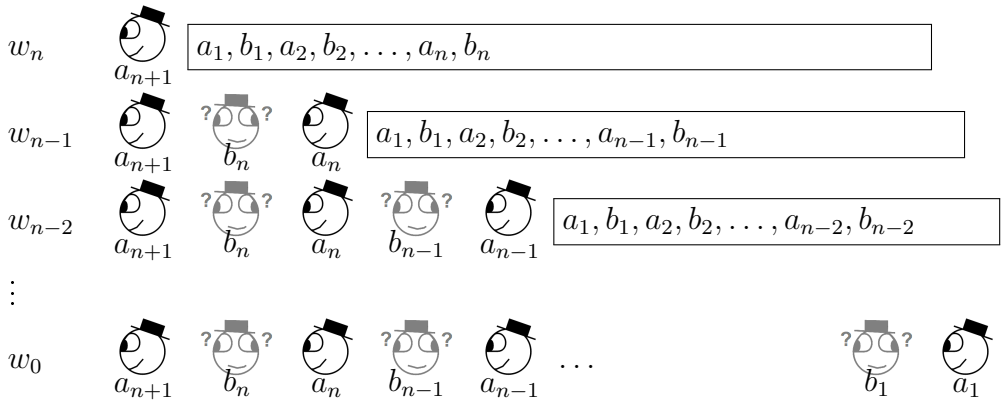


Figure 4.6: w_n will be step-by-step transformed into linewords of the form $w_{n-1}, w_{n-2}, \ldots w_0$.

Figure 4.7: Agent a_1 alone.

where for all positive integers i, if $i \leq n$ then $put_{a_i} = \bigwedge_{j=1}^{i-1} \neg a_i \rhd a_j \wedge \bigwedge_{j=1}^{i-1} \neg a_i \rhd b_j \wedge a_i \rhd a_{i+1} \wedge a_i \rhd b_i$.

For all positive integers i, if $i \leq n$ then the guard $K_{a_{i+1}}(put_{a_i} \rightarrow \ldots)$ corresponds to the Boolean quantifier $Q_i = \forall$ and the guard $\hat{K}_{a_{i+1}}(put_{a_i} \wedge \ldots)$ corresponds to the Boolean quantifier $Q_i = \exists$. Successively interpretating these guards, the reader may easily verify that w_n will be step-by-step transformed into linewords of the form $w_{n-1}, w_{n-2}, \ldots, w_0$ described in Figure 4.6.

During the process leading to w_0, put_{a_i} means "the relative positions and the directions of $a_{n+1}, a_n, b_n, a_{n-1}, b_{n-1}, \ldots a_i, b_i$ are fixed whereas the relative positions and the directions of $a_{i-1}, b_{i-1}, \ldots a_1, b_1$ are still to be chosen". Obviously, the lineworld w_n and the formula φ_n in $\mathcal{L}_{PK}$ can be computed in logarithmic space. We claim that $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots, p_n)$ holds iff $w_n \models \varphi_n$. To prove this claim, we proceed by induction on the nonnegative integer n.

Basis Suppose $n = 0$. Hence ψ is equivalent either to $\bot$ or $\top$. Moreover, w_n is the finite lineworld described in Figure 4.7.

Finally φ_n is nothing but ψ. As the reader is asked to show, φ_n holds iff $w_n \models \varphi_n$.

Hypothesis

Let n be a nonnegative integer such that for all Boolean quantifiers $Q_1, \ldots Q_n$, for all pairwise distinct Boolean variables $p_1, \ldots p_n$ and for all Boolean formulas

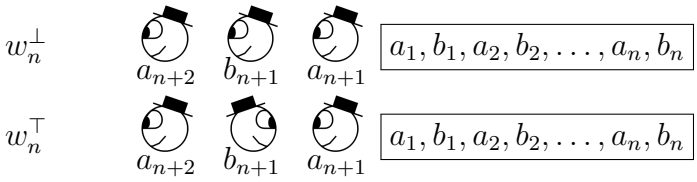


Figure 4.8: The worlds $w_n^{\perp}$ and $w_n^{\top}$.

$\psi(p_1, \ldots, p_n)$, $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n)$ holds iff the corresponding lineworld w_n and the corresponding formula φ_n in $\mathcal{L}_{PK}$ are such that $w_n \models \varphi_n$.

Step Let $Q_1, Q_2, \ldots Q_n, Q_{n+1}$ be Boolean quantifiers $p_1, p_2, \ldots p_n, p_{n+1}$ be pairwise distinct Boolean variables and $\psi(p_1, p_2, \ldots, p_n, p_{n+1})$ be a Boolean formula. Let w_{n+1} be the corresponding lineworld and φ_{n+1} be the corresponding formula in $\mathcal{L}_{PK}$. We consider two cases: $Q_{n+1} = \forall$ and $Q_{n+1} = \exists$. The case $Q_{n+1} = \exists$ is similar to the case $Q_{n+1} = \forall$. For this reason we only give the proof for $Q_{n+1} = \forall$.

In the case $Q_{n+1} = \forall$, $Q_{n+1} p_{n+1} Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, p_{n+1})$ holds iff both $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, \perp)$ and $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, \top)$ holds. Let $w_n^{\perp}$ and $w_n^{\top}$ be the lineworlds described in Figure 4.8.

Let $\varphi_n^{\perp}$ and $\varphi_n^{\top}$ be the formulas in $\mathcal{L}_{PK}$ corresponding respectively to $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, \perp)$ and $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, \top)$. By induction hypothetis, $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, \perp)$ holds iff $w_n^{\perp} \models \varphi_n^{\perp}$ and $Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, \top)$ holds iff $w_n^{\top} \models \varphi_n^{\top}$. Hence $Q_{n+1} p_{n+1} Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, p_{n+1})$ holds iff $w_n^{\perp} \models \varphi_n^{\perp}$ and $w_n^{\top} \models \varphi_n^{\top}$. Now, obviously, $w_{n+1} \models \varphi_{n+1}$ iff $w_n^{\perp} \models \varphi_n^{\perp}$ and $w_n^{\top} \models \varphi_n^{\top}$. Finally, $Q_{n+1} p_{n+1} Q_n p_n \ldots Q_1 p_1 \psi(p_1, \ldots p_n, p_{n+1})$ holds iff $w_{n+1} \models \varphi_{n+1}$.

This terminates the proof that the Lineland model checking problem is PSPACE-hard. To demonstrate that the satisfiability problem in Lineland is PSPACE-hard too, it suffices to prove that the Lineland model checking problem is reducible to the satisfiability problem in Lineland. Let w be a finite lineworld and φ be a formula in $\mathcal{L}_{PK}$. We define $G = \{b \in AGT \mid b \text{ occurs in } \varphi\}$. Let Φ be the formula $\bigwedge_{a,b \in G, w \models a \rhd b} a \rhd b \wedge \bigwedge_{a,b \in G, w \not\models a \rhd b} \neg a \rhd b$. This formula Φ is a G-maximal conjunction. We have $w \models \varphi$ iff the formula $\varphi \wedge \Phi$ is satisfiable. Indeed, from left to right it follows directly from the fact that $w \models \Phi$. Reciprocally, if $\varphi \wedge \Phi$ is satisfiable, there exists a world u such that $u \models \varphi \wedge \Phi$. But then, as $w \models \Phi$, the Lemma 2 gives that $w \models \varphi$.

■

Finally:

Corollary 1 *The model checking and the satisfiability problems in the epistemic language $\mathcal{L}_{PK}$ are PSPACE-complete.*

The PSPACE-hardness of the model checking problem in the language $\mathcal{L}_{PK}$ is related to the fact that *one* lineworld implicitly defines an exponential number of possible lineworlds.

4.4 Axiomatization

4.4.1 Perception fragment

The following axiomatics describes the geometry of Lineland that agents perceive.

Definition 22 (theory $\mathcal{P}$)
We define $\mathcal{P}$ as the smallest set of formulas of $\mathcal{L}_P$ closed by modus ponens and containing all Boolean tautologies and also the following formulas as proper axioms:

(Ax_1) $\neg a \rhd a$;

(Ax_2) $a \rhd b, c \wedge (c \rhd a \leftrightarrow c \rhd b) \rightarrow (\neg b \rhd a \leftrightarrow b \rhd c)$;

(Ax_3) $\neg a \rhd b \wedge \neg a \rhd c \wedge (c \rhd a \leftrightarrow \neg c \rhd b) \rightarrow (b \rhd c \leftrightarrow b \rhd a)$;

(Ax_4) $\neg a \rhd b \wedge a \rhd c \rightarrow (b \rhd a \leftrightarrow b \rhd c)$;

(Ax_5) $\neg a \rhd b \wedge \neg a \rhd c \wedge \neg a \rhd d \wedge (b \rhd a \leftrightarrow b \rhd c) \wedge (c \rhd a \leftrightarrow c \rhd d) \rightarrow (b \rhd a \leftrightarrow b \rhd d)$;

(Ax_6) $a \rhd b \wedge a \rhd c \wedge a \rhd d \wedge (b \rhd a \leftrightarrow \neg b \rhd c) \wedge (c \rhd a \leftrightarrow \neg c \rhd d) \rightarrow (b \rhd a \leftrightarrow \neg b \rhd d)$

for all $a, b, c, d \in AGT$.

The axiom Ax_1 means that an agent never sees herself. There are no mirrors in Lineland and the axiom Ax_2 means that if b is between a and c, then b sees either a or c. The axioms Ax_3 and Ax_4 are true because agents are transparent. The axioms Ax_5 and Ax_6 means that the ordering on the line is transitive.

Theorem 17 *The axiomatics is sound: each formula of $\mathcal{P}$ is valid.*

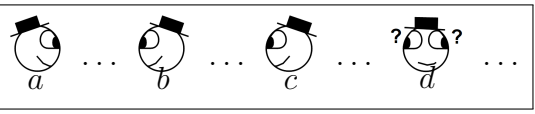
Proof.
[sketch] The soundness is only verification. We only give here the proof that Ax_6 $a \rhd b \wedge a \rhd c \wedge a \rhd d \wedge (b \rhd a \leftrightarrow \neg b \rhd c) \wedge (c \rhd a \leftrightarrow \neg c \rhd d) \rightarrow (b \rhd a \leftrightarrow \neg b \rhd d)$ is valid.

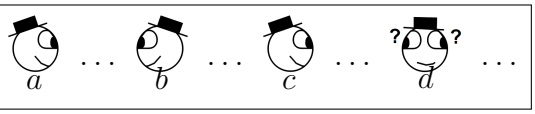
Let $w \in W$ be such that (*) $w \models a \rhd b \wedge a \rhd c \wedge a \rhd d \wedge (b \rhd a \leftrightarrow \neg b \rhd c) \wedge (c \rhd a \leftrightarrow \neg c \rhd d)$. Suppose that without loss of generality agent a is looking right. As a sees b, c, and d, the world w looks like  ....

Now we consider two cases on the truth of $b \rhd a$ in w:

- $b \rhd a$ is true in w: (*) implies $w \models \neg b \rhd c$: w looks like 

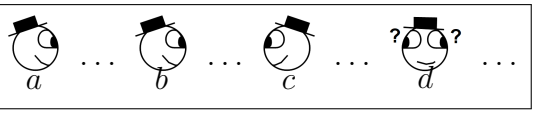
  Now we consider two cases on the truth of $c \rhd a$:

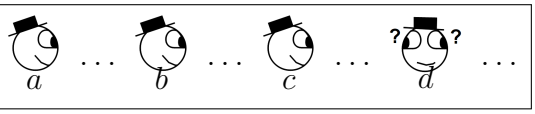
  - $c \rhd a$ is true in w. Therefore, $w \models \neg c \rhd d$: $w =$ 

  - $\neg c \rhd a$ and $c \rhd d$: $w =$ 

- $\neg b \rhd a$ and $b \rhd c$: $w =$ 

  Case on $c \rhd a$:

  - $c \rhd a$ is true and $\neg c \rhd d$: $w =$ 

  - $\neg c \rhd a$ and $c \rhd d$: $w =$ 

In all framed images, we can check that $w \models (b \rhd a \leftrightarrow \neg b \rhd d)$. ■

As usual, a set Γ of formulas is $\mathcal{P}$-*consistent* iff there is no finite subset $\{\varphi_1, \ldots, \varphi_n\} \subseteq \Gamma$ such that $\varphi_1 \wedge \cdots \wedge \varphi_n \rightarrow \bot \in \mathcal{P}$. Such a $\mathcal{P}$-consistent set Γ is called *maximal* iff there is no $\mathcal{P}$-consistent set Γ' such that $\Gamma \subsetneq \Gamma'$. We suppose the reader to be familiar with the Lindenbaum's lemma and properties of maximal consistent set. For details, see [BDRV02].

Now, we define the canonical lineworld of a maximal consistent set. Given a maximal consistent set Γ and an agent a_0, the lineworld $w_\Gamma^{a_0}$ denotes a lineworld where a_0's direction is right and where all formulas in Γ are true. Note that the following Definition looks like the condition of Proposition 2.

Definition 23 (canonical model)
Let $a_0 \in AGT$. Let Γ be a maximal $\mathcal{P}$-consistent set. We define the lineworld $w_\Gamma^{a_0} = \langle <, \vec{dir} \rangle$, called *canonical lineworld of* Γ, by:

- For all $b \in AGT$, $b < a_0$ iff $b \neq a_0$ and $\neg a_0 \rhd b \in \Gamma$;
- For all $c \in AGT$, $a_0 < c$ iff $a_0 \rhd c \in \Gamma$;
- For all $b, c \in AGT$ such that $b \neq a_0$ and $c \neq a_0$,

$b < c$ iff $(b \neq c)$ and
[either (1) $\neg a_0 \rhd b, \neg a_0 \rhd c \in \Gamma$ and $b \rhd a_0 \leftrightarrow b \rhd c \in \Gamma$
or (2) $\neg a_0 \rhd b, a_0 \rhd c \in \Gamma$
or (3) $a_0 \rhd b, a_0 \rhd c \in \Gamma$ and $b \rhd a_0 \leftrightarrow \neg b \rhd c \in \Gamma$];

- For all $b \in AGT$,

$\vec{dir}(b) =$ `Right` if $a_0 \rhd b \leftrightarrow \neg b \rhd a_0 \in \Gamma$ or $b = a_0$;
$=$ `Left` if $a_0 \rhd b \leftrightarrow b \rhd a_0 \in \Gamma$ and $b \neq a_0$.

Now we just have to check that the canonical lineworld is a lineworld in the sense of Definition 14: $<$ is a total order and $\vec{dir}$ is well-defined. This will be ensured by the proper axioms of the theory $\mathcal{P}$.

Proposition 3 *For all $a_0 \in AGT$, for all Γ maximal $\mathcal{P}$-consistent set, $w_\Gamma^{a_0} \in W$.*

PROOF.
Given $w_\Gamma^{a_0} = \langle <, \vec{dir} \rangle$, we check that $<$ is a strict total order.

$<$ is strict. By Definition, it is obvious.

$<$ is trichotomous. Let $b, c \in AGT$ be such that $b \neq c$. Let us consider the simple case: $b = a_0$ or $c = a_0$. Without loss of generality, suppose $b = a_0$. If $a_0 \rhd c \in \Gamma$ then $a_0 < c$. If $\neg a_0 \rhd c \in \Gamma$ then $c < a_0$.

Now we are treating the general case where $b \neq a_0$, $c \neq a_0$ and $b \neq c$. Let us prove it by contradiction. Suppose we have $b \not< c$ and $c \not< b$.

Now let us consider the four different following cases (depending on whether $a_0 \rhd b \in \Gamma$ or not and whether $a_0 \rhd c \in \Gamma$ or not):

- First case: $a_0 \rhd b, \neg a_0 \rhd c \in \Gamma$. The condition (2) in the Definition of $c < b$ (see Definition 23) is true. So we have $c < b$ hence contradiction.
- Second case: $\neg a_0 \rhd b, a_0 \rhd c \in \Gamma$. In the same way, the condition (2) in the Definition of $b < c$ is true. So we have $b < c$ hence contradiction.
- Third case: $a_0 \rhd b, a_0 \rhd c \in \Gamma$.
  As $b \not< c$ and $c \not< b$, the condition (3) of the definition of $b < c$ and the condition (3) of the definition of $c < b$ are false. So we have: $b \rhd a_0 \leftrightarrow b \rhd c \in \Gamma$ (*) and $c \rhd a_0 \leftrightarrow c \rhd b \in \Gamma$. As $a_0 \rhd b \wedge a_0 \rhd c \wedge (c \rhd a_0 \leftrightarrow c \rhd b) \rightarrow (\neg b \rhd a_0 \leftrightarrow b \rhd c) \in \Gamma$ (Ax_2), modus ponens gives $(\neg b \rhd a_0 \leftrightarrow b \rhd c) \in \Gamma$. This contradicts (*).

– Fourth case: $\neg a_0 \rhd b, \neg a_0 \rhd c \in \Gamma$. As $b \not< c$ and $c \not< b$, the condition (1) of the definition of $b < c$ and the condition (1) of the definition of $c < b$ are false. So we have: $b \rhd a_0 \leftrightarrow \neg b \rhd c \in \Gamma$ (*) and $c \rhd a_0 \leftrightarrow \neg c \rhd b \in \Gamma$ $\neg a_0 \rhd b \wedge \neg a_0 \rhd c \wedge (c \rhd a_0 \leftrightarrow \neg c \rhd b) \rightarrow (b \rhd c \leftrightarrow b \rhd a_0) \in \Gamma$ (Ax_3). So $(b \rhd c \leftrightarrow b \rhd a_0) \in \Gamma$. This contradicts (*).

$<$ is transitive. Suppose that $b < c$ and $c < d$ and let us prove that $b < d$. The proofs when $b = a_0$ or $c = a_0$ are left to the reader. We only consider here the complex case when $b \neq a_0$ and $c \neq a_0$. By Definition 23, $b < c$ implies that:

– either (1) $\neg a_0 \rhd b, \neg a_0 \rhd c \in \Gamma$ and $b \rhd a_0 \leftrightarrow b \rhd c \in \Gamma$;

– or (2) $\neg a_0 \rhd b, a_0 \rhd c \in \Gamma$;

– or (3) $a_0 \rhd b, a_0 \rhd c \in \Gamma$ and $b \rhd a_0 \leftrightarrow \neg b \rhd c \in \Gamma$.

In the same way, $c < d$ implies that:

– either (1') $\neg a_0 \rhd c, \neg a_0 \rhd d \in \Gamma$ and $c \rhd a_0 \leftrightarrow c \rhd d \in \Gamma$;

– or (2') $\neg a_0 \rhd c, a_0 \rhd d \in \Gamma$;

– or (3') $a_0 \rhd c, a_0 \rhd d \in \Gamma$ and $c \rhd a_0 \leftrightarrow \neg c \rhd d \in \Gamma$.

Hence, we have to consider the following 9 cases.

– (1) (1')We have $\neg a_0 \rhd b, \neg a_0 \rhd c, \neg a_0 \rhd d \in \Gamma$. We have $b \rhd a_0 \leftrightarrow b \rhd c$ and $c \rhd a_0 \leftrightarrow c \rhd d$. It suffices to prove that $b \rhd a_0 \leftrightarrow b \rhd d \in \Gamma$. This follows from the axiom of transitivity (Ax_5) $\neg a_0 \rhd b \wedge \neg a_0 \rhd c \wedge \neg a_0 \rhd d \wedge (b \rhd a_0 \leftrightarrow b \rhd c) \wedge (c \rhd a_0 \leftrightarrow c \rhd d) \rightarrow (b \rhd a_0 \leftrightarrow b \rhd d) \in \Gamma$.

– (1) (2') It gives directly (2) for $b < d$.

– (1) (3') We have $\neg a_0 \rhd b, \neg a_0 \rhd c, a_0 \rhd c, a_0 \rhd b \in \Gamma$ and this is simply impossible because Γ is consistent.

– (2) (1') We have $a_0 \rhd c, \neg a_0 \rhd c \in \Gamma$ and this is impossible.

– (2) (2') We have $a_0 \rhd c, \neg a_0 \rhd c \in \Gamma$ and this is impossible.

– (2) (3') We have $\neg a_0 \rhd b, a_0 \rhd c, a_0 \rhd d$ and $c \rhd a_0 \leftrightarrow \neg c \rhd d$. Sure, we have $\neg a_0 \rhd b$ and $a_0 \rhd d$. So we have the point (2) of the Definition of $b < d$.

– (3) (1') $a_0 \rhd c, \neg a_0 \rhd c \in \Gamma$ and this is impossible.

– (3) (2') $a_0 \rhd c, \neg a_0 \rhd c \in \Gamma$ and this is impossible.

– (3) (3) This case looks like the case (1) (1) except that we use here axiom (Ax_6).

If all the possible cases, we have $b < d$.

■

Now we prove that the formulas true in $w_\Gamma^{a_0}$ are exactly the formulas in Γ.

Lemma 3 (truth lemma) *For all $a_0 \in AGT$, for all Γ maximal $\mathcal{P}$-consistent set, for all $\varphi \in \mathcal{L}_P$, $\varphi \in \Gamma$ iff $w_\Gamma^{a_0} \models \varphi$.*

PROOF.
By induction on φ.

$\boxed{a_0 \rhd c}$ We prove that $a_0 \rhd c \in \Gamma$ iff $w_\Gamma \models a_0 \rhd c$.

- $\boxed{\Rightarrow}$ If $a_0 \rhd c \in \Gamma$, then by definition of $<$, we have $a_0 < c$. Moreover $\vec{dir}(a_0) = \texttt{Right}$. Hence $w_\Gamma \models a_0 \rhd c$.
- $\boxed{\Leftarrow}$ Reciprocally, if $w_\Gamma \models a_0 \rhd c$, it implies $a_0 < c$ because a_0 is looking right. Hence, by definition of $<$, $a_0 \rhd c \in \Gamma$.

$\boxed{b \rhd a_0}$ We prove that $b \rhd a_0 \in \Gamma$ iff $w_\Gamma \models b \rhd a_0$.

- $\boxed{\Rightarrow}$ Suppose that $b \rhd a_0 \in \Gamma$.
  - First case : $a_0 \rhd b \in \Gamma$.
    By definition of $<$, we have $a_0 < b$. By definition of $\vec{dir}(b)$, as $a_0 \rhd b \leftrightarrow b \rhd a_0 \in \Gamma$, we have $\vec{dir}(b) = \texttt{Left}$. Hence, by Definition of the truth condition we have $w_\Gamma \models b \rhd a_0$.
  - Second case : $\neg a_0 \rhd b \in \Gamma$. This case is similar. By definition of $<$, we have $b < a_0$. By definition of $\vec{dir}(b)$, $\vec{dir}(b) = \texttt{Right}$. So $w_\Gamma \models b \rhd a_0$.
- $\boxed{\Leftarrow}$ And reciprocally, suppose that $w_\Gamma \models b \rhd a_0$.
  - Case $b < a_0$: we must have $\vec{dir}(b) = \texttt{Right}$. So by Definition of $\vec{dir}$, we have $a_0 \rhd b \leftrightarrow \neg b \rhd a_0 \in \Gamma$. By Definition of $b < a_0$, we have $\neg a_0 \rhd b \in \Gamma$. Hence $b \rhd a_0 \in \Gamma$.
  - Case $a_0 < b$: Similar to the previous case.

$\boxed{b \rhd c}$ Let us prove the truth lemma for the case $b \rhd c$ where $b, c \neq a_0$.

- $\boxed{\Rightarrow}$ Suppose $b \rhd c \in \Gamma$ and let us prove that $w_\Gamma \models b \rhd c$.
  We have to consider two cases: either $b < c$ or $c < b$. Let us study the case $b < c$.

* Suppose (2). Thus $\neg a_0 \rhd b, a_0 \rhd c \in \Gamma$, i.e. by Definition of $<$: $b < a_0 < c$. Let us prove that $\vec{dir}(b) = \texttt{Right}$, i.e. we have to prove that $a_0 \rhd b \leftrightarrow \neg b \rhd a_0 \in \Gamma$. As we have $\neg a_0 \rhd b \in \Gamma$ we have to prove that $b \rhd a_0 \in \Gamma$.
  We have $a \rhd c \wedge \neg a \rhd b \rightarrow (b \rhd a \leftrightarrow b \rhd c) \in \Gamma$ (Ax_4) and $b \rhd c \in \Gamma$ (hypothesis) so $b \rhd a_0 \in \Gamma$.
* Suppose (3). Thus $a_0 \rhd b, a_0 \rhd c \in \Gamma$, i.e. by Definition of $<$: $a_0 < b < c$. Let us prove that $\vec{dir}(b) = \texttt{Right}$, i.e. we have to prove that $a_0 \rhd b \leftrightarrow \neg b \rhd a_0 \in \Gamma$. But $a_0 \rhd b \in \Gamma$. So we have to prove $\neg b \rhd a_0 \in \Gamma$. But by (3), we have $\neg b \rhd a_0 \leftrightarrow b \rhd c \in \Gamma$ and $b \rhd c \in \Gamma$.
* Suppose (1) $(b < c < a_0)$. Let us prove that $\vec{dir}(b) = \texttt{Right}$. We have to prove that $a_0 \rhd b \leftrightarrow \neg b \rhd a_0 \in \Gamma$. But $\neg a_0 \rhd b \in \Gamma$. So we have to prove $b \rhd a_0 \in \Gamma$. But by (1), we have $b \rhd a_0 \leftrightarrow b \rhd c \in \Gamma$ and $b \rhd c \in \Gamma$!

$\boxed{\Leftarrow}$ Reciprocally, suppose that $w_\Gamma \models b \rhd c$ and let us prove that $b \rhd c \in \Gamma$.
We have to consider the following two cases:

* The case $\vec{dir}(b) = \texttt{Right}$ and $b < c$: we have $a_0 \rhd b \leftrightarrow \neg b \rhd a_0 \in \Gamma$. We have to consider again two cases:
  1. $\neg a_0 \rhd b \in \Gamma$: hence $b \rhd a_0 \in \Gamma$ because $\vec{dir}(b) = \texttt{Right}$.
     We have to consider again two cases:
     * $\neg a_0 \rhd c \in \Gamma$. Then by Definition of $b < c$, we have $b \rhd a_0 \leftrightarrow b \rhd c \in \Gamma$, so we have $b \rhd c \in \Gamma$;
     * $a_0 \rhd c \in \Gamma$: But $a_0 \rhd c \wedge \neg a_0 \rhd b \rightarrow (b \rhd a_0 \leftrightarrow b \rhd c) \in \Gamma$ (Ax_4) we have $b \rhd c \in \Gamma$.
  2. $a_0 \rhd b \in \Gamma$: left to the reader.
* The case $\vec{dir}(b) = \texttt{Left}$ and $c < b$ is similar and left to the reader.

(Boolean cases) They are left to the reader.

■

Corollary 2 *Valid formulas of $\mathcal{L}_P$ are exactly formulas in $\mathcal{P}$.*

In the previous Definition 23, we have based the construction of the canonical lineworld to a particular agent a_0. In fact, the canonical model does not depend on the choice of $a_0 \in AGT$. More precisely:

Proposition 4 *For all a_0, $b_0 \in AGT$, for all Γ maximal $\mathcal{P}$-consistent sets, we have $w_\Gamma^{a_0} \approx w_\Gamma^{b_0}$.*

PROOF.
By Proposition 13 and Lemma 3. ■

The previous Proposition will be useful in next Subsection for the axiomatization with knowledge operators.

4.4.2 Perception and knowledge

The axiomatics for the perception and epistemic modal logic lies on the notion of G-what-a-perceives-mc. It corresponds to a conjunction which specifies exactly the factual information agent a knows about agents in G.

Definition 24 (G-what-a-perceives-mc)
Let $G \subseteq AGT$ be such that G is finite and non-empty. Let $a \in G$. We say that φ is a *G-what-a-perceives-mc* iff φ is a conjunction of litterals such that there exists a subset $V \subseteq G$ such that:

- for all $b \in V$, $a \rhd b$ appears in φ;
- for all $b \in G \setminus V$, $\neg a \rhd b$ appears in φ;
- for all $b \in V$, for all $c \in G$, either $b \rhd c$ or $\neg b \rhd c$ appears in φ;
- φ is satisfiable.

In the previous Definition, the set V represents the set of agents seen by agent a. The first and second items correspond to the information about agents that agent a sees and does not see. The third item corresponds to what agents visible to a see. The fourth item implies that $a \notin V$. Now we give an axiomatization describing the interaction between knowledge and perception.

Definition 25 (theory $\mathcal{PK}$)
We define $\mathcal{PK}$ as the smallest set of formulas of $\mathcal{L}_{PK}$ closed by modus ponens and necessitation rules and containing all Boolean tautologies, all proper axioms of $\mathcal{P}$ and also the following formulas as proper axioms:

(Ax_K) $K_a(\varphi \to \psi) \to (K_a\varphi \to K_a\psi)$;

(Ax_7) $a \rhd b \to K_a a \rhd b$;

(Ax_8) $\neg a \rhd b \to K_a \neg a \rhd b$;

(Ax_9) $a \rhd b \wedge b \rhd c \to K_a b \rhd c$;

(Ax_{10}) $a \rhd b \wedge \neg b \rhd c \to K_a \neg b \rhd c$;

(Ax_{11}) $\varphi \rightarrow \hat{K}_a\Phi$ where φ is G-what-a-perceives-mc and Φ is any G-describing conjunction containing φ.

The axiom Ax_7 says that an agent knows when she is seeing somebody (positive introspection). The axiom Ax_8 says that an agent knows when she is not seeing somebody (negative introspection). In that sense, an agent a is aware of the existence of the other agents and knows whether she sees b them or not. The axiom Ax_9 says that a is aware of the perception of the agents she sees. The axiom Ax_{10} says that a is aware of the non-perception of the agents she sees. The axiom Ax_{11} says that a always can imagine all possible situations compatible with her perceptions. The last axiom can be recursively enumerated since deciding satisfiability of a formula in $\mathcal{L}_P$ is in NP (Theorem 14). Actually, we do not know if Ax_{11} can be replaced by a finite set of axioms.

Note that any instances of T, 4, 5 for K_a are in $\mathcal{PK}$. This follows from Corollary 3 and the fact that such instances are valid.

Example 11 *Let us consider* $G = \{a, b, c\}$. *The formula* $\varphi = \neg a \rhd a \wedge a \rhd b \wedge \neg a \rhd c \wedge b \rhd a \wedge \neg b \rhd b \wedge b \rhd c$ *is a* G*-what-*a*-perceives-mc. The formula* $\Phi = \neg a \rhd a \wedge a \rhd b \wedge \neg a \rhd c \wedge b \rhd a \wedge \neg b \rhd b \wedge b \rhd c \wedge c \rhd a \wedge c \rhd b \wedge \neg c \rhd c$ *is an* G*-describing conjunction which subsumes* φ. *So* $\varphi \rightarrow \hat{K}_a\Phi$ *is an axiom of the theory* $\mathcal{PK}$.

We let the reader check that the axiomatics is sound: each formula of the theory $\mathcal{PK}$ is valid.

The notion of maximal $\mathcal{PK}$-consistency is defined as usual. Obviously, given a maximal $\mathcal{PK}$-consistent set Γ of formulas in $\mathcal{L}_{PK}$, $\Gamma \cap \mathcal{L}_P$ is maximal and $\mathcal{P}$-consistent in $\mathcal{L}_P$. Therefore, for all $a_0 \in AGT$, one may associate to Γ the lineworld $w_\Gamma^{a_0}$ as in Definition 23. Note that since Γ is maximal and consistent, given an agent a and given a finite set of agents G, Γ constains a unique G-what-a-perceives-mc. Now let us demonstrate that the formulas true in $w_\Gamma^{a_0}$ are exactly the formulas in Γ.

Lemma 4 (truth lemma) *For all* $a_0 \in AGT$, *For all* $\varphi \in \mathcal{L}_{PK}$, *for all* Γ *maximal* $\mathcal{PK}$*-consistent set,* $\varphi \in \Gamma$ *iff* $w_\Gamma^{a_0} \models \varphi$.

Proof.

We prove the truth lemma by induction on the modal degree of the formula φ. If φ is a formula in $\mathcal{L}_P$ then by Lemma 3, $\varphi \in \Gamma$ iff $w_\Gamma^{a_0} \models \varphi$. Boolean cases are left to the reader. We are left with the case $\varphi = K_a\psi$.

$\boxed{\Rightarrow}$ Suppose that $K_a\psi \in \Gamma$. We have to prove that $w_\Gamma^{a_0} \models K_a\psi$. In other words, we have to prove that for all $u \in R_a(w_\Gamma^{a_0})$, $u \models \psi$. Let $G = \{b \in AGT$

$\mid b$ occurs in $\varphi\}$. Let χ be the G-what-a-perceives-mc contained in Γ. Let us take $u \in W$ such that $w_\Gamma^{a_0} R_a u$. Let Φ be the G-describing conjunction true in u. By Definition of R_a, we have that Φ subsumes χ. Thus, the instance $\chi \to \hat{K}_a\Phi$ of Ax_{11} is in Γ. Hence $\hat{K}_a\Phi \in \Gamma$. Since $K_a\psi \in \Gamma$, then by axiom Ax_K with K_a, the necessitation rule with K_a and modus ponens, the singleton $\{\Phi \wedge \psi\}$ is consistent. By Lindenbaum's lemma, there exists a maximal $\mathcal{PK}$-consistent set Δ containing the formula $\Phi \wedge \psi$. The modal degree of the formula $\Phi \wedge \psi$ is strictly less that the modal degree of $K_a\psi$. So we can apply the induction hypothetis with $\Phi \wedge \psi$: we have $w_\Delta^{a_0} \models \Phi \wedge \psi$. But $u \models \Phi$. Hence by Lemma 2, $u \models \psi$. This holds for all $u \in R_a(w_\Gamma^{a_0})$. Finally $w_\Gamma^{a_0} \models K_a\psi$.

$\boxed{\Leftarrow}$ Reciprocally suppose we have $\hat{K}_a\psi \in \Gamma$. And let us prove that $w_\Gamma^{a_0} \models \hat{K}_a\psi$. By Proposition 4, up to $\approx$-equivalence, there is only one Γ-canonical lineworld. For this reason, it suffices to prove that $w_\Gamma^a \models \hat{K}_a\psi$. Using Ax_K with K_a, the necessitation rule with K_a and modus ponens, it follows that the set $S = \{\psi\} \cup \{\chi \mid K_a\chi \in \Gamma\}$ is consistent. Let Δ be a maximal $\mathcal{PK}$-consistent set containing S. By induction, we obtain $w_\Delta^a \models \psi$. So it suffices to prove that $w_\Gamma^a R_a w_\Delta^a$. Let $V_\Gamma = \{b \in AGT \mid w_\Gamma^a \models a \rhd b\}$ and $V_\Delta = \{b \in AGT \mid w_\Delta^a \models a \rhd b\}$.

We have $V_\Gamma = V_\Delta$. Indeed:

$\boxed{\subseteq}$ If $w_\Gamma^a \models a \rhd b$ then $a \rhd b \in \Gamma$. By Ax_7 we obtain $K_a a \rhd b \in \Gamma$. Hence by definition of Δ, $a \rhd b \in \Delta$. Hence by induction, $w_\Delta^a \models a \rhd b$.

$\boxed{\supseteq}$ If $w_\Gamma^a \not\models a \rhd b$ then $\neg a \rhd b \in \Gamma$. By Ax_8 we obtain $K_a \neg a \rhd b \in \Gamma$. Hence by definition of Δ, $\neg a \rhd b \in \Delta$. Hence, $w_\Delta^a \not\models a \rhd b$.

For all $b \in V_\Gamma$, for all $c \in AGT$, we have the following equivalence referred as (*): $b \rhd c \in \Gamma$ iff $b \rhd c \in \Delta$. Indeed:

$\boxed{\Rightarrow}$ If $b \rhd c \in \Gamma$, since $a \rhd b \in \Gamma$ the axiom Ax_9 and modus ponens give that $K_a b \rhd c \in \Gamma$. Hence by Definition of Δ, $b \rhd c \in \Delta$.

$\boxed{\Leftarrow}$ If $\neg b \rhd c \in \Gamma$, by Ax_{10}, we have $K_a \neg b \rhd c \in \Gamma$. Hence $\neg b \rhd c \in \Delta$.

We are going to check differents points of the Definition 17 using the Definition of w_Γ^a and w_Δ^a (Definition 23):

- First $V(a)_{w_\Gamma^a} = V_\Gamma = V_\Delta = V(a)_{w_\Delta^a}$;
- We have for all $b, c \in V(a)_{w_\Gamma^a}$, $b <_{w_\Gamma^a} c$ iff $b <_{w_\Delta^a} c$ because $V_\Gamma = V_\Delta$ and (*);
- For all $b \in V(a)_{w_\Gamma^a}$, $\vec{dir}_{w_\Gamma^a}(b) = \vec{dir}_{w_\Delta^a}(b)$ because $V_\Gamma = V_\Delta$ and (*).

Hence $w^a_\Gamma R_a w^a_\Delta$. Finally, we proved that $w^a_\Gamma \models \hat{K}_a\psi$.

■

Corollary 3 *Valid formulas of $\mathcal{L}_{PK}$ are exactly formulas in $\mathcal{PK}$.*

4.5 Conclusion and perspectives

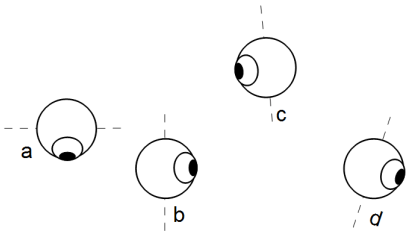


Figure 4.9: Example of a flatworld w.

We have studied an epistemic logic interpreted over lineworlds where knowledge of agents is based on what they can see. We have given a complete axiomatization and tight decision procedures for model checking and satisfiability problems.

We do not know if our epistemic logic is finitely axiomatizable. In other respects, it is unknown whether PSPACE-hardness of the model checking and the satisfiability problems still hold when we the construction $K_a\varphi$ is allowed for agents a which belongs to a finite set $AGT' \subseteq AGT$.

In the next Chapter, we extend our work to *Flatland* [Abb84], i.e. interpreting formulas of $\mathcal{L}_{PK}$ in *flatworlds*: a flatworld is specified by giving to any agent a position in the plane and a direction the agent is looking in. For example, in Figure 4.9, agent a sees b and d but cannot see c.

4.6 Implementation

Algorithms to solve the model-checking problem and the satisfiability problem in Lineland has been implemented in Scheme/Java.

4.6.1 Pedagogical motivation

In addition to robotics and video games application, Lineland (and also Flatland etc.) is a pedagogical tool. Epistemic logic has the syntax and the semantics of

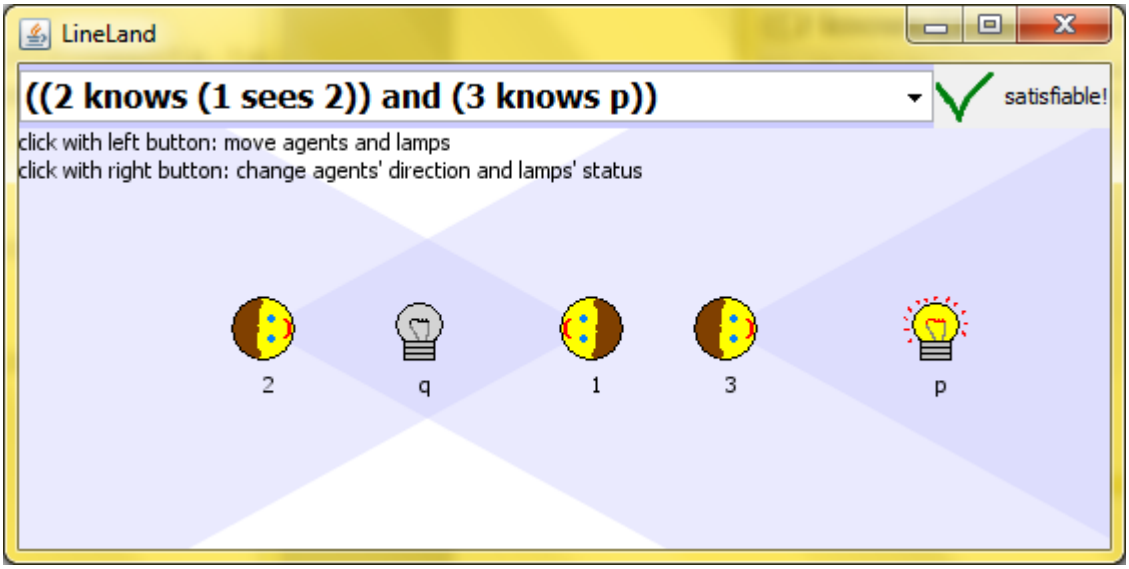


Figure 4.10: Screenshot of the model-checker for Lineland

modal logic and contrarily to temporal logics, it is perhaps more difficult to explain where the possible worlds come from to students who lack a strong background in logic. This is a reason why we study a concrete example of multi-agent system: we put agents in a space (here a line) and then ask "what do agents know about lamps, about the knowledge of other agents about lamps and so on.?" Our logic is implemented as a pedagogical tool in order to illustrate any epistemic logic course. Indeed, students can easily understand some epistemic logic on concrete examples, like the Muddy-children puzzle where each child must guess whether her forehead is muddy or not by considering the others' and knowing that at least one of them is muddy.

Our approach can also be compared to the pedagogical approach in [BE93] where there are objects like cubes and pyramids and where one can write formulas in first order logic to check properties of and relations between these objects. Here our approach is similar: we put agents and lamps in flatland and then, we can write formulas in epistemic logic to check whether some property is true.

4.6.2 How deos it work ?

As you can show in the Figure 4.10, the graphical user interface is divided in two parts:

- the line you can write the formula you want to check. The language is similar to the theoretical language and based on the syntax of Scheme;

- a drawing of a lineworld. You can move agents and lamps with the mouse. You can also change direction of an agent or the state of a lamp.

The language for formulas is the following:

- Agents are natural numbers;
- lamps are a lower-case character;
- (... sees ...);

  (... **or** ...);
- (... **and** ...);

  (... **or** ...);
- (... knows ...);

  (announce ... ...).

4.6.3 Technical information

4.6.3.1 The engine in Scheme

The language Scheme [EA93] is adapted to write this kind of algorithms for many reasons:

- Pattern matching of expressions (formulas) is supported;
- The syntax of Scheme is such that we can directly use a syntax for formulas closed to the formal definition;
- The syntax is simple and the language is dynamically typed so it is suitable for a prototype;
- Scheme can embedded into a Java application via Kawa so it is multi-platform too.

The Figure 4.11 shows the main function of the program written in Scheme: the function is a model-checker. It checks if the formula is true in the world, provided the context. The context is here in order to deal with a sequence of public announcements.

You can download the program on the Internet for more information: `http://www.irit.fr/~Francois.Schwarzentruber/lineland/`.

```
(define (mc-with-context world context formula)
  (match formula
    ('top #t)
    ((phi 'or psi)
     (or  (mc-with-context world context phi)
          (mc-with-context world context psi)))
    (('not phi1)
     (not (mc-with-context world context phi1)))
    ((phi1 'and phi2)
     (and (mc-with-context world context phi1)
          (mc-with-context world context phi2)))
    ((phi1 'implies phi2)
     (or (not (mc-with-context world context phi1))
         (mc-with-context world context phi2)))
    ((a 'knows phi)
     (validin-with-context
      (worldset-delete-not-satisfying
       (world-getpossibleworlds world a) context)
      context
      phi))
    (('announce phi psi)
     (let ((newcontext (list context 'and phi)))
       (if (mc-with-context world context phi)
           (mc-with-context world newcontext psi)
           #t)))
    ((a 'sees b)
     (sees? world a b))
    (p (world-getvalue world p))))
```

Figure 4.11: The function for the model-checking in Scheme

4.6.3.2 The front end in Java

The language Java is adapted in order to create the graphic user interface for many reasons:

- Java offers the suitable and easy to use Application Programming Interface (API) Swing [ELW+98] adapted to design graphic user interface;
- Java is multi-platform via a virtual machine [LY99].

The interface between Java and Scheme is the library kawa: `http://www.gnu.org/software/kawa/`. This library provides a Scheme interpreter that enables to execute Scheme code from a Java program.

Open questions

- Is the logic of Lineland finitely axiomatizable?
- What is the complexity of the satisfiability problem if we restrict constructions of the form $K_a\varphi$ so that a belongs to a finite and fixed set of agents?

Chapter 5

Knowledge in Flatland

5.1 Introduction

In the previous Chapter we have studied the logic of perception and knowledge when the dimension of the space is one. In this Chapter we are interested in the logic of perception and knowledge when the dimension is two: Flatland.

This chapter is organized as follows:

- We present again the epistemic language $\mathcal{L}_{PK}$ in Section 5.2;
- We recall geometric standard notations in Section 5.3,;
- We present the semantics of $\mathcal{L}_{PK}$ in Section 5.4;
- We deal with decidability of the logic in Section 5.5;
- We add public announcements in Section 5.6;
- We talk about perspectives in Section 5.9.

As for Lineland, initially, the first formalization of Flatland [BGS10] used agents and *lamps*. The aim of the lamps was to denote propositions in the world. For instance, as you can see in the muddy-children puzzle, Figure 5.1, lamps are kind of propositions and here represents the state of foreheads. If ℓ_i is on, then it means that the forehead of agent a_i is dirty. In Figure 5.1, the forehead of agent a_2 is clean but the foreheads of the other agents are dirty. As for Lineland, we can easily encode the state of a lamp by extra agents as following: a lamp ℓ_i is on (Figure 5.1) iff the agent ℓ_i is looking at agent a_i (Figure 5.2) That is why we have decided to delete lamps from the language and the models.

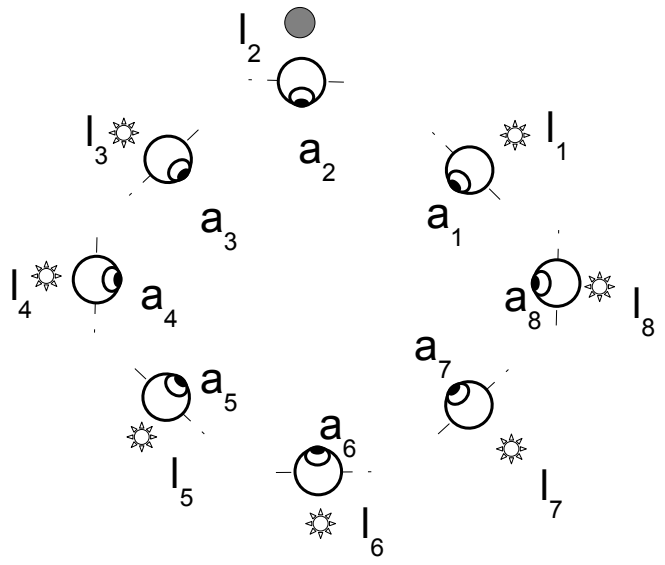


Figure 5.1: Muddy-children in flatland with lamps (denoting the state of foreheads)

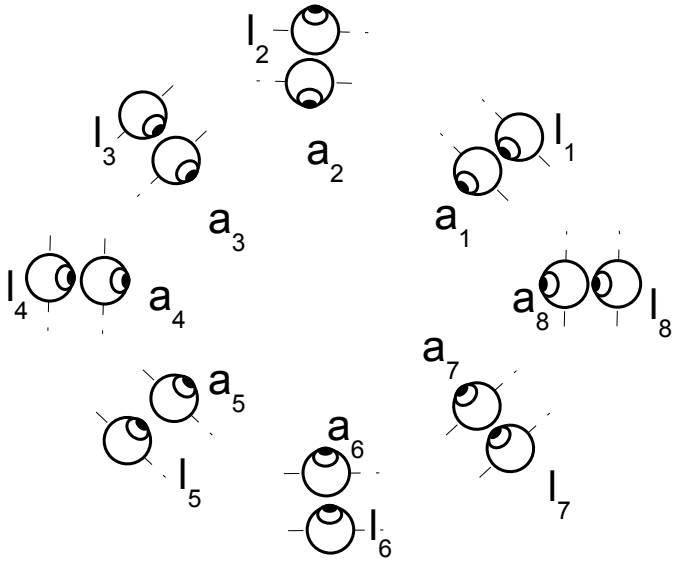


Figure 5.2: Muddy-children in flatland with only agents (the state of foreheads are encoded by agents l_k)

5.2 Syntax

In this Section, we recall the language of perception and knowledge introduced in Subsection 4.2.1 for Lineland.

Let $AGT = \{a, b, c \ldots\}$ be a countable set of *agents*.

Definition 26 (language)
The *language* $\mathcal{L}_{PK}$ is defined by the following BNF:

$$\varphi ::= a \rhd b \mid \bot \mid \neg\varphi \mid (\varphi \vee \varphi) \mid K_a\psi$$

where $a, b \in AGT$.

As usual, $(\varphi \wedge \psi) =^{\text{def}} \neg(\neg\varphi \vee \neg\psi)$. $\hat{K}_a\psi =^{\text{def}} \neg K_a \neg\psi$. We follow the standard rules for omission of parentheses. Let $agt(\varphi)$ be the set of all agents occurring in φ. The formula $K_a b \rhd c$ is read "agent a knows that b sees agent c".

As for Lineland, $\mathcal{L}_P$ denotes the set of formulas without epistemic modality.

5.3 Notations

In this Section, we recall some basic notions of geometry. We note $\mathbb{N}$ the set of *natural numbers* and $\mathbb{R}$ the set of *real numbers*. We note $\mathbb{R}^2$ the *real plane*. If $a \in \mathbb{R}^2$, we note $a = (a_x, a_y)$ where $a_x, a_y \in \mathbb{R}$: a_x is called *abscise* of a and a_y is called *ordinate* of a. If $a, b \in \mathbb{R}^2$, we define $\vec{ab} \in \mathbb{R}^2$ as $\vec{ab} = (b_x - a_x, b_y - a_y)$.

The *scalar product* of a and b is $a \bullet b = a_x \times b_x + a_y \times b_y \in \mathbb{R}$. If $x \in \mathbb{R}^2$, we define $||x|| = \sqrt{x \bullet x}$. $||x||$ is called *euclidian norm* of the vector x.

Let $U = \{x \in \mathbb{R}^2 \mid ||x|| = 1\}$. U is called *unit circle*.

5.4 Concrete semantics

The semantics is not defined with a class of models but directly with a concrete flatland situation. A *flatworld* is a situation where all *agents* have a *location* (*position* and *direction* at which they look) in the plane , all *lamps* have a *position* and a *state* (on or off). From it, we will obtain a spatially grounded epistemic logic. Formally:

Definition 27 (flatworld)
A *flatworld* w is a tuple $\langle pos, \vec{dir} \rangle$ where:

- $pos : AGT \to \mathbb{R}^2$;
- $\vec{dir} : AGT \to U$;

The set of all worlds is noted W.

In a flatworld $w = \langle pos, \vec{dir} \rangle$, for all agent a, $pos(a)$ is the position of agent a in the plane. For all agent a, the vector $\vec{dir}(a)$ of norm 1 denotes the direction where agent a is looking. The agent see all the closed half-plane in the direction $\vec{dir}(a)$.

Example 12 *The Figure 2.1 of Muddy children is a flatworld in the sense of Definition 27:*

- $AGT = \{a_1, \ldots, a_8, \ell_1, \ldots, \ell_8\}$;
- $pos(a_k) = \big(\cos(\frac{k\pi}{4}), \sin(\frac{k\pi}{4})\big)$; *(positions of agents a_k)*
- $pos(\ell_k) = \big(1.1{\times}\cos(\frac{k\pi}{4}), 1.1{\times}\sin(\frac{k\pi}{4})\big)$; *(positions of agents ℓ_k)*
- $\vec{dir}(a_k) = \big(-\cos(\frac{k\pi}{4}), -\sin(\frac{k\pi}{4})\big)$; *(directions of a_k)*
- $\vec{dir}(a_k) = \big(-\cos(\frac{k\pi}{4}), -\sin(\frac{k\pi}{4})\big)$; *(directions of a_k for all $k \neq 2$)*
- $\vec{dir}(a_2) = (0, 1)$.

Now we define the set of all points that an agent a sees.

Definition 28 (cone)
Let us consider a flatworld $w = \langle pos, \vec{dir}, val \rangle$. For all $a \in AGT$, we note $cone_w(a)$ the set $\{x \in \mathbb{R}^2 \mid \vec{dir}(a) . \vec{pos(a)}x \geq 0\}$.

As depicted in the Figure 5.3, $cone_w(a)$ is the *closed half-plane* of all points x such that $\vec{dir}(a) . \vec{pos(a)}x \geq 0$. We could change this Definition:

- Agent can see the *open half-plane*;
- Agent can see only a cone of angle α: $cone_w(a) = \{x \in \mathbb{R}^2 \mid \vec{dir}(a) . \vec{pos(a)}x \geq cos(\alpha)||\vec{pos(a)}$
- An agent can be *myopic* and does not see what is at a distance greater than r, etc. $cone_w(a) = \{x \in \mathbb{R}^2 \mid \vec{dir}(a) . \vec{pos(a)}x \geq 0$ and $||\vec{pos(a)}x|| \leq r\}$.
- Agent can see the open half-plane: $cone_w(a) = \{x \in \mathbb{R}^2 \mid \vec{dir}(a) . \vec{pos(a)}x > 0\}$;
- Agent can see only a cone of angle α: $cone_w(a) = \{x \in \mathbb{R}^2 \mid \vec{dir}(a) . \vec{pos(a)}x \geq cos(\alpha)||\vec{pos(a)}$

Here for the sake of simplicity, we adopt the Definition 28.

Now we define the epistemic relation over worlds. For all $w, u \in W$, wR_au means that agent a cannot distinguish w from u, i.e. agent a sees the same objects in w and u. Differences between w and u only lie in positions, directions, states of objects that agent a does not see. Formally:

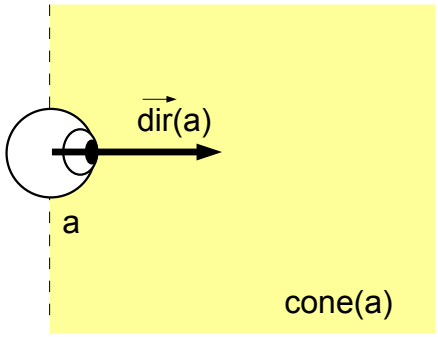


Figure 5.3: $cone_w(a)$ = closed half-plane seen by agent a

Definition 29 (epistemic relation)
Let $a \in AGT$. We define the *relation* R_a over worlds $w = \langle pos, \vec{dir} \rangle$ and $u =: \langle pos', \vec{dir}' \rangle$: wR_au iff:

- for all $b \in AGT$,

$$pos(b) \in cone_w(a) \text{ iff } pos'(b) \in cone_u(a);$$

- and for all $b \in AGT$, if $pos(b) \in cone_w(a)$ then

$$pos(b) = pos'(b) \text{ and } \vec{dir}(b) = \vec{dir}'(b).$$

The Figure 5.4 presents two worlds linked by R_b: agents c, d are seen by agent b and so they have the same positions and directions in both worlds. But, agent a can change directions and positions provided she remains invisible from agent a.

Obviously, the relation R_a is an equivalence relation.

Definition 30 (truth conditions)
Let $w = \langle pos, \vec{dir}, val \rangle \in W$ and φ be a formula of $\mathcal{L}_{PK}$. We define $w \models \varphi$ by induction:

- $w \not\models \bot$;
- $w \models a \rhd b$ iff $pos(b) \in cone_w(a)$;
- $w \models \varphi \vee \psi$ iff $w \models \varphi$ or $w \models \psi$;
- $w \models \neg\varphi$ iff $w \not\models \varphi$;

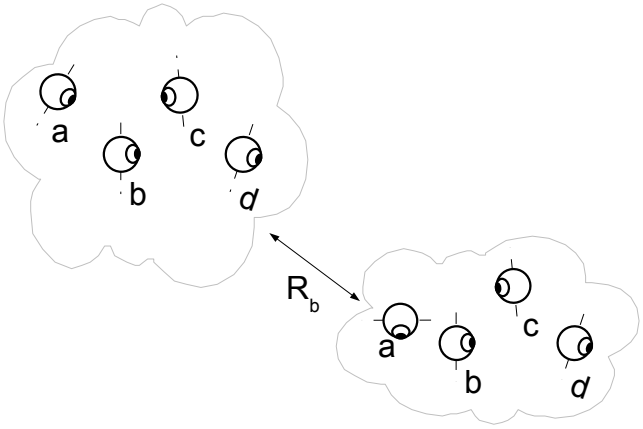


Figure 5.4: Two worlds linked by R_a

- $w \models K_a\psi$ iff for all $u \in W$, wR_au implies $u \models \psi$.

We shall say that a formula φ is *satisfiable* iff there exists a flatworld $w \in W$ such that $w \models \varphi$. Formula φ is said to be *valid* iff for all worlds $w \in W$, $w \models \varphi$. Since R_a is an equivalence relation on W, then the axioms of classical epistemic logic $S5_n$ are valid:

- $K_a\varphi \rightarrow \varphi$;
- $K_a\varphi \rightarrow K_aK_a\varphi$;
- $\neg K_a\varphi \rightarrow K_a\neg K_a\varphi$.

5.5 Two decision problems

In Subsection 5.5.1, we see that finding a qualitative semantics is not so trivial. That is why we remain with a quantitative representation of worlds: in Subsection 3.2.1.1 , we recall the theory of real numbers enabling us to reduce the model-checking and satisfiability of an epistemic formula in flatland in subsection 5.5.2 . Let us recall the Definitions of the problem of model-checking and satisfiability.

Definition 31 (model-checking in flatland)

We call *model-checking in Flatland* the following problem:

- Input: a formula $\varphi \in \mathcal{L}_{PK}$, a *description of the flatworld* w;

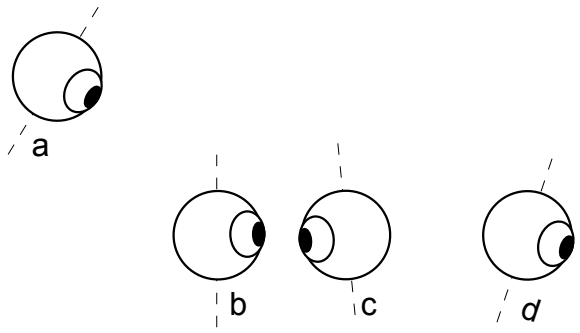


Figure 5.5: flatworld where agents b, a, lamps ℓ, m,n are aligned

- Output: Yes iff we have $w \models \varphi$. No, otherwise.

In Definition 31, the *description of the flatworld* w only objects occurring in φ are taken in account. Positions of agents are supposed to have *rational coordinates* because we need a *data structure* to represent positions of agents. In the same way, we need a data structure to represent directions of agent.. We do not represent directions with angles... because the logic of real numbers with the function cos is... undecidable. Indeed, if we cos, we can define π and then define integers and Peano's arithmetic is undecidable. We represent $\vec{dir}(a)$ by the abscise $\vec{dir}(a)_x$, supposed to be rational and the sign of the ordinate $\vec{dir}(a)_y$. The value of $\vec{dir}(a)_x$ and the sign of $\vec{dir}(a)_y$ entirely determine $\vec{dir}(a)$ because $\vec{dir}(a)_y^2 = 1 - \vec{dir}(a)_y^2$.

Definition 32 (flatland-satisfiability problem)
The *flatland-satisfiability problem* is the following problem:

- Input: a formula $\varphi \in \mathcal{L}_{PK}$;
- Output: Yes iff there exists a flatworld w such that $w \models \varphi$.

5.5.1 A non-successful qualitative semantics

One idea could be that facts of the form $a \rhd b$ are sufficient to represent a situation. The Figures 5.5 and 5.6 gives us two worlds where we have the same valuation (*) for those facts. In both situations:

- $a \rhd b$, $a \rhd c$, $a \rhd d$ are true;

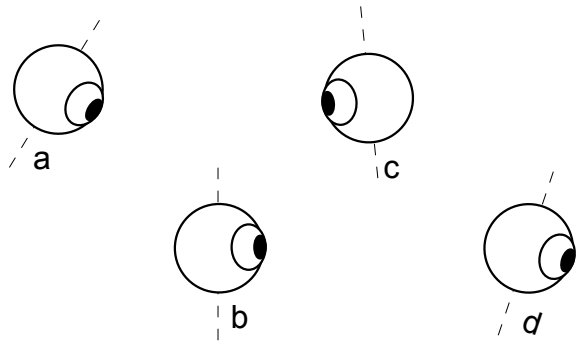


Figure 5.6: flatworld where objects b, a, ℓ, m,n are not aligned

- $\neg b \rhd a$, $b \rhd c$, $b \rhd d$ are true;
- $c \rhd a$, $c \rhd b$, $\neg c \rhd d$ are true;
- $\neg d \rhd a$, $\neg d \rhd b$, $\neg d \rhd c$ are true.

Nevertheless, in the situation of the Figure 5.6, the formula $\varphi = \hat{K}_b(a \rhd b \wedge a \rhd d \wedge \neg a \rhd c)$ holds whereas this formula φ is false in the situation of the Figure 5.5. Indeed:

- φ holds in the flatworld of Figure 5.6 because agent a can imagine the flatworld of the Figure 5.4.
- φ does not hold in Figure 5.5 because as $pos(b)$, $pos(c)$, $pos(d)$ are aligned for all possible worlds u for agent b, $pos(b) \in cone_w(a)_u$ and $pos(d) \in cone_w(a)_u$ implies $pos(c) \in cone_w(a)_u$ because $cone_u(a)$ is a *convex* set. Hence in the flatword of Figure 5.5 the formula $K_b(a \rhd b \wedge a \rhd d \rightarrow a \rhd c)$ holds.

To sum up, we have exhibited two worlds (one of Figure 5.6 and one of 5.5), satisfying the same extra propositions of the form $a \rhd b$ but not the same epistemic formulas. What agents see or not does not determine a unique epistemic situation. This means that representing a flatland situation is not trivial. In other worlds, the Lemma 2 is no longer true in Flatland. We tried other formalization that takes into account that such and such agents are aligned, or that the former is on the left of the latter, etc. but unsuccessfully. The existence of a qualitative and complete representation of a flatworld remains an open question.

5.5.2 Translation into real numbers

In order to translate a formula of $\mathcal{L}_{PK}$ into the logic of real numbers, we will introduce numerated situations to simulate the truth condition of $K_a\psi$. (see Definition 33) We need the set of variables $\mathbb{VAR}$ to contain some extra variables (written in bold face):

- For all $n \in \mathbb{N}$ and for all $a \in AGT$, the set $\mathbb{VAR}$ contains the variables $\mathbf{pos}^n_{x,a}$ and $\mathbf{pos}^n_{y,a}$. They will be equal respectively to the abscise and ordinate of the position of the agent a in the situation number n.
- For all $n \in \mathbb{N}$ and for all $a \in AGT$, the set $\mathbb{VAR}$ contains the variables $\mathbf{dir}^n_{x,a}$ and $\mathbf{dir}^n_{y,a}$. They will be respectively equal to the abscise and ordinate of the direction of the agent a in the situation number n.

We define the following abbreviations:

- $DIR(n,\varphi)$ says the variables of directions of agents of the formula φ represent vectors of the unit circle U:
  $DIR(n,\varphi) = \bigwedge_{b\in agt(\varphi)} ||\vec{\mathbf{dir}}^n_b||^2 = 1$ where $||\vec{\mathbf{dir}}^n_b||^2$ is the expression $\mathbf{dir}^n_{x,b}\times \mathbf{dir}^n_{x,b} + \mathbf{dir}^n_{y,b}\times \mathbf{dir}^n_{y,b}$.
- $FORALL(n,\varphi) = \forall\ \mathbf{pos}^n_{x,a_1}, \ldots \forall\ \mathbf{pos}^n_{x,a_k} \forall\ \mathbf{pos}^n_{y,a_1}, \ldots \forall\ \mathbf{pos}^n_{y,a_k}$
  $\forall\ \mathbf{dir}^n_{x,a_1}, \ldots \forall\ \mathbf{dir}^n_{x,a_l} \forall\ \mathbf{dir}^n_{y,a_1}, \ldots \forall\ \mathbf{dir}^n_{y,a_l}$
  where $agt(\varphi) = \{a_1, \ldots, a_l\}$.
  $FORALL(n,\varphi)$ is a "for all quantifier" over all variables interpreted as positions of objects, directions of agents and valuations of atoms of φ in the situation number n.
- The formula $EPI(n,a,\varphi)$ will be a formula of $\mathcal{L}_{\mathbb{R}}$ saying that the situations number n and $n+1$ are linked by the epistemic relation R_a, i.e. are similar w.r.t. what agent a sees. More precisely, the variables representing the situations number n and $n+1$ are satisfying the constraints of Definition 29. Moreover, in $EPI(n,a,\varphi)$ we are only interested about objects of the formula φ. Formally:
  $EPI(n,a,\varphi) = \bigwedge_{b\in agt(\varphi)}[\vec{\mathbf{dir}}^n_a \cdot \mathbf{pos}^n_a \vec{\mathbf{pos}}^n_b \geq 0 \leftrightarrow$
  $\vec{\mathbf{dir}}^{n+1}_a \cdot \mathbf{pos}^{n+1}_a \vec{\mathbf{pos}}^{n+1}_b \geq 0] \wedge$
  $\bigwedge_{b\in agt(\varphi)}[\vec{\mathbf{dir}}^n_a \cdot \mathbf{pos}^n_a \vec{\mathbf{pos}}^n_b \geq 0$
  $\rightarrow (\ \mathbf{pos}^{n+1}_{x,b} = \mathbf{pos}^n_{x,b} \wedge\ \mathbf{pos}^{n+1}_{y,b} = \mathbf{pos}^n_{y,b} \wedge$

$$\mathbf{dir}^{n+1}_{x,b} = \mathbf{dir}^{n}_{x,b} \wedge \mathbf{dir}^{n+1}_{y,b} = \mathbf{dir}^{n}_{y,b})].$$

where $\vec{\mathbf{dir}}^{\,n}_{a}.\mathbf{pos}^{n}_{a}\vec{\mathbf{pos}}^{\,n}_{a}$ is the expression $\mathbf{dir}^{n}_{x,a} \times (\mathbf{pos}^{n}_{x,a} - \mathbf{pos}^{n}_{x,a}) + \mathbf{dir}^{n}_{y,a} \times (\mathbf{pos}^{n}_{y,a} - \mathbf{pos}^{n}_{y,a})$ etc.

Now we we can give the translation of a formula of $\mathcal{L}_{PK}$ into $\mathcal{L}_{\mathbb{R}}$. Given $\varphi \in \mathcal{L}_{PK}$ and $n \in \mathbb{N}$, we introduce $\tau(n, \varphi) \in \mathcal{L}_{\mathbb{R}}$ whose meaning is "φ is true in the situation number n".

Definition 33 (translation)
We define the *translation* $\tau : \mathbb{N} \times \mathcal{L}_{PK} \to \mathcal{L}_{\mathbb{R}}$ by, for all $n \in \mathbb{N}$:

- $\tau(n, \bot) = \bot$;
- $\tau(n, a \rhd b) = (\vec{\mathbf{dir}}^{\,n}_{a}.\mathbf{pos}^{n}_{a}\vec{\mathbf{pos}}^{\,n}_{b})$;
- $\tau(n, \varphi_1 \vee \varphi_2) = \tau(n, \varphi_1) \vee \tau(n, \varphi_2)$;
- $\tau(n, \neg\varphi) = \neg\tau(n, \varphi)$;
- $\tau(n, K_a\varphi) = FORALL(n+1, \varphi)$
$$DIR(n+1, \varphi) \wedge EPI(n, a, \varphi) \to \tau(n+1, \varphi).$$

where $FORALL(n+1, \varphi)$, $DIR(n+1, \varphi)$, $EPI(n, a, \varphi)$ are defined above.

Let us explain $\tau(n, K_a\varphi)$: the variables of the situation number n are such that for all situation number $n+1$ $[FORALL(n+1, \varphi)]$, if the directions are correct $[DIR(n+1, \varphi)]$ and if the situation number $n+1$ is linked to the situation number n by R_a $[EPI(n, a, \varphi)]$ then the formula φ is true in the situation number $n+1$ $[\tau(n+1, \varphi)]$. The Definition $\tau(n, K_a\varphi)$ simulates the truth condition of $K_a\psi$ (Definition 30) by using numerated situations.

Proposition 5 *For all $\varphi \in \mathcal{L}_{PK}$, for all $n \in \mathbb{N}$, for all $w = \langle pos, \vec{dir}\rangle \in W$, we have:*

$w \models \varphi$ iff $I \models \tau(n, \varphi)$ for all $I \in Inter(n, w, \varphi)$ where $Inter(n, w, \varphi)$ is the set of all interpretations I such that:

- $I(\ \boldsymbol{pos}^{n}_{x,a}) = pos(\beta)_x$, $I(\ \boldsymbol{pos}^{n}_{y,a}) = pos(\beta)_y$ *for all* $a \in agt(\varphi)$;
- $I(\ \boldsymbol{dir}^{n}_{x,a}) = \vec{dir}(a)_x$, $I(\ \boldsymbol{dir}^{n}_{y,a}) = \vec{dir}(a)_y$ *for all* $a \in agt(\varphi)$;

PROOF.

Let $V^{n,\varphi} = \{\vec{\mathbf{pos}}^n, \vec{\mathbf{dir}}^n\}$. By induction on φ. The property is $\mathcal{P}(\varphi)$ ="for all $n \in \mathbb{N}$, for all $w \in W$, we have $w \models \varphi$ iff $I \models \tau(n, \varphi)$".

$\boxed{\varphi = a \rhd b}$

Left to the reader.

$\boxed{\neg\varphi}$

$w \models \neg\varphi$ iff $w \not\models \varphi$ iff not[for all $I \in Inter(n, w, \varphi)$, $I \models \tau(n, \varphi)$] iff there exists I such that (*) and $I \not\models \tau(n, \varphi)$ iff there exists $I \in Inter(n, w, \varphi)$ and $I \models \neg\tau(n, \varphi)$ iff there exists $I \in Inter(n, w, \varphi)$ and $I \models \tau(n, \neg\varphi)$ for all $I \in Inter(n, w, \varphi)$ and $I \models \tau(n, \neg\varphi)$. (because the interpretation of $\tau(n, \neg\varphi)$ only depends of $\vec{\mathbf{pos}}^n, \vec{\mathbf{dir}}^n$).

$\boxed{K_a\varphi}$

$w \models K_a\varphi$ iff for all $u \in R_a(w)$ $u \models \varphi$ iff for all $u \in W$ such that Definition 29, $u \models \varphi$. By induction, it is equivalent to for all $u \in W$ such that Definition 4, for all $J \in Inter(n+1, u, \varphi)$, $J \models \tau(n+1, \varphi)$. (1)

For all $I \in Inter(n, K_a\varphi)$, $I \models \tau(n, K_a\varphi)$. It is equivalent to for all $I \in Inter(n, w, K_a\varphi)$, for all interpretation J such that $I_{|V^{n,\varphi}} = J_{|V^{n,\varphi}}$ and $J \models epi(n, a, \varphi)$ we have $J \models \tau(n+1, \varphi)$. (2)

Let us prove (1) $\Leftrightarrow$ (2). Suppose (1). Let $I \in Inter(n, w, K_a\varphi)$ and J such that $I_{|V^{n,\varphi}} = J_{|V^{n,\varphi}}$ and $J \models epi(n, a, \varphi)$. Let $u \in W$ such that:

- $I(\mathbf{pos}_\beta^{n+1}) = p_\beta$ for all $\beta \in AGT \cap \varphi$;
- $I(\vec{\mathbf{dir}}_b^{\,n+1}) = d_b$ for all $b \in AGT \cap \varphi$;

and for all objects it is like in w. We can prove that $u \in R_a(w)$. And as $J \in Inter(n+1, u, \varphi)$, we have by (1), we have $J \models \tau(n+1, \varphi)$ and we have proved (2).

On the contrary, suppose (2). Let $u \in R_a(w)$ and $J \in Inter(n+1, u, \varphi)$. Let $I \in Inter(n, w, K_a\varphi)$ and let K be as J plus $K_{|V^{n,\varphi}} = I_{|V^{n,\varphi}}$. As $u \in R_a(w)$, we have $K \models epi(n, a, \varphi)$. So $K \models \tau(n+1, \varphi)$. As $J_{|V^{n+1,\varphi}} = K_{|V^{n+1,\varphi}}$, $J \models \tau(n+1, \varphi)$. We have proved (1).

■

Corollary 4 *The $\mathcal{L}_{PK}$-satisfiability problem is decidable.*

PROOF.

Given $\varphi \in \mathcal{L}_{PK}$ and $w = \langle p, d, \pi\rangle$, we have φ is satisfiable iff $DIR(0, \varphi) \wedge \tau(0, \varphi)$ is $\mathbb{R}$-satisfiable. And τ is computable (and in polynomial time!). ■

Corollary 5 *The model-checking in flatland in decidable.*

PROOF.

Given $\varphi \in \mathcal{L}_{PK}$ and $w = \langle pos, \vec{dir} \rangle$ such coordinates of positions are rational and abscises of directions are rational. We have $w \models \varphi$ iff $INIT(w, \varphi) \wedge \tau(0, \varphi)$ is $\mathbb{R}$-satisfiable where $INIT(w)$ is a formula saying that the flatworld number 0 is the flatworld w. More precisely:

$INIT(w, \varphi) =$

$$\bigwedge_{a \in agt(\varphi)} (\ \mathbf{pos}^0_{x,a} = pos(a)_x) \wedge (\ \mathbf{pos}^0_{y,a} = pos(a)_y) \wedge$$
$$\bigwedge_{a \in agt(\varphi)} [(\ \mathbf{dir}^0_{x,a} = \vec{dir}(a)_x) \wedge (\ \mathbf{dir}^0_{y,a} \triangle_a 0)] \wedge$$
$$DIR(0, \varphi)$$

where:

- $\triangle_a =$ " $=$" iff $\vec{dir}(a)_y = 0$;
- $\triangle_a =$ " $>$" iff $\vec{dir}(a)_y > 0$;
- $\triangle_a =$ " $<$" iff $\vec{dir}(a)_y < 0$.

■

We have tried to solve the satisfiability problem for small formulas of $\mathcal{L}_{PK}$ by treating the translation of it with the solver REDLOG for the real logic [Wei93]: it is slow[1]! In the Chapter 4, we have proved that the model-checking problem and the satisfiability problem are PSPACE-complete in *lineland*. We conjecture (and hope!) that, in *flatland*, these decision problems are PSPACE-complete too.

5.6 Public announcement

As done in [Pla07] we can extend our framework with public announcements. This is essentially motivated by modeling examples like Muddy children. With public announcements, an agent will be able to learn something about the part of the actual flatworld which he can not see. The technique is classical: we add an operator $[\varphi!]$ and we define semantics as in $S5_n$.

5.6.1 Definitions

Our new language $\mathcal{L}^!$ is defined by the following rule:

$$\varphi ::= a \rhd b \mid \bot \mid \neg\varphi \mid (\varphi \vee \varphi) \mid K_a\psi \mid [\varphi!]\varphi$$

where $a, b \in AGT$.

[1] 130 seconds to solve the non-validity of $K_a on_\ell$ on a 1.5Ghz processor!

The formula $[\varphi!]\psi$ says that if φ holds in the current situation then publicly announcing φ, i.e. restricting the current situation to the set of all worlds where φ holds, creates a new situation where ψ holds. From now, we write $U, w \models \varphi$ and it means that φ is true in w given that U is the set of all worlds compatible with all announcements already made.

Definition 34 (truth conditions)
Let U be a set of worlds ($U \subseteq W$). Let $w \in U$. We define $U, w \models \varphi$ by induction:

- $U, w \models a \rhd b$ iff $w \models a \rhd b$;
- $U, w \models \varphi \vee \psi$ iff $U, w \models \varphi$ or $U, w \models \psi$;
- $U, w \models \neg\varphi$ iff $U, w \not\models \varphi$;
- $U, w \models K_a\psi$ iff for all $w' \in U$, wR_aw' implies $U, w' \models \psi$;
- $U, w \models [\varphi!]\psi$ iff $U, w \models \varphi$ implies $U', w \models \psi$ where $U' = \{w' \in U \mid U, w' \models \varphi\}$.

Example 13 (Muddy-children) *Let us consider the flatworld w depicted in the Figure 2.1. We have*

$$W, w \models [\bigwedge_{i\in\{1...8\}} \bigwedge_{j\in\{1...8\},j\neq i} a_i \rhd \ell_j!][\bigvee_{i\in\{1...8\}} \ell_i \rhd a_i!][\bigwedge_{i\in\{1...8\}} \neg K_{a_i}\ell_i \rhd a_i!]^7 K_{a_1}\ell_1 \rhd a_1.$$

where the construction $a_i \rhd \ell_j$ models the proposition "agent i sees the forehead of agent j", the formula $\bigwedge_{i\in\{1...8\}}\bigwedge_{j\in\{1...8\},j\neq i} a_i \rhd \ell_j$ means that all agents i sees the forehead of all agents j and the construction $\ell_i \rhd a_i$ means "the forehead of agent i is dirty".

We shall say that a formula φ is *satisfiable* iff there exists a flatworld $w \in W$ such that $W, w \models \varphi$. Formula φ is said to be *valid* iff for all worlds $w \in W$, $W, w \models \varphi$.

5.6.2 Decidability

As in the previous Section, we define a translation from $\mathcal{L}^!$ into $\mathcal{L}_{\mathbb{R}}$. Here, we need lists of formula. $list(\mathcal{L}^!)$ denotes the set of lists of formula in $\mathcal{L}^!$. The empty list is noted []. Given $\psi \in \mathcal{L}^!$ and a list $L \in list(\mathcal{L}^!)$, we denote by $[\psi : L]$ the list whose first element is ψ and whose queue is L.

Definition 35 (translation)
We define the translation $\tau : \mathbb{N} \times list(\mathcal{L}^!) \times \mathcal{L}_{PK} \to \mathcal{L}_{\mathbb{R}}$ by: for all $n \in \mathbb{N}$,

- $\tau(n, L, \bot) = \bot$;
- $\tau(n, L, a \rhd b) = \tau(n, a \rhd b)$;
- $\tau(n, L, \varphi_1 \vee \varphi_2) = \tau(n, L, \varphi_1) \vee \tau(n, L, \varphi_2)$;
- $\tau(n, L, \neg\varphi) = \neg\tau(n, L, \varphi)$;
- $\tau(n, [], K_a\varphi) = FORALL(n+1, \varphi)$
  $DIR(n+1, \varphi) \wedge EPI(n, a, \varphi) \to \tau(n+1, \varphi).$;
- $\tau(n, [\psi : L], K_a\varphi) = FORALL(n+1, \varphi)$,
  $DIR(n+1, \varphi) \wedge EPI(n, a, \varphi) \wedge \tau(n, L, \psi)$
  $\to \tau(n+1, \varphi).$;
- $\tau(n, L, [\psi!]\varphi) = \tau(n, L, \psi) \to \tau(n, [\psi : L], \varphi)$.

where $FORALL(n+1, \varphi)$, $DIR(n+1, \varphi)$, $EPI(n, a, \varphi)$ are defined in Subsection 5.5.2.

Here is the Proposition of correctness of the translation:

Proposition 6 *For all $\varphi \in \mathcal{L}^!$, for all $n \in \mathbb{N}$, for all $w \in W$, we have: $w \models \varphi$ iff $I \models \tau(n, \varphi)$ for all $I \in Inter(n, w, \varphi)$.*

In the same way, reasoning about knowledge and public announcements in flatland is *decidable*.

5.7 Weaker semantics

With the Definition 29, agents are "very clever". They can make the difference between aligned points, points exactly settled on the conic of equation $x^2 - 3y^2 = \frac{1}{4}$. In real life, humans and also robots have not this capabilities. For instance, stars in the sky seems to appear on a sphere (the celestial sphere) but stars are not settled on a sphere at all. Since we do not have such perfect abilities, we decide to make the Definition 29 weaker.

Let us define $V_w(a) = \{b \in AGT \mid pos(b) \in cone_w(a)\}$.

Here is a version of a Definition for the epistemic relations where agents can only know whether an agent a sees an agent b or not but they have no information about the exact positions of agents in mind:

Definition 36 (epistemic relation)
Let $a \in AGT$. We define the *relation* R_a over worlds $w = \langle pos, \vec{dir} \rangle$ and $u =: \langle pos', \vec{dir}' \rangle$: wR_au iff:

- $V_w(a) = V_u(a)$;
- For all agent $b, c \in V_w(a)$, $c \in V_w(b)$ iff $c \in V_u(b)$.

From now we have two semantics for flatland:

- the initial one where agents are clever and knows the exact positions of agents with Definition 29;
- the new one where agents are stupid and only take care about what agents sees with Definition 36. We note $\models_{\text{stupid}}$ the satisfiability symbol for this variant.

Theorem 18 *The model checking:*

- *input: a flatworld w and a formula φ;*
- *output: yes if w such that $w \models_{stupid} \varphi$; no otherwise.*

and the satisfiability problem:

- *input: a formula φ;*
- *output: yes if there exists a flatworld w such that $w \models_{stupid} \varphi$; no otherwise.*

are PSPACE.

Proof.
The Figure 5.7 provides an algorithm to solve the model-checking of Flatland with $\models_{\text{stupid}}$. A flatworld w is now represented by the set of all literals of the form $a \rhd b$ which are true, where a and b are agents appearing in the formula φ.
In order to test if flatworld w is satisfiable we simply use the PSPACE-procedure provided by 9. ■

5.8 Comparisons

Proposition 7 *Let $\varphi \in \mathcal{L}_P$ (i.e. formula has no modal operator but only literals of the form $a \rhd b$).*
We have equivalence between φ is valid in the clever version Flatland and φ is valid in the stupid version of Flatland. If φ is valid in (clever/stupid) Flatland then φ is valid in Lineland.

PROOF.
We can represent a lineworld as a flatworld. ■

Proposition 8 *$K_d(\neg a \rhd b \wedge a \rhd c \rightarrow (b \rhd a \leftrightarrow b \rhd c))$ is valid in Lineland but not in Flatland (both versions).*

$\neg d \rhd a \wedge \neg d \rhd b \wedge \neg d \rhd c \rightarrow \neg K_d(\neg a \rhd b \wedge a \rhd c \rightarrow (b \rhd a \leftrightarrow b \rhd c))$ is valid in Flatland (both versions) but is not valid in Lineland.

Proposition 9 *There exists a valid formula for clever agents which is not valid for stupid agents.*

There exists a valid formula for stupid agents which is not valid for clever agents.

PROOF.
Let $\varphi = b \rhd c \wedge b \rhd d \wedge \neg b \rhd a \rightarrow \hat{K}_b(a \rhd b \wedge a \rhd d \wedge \neg a \rhd c)$. The formula φ is valid for stupid agents but not for clever agents.

$\neg a \rhd b \wedge \neg a \rhd c \wedge \neg a \rhd d \rightarrow \hat{K}_a\varphi$ is valid for clever agents but not for stupid agents.

■

5.9 Perspectives

There are many perspectives emerging from this work, some of them already brought up in the paper are long range perspectives: enrich the situation by adding obstacles or indirect sight (like mirrors), and take into account the shape of objects. At shorter term, we aim at solving the questions concerning the exact complexity class of flatland-satisfiability and model-checking problems, and concerning the decidability and complexity issues with the common knowledge operator CK_J. This is a way to compare Lineland, Flatland and *Spaceland*. Finally, we aim at implementing an efficient a flatland solver especially to have a good pedagogical tool for students in epistemic modal logic.

Acknowledgment. Thanks to Andreas Herzig and Emiliano Lorini for their advices.

5.10 Open questions

- Is the logic of Flatland axiomatizable?
- Is the logic of Flatland in PSPACE?
- Is the logic of Flatland in PSPACE-hard?

```
procedure istrue(w, φ)
    match (φ)
        ⊤: accept ;
        a ▷ b:
        accept iff a ▷ b ∈ w
        ψ1 ∨ ψ2:
            choose (∃)i ∈ {1, 2};
            call istrue(w, φi);
        ¬ψ: call isfalse(w, ψ);
        Kaψ:
            choose (∀)u ∈ W;
            if u ∈ Ra(w) then
                if u satisfiable then
                    call istrue(u, ψ)
                else
                    accept
                endIf
            else
                accept
            endIf
    endMatch
endProcedure
```

```
procedure isfalse(w, φ)
    match (φ)
        ⊤: reject ;
        a ▷ b:
        accept iff a ▷ b ∈ w
        ψ1 ∨ ψ2:
            choose (∀)i ∈ {1, 2};
            call isfalse(w, φi);
        ¬ψ: call istrue(w, ψ);
        Kaψ:
            choose (∃)u ∈ W;
            if u ∈ Ra(w) then
                if u satisfiable then
                    call isfalse(u, ψ)
                else
                    reject
                endIf
            else
                reject
            endIf
    endMatch
endProcedure
```

Figure 5.7: Algorithm for stupid agents

- How to implement Flatland?
- Combine the operator $\Diamond\varphi$ "the agent make an effort such that φ" ([MP92], [PMS07] and [Hei06]) and Flatland: the semantics of $\Diamond$ may consist in widen the vision cone of an agent.

Part II

Doing

Chapter 6

Towards the logic STIT

Actions are omnipresent in real life and in computer science. For instance in chess game "moving a pawn" or "castling" are actions. For instance in robotics, actions can be "walk" or "turn the head".

In this chapter we first present well-known formalizations in modal logics dealing with actions and/or choice of actions by agents. We will see the drawbacks of those formalism in terms of expressivity. Then we introduce individual STIT ("sees-to-it-that') logic and finally the group STIT logic.

6.1 PDL

In this section we introduce the famous logic PDL [FL79] devoted to deal with *actions* and even more precisely *programs*. Let us consider a countably infinite set of atomic propositions ATM and a countably infinite set of atomic actions ACT. The language of PDL is defined in the following way:

$$\varphi \quad ::= \quad p \quad | \quad \neg\varphi \quad | \quad (\varphi \wedge \varphi) \quad | \quad [\pi]\varphi$$

where p ranges over ATM and π ranges over the set of all *regular expressions* formed over the set of atomic actions ACT and with operators ; and *.[1]

We do not give the semantics, axiomatization here. You can find an axiomatization and more information in [BDRV02]. Nevertheless we give an intuition of the semantics:

- An atomic proposition p is true means that the interpretation of it in the current state of the program/device/game is true;

[1] There are more complex version of PDL with $\cup$, $\cap$, $\varphi?$, etc.

- $[a]\varphi$ means that φ is true in all states reachable from the current state by an execution of the atomic action a (for instance "set a variable v to 1", "push the button", etc.). We simply say that $[a]\varphi$ iff after executing a we have φ.
- $[\pi_1;\pi_2]\varphi$ is true means φ is true after executing φ_1 then φ_2 we have φ;
- $[\pi^*]\varphi$ is true means φ is true after executing a finite number of times the action π.

As you can see, this logic is quite interesting to speak about actions: we can tell the outcomes/effects of an action. But it does not talk at all about agents.

Drawbacks of PDL In PDL, there is only one agent: the "computer" executing the program. As you can see, there are no agents mentioned in the language and we can not express interaction between different agents. For instance we can not express *cooperation* between several agents such that if agent 1 executes π_1 and if agent 2 executes π_2 they ensure a property φ whatever the program of agent 3 is.

6.2 Coalition Logic

In this section, we present the logic of Coalition [Pau02]. This logic is inspired from PDL in the sense that this logic speaks about the outcomes/effects of actions performed by agents. In this logic, no actions are mentioned but only group of agents. Coalition Logic provides a construction of the form $\langle\langle J\rangle\rangle\varphi$ meaning that "the group of agents J can ensure the property φ in the next state" or more precisely "agents of J have actions in their repertoire such that if they execute those actions then they ensure the property φ in the next time whatever the other agents do".

More precisely the language of Coalition logic is defined by the following rule:

$$\varphi \quad ::= \quad p \mid \neg\varphi \quad | \quad (\varphi \wedge \varphi) \quad | \quad \langle\langle J\rangle\rangle\varphi$$

As designed by Pauly [Pau02], semantics of Coalition Logic is in terms of neighborhood models, that is, models providing a *neighborhood function*, associating a world to a set of neighborhoods, or clusters. (See [Che80, Chap. 7] for details about those models.)

Definition 37 (effectivity function)
Given a nonempty set of states S, an *effectivity function* is a function $E : 2^{AGT} \to 2^{2^S}$. An effectivity function is said to be:

- *J-maximal* iff for all $X \subseteq S$, if $S \setminus X \notin E(AGT \setminus J)$ then $X \in E(J)$;

- *outcome monotonic* iff for all $X, X' \subseteq S$ and for all $J \subseteq AGT$, if $X \in E(J)$ and $X \subseteq X'$ then $X' \in E(J)$;
- *superadditive* iff for all J_1, J_2, if $J_1 \cap J_2 = \emptyset$ then for all $X_1, X_2 \subseteq S$, if $X_1 \in E(J_1)$ and $X_2 \in E(J_2)$ then $X_1 \cap X_2 \in E(J_1 \cup J_2)$.

The function E intuitively associates every coalition J to a set of subsets of S (or set of outcomes) for which J is effective. That is, J can force the world to be in some state of X, for each $X \in E(J)$.

Definition 38 (playable effectivity function)
Given a nonempty set of states S, an effectivity function $E : 2^{AGT} \to 2^{2^S}$ is said to be *playable* iff the following conditions hold:

1. for all J, $\emptyset \notin E(J)$ (Liveness)
2. for all J, $S \in E(J)$ (Termination)
3. E is AGT-maximal
4. E is outcome-monotonic
5. E is superadditive

A coalition model is a pair $((S, E), V)$ where:

- S is a nonempty set of states;
- $E : S \to (2^{AGT} \to 2^{2^S})$ associates every state s with a playable effectivity function $E(s)$.
- $V : S \to 2^{Prop}$ is a valuation function.

We will write $E_s(J)$ instead of $E(s)(J)$ to denote the effectivity of the group J at the state s.

Truth conditions are standard for Boolean operators. We evaluate the coalitional operators against a coalition model M and a state s as follows:

$$M, s \models \langle\langle J \rangle\rangle \varphi \text{ iff } \{t \mid M, t \models \varphi\} \in E_s(J)$$

Alur et al. propose a similar formalism called *Alternating-Time Logic* (ATL) in [AHK99] which is an extension of Coalition Logic plus time expressivity. The idea is namely the same than Coalition Logic except that ATL deals with long-term strategy.

6.3 Drawbacks of Coalition Logic

Compared to PDL, Coalition Logic enables us to speak about ability. Nevertheless Coalition Logic has some drawbacks.

6.3.1 Combining with epistemic logic: de dicto VS de re

When an agent make a plan in order to get a certain property φ he must take in account its own knowledge/belief about the world and about other agents' action. We can distinguish essentially three different situations mixing abilities and knowledge.

1. The agent has an action a in his repertoire to ensure φ but she does not know that she has this action a.

   Example 14 *Player Marwais playing chess and actually she has a strategy to perform a checkmate but as she is beginner in chess she does not know she can perform a checkmate.This situation can actually be represented in modal logic using a Coalition Logic operator and an epistemic operator:*

   $$\langle\langle Marwa\rangle\rangle chessmate \wedge \neg K_{Marwa}\langle\langle Marwa\rangle\rangle chessmate.$$

2. The agent knows that she has an action to ensure φ and she knows exactly which action she has to execute to ensure φ. This situation is called "*de re*" (of the thing): the agent has a specific action in mind.

   Example 15 *Imagine the situation where Marwa is near a lamp which is off and a button [BHT07a]. Suppose that Marwa knows that the button controls the lamp. Then Marwa knows that she has an action (toggling the button) in order to get the lamp on. Furthermore she knows that the action is "toggling".*

3. The agent knows that she has an action to ensure φ but she can not identify this action. This situation is called "de dicto" (of the word): the agent can give a word to this action but can not associate this word to a specific action ensuring φ.

   Example 16 *Imagine the situation where Marwa has the credit card of Bilal but she does not know its 4-digit pin code. In this situation, Marwa knows that she can have money from a cash machine. Actually she knows that she has an action in order to get money, that is to say to write the correct 4-digit pin code but she does not know which pin code she has to write.*

Example 17 *[HT06] Imagine the situation where Marwa is blind and is located into a room where the light is off. She is near a button enabling to switch the state of the light. Marwa can ensure the light to be on by pressing the button but as she does not know the state of the light she does not whether she has to toggle the button or not.*

Whatever you try to combine the epistemic operator and the coalition logic operator you will not be able to express the difference between the "de re" and the "de dicto" situations. The Coalition logic operator is not expressive enough.

6.3.2 Counterfactual emotions

As you can see in Chapter 10, formalization of regret is made up of two ingredients:

- the agent who regret φ now knows that φ is true;
- She also knows that she could have prevented φ, that is to say she would have an action a in her repertoire such that φ would be false if she would performed the action a (the actions of other agents are fixed).

We claim that Coalition Logic is not expressive enough to express counterfactual emotions like regret: the notion of "agent a could have prevented φ" is different from the Coalition Logic. Indeed "agent a could have prevented φ" requires to have an operator enabling to examine a change of action a by continuing to fix actions of other agents.

6.3.3 Solutions

Many logicians [JÅ07, Jon03, JvdH04, VOJ05, Sch04] have studied quite elegant adapted version of Coalition Logic in order to capture the notion of "de re" and "de dicto".

For instance in [JÅ07], the authors provide "ad hoc" non-standard operators:

- $\mathbb{K}_a\langle\langle a\rangle\rangle\varphi$: agent a knows that she can ensure φ and knows a specific strategy in order to get φ ("de re");
- $K_a\langle\langle a\rangle\rangle\varphi$: agent a knows that she can ensure φ but not necessarily she knows about a specific strategy ("de dicto").

Coalition logic is a non-normal logic in the sense that the semantics is not describe in the natural way of relations and standard truth conditions. As you can see in the truth conditions of $\langle\langle J\rangle\rangle\varphi$, the semantics can be reformulated as follows:

- there exists a set of points $A \in E_s(J)$ (this set corresponds to the choice of actions for all agents in J);
- such that for all state $t \in A$ we have $M, t \models \varphi$. (that is to say, whatever the other agents are doing, φ will be true)

In this thesis we study the point of view of [HT06]: we claim that we can model "de re" and "de dicto" principles and counterfactual emotions (see Chapter 10) with *standard* epistemic logic using only normal modal logic. This leads to the idea to decompose the $\langle\langle J\rangle\rangle$ operator into three normal operators in the same principle than in [GH93] and more precisely [BHT05]:

- a "diamond" operator $\Diamond$ to model the existential part of choosing an action for agents of J;
- a "box" operator $[J]$ in order to browse all actions of agents that are not in J.
- and finally a "next" operator X in order to model time.

In this formalism, the "de re" is formalized by

$$\Diamond K_a[J]X\varphi$$

and "de dicto" is formalized by

$$K_a\Diamond[J]X\varphi.$$

In this thesis, we will be interested about the operators $\Diamond$ and $[J]$ without time operators. Those operators are operators of the *sees-to-it-that* modal logic.

Broersen et al. showed that ATL can be embedded into a strategic version of Chellas STIT, by identifying $\langle\langle J\rangle\rangle X\varphi$ with $\langle\emptyset\rangle[J]X\varphi$ and $\langle\langle J\rangle\rangle(\varphi U\psi)$ with $\langle\emptyset\rangle[J](\varphi U\psi)$. [BHT06a]. This highlights that the modal operators of CL [Pau02] and ATL [AHK99] are nothing but fusions of three modal operators. STIT-logics are therefore the most general formal framework for agency, allowing not only to reason about what agents *can do*, but also about what they *do*, contrary to CL and ATL.

6.4 The STIT logic

In philosophy of action constructions of the form $[i\ stit : \varphi]$ were introduced by Belnap et col. [BPX01], read "agent i sees to it that φ" or "i brings it about that φ". In this paper, we focus on the basic version that is called Chellas STIT

[Che92] (thus baptized by [HB95]), noted [*i cstit* : φ] in the literature. (The original operator defined by Chellas is nevertheless notably different since it does not come with the principle of independence of agents that plays a central role in STIT theory.) The Chellas STIT was extended to group agency in [BPX01, Section 10.C] and [Hor01b, Section 2.4]. For a set of agents J, the formula [J *cstit* : φ] reads "group J sees to it that φ". We here write $[J]\varphi$ instead of [J *stit* : φ]. These logics moreover have a modal operator of historical necessity that is identified with $[\emptyset]$.

We present two semantics to interpret formulas in the language of group STIT with the "next" operator. In this section, we recall the orginal semantics in terms of Belnap's *branching-time models* [BPX01]. In the next section, we will define an equivalent semantics closer to standard presentations of *Kripke models*. For that matter you can already find such a semantics in terms of Kripke models in [BHT08] and [HS08] for the STIT without temporal operators.

6.4.1 Syntax

Let ATM be an enumerable non empty set of atomic propositions, let n be a positive integer and let $AGT = \{1, \ldots n\}$ be a finite (non empty) set of agents. The language $\mathcal{L}_{\mathsf{XCSTIT}}$ of logic XCSTIT is defined by the following BNF:

$$\varphi \ ::= \ \bot \ \mid \ p \ \mid \ (\varphi \vee \varphi) \ \mid \ \neg\varphi \ \mid \ [J]\varphi \ \mid \ X\varphi$$

where $p \in ATM$ and J ranges over 2^{AGT}.

The construction $[J]\varphi$ is read "group J sees to it that φ is true". When $J = \emptyset$, the construction $[\emptyset]\varphi$ means that φ is historically necessary. The construction $X\varphi$ reads "φ will be true in the next moment". We define the following standard abbreviations: $\top =^{\text{def}} \neg\bot$, $(\varphi \wedge \psi) =^{\text{def}} \neg(\neg\varphi \wedge \neg\psi)$ and $\langle J\rangle\varphi =^{\text{def}} \neg[J]\neg\varphi$. We follow the standard rules for omission of parentheses.

Remark 5 *The version of STIT is called "Chellas' STIT". There exists another version of STIT called individual deliberative STIT providing a construction of the form [adstit:φ] meaning that "agent a sees to it that φ is true and φ is not necessarily true". Of course we can define this operator in our language by* $[adstit{:}\varphi] = [\{a\}]\varphi \wedge \neg[\emptyset]\varphi$. *For more details, see [BPX01], [Wan06].*

6.4.2 Traditional semantics with Branching time structure

Semantics is given to formulas of $\mathcal{L}_{\mathsf{XCSTIT}}$ in terms of a branching-time (BT) structure augmented by an agent choice (AC) function. Let us introduce first the STIT-branching time structure. As we deal with the "next" operator, we suppose in this paper the time to be discrete. Moreover, we suppose the time to be without endpoints.

6.4.2.1 STIT-branching time structure

Definition 39 (STIT-branching time structure)
[BPX01][p. 30] A discrete STIT-branching time structure without endpoints (BT-structure) is a tuple $(M, \leq)$ where:

- M is a non empty set of *moments*;
- $\leq$ is *tree-like* that is to say:
  1. for all $m \in M$, $m \leq m$ (reflexive);
  2. for all $m_1, m_2, m_3 \in M$, $m_1 \leq m_2$ and $m_2 \leq m_3$ implies $m_1 \leq m_3$ (transitive);
  3. for all $m_1, m_2 \in M$, $m_1 \leq m_2$ and $m_2 \leq m_1$ implies $m_1 = m_2$ (antisymmetric);
  4. for all $m_1, m_2, m_3 \in M$, $m_1 \leq m_3$ and $m_2 \leq m_3$ implies $m_1 \leq m_2$ or $m_2 \leq m_1$ (unique past);
  5. for all $m_2, m_3 \in M$, there exists $m_1 \in M$ such that $m_1 \leq m_2$ and $m_1 \leq m_3$ (historical connection);
  6. for all $m_1, m_2 \in M$, if $m_1 < m_2$ then there exists m_3 such that $m_1 < m_3$, $m_3 \leq m_2$ and there is no $m' \in M$ such that $m_1 < m' < m_3$ (discreteness);
  7. for all m_1, there exists m_2 such that $m_1 < m_2$ (seriality).

You can think of moments as states. $m_1 \leq m_2$ means that the state m_1 is before or equal to m_2. This relation is transitive (Item 2.) Item 3. says that if m_1 is before m_2 and m_2 before m_1 then m_1 and m_2 are the same moment. Items 4. and 5. entail a tree structure. Item 6. means that the relation is *discrete*, that is to say, given a moment m_1, we can speak about moments which strictly just after m_1. Item 7. says there is a future.

Definition 40 (history)

A *history* of a BT-structure $(M, \leq)$ is a maximal set of linearly ordered moments from M.

Notation 1 *The set of all histories of* $(M, \leq)$ *is noted* H^M *or simply* H.

Notation 2 *The set of all histories passing through* $m \in M$ *is noted* H_m^M *or simply* H_m. *Formally:* $H_m^M = \{h \in H^M \mid m \in h\}$.

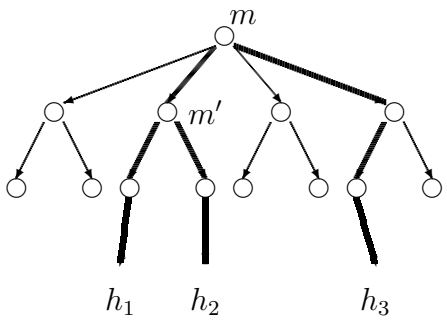


Figure 6.1: Undivided and divided histories

Definition 41 (undivided histories)
Let $(M, \leq)$ be a BT-structure. Let $m \in M$. Let $h_1, h_2 \in H_m$. We say that h_1 and h_2 are *undivided histories* at m iff there exists $m' \in M$ such that $m < m'$ and $m' \in h_1 \cap h_2$.

Example 18 *Let us consider the BT-structure of the Figure 6.1. The histories h_1 and h_2 are undivided at the moment m but they are divided at the moment m'. The histories h_1 and h_3 are divided at the moment m.*

Definition 42 (next moment)
Given $h \in H^m$, we define $next_h(m)$ as the smallest $m' \in h$ such that $m < m'$.

The function $next_h$ is total because of seriality and discreteness. Notice also that if h and h' are undivided at m then $next_h(m) = next_{h'}(m)$.

6.4.2.2 Adding choices

Now we introduce the model to interpret formulas of $\mathcal{L}_{\mathsf{XCSTIT}}$. It consists of a branching-time structure augmented by agents' choices at every moment and a valuation. Such structure are called BT+AC.

Definition 43 (CSTIT-branching time and choices model)
A CSTIT-branching time and choices model (BT+AC-model) is a tuple $\mathcal{M} = (M, \leq, C, V)$ where:

- $(M, \leq)$ is a BT-structure;
- $C : 2^{AGT} \times M \to 2^{H \times H}$ such that for all $m \in M$:
    1. For all $J \subseteq AGT$ and $m \in M$, $C_{J,m}$ is an equivalence relation over H_m;

2. $C_{\emptyset,}m = H_m \times H_m$;
3. For all $J \subseteq AGT$, $C_{J,m} = \bigcap_{j \in J} C_{\{j\},m}$;
4. For all $J \subseteq AGT$, for all $h, h' \in H_m$, if h and h' are undivided at m, then $hC_{J,m}h'$;
5. For all $(h_1, \ldots, h_n) \in H_m^n$, $\bigcap_{j \in AGT} C_{\{j\},m}(h_j) \neq \emptyset$.

- $V : M \times H \to 2^{ATM}$.

Time is branching due to choices that agents do. At each moment m, agents make choices which lead to different histories. The choices are represented in C: we classify histories passing through m according to the choices made by the agents. Informally, given two histories h and h' in H_m, we have $hC_{J,m}h'$ iff the choices of agents in the group J at the moment m are the same in h and h'. Obviously this informal intuition of $C_{J,m}$ justifies that $C_{J,m}$ is an equivalence relation over H_m (Item 1). The Item 2. is coherent with the informal intuition of $C_{J,m}$. Item 3. means that the choices of agents in the group J is made up of the choices of each individual agent. We call this property *additivity*. Item 4. corresponds to the property of *no choice between undivided histories*: agents cannot make a choice in the moment m between two histories which are undivided at m.

A formula is evaluated with respect to a model and a moment-history pair:

Definition 44 (truth conditions)
Given a BT+AC-model $\mathcal{M} = (M, \leq, C, V)$, a moment $m \in M$ and a history $h \in H_m$,

- $\mathcal{M}, m, h \models p$ iff $p \in V(m, h)$;
- $\mathcal{M}, m, h \models X\varphi$ iff $\mathcal{M}, next_h(m), h \models \varphi$;
- $\mathcal{M}, m, h \models [J]\varphi$ iff for all $h' \in C_{J,m}(h)$, we have $\mathcal{M}, m, h' \models \varphi$.

You can find a discussion in [BPX01] (p.31) about the fact that atomic propositions p are evaluated on a pair m/h and not only a moment. Broadly speaking, atomic propositions not only describe the physical state of a world supposed to be the same in all pairs m/h' where $h' \in H_m$. Atomic propositions can also represent choices such that "the agents are going to organize a party" etc. Actually this kind of propositions does depend on the history $h \in H_m$ under consideration and not just on the moment m.

As said before, in STIT theory, time is branching. But as we also consider an history h, the "next" operator is the "next" operator of *linear* temporal logic interpreted in the current history h while the operators $[J]$ are devoted to change the current history but they do not change the current moment.

As usual, a XCSTIT-formula φ is said to be CSTIT-satisfiable if we can find a BT+AC-model $\mathcal{M}$, a moment m, and a history h containing m such that $\mathcal{M}, m, h \models \varphi$. A XCSTIT-formula φ is said to be CSTIT-valid iff $\neg\varphi$ is not CSTIT-satisfiable.

6.5 A semantics with Kripke structures

In this section, we give an alternative semantics to BT+AC-model (Definition 43) in terms of Kripke structure. In other words, we extend the papers [HS08] and [BHT08] to STIT with the temporal "next" operator . The aim is provide a good framework to prove completeness and complexity results.

6.5.1 Definition

Definition 45 (XCSTIT-Kripke model)
A XCSTIT-Kripke model is a tuple $\mathcal{M} = (W, R_X, \{R_J\}_{J \subseteq AGT}, V)$ where:

- W is a non-empty set of *possible worlds*;
- $R_X : W \to W$ is a total function;
- for all $J \subseteq AGT$, $R_J \subseteq W \times W$ is an equivalence relation such that:
  1. $R_J \subseteq R_\emptyset$;
  2. $R_J = \bigcap_{j \in J} R_{\{j\}}$;
  3. for all $w \in W$, for all $w_1, \ldots w_n \in R_\emptyset(w)$, $\bigcap_{j \in AGT} R_{\{j\}}(w_j) \neq \emptyset$;
  4. $R_X \circ R_\emptyset \subseteq R_{AGT} \circ R_X$.
- $V : W \to 2^{ATM}$.

A world $w \in W$ corresponds to a couple moment/history in a BT-structure. Intuitively wR_Xu means that w and u share some same history and the moment of u is the next moment of the moment w in that history. Intuitively, R_J is nothing but the equivalence relation corresponding to the partition of $C_{\{j\},m}$ in Definition 43. wR_Ju means that w and u share the same moment and that their corresponding histories are in the same choice of the group J. $wR_\emptyset u$ simply means that w and u are in the same moment. Condition 2., called *additivity*, means that the choices of agents in the group J is made up of the choices of each individual agent and no more. We will see later that this condition 2. can be weakened (see Definition 50 and Definition 51). Condition 3. corresponds to the *independence of agents*. Figure 6.2 explains this property in the case of two agents ($AGT = \{1, 2\}$):

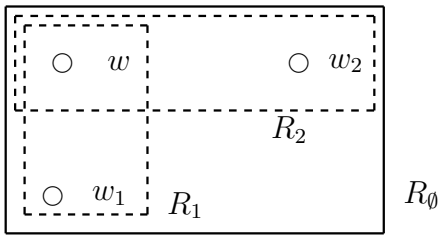


Figure 6.2: independence of agents

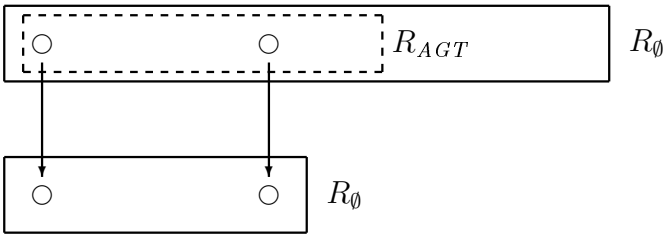


Figure 6.3: No Choice between undivided history

if you have two worlds w_1 and w_2 such that $w_1 R_\emptyset w_2$ then there exists a world $w \in R_1(w_1) \cap R_2(w_2)$. Condition 4. corresponds to the property of *no choice between undivided histories* (cf. Figure 6.3).

The truth condition is as usual in modal logic:

Definition 46 (Truth conditions)
Given a STIT-Kripke model $\mathcal{M} = (W, R_X, \{R_J\}_{J \subseteq AGT}, V)$,

- $\mathcal{M}, w \models p$ iff $p \in V(w)$;
- $\mathcal{M}, w \models X\varphi$ iff $\mathcal{M}, R_X(w) \models \varphi$;
- $\mathcal{M}, w \models [J]\varphi$ iff for all $w' \in W$, wR_Jw' implies $\mathcal{M}, w' \models \varphi$.

The operator X is the "next" operator of Linear Temporal Logic because the relation R_X is a function. The BT-structure has been blown up as you can see of Figures 6.5 and 6.4: a moment of a BT-structure is represented by a $R_\emptyset$-equivalence class of possible worlds in a STIT-Kripke model.

6.5.2 Equivalence

Now we prove the equivalence between the semantics of BT+AC-models and of Kripke model that is to say we obtain the same satisfiable formulas with the both

semantics. Let us consider the easy half: we construct a Kripke model from a BT+AC-structure.

Theorem 19 *Let $\varphi \in \mathcal{L}_{\mathsf{CSTIT}}$. If φ is satisfiable in a BT+AC-model then φ is satisfiable in XCSTIT-Kripke model.*

Proof.

We transform a BT+AC-model (Figure 6.4) into a XCSTIT-Kripke model (see Figure 6.5). Worlds of the CSTIT-Kripke model will be moment/history pairs of the BT+AC-model.

Let $\mathcal{M} = (M, \leq, C, V)$ be a BT+AC-model, $m \in M$ and $h \in H_m$ such that $\mathcal{M}, m, h \models \varphi$. Now we define $\mathcal{M}' = (W, R_X, \{R_J\}_{J \subseteq AGT}, V')$ where:

- $W = \{(m, h) \in M \times H^{\mathcal{M}} \mid m \in M \text{ and } h \in H_m\}$;
- $\begin{array}{rccc} R_X : & W & \to & W \\ & (m,h) & \mapsto & (next_h(m), h) \end{array}$
- $R_J = \{\langle (m,h), (m,h') \rangle \mid m \in M, h, h' \in H_m \text{ and } h' \in C_{J,m}(h)\}$;
- $V' = V$.

Now we prove that $\mathcal{M}'$ is a CSTIT-Kripke model and that $\mathcal{M}', (m, h) \models \varphi$.

Proof of "$\mathcal{M}'$ is a CSTIT-Kripke model"

We check that all properties required by the Definition 45 are true.

Let us prove the condition $R_X \circ R_\emptyset \subseteq R_{AGT} \circ R_X$ (Item 4). Let (m, h) and (m', h') be such that

$$(m,h) R_X (next_h(m), h) \text{ and } (next_h(m), h) R_\emptyset (m', h').$$

By definition of $R_\emptyset$, $m' = next_h(m)$. Now let us see that h and h' are undivided histories in m (Definition 41): indeed we have $m < m'$, $m' \in h$ and $m' \in h'$ (because $h' \in H_{m'}$). Furthermore, $m' = next_h(m) = next_{h'}(m)$. So by Definition 43, $h C_{AGT,m} h'$. To sum up we have:

$$(m,h) R_{AGT} (m, h') \text{ and } (m, h') R_X (m', h').$$

We leave it to the reader to check the other conditions of Definition 45.

Proof of $\mathcal{M}', (m, h) \models \varphi$

We prove by induction on ψ the property $\mathcal{P}(\psi) =$ "for all $(m, h) \in W$, $\mathcal{M}', (m, h) \models \psi$ iff $\mathcal{M}, m, h \models \psi$".

(Propositions) $\mathcal{M}', (m, h) \models p$ iff $p \in V'(m, h)$ iff $p \in V(m, h)$ iff $\mathcal{M}, m, h \models p$.

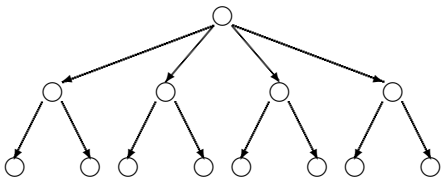


Figure 6.4: A BT-structure

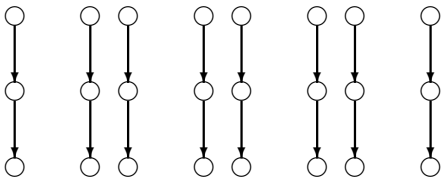


Figure 6.5: A CSTIT-Kripke model

(Booleans) We leave the cases of Boolean operators to the reader.

$(X\theta)$ $\mathcal{M}', (m,h) \models X\theta$ iff $\mathcal{M}', (next_h(m), h) \models \theta$
iff $\mathcal{M}, next_h(m), h \models \theta$
iff $\mathcal{M}, m, h \models X\theta$.

$([J]\theta)$ $\mathcal{M}', (m,h) \models [J]\theta$ iff for all $(m,h') \in R_J(m,h)$, $\mathcal{M}', (m,h') \models \theta$
iff for all $h' \in C_{J,m}(h)$, $\mathcal{M}', (m,h') \models \theta$
iff for all $h' \in C_{J,m}(h)$, $\mathcal{M}, m, h' \models \theta$
iff $\mathcal{M}, m, h \models [J]\theta$.

■

The other direction, that is to say to extract a BT+AC-model from a XCSTIT-Kripke model is much more difficult. The proof of that point is similar to the proof in [HL10]. Before starting the proof of the previous theorem, let us consider the example of Figure 6.6 to see where the difficulties are.

In this model, there is one $R_\emptyset$-equivalence class made up of two worlds w and u. The relation R_X links w to u and u to w.

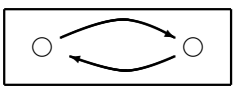


Figure 6.6: A XCSTIT-Kripke structure with two worlds

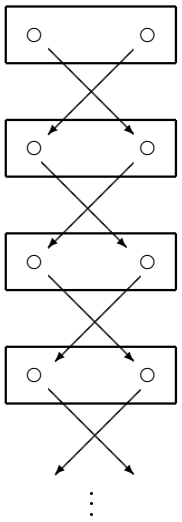


Figure 6.7: An unraveled CSTIT-Kripke model

Broadly speaking, the relation $R_\emptyset$ corresponds to changing history passing through the current moment. The relation R_X corresponds to going to the next moment in the current history. A moment of a BT-structure corresponds to a $R_\emptyset$-equivalence class. As here there is only one equivalence class, the relation R_X is such that the next moment is equal to the current moment. This contradicts the property of discreteness.

Solution 1. We have to unravel the CSTIT-Kripke model to get a legal BT+AC-model.

Now let us have a look at the unraveling of the model of Figure 6.6 that is depicted in Figure 6.7. We have here a curious contradiction:

- On the one hand, there is only one history;
- On the other hand, let us consider one $R_\emptyset$-class (supposed to represent a moment m). It contains two worlds so that there are two histories passing through m.

Our aim is to give a solution to avoid this contradiction. Given a formula φ, we are only interested in worlds with R_X-distance from the root at most the X-modal depth of φ. The X-modal depth of φ is the maximal number of nested X-operators in the formula φ.

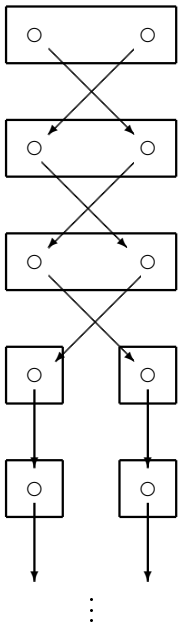


Figure 6.8: An unraveled and corrected CSTIT-Kripke model

Solution 2. For instance, if we deal with a formula of X-modal depth 3 (as $Xp \wedge X(p \vee [J](Xq \vee Xr))$, we correct the unraveled model to have singleton $R_\emptyset$-equivalence classes when the X-depth is greater than 3 as depicted in Figure 6.8 in order to have as many histories as there are points in $R_\emptyset$-equivalence classes.

Theorem 20 *Let $\varphi \in \mathcal{L}_{CSTIT}$. φ is satisfiable in CSTIT-Kripke models implies φ is satisfiable in CSTIT-Kripke models where the relation R_X is injective implies φ is satisfiable in BT+AC-models.*

Proof.

We have to prove that we can transform a CSTIT-Kripke model into a BT+AC-model. We do so in three steps:

1. First we unravel the CSTIT-Kripke model. This unraveling is done as usual up to depth N (cf. Solution 1) then we simply have singleton equivalence classes (cf Solution 2). This ensures that the resulting BT+AC-model has as many histories as there are points in the first equivalence class $R_\emptyset(w)$;

2. Finally we transform the unraveled CSTIT-Kripke frame into BT+AC-model: moments are $R_\emptyset$-equivalence classes.

Let us do it.

1. Let $\mathcal{M} = (W, R_X, \{R_J\}_{J \subseteq AGT}, V)$ be a CSTIT-Kripke frame and $w \in W$ such that $\mathcal{M}, w \models \varphi$. Let N be the X-modal depth of φ.

   We define $\mathcal{M}' = (W', R'_X, \{R'_J\}_{J \subseteq AGT}, V')$ as follows:

   - W' is the set of all finite sequences $w_0 \dots w_n$ such that $w_0 \in R_\emptyset(w)$, $n \in \mathbb{N}$ and for all $i \in \{0, \dots, n-1\}$, $w_i R_X w_{i+1}$;
   - $w_0 \dots w_n R'_X w_0 \dots w_{n+1}$;
   - $w_0 \dots w_n R'_J v_0 \dots v_m$ iff $n = m$ and
     $$\begin{cases} w_i R_{AGT} v_i \text{ for all } i < n \text{ and } w_n R_J v_n & \text{if } n \leq N \\ w_0 \dots w_n = v_0 \dots v_n & \text{if } n > N; \end{cases}$$
   - $V'(w_0 \dots w_n) = V(w_n)$.

   Notice that the relation R'_X is injective. The structure $\mathcal{M}'$ is a CSTIT-Kripke frame. Here we only prove the item 4 of Definition 45: $R'_X \circ R'_\emptyset \subseteq R'_{AGT} \circ R'_X$. The other constraints are left to the reader.

   Let us consider $w_0 \dots w_n R'_X \circ R'_\emptyset v_0 \dots v_m$. By definition of R'_X and $R'_\emptyset$, $m = n+1$, there exists w_{n+1} such that $w_n R_X w_{n+1}$ and $w_{n+1} R_\emptyset v_{n+1}$ and for all $i < n+1$, $w_i R_{AGT} v_i$. Hence we have $w_0 \dots w_n R'_{AGT} v_0 \dots v_n$ and $v_0 \dots v_n R'_X v_0 \dots v_n v_{n+1}$.

   We can prove by induction on k that for all integers $k \in \{0, n\}$, for all formulas ψ of X-modal depth k, for all sequence $w_0 \dots w_k$ we have $\mathcal{M}, w_k \models \psi$ iff $\mathcal{M}', w_0 \dots w_k \models \psi$. So we have $\mathcal{M}', w \models \varphi$.

2. Let φ be a satisfiable in CSTIT-Kripke models and N be the X-modal depth φ. Let $\mathcal{M}' = (W', R'_X, \{R'_J\}_{J \subseteq AGT}, V')$ be a CSTIT-Kripke model, $w \in W$ such that $\mathcal{M}', w \models \varphi$, for all $u \in R_\emptyset(w)$, for all $i > N$, $R'_\emptyset({R'_X}^i(u)) = \{{R'_X}^i(u)\}$ and such that R'_X is injective. (like the model $\mathcal{M}'$ created in step 1.) As R'_X is injective, we use the notation ${R'_X}^-$ to denote the converse of R'_X in the sequel. We now define a BT+AC-model $\mathcal{M} = (M, \leq, C, V)$ as follows.

   First we define the set of moments: $M = \{R_\emptyset(u) \mid u \in W\}$. The relation $\leq$ is defined as follows: for all $m, m' \in M$, $m \leq m'$ iff there exists $u \in m$ and $u' \in m'$ such that $u {R'_X}^* u'$ where ${R'_X}^*$ denotes the reflexive and transitive closure of R'_X;

   From now on, we introduce $\mathtt{w}(m, h)$ denoting the world in W' corresponding to the moment $m \in M$ and the history $h \in H_m$. Formally for all $m \in M$ and $h \in H_m$, the moment $next_h^N(m)$ (which is also an $R'_\emptyset$-equivalence class) is a singleton $\{x_{h,m}\}$ (because R'_X is injective). We define $\mathtt{w}(m, h) =^{\text{def}}$

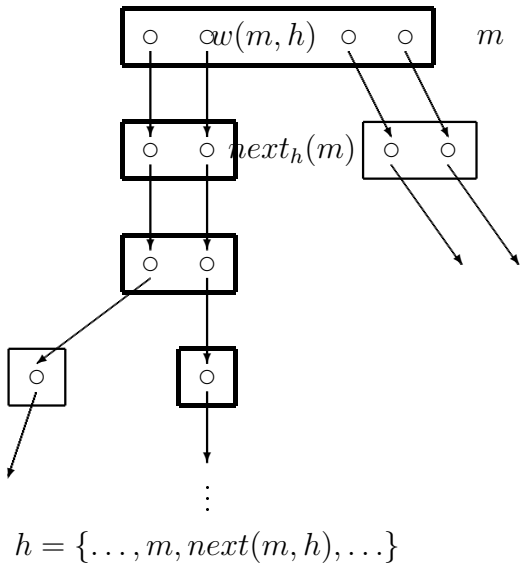


Figure 6.9: A BT-structure from a Kripke model: moments are $R_\emptyset$-equivalence classes.

$R_X^{-N}(x_{h,m})$. Of course $\mathtt{w}(m,h) \in m$. We have there exists $u \in next_h^n(m)$ such that $\mathtt{w}(m,h)R'_X u$. Note that for all $w \in W$, we have $w = \mathtt{w}(m,h)$ where $m = R'_\emptyset(w)$ and $h = \{R'_\emptyset(v) \mid v \in {R'_X}^* \circ ({R'_X}^-)^*(w)\}$ where $({R'_X}^-)^*$ is the reflexive and transitive closure of the converse of R'_X. All these notions are depicted in Figure 6.9.

Finally we define $C_{J,m}$ and V:

- For all $J \in 2^{AGT}$, for all $h, h' \in H_m$, $hC_{J,m}h'$ iff $\mathtt{w}(m,h)R'_J\mathtt{w}(m,h')$;
- $V(m,h) = V'(\mathtt{w}(m,h))$.

Now we have to prove two things:

(a) $\mathcal{M}'$ is a BT+AC-model;

(b) $\mathcal{M}', R'_\emptyset(w), \mathtt{w}(R'_\emptyset(w), \{R'_\emptyset(v) \mid v \in {R'_{X^\pm}}^*(w)\}) \models \varphi$.

For (a) we just check the different properties of Definition 43.

Let us prove that $C_{\emptyset,m} = H_m \times H_m$. Let $h, h' \in H_m$. $\mathtt{w}(m,h)$ and $\mathtt{w}(m,h')$ are both in m which is a $R_\emptyset$-equivalence class. In other words, $\mathtt{w}(m,h)R_\emptyset\mathtt{w}(m,h')$. So $hC_{\emptyset,m}h'$.

Let us prove that for all $J \in 2^{AGT}$ and $h, h' \in H_m$, if h and h' are undivided at m, then $hC_{J,m}h'$. If h and h' are undivided at m, then $next_h(m) =$

$next_{h'}(m) = m'$. There exists u, v such that $u, v \in m'$ such that $\mathtt{w}(m,h)R_X u$ and $\mathtt{w}(m,h')R_X v$. We have $\mathtt{w}(m,h)R_X \circ R_\emptyset v$. As $R_X \circ R_\emptyset \subseteq R_{AGT} \circ R_X$, there exists $z \in W$ such that $\mathtt{w}(m,h)R_{AGT}z$ and $zR_X v$. As R_X is injective, we have $\mathtt{w}(m,h') = z$. Hence, $\mathtt{w}(m,h)R_{AGT}\mathtt{w}(m,h')$. Hence $\mathtt{w}(m,h)R_J\mathtt{w}(m,h')$ that is to say $hC_{J,m}h'$.

Let us prove that for all $h_1, \ldots, h_n \in H_m$, $\bigcap_{j \in AGT} C_{\{j\},m}(h_j) \neq \emptyset$.

For all $h_1, \ldots, h_n \in H_m$, we have $\bigcap_{j \in AGT} R_{\{j\}}(\mathtt{w}(m,h_j)) \neq \emptyset$. In other worlds, it contains a world $w = \mathtt{w}(m,h)$ where $h = \{R'_\emptyset(v) \mid v \in {R'_X}^* \circ ({R'_X}^-)^*(w)\}$. We have finally $h \in \bigcap_{j \in AGT} C_{\{j\},m}(h_j)$.

For (b) we prove by induction on ψ the property

$$\mathcal{P}(\psi) = \text{“for all } u \in W,\ \mathcal{M}', \mathtt{w}(m,h) \models \psi \text{ iff } \mathcal{M}, m, h \models \psi\text{”}$$

(Propositions) Left to the reader.

(Boolean cases) Left to the reader.

$(X\theta)$ $\mathcal{M}', \mathtt{w}(m,h) \models X\theta$ iff $\mathcal{M}', R'_X(\mathtt{w}(m,h)) \models \theta$
iff $\mathcal{M}, next_h(m), h \models \theta$
iff $\mathcal{M}, m, h \models X\theta$.

$([J]\theta)$ $\mathcal{M}', \mathtt{w}(m,h) \models [J]\theta$ iff for $x \in R_J(\mathtt{w}(m,h))$, $\mathcal{M}', x \models \theta$
iff for all $h' \in H_m$,
$\mathtt{w}(m,h)R_J\mathtt{w}(m,h')$ implies $\mathcal{M}', \mathtt{w}(m,h') \models \theta$
iff for all $h' \in H_m$,
$hC_{J,m}h'$ implies $\mathcal{M}, m, h' \models \theta$
iff $\mathcal{M}, m, h \models [J]\theta$.

■

6.6 Conclusion

In this Chapter we have considered some logics of the literature dealing with actions, cooperations etc. like PDL, Coalition Logic, ATL, etc. We have seen that concerning emotions, “de dicto” and “de re” principles, those logics are not adapted. The main reason is because we need not only to express the concept of capabilities but also to express what agents actually *do*.

The logic STIT is suitable to express what agents actually *do* because it provides a modal construction of the form $[J]\varphi$ saying that the coalition J sees to it that φ is true.

We have given a semantics of group STIT with “next operator” in terms of Kripke structure. This is interesting for researchers who want to add other modal

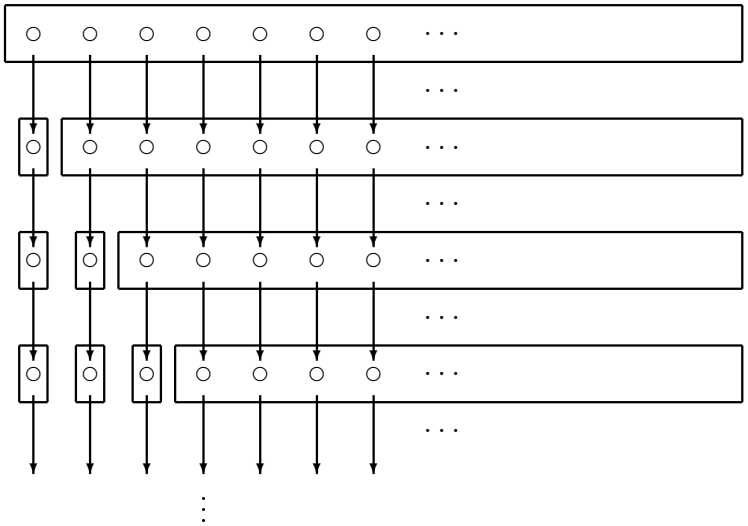


Figure 6.10: A Kripke structure

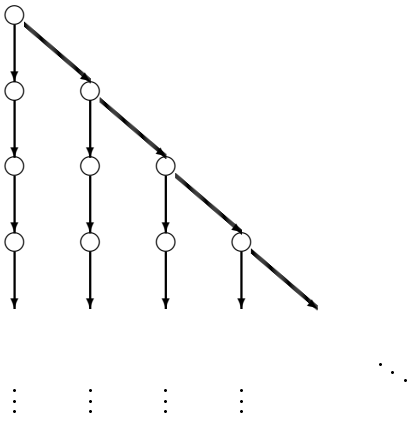


Figure 6.11: A BT-structure

operators to STIT like "knowledge", "obligations", "intentions" etc. and will make it easy to give the corresponding semantics.

One major perspective is to study STIT with the time operator "eventually". With this kind of operator, there are some difficulties about finding a Kripke semantics and proving its equivalence with classical branching time structure. Indeed, one major problem is that we do not have a control over the depth in the model. With an operator "eventually", we maybe do not only deal with worlds of depth at most n as we have done with the X-modal depth of the formula is n in Subsection 6.5.2. This fact leads to a problem called the problem of *hidden histories* (the "bundled model" in [Rey05]). We are going to explain this problem on an example shown on Figure 6.10 and Figure 6.11. Figure 6.10 shows a Kripke structure. All "histories" are such that after a finite number of steps there is no more branching. When you try to construct the corresponding branching time structure you obtain Figure 6.11. In this branching time, it appears a "new" history (in bold) in which there is always branching possible.

Chapter 7

Satisfiability problem and axiomatization of fragments of STIT

In this Chapter we are going to study the satisfiability problem of a given STIT-formula. We will also deal with axiomatization. We first begin to study the case where formulas does not contain any temporal operator.

7.1 Forget time for a while

In this section, we investigate some fragments of CSTIT without the "next" operator. We recall that the whole group STIT without temporal operators, called $\mathcal{L}_{[\{J\}]}$, that is to say the logic XCSTIT restricted to this fragment defined by the following rule

$$\varphi \ ::= \ p \ \mid \ (\varphi \vee \varphi) \ \mid \ \neg\varphi \ \mid \ [J]\varphi$$

where p ranges over ATM and J ranges over 2^{AGT}.

The semantics is given by STIT-Kripke model (Definition 45) but the relation for the time is useless:

Definition 47 (STIT-Kripke model)
A STIT-Kripke model $\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ is a tuple where:

- W is a set of worlds;
- for all $J \subseteq AGT$, R_J is a equivalence relation such that:
    1. $R_J \subseteq R_\emptyset$;

2. $R_J = \bigcap_{j \in J} R_{\{j\}}$;
3. for all $w \in W$, for all $(w_j)_{j \in AGT} \in R_\emptyset(w)^n$, $\bigcap_{j \in AGT} R_{\{j\}}(w_j) \neq \emptyset$;

- $V : W \to 2^{ATM}$.

7.1.1 Group STIT

In this section, we study the satisfiability problem and the axiomatization of group STIT. You can also refer to [HS08]. We are going to reduce the problem of satisfiability in $S5^n$ to the problem of satisfiability in STIT^G_n. The range of our translation is the set of formulas where only "anti-individuals" occur, i.e. groups J such that $J = AGT \setminus \{i\}$. Noting $\bar{i}$ such a set, we define the following translation:

Definition 48 (translation from $\mathcal{L}_{S5^n}$ to $\mathcal{L}_{\mathsf{STIT}^G_n}$)
Let $tr : \mathcal{L}_{S5^n} \longrightarrow \mathcal{L}_{\mathsf{STIT}^G_n}$ be defined by:
$tr(p) = p$
$tr(\neg\varphi) = \neg tr(\varphi)$
$tr(\varphi \wedge \psi) = tr(\varphi) \wedge tr(\psi)$
$tr(\Box_i \varphi) = [\bar{i}]tr(\varphi)$, where $\bar{i} = AGT \setminus \{i\}$.

From now, we note φ instead of $tr(\varphi)$.

Theorem 21 *For any $\varphi \in \mathcal{L}_{S5^n}$, those propositions are equivalent:*

- $\models_{S5^n} \varphi$*;*
- $\models_{\mathcal{C}} \varphi$ *where $\mathcal{C}$ is the class of $STIT^G_n$ models where $R_{AGT} = id_W$;*
- $\models_{STIT^G_n} [\emptyset](\bigwedge_{p \in atm(\varphi)} [AGT]p \leftrightarrow p) \to \varphi$ *where $atm(\varphi)$ be the set of all atomic propositions occurring in φ.*

Notice that $[\emptyset]$ has to be read $[\overline{1}] \dots [\overline{n}]$.
PROOF.
We are going to prove that the following statements are equivalent:

1. φ is $S5^n$-satisfiable ;
2. φ is satisfiable in a STIT^G_n-model where $R_{AGT} = id_W$;
3. $[\emptyset](\bigwedge_{p \in atm(\varphi)} [AGT]p \leftrightarrow p) \wedge \varphi$ is STIT^G_n-satisfiable.

$\boxed{1. \Rightarrow 2.}$ Let $\varphi \in \mathcal{L}_{S5^n}$ such that it exists a $S5^n$-model $\mathcal{M} = \langle X, R, V \rangle$ where:

- $X = X_1 \times X_2 \times \dots X_n$;

- R is mapping associating to every $i \in AGT$ the equivalence relation R_i defined by $R_i = \{((x_1, ...x_n), (y_1, ...y_n)) \in X^2$ where for all $j \neq i, x_j = y_j\}$

and a point $x \in X$ such that $\mathcal{M}, x \models \varphi$.

Now, we are going to define a triple $\mathcal{M}' = \langle W', R', V' \rangle$ as follows:

- $W' = X$;
- R' is a mapping associating to every $i \in AGT$ the equivalence relation $R'_i = \{\langle (x_1, ...x_n), (y_1, ...y_n)\rangle \in W'^2$ where $x_i = y_i\}$ on W';
- $V' = V$.

We can check that for all $(x_1, x_2, \ldots, x_n) \in W^n, \bigcap_{i \in AGT} R_i(x_i) \neq \emptyset$. Thus, $\mathcal{M}'$ is a STIT-Kripke structure. We can see that

$$\begin{array}{rcl} R'_{\bar{i}} & = & \bigcap_{j \in \bar{i}} R'_j \text{ (see definition 47)} \\ & = & \bigcap_{j \in \bar{i}} \{\langle (x_1, ...x_n), (y_1, ...y_n)\rangle \in W'^2 \text{ where } x_j = y_j\} \\ & = & \{\langle (x_1, ...x_n), (y_1, ...y_n)\rangle \in W^2 \text{ where for all } j \neq i, x_j = y_j\} \\ & = & R_i \end{array}$$

and that

$$\begin{array}{rcl} R'_{AGT} & = & \bigcap_{j \in AGT} R'_j \text{ (see definition 47)} \\ & = & \bigcap_{j \in AGT} \{\langle (x_1, ...x_n), (y_1, ...y_n)\rangle \in W'^2 \text{ where } x_j = y_j\} \\ & = & \{\langle (x_1, ...x_n), (x_1, ...x_n)\rangle \in W^2\} \\ & = & id_W \end{array}.$$

We can check that for all $z \in W, \mathcal{M}, z \models \varphi$ iff $\mathcal{M}', z \models \varphi$ by induction on φ.

2. $\Rightarrow$ 3.

Let $\varphi \in \mathcal{L}_{S5^n}$ such that there exists a STIT^G_n-model $\mathcal{M} = \langle W, R, V\rangle$ and a point $w \in W$ with $R_{AGT} = id_W$ such that $\mathcal{M}, w \models \varphi$. As $R_{AGT} = id_W$, we have $\mathcal{M}, w \models [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p)$. Thus $\mathcal{M}, w \models [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \wedge \varphi$.

3. $\Rightarrow$ 1.

Let $\varphi \in \mathcal{L}_{S5^n}$ such that $[\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \wedge \varphi$ is satisfiable in STIT^G_n.

Lemma 5 *There exists a STIT^G_n-model $\mathcal{M} = \langle W, R, V\rangle$ and a point $w \in W$ with $R_{AGT} = id_W$ such that $\mathcal{M}, w \models \varphi$.*

Proof.

As $[\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \wedge \varphi$ is satisfiable in STIT^G_n, there exists a STIT^G_n-model $\mathcal{M}' = \langle W', R', V'\rangle$ and a point $w \in W'$ such that $\mathcal{M}', w \models [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \wedge \varphi$. Now, we are going to define a STIT^G_n-model $\mathcal{M} = \langle W, R, V\rangle$ as follows:

- $W = \{R'_{AGT}(x) \mid x \in W'\}$;
- $R_i = \{(R'_{AGT}(x), R'_{AGT}(y)) \mid (x, y) \in R'_i\}$;

- $V(p) = \{U \in W \mid U \subseteq V'(p)\}$.

Notice than $R_J = \{(R'_{AGT}(x), R'_{AGT}(y)) \mid (x,y) \in R'_J\}$. We can check that $R_{AGT} = id_W$. We can check that for all $z \in W'$, for all subformulas ψ of φ that

$$\mathcal{M}', z \models \psi \text{ iff } \mathcal{M}, R'_{AGT}(z) \models \psi$$

$$\begin{array}{lll} \mathcal{M}', z \models p & \text{iff} & \mathcal{M}', z \models [AGT]p \text{ (because } \mathcal{M}', z \models [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p)) \\ & \text{iff} & \mathcal{M}', y \models p \text{ for all } y \in R'_{AGT}(z) \\ & \text{iff} & y \in V'(p) \text{ for all } y \in R'_{AGT}(z) \\ & \text{iff} & R'_{AGT}(z) \subseteq V'(p) \\ & \text{iff} & R'_{AGT}(z) \in V(p) \\ & \text{iff} & \mathcal{M}, R'_{AGT}(z) \models p \end{array}$$

$$\begin{array}{lll} \mathcal{M}', z \models [\bar{i}]\psi & \text{iff} & \mathcal{M}', y \models \psi \text{ for all } y \in R'_{\bar{i}}(z) \\ & \text{iff} & \mathcal{M}, R'_{AGT}(y) \models \psi \text{ for all } y \in R'_{\bar{i}}(z) \\ & \text{iff} & \mathcal{M}, R'_{AGT}(y) \models \psi \text{ for all } R'_{AGT}(y) \in R_{\bar{i}}(R'_{AGT}(z)) \\ & \text{iff} & \mathcal{M}, R'_{AGT}(z) \models [\bar{i}]\psi \end{array}$$

■

Let be $\mathcal{M} = \langle W, R, V\rangle$ a STIT^G_n-model and a point $w \in W$ with $R_{AGT} = id_W$ such that $\mathcal{M}, w \models \varphi$. We define the $S5^n$-model $\mathcal{M}' = \langle X', R', V'\rangle$ as follows:

- $X' = X_1 \times X_2 \times \ldots X_n$ where for all $i \in AGT$, $X_i = \{R_i(x), x \in W\}$;
- R' is a mapping associating to every $i \in AGT$ the equivalence relation $R'_i = \{((c_1, ...c_n), (d_1, ...d_n)) \in X'^2$ where for all $j \neq i, c_j = d_j\}$ on X';
- $V'(p) = \{(c_1, \ldots c_n) \mid \bigcap_{i \in AGT} c_i \in V(p)\}$. (notice that we confuse $\bigcap_{i \in AGT} c_i = \{y\}$ and y)

We can check that for all $z \in W$,

$$\mathcal{M}', (R_1(z), R_2(z), \ldots, R_n(z)) \models \varphi \text{ iff } \mathcal{M}, z \models tr(\varphi)$$

.

$$\begin{array}{lll} \mathcal{M}', (R_1(z), R_2(z), \ldots, R_n(z)) \models p & \text{iff} & (R_1(z), R_2(z), \ldots, R_n(z)) \in V'(p) \\ & \text{iff} & \bigcap_{i \in AGT} R_i(z) \in V(p) \\ & \text{iff} & z \in V(p) \text{ (notice that } \bigcap_{i \in AGT} R_i(z) = \{z\}) \\ & \text{iff} & \mathcal{M}, z \models p \end{array}$$

$\mathcal{M}', (R_1(z), R_2(z), \ldots, R_n(z)) \models \Box_i \psi$
iff $\mathcal{M}', (R_1(z), R_2(z), \ldots, R_i(y), \ldots R_n(z)) \models \psi$ for all $y \in W$

iff $\mathcal{M}', (R_1(x), R_2(x), \ldots, R_i(x), \ldots R_n(x)) \models \psi$
where $R_1(z) \cap R_2(z) \cap \ldots R_i(y) \ldots R_n(z) = \{x\}$ for all $y \in W$

iff $\mathcal{M}', (R_1(x), R_2(x), \ldots, R_i(x), \ldots R_n(x)) \models \psi$ for all $x \in R_{\bar{i}}(z)$

iff $\mathcal{M}, x \models \psi$ for all $x \in R_{\bar{i}}(z)$

iff $\mathcal{M}, z \models [\bar{i}]\psi$

■

7.1.1.1 Group STIT is undecidable

Theorem 22 *If $n \geq 3$, the problem of satisfiability of a formula of $STIT_n^G$ is undecidable.*

Proof.
By theorem 8 and 21. ■

7.1.1.2 Group STIT is non-axiomatizable

Theorem 23 *If $n \leq 3$, the logic $STIT_n^G$ is not finitely axiomatizable.*

Proof.
Suppose for a contradiction that STIT_n^G is finitely axiomatizable. There exists a finite set of axioms Ax such that for all STIT_n^G-formula φ, we have $\models_{\mathsf{STIT}_n^G} \varphi$ iff $Ax \vdash \varphi$. Let us define an axiomatics Ax' obtained from Ax by removing $[AGT]$ symbols. We are going to prove that for all formulas $\varphi \in \mathcal{L}_{S5^n}$:

$$\vdash_{Ax'} \varphi \text{ iff } \models_{S5^n} \varphi$$

.
Hence, $S5^n$ would be axiomatizable and there is a contradiction.
First:

Lemma 6 *$\vdash_{Ax'} \varphi$ implies $\models_{S5^n} \varphi$.*

Proof.
We are going to prove that each instance of Ax' is valid in $S5^n$. Let us consider an instance ψ' of an axiom of Ax'. ψ' is obtained from an instance ψ of Ax by removing $[AGT]$ symbols. We have $\models_{\mathsf{STIT}_n^G} \psi$. Therefore, ψ is valid in the class of

STIT^G_n-models where $R_{AGT} = id_W$. Hence, ψ' is valid in the class of STIT^G_n-model where $R_{AGT} = id_W$. Hence $\models_{S5^n} \varphi$. ■

Here is an outline of the $\boxed{\Leftarrow}$-sense proof. For all $S5^n$-formulas φ,

$\models_{S5^n} \varphi$	iff	$\models_{\mathsf{STIT}^G_n} [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \rightarrow \varphi$
	iff	$\vdash_{Ax} [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \rightarrow \varphi$
	implies (1)	$\vdash_{Ax,[AGT]\psi \leftrightarrow \psi} \varphi$
	implies (2)	$\vdash_{Ax'} \varphi$

We just have to prove (1) and (2). They are provided by the two following lemmas:

Lemma 7 $\vdash_{Ax} [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \rightarrow \varphi$ *implies* $\vdash_{Ax,[AGT]\psi \leftrightarrow \psi} \varphi$.

PROOF.

We have:

$$\dfrac{\dfrac{\text{... (necessitation and principles of classical propositional logic)}}{\vdash_{Ax,[AGT]\psi \leftrightarrow \psi} [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p)} \qquad \dfrac{Hyp}{\vdash_{Ax,[AGT]\psi \leftrightarrow \psi} [\emptyset](\bigwedge_{p \in atm(\varphi)}[AGT]p \leftrightarrow p) \rightarrow \varphi}}{\vdash_{Ax,[AGT]\psi \leftrightarrow \psi} \varphi}$$

■

Lemma 8 $\vdash_{Ax,[AGT]\psi \leftrightarrow \psi} \varphi$ *implies* $\vdash_{Ax'} \varphi$.

PROOF.

Assume that $\vdash_{Ax,[AGT]\psi \leftrightarrow \psi} \varphi$. There exists a proof of φ, that is to say a sequence $(\varphi_1, \ldots, \varphi_k)$ such that for $1 \leq i \leq k$, one of the following holds:

- φ_i is a tautology, an instance of an axiom in Ax or an instance of $[AGT]\psi \leftrightarrow \psi$;
- φ_i is obtained by necessitation from φ_j where $j < i$;
- φ_i is obtained by modus ponens from φ_j and φ_k where $j, k < i$;
- $\varphi_k = \varphi$.

Now, we construct $(\varphi'_1, \ldots, \varphi'_n)$ where φ'_i is φ_i in which we have removed $[AGT]$ symbols. The reader can check that $(\varphi'_1, \ldots, \varphi'_n)$ is a proof of φ.

■

■

Unfortunately the set of formulas valid in STIT-Kripke models is not finitely axiomatizable and its satisfiability problem is undecidable when the number of agents is at least 3. Therefore we are interested in *syntactic restrictions* that are as expressible as possible while having good mathematical properties like good complexity of the satisfiability problem or axiomatizability.

First we add the grand coalition operator $[AGT]$ to the individual STIT studied in [Xu98] and [BHT08]. Then we study a generalisation where coalitions belong to the lattice of Figure 7.3. Finally, we study STIT where the coalitions form a chain $J_1 \subset J_2 \subset J_3 \ldots$.

7.1.2 Individual STIT plus the grand coalition without time

In [Xu98] and [BHT08], it was proved that the satisfiability problem of the atemporal individual STIT logic, that is to say the fragment of $\mathcal{L}_{[\{J\}]}$ where all $[J]$ operators are of the form $[\{i\}]$ where i ranges over AGT, is in NEXPTIME and that if a formula φ of $\mathcal{L}_{[\{i\}]}$is satisfiable then it is satisfiable in a a CSTIT-model where the number of worlds in bounded by $2^{|\varphi|}$. In [Wan06], you can also find a tableau method for the individual deliberative STIT (see Remark 5. In this subsection, we extend the results of [Xu98] , [BHT08] [Wan06] by adding the operator $[AGT]$ where AGT is the *grand coalition*.

In this subsection, we suppose that $card(AGT) \geq 2$.

7.1.2.1 Definition

We define now the fragment of the whole language $\mathcal{L}_{\mathsf{XCSTIT}}$ we are interested in, denoted as $\mathcal{L}_{[\{i\}],[AGT]}$, by the following rule:

$$\varphi \ ::= \ \bot \ | \ p \ | \ (\varphi \vee \varphi) \ | \ \neg\varphi \ | \ [\{i\}]\varphi \ | \ [AGT]\varphi$$

where $p \in ATM$ and $i \in AGT$.

7.1.2.2 Semantics

We recall the definition of a standard Kripke model but we only give the element correspond to the semantics of $[\{i\}]$ and $[AGT]\varphi$. The following Definition 49 is exactly the Definition 45 but restricted to operators $[\{i\}]$ and $[AGT]\varphi$.

Definition 49 ($\mathcal{L}_{[\{i\}],[AGT]}$-Kripke model)
A $\mathcal{L}_{[\{i\}],[AGT]}$-Kripke model $\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ is a tuple where:

- W is a set of worlds;
- for all $J \subseteq AGT$, R_J is a equivalence relation such that:
  1. $R_{\{j\}} \subseteq R_\emptyset$;
  2. $\boxed{R_{AGT} = \bigcap_{AGT \in J} R_{\{j\}};}$
  3. for all $w \in W$, for all $(w_j)_{j \in AGT} \in R_\emptyset(w)^n$, $\bigcap_{j \in AGT} R_{\{j\}}(w_j) \neq \emptyset$;

- $V : W \to 2^{ATM}$.

We also introduce a new class of models by weakening the condition of additivity (Condition 2 of Definition 49): the class of $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke models. In these models, we only force the inclusion $R_{AGT} \subseteq \bigcap_{j \in J} R_{\{j\}}$ but not the equality. In other words, *super-additivity* means that the choices of agents in the group J is *more* than the choices of each individual agent. This class of models will be helpful to provide axiomatization of all valid formulas of the fragment $\mathcal{L}_{[\{i\}],[AGT]}$. More precisely:

Definition 50 ($\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model)
A $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model $\mathcal{M} = (W, R_X, \{R_J\}_{J \subseteq AGT}, V)$ is a tuple where:

- W is a set of worlds;
- for all $J \subseteq AGT$, R_J is a equivalence relation such that:
  1. $R_{\{j\}} \subseteq R_\emptyset$;
  2. $\boxed{R_{AGT} \subseteq \bigcap_{j \in AGT} R_{\{j\}};}$
  3. for all $w \in W$, for all $(w_j)_{j \in AGT} \in R_\emptyset(w)^n$, $\bigcap_{j \in AGT} R_{\{j\}}(w_j) \neq \emptyset$;
- $V : W \to 2^{ATM}$.

We show that the language $\mathcal{L}_{[\{i\}],[AGT]}$ cannot distinguish the standard Kripke model and the super-additive variant. In other worlds, concerning the property of additivity and super-additivity the language $\mathcal{L}_{[\{i\}],[AGT]}$ is as poor as Coalition Logic [Pau02]: Pauly proved that all effectivity function is playable (hence superadditive) is the effectivity function of a strategic game (additive). Of course a $\mathcal{L}_{[\{i\}],[AGT]}$-Kripke model is a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive Kripke model. But the contrary is false. Given a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive Kripke model, we are going to transform it into a $\mathcal{L}_{[\{i\}],[AGT]}$-Kripke model using the technique of [Vak]. This technique consists in copyings worlds in the model without changing the truth of formulas. We also state the finite model property for both classes.

Theorem 24 *Let φ be a formula in the fragment $\mathcal{L}_{[\{i\}],[AGT]}$. We have equivalence between:*

1. *φ is satisfiable in a $\mathcal{L}_{[\{i\}],[AGT]}$-Kripke model;*
2. *φ is satisfiable in a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model;*
3. *φ is satisfiable in a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model with $2^{|\varphi|}$ worlds;*

4. φ is satisfiable in a $\mathcal{L}_{[\{i\}],[AGT]}$-Kripke model with $2^{(n+2)|\varphi|}$ worlds.

PROOF.

2. $\Rightarrow$ 3. Let φ be a satisfiable formula in a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model: there exists a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model $\mathcal{M}' = (W', R', V')$ and $w \in W'$ such that $\mathcal{M}', w \models \varphi$.

As usual, we filtrate by the set of subformulas of φ: two worlds are in the same equivalence class iff they satisfy the same subformulas of φ. Let $|w|$ be the equivalence class of w. Now we define the structure $\mathcal{M} = (W, R, V)$:

- $W = \{|w| \mid w \in W'\}$ is the set of equivalence classes;
- $|u| R_{\{i\}} |v|$ iff for all subformulas $[\{i\}]\psi$ of φ, $\mathcal{M}', u \models [\{i\}]\psi$ iff $\mathcal{M}', v \models [\{i\}]\psi$;
- $|u| R_{AGT} |v|$ iff for all $i \in AGT$, $|u| R_{\{i\}} |v|$ and for all subformulas $[AGT]\psi$ of φ, $\mathcal{M}', u \models [AGT]\psi$ iff $\mathcal{M}', v \models [AGT]\psi$;
- $V(|u|) = \{p \in ATM \mid p \text{ occurs in } \varphi \text{ and } p \in V(u)\}$.

The resulting model $\mathcal{M}$ is a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model with at most $2^{|\varphi|}$ worlds. We prove by induction on ψ that for all $\psi \in SF(\varphi)$, for all $u \in W'$ we have $\mathcal{M}', u \models \psi$ iff $\mathcal{M}, |u| \models \psi$. In particular we have $\mathcal{M}', w \models \varphi$ hence $\mathcal{M}, |w| \models \varphi$.

3. $\Rightarrow$ 4.

Let φ a $\mathcal{L}_{[\{i\}],[AGT]}$-formula, let $\mathcal{M} = (W, R, V)$ be a $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-Kripke model and $w \in W$such that $\mathcal{M}, w \models \varphi$ and $card(W) \leq 2^{|\varphi|}$.

Now we are going to transform the model $\mathcal{M}$ into a $\mathcal{L}_{[\{i\}],[AGT]}$-Kripke model $\mathcal{M}'$ with $2^{(n+2)|\varphi|}$ worlds. Figure 7.1 explains this transformation.

Let us consider:

- $\mathcal{C}_i$ denotes the set of R_i-equivalence classes in $\mathcal{M}$;
- $\vec{\mathcal{C}} = \prod_{i \in AGT} \mathcal{C}_i$;
- For all $\vec{C} \in \vec{\mathcal{C}}$, the set $\mathcal{A}_{\vec{C}}$ denotes the set of R_{AGT}-equivalence classes included in $\bigcap_{i \in AGT} C_i$;
- As $\mathcal{M}$ is finite, $\vec{\mathcal{C}}$ is finite and each set $\mathcal{A}_{\vec{C}}$ is finite. Moreover, the cardinality of $\mathcal{A}_{\vec{C}}$ is uniformly bounded by an integer l. Note that $l \leq 2^{|\varphi|}$. We write $\mathcal{A}_{\vec{C}} = \{A^0_{\vec{C}}, \ldots, A^{l-1}_{\vec{C}}\}$ (sequence may be with repetitions). In the example of Figure 7.1, $l = 3$.

We define $\mathcal{M}' = (W', R', V')$ as follows:

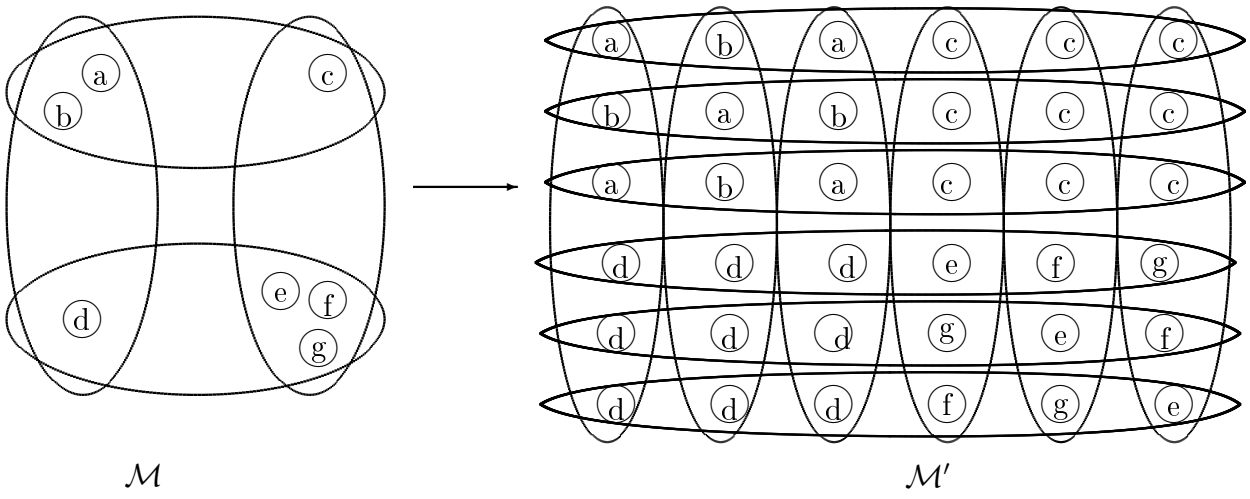


Figure 7.1: Proof of 3. $\Rightarrow$ 4.: "a", ..., "g" denote R_{AGT}-classes. The figure explains how these classes are duplicated from $\mathcal{M}$ to obtain $\mathcal{M}'$.

- $W' = \{(\vec{C}, i_1, \ldots, i_n, w) \mid \vec{C} \in \vec{\mathcal{C}}, (j_1, \ldots j_n) \in \{0, \ldots, l-1\}^n$ and $w \in A_{\vec{C}}^{\sum_{j \in AGT} i_j[l]}\}$ where the notation $x[l]$ stands for "x modulo l".
- $(\vec{C}, i_1, \ldots, i_n, w) R'_{\{j\}} (\vec{C'}, i'_1, \ldots, i'_n, w')$ iff $C_j = C'_j$ and $i'_j = i_j$;
- $(\vec{C}, i_1, \ldots, i_n, w) R'_{AGT} (\vec{C'}, i'_1, \ldots, i'_n, w')$ iff $\vec{C} = \vec{C'}$ and for all $j \in AGT$, $i'_j = i_j$;
- $V'((\vec{C}, i_1, \ldots, i_n, w)) = V(w)$.

Remark that we have $R'_{AGT} = \bigcap_{j \in AGT} R_{\{j\}}$. The map $f : (\vec{C}, i_1, \ldots, i_n, w) \mapsto w$ is a bounded-morphism [BDRV02, Definition 2.10]. Indeed:

1. for all $x \in W'$, $V'(x) = V(f(x))$;
2. $(\vec{C}, i_1, \ldots, i_n, w) R'_{\{j\}} (\vec{C'}, i'_1, \ldots, i'_n, w')$ implies $w R_{\{j\}} w'$ and
   $(\vec{C}, i_1, \ldots, i_n, w) R'_{AGT} (\vec{C'}, i'_1, \ldots, i'_n, w')$ implies $w R_{AGT} w'$;
3. If $w R_{\{j\}} w'$ then we have
   $(R_{\{1\}}(w), \ldots, R_{\{n\}}(w), i_1, \ldots, i_n, w) R'_{\{j\}} (R'_{\{1\}}(w'), \ldots, R'_{\{n\}}(w'), i'_1, \ldots, i'_n, w')$
   where $i_j = i'_j$ and other i_k's are such that $w \in A_{\vec{C}}^{\sum_{j \in AGT} i_j[l]}$ and $w' \in A_{\vec{C'}}^{\sum_{j \in AGT} i'_j[l]}$;

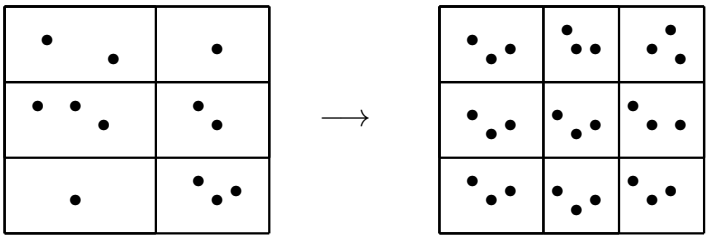


Figure 7.2: Transforming a finite STIT-model into a model where the number of worlds in each R_{AGT}-classes and the number of $R_{\{i\}}$-class for all $i \in AGT$ are equal to N ($N = 3$ here).

4. If $wR_{AGT}w'$ then we have

   $(R_{\{1\}}(w), \ldots, R_{\{n\}}(w), i, 0 \ldots, 0, w) R'_{AGT} (R_{\{1\}}(w'), \ldots, R_{\{n\}}(w'), i, 0, \ldots, 0, w')$
   where i is such that $w, w' \in A^i_{\vec{C}}$.

As f is a *bounded morphism*, for all $x \in W'$, for all formulas ψ, $\mathcal{M}', x \models \psi$ iff $\mathcal{M}, f(x) \models \psi$. Moreover the size of W' is bounded by $2^{|\varphi|} \times 2^{|\varphi|^n} \times 2^{|\varphi|} = 2^{(n+2)|\varphi|}$.
■

The above Theorem 24 allows us to give complexity result and axiomatization. Now we give a result helping for the generalization in the next subsection:

Theorem 25 *If a formula is satisfiable in STIT-model, then we can suppose it is satisfiable in STIT-model such that:*

- *the number of worlds in each R_{AGT}-classes is equal to N;*
- *the number of $R_{\{i\}}$-classes for all $i \in AGT$ is equal to N.*

where $N = 2^{(n+2)|\varphi|}$.

PROOF.

Let $\mathcal{M} = (W, R, V)$ a STIT-Kripke model satisfying φ. We can suppose that the number of worlds in $\mathcal{M}$ is bounded by $2^{(n+2)|\varphi|}$ by Theorem 24. Thus the number of $R_{\{i\}}$-classes and the number of points in each R_{AGT}-class in bounded by $2^{(n+2)|\varphi|}$. Now the operation consists in adding $R_{\{i\}}$-classes and points to R_{AGT}-classes. The transformation is depicted Figure 7.2: we first "fill up" each R_{AGT}-class by adding copies of worlds and then adjust the number of $R_{\{i\}}$-classes by adding copies of R_{AGT}-classes. The worst we can have is to have at the end of the process:

- $2^{(n+2)|\varphi|}$ points in each R_{AGT}-classes;
- $2^{(n+2)|\varphi|}$ $R_{\{i\}}$-classes.

This makes at most $2^{(n+1)(n+2)|\varphi|}$ worlds in the final model. The formal proof of this result is fastidious and based on the same idea of p-morphism that in Theorem 24. ■

7.1.2.3 Complexity

The finite model property of the previous Theorem 24 leads to the following result:

Theorem 26 *The problem of satisfiability of a given formula in* $\mathcal{L}_{[\{i\}],[AGT]}$ *is:*

- *NP-complete when* $\mathrm{card}(AGT) = 1$*;*
- *NEXPTIME-complete when* $\mathrm{card}(AGT) \geq 2$*.*

PROOF.
The case $card(AGT) = 1$ will be established in Corollary 6. Let us consider the case $card(AGT) \geq 2$. First it is NEXPTIME-hard because the satisfiability problem of the individual CSTIT has already been proved NEXPTIME-hard [BHT08]. It is in NEXPTIME because we have an algorithm to solve the satisfiability problem of a given formula φ:

- guess a super-additive-Kripke structure $\mathcal{M}$ of $2^{|\varphi|}$ worlds;
- check if φ is true in some point of $\mathcal{M}$, which can be done in time linear in the size of $\mathcal{M}$.

This algorithm is sound and complete because of Theorem 24.
■

7.1.2.4 Axiomatization

For the individual CSTIT, Xu [BPX01, Chapter 17] gave the following axioms:

S5($[\emptyset]$) the axiom schemas of S5 for $\Box$;

S5($[i]$) the axiom schemas of S5 for $[i]$ for all $i \in AGT$;

($\emptyset \rightarrow i$) $[\emptyset]\varphi \rightarrow [i]\varphi$, for all $i \in AGT$;

(AIA_n) $(\Diamond[1]\varphi_1 \wedge \ldots \wedge \Diamond[n]\varphi_n) \rightarrow \Diamond([1]\varphi_1 \wedge \ldots \wedge [n]\varphi_n)$.

The characterization of Theorem 24 enties us to add the following axioms:

S5($[AGT]$) the axiom schemas of S5 for $[AGT]$;

($i \rightarrow AGT$) $[\{i\}]\varphi \rightarrow [AGT]\varphi$;

Theorem 27 *A formula φ of $\mathcal{L}_{\mathsf{CSTIT}}$ is CSTIT-valid iff φ is provable from the schemas S5($[\emptyset]$), S5($[i]$), ($\emptyset \rightarrow i$), and (AIA_n), S5($[AGT]$), ($i \rightarrow AGT$) and by the rules of modus ponens and $[\emptyset]$-necessitation.*

Proof.

We have a correspondence between each axiom and its semantical constraint. According to the Theorem of Sahlqvist. [BDRV02, Theorem 4.42] this axiomatization is sound and complete.

■

7.1.3 A generalization of individual coalitions

In this subsection, we investigate the complexity of the satisfiability problem of a richer fragment of CSTIT without time. We only authorize coalitions of the lattice *Lat* (or Hasse diagram) presented in Figure 7.3.

More precisely, in this subsection we suppose that $AGT = \{1, \ldots, km + m - 1\}$ and we are interested in the following fragment of the whole language, denoted as $\mathcal{L}_{lat}$, defined by the following rule:

$$\varphi \ ::= \ \bot \ | \ p \ | \ (\varphi \vee \varphi) \ | \ \neg\varphi \ | \ [J]\varphi$$

where $p \in ATM$ and J is a coalition of the lattice *Lat* presented in the Figure 7.3.

Definition 51 ($\mathcal{L}_{lat}$-super-additive-Kripke model)
A $\mathcal{L}_{lat}$-super-additive-Kripke model $\mathcal{M} = (W, R_X, \{R_J\}_{J \in Lat}, V)$ is a tuple where:

- W is a non-empty set of possible worlds;
- for all $J \in Lat$, R_J is a equivalence relation such that:
  1. if $J \subseteq J'$, $R'_J \subseteq R_J$;
  2. for all k, for all $w \in W$, for all $(w_j)_{j \in \{1,\ldots,m-1\}} \in R_{\{1,\ldots,km\}}$,
$$\bigcap_{j \in \{1,\ldots,m-1\}} R_{\{1,\ldots,km\} \cup \{km+j\}}(w_j) \neq \emptyset;$$
- $V : W \rightarrow 2^{ATM}$.

We introduce some notions:

- As it is depicted on the Figure 7.3, coalitions are classified according to their types. $\emptyset, \{1\}, \ldots, \{m-1\}$ and $\{1, \ldots, m-1\}$ are of type 0, ..., $\{1, \ldots, km\}$, $\ldots$, $\{1, \ldots, km\} \cup \{km+1\}$ and $\{1, \ldots, km\} \cup \{km+m-1\}$ are of type k, and so on.

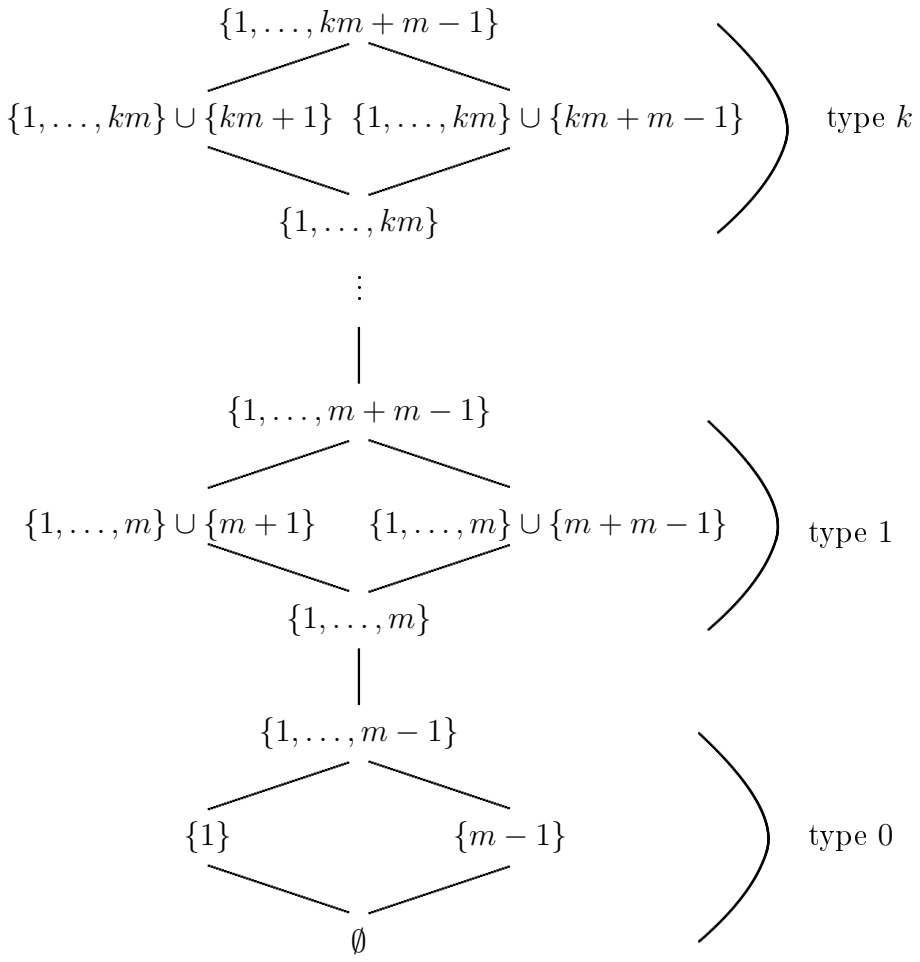


Figure 7.3: The *lattice* of coalitions

- $\Delta\varphi$ denotes the difference between the maximal type and the minimal type of operators appearing in φ.

In this subsection, we will see that STIT-models and $\mathcal{L}_{lat}$-super-additive-models provide in fact the same set of validities. First we explore the easy implication: every STIT-model (Definition 47) can obviously be seen as a $\mathcal{L}_{lat}$-super-additive-model. So if φ is satisfiable in a STIT-model then φ is satisfiable in a $\mathcal{L}_{lat}$-super-additive-model. The following theorem provides a finite model property for $\mathcal{L}_{lat}$-super-additive-models.

Theorem 28 *Let φ a formula satisfiable in a $\mathcal{L}_{lat}$-super-additive-Kripke model. Then φ is satisfiable in a $\mathcal{L}_{lat}$-super-additive-Kripke model where the number of worlds in bounded by $2^{\Delta\varphi O(|\varphi|^2)}$.*[1]

Proof.

We prove the following statement by induction on $\Delta\varphi$. Let $P(K)$ be the following property: "For all satisfiable formulas φ such that $\Delta\varphi = K$, φ is satisfiable in a model of size $2^{K \max(2L_\varphi M_\varphi + 3\alpha_\varphi, |\varphi|)}$ where L_φ is the number of subformulas of the form $[C]\psi$ of φ and M_φ is the maximal size of a subformulas of the form $[C]\psi$ of φ and α_φ is the number of atomic propositions in φ".

The basic case correspond to $\Delta\varphi = 0$. Without loss of generality we can suppose that the formula φ only contains operators of type 0. We are in the case of a formula with individual plus the grand coalition and the theorem is true for this case (see Theorem 24).

Now let us consider the inductive case. Without loss of generality we can suppose that the minimal type appearing in the formula φ is 0. If it is not the case, and if the minimum type is k, simply read this proof by replacing $\emptyset$ by $\{1, \ldots, km\}$, and $\{i\}$ by $\{1, \ldots, km\} \cup \{km + i\}$ for all $i \in \{1, \ldots, m-1\}$. Suppose the formula φ is satisfiable in a super-additive-model $\mathcal{M}, w$ such that $\mathcal{M} = (W, R, V)$. The proof is done in four steps:

1. From $\mathcal{M}$, we are only interested in what is going on concerning the relation of type 0: $R_\emptyset$, $R_{\{i\}}$ and $R_{\{1,\ldots,m-1\}}$. For that reason we introduce the model $\mathcal{M}^f$ obtained from $\mathcal{M}$ by dropping all the accessibility relations and we filter it almost as in the proof of Theorem 24: we get a super-additive-model $\mathcal{M}' = (W', R', V')$; (Figure 7.4)

2. For all $u' \in W'$, we define a formula $propagation(\varphi, u')$ as the formula that must be true in the point u'. This formula sums up the constraints dued to operators of type > 0;

[1] $O(\ldots)$ is the Big Oh Notation. See [Pap03]

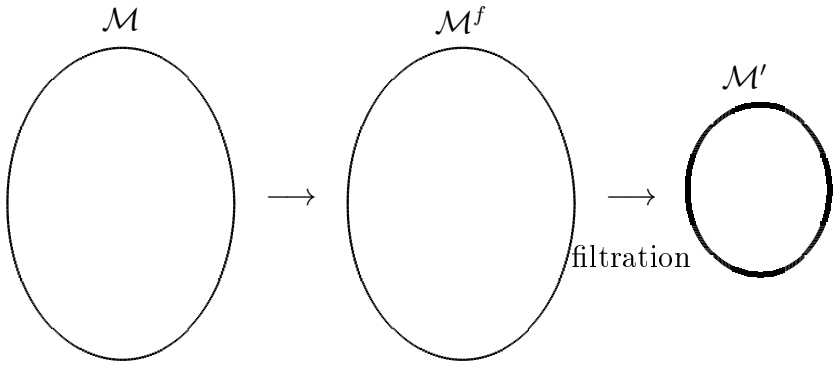


Figure 7.4: $\mathcal{M}^f$: the model $\mathcal{M}$ by removing information about coalitions of type > 0; $\mathcal{M}'$: the filtered model of $\mathcal{M}^f$

3. We exhibit super-additive-models $\mathcal{M}_{u'}$ satisfying $propagation(\varphi, u')$ for all $u' \in W'$;

4. We define a new super-additive-model $\mathcal{M}^*$ by gluing the super-additive-models $\mathcal{M}_u$ alltogether satisfying φ. (Figure 7.5)

Step 1 For all formulas $[B]\psi$ where B is a coalition of type > 0, we are going to introduce extra atomic propositions $p_{[B]\psi}$ in the language. Suppose that there exists a super-additive-model $\mathcal{M} = (W, R, V)$ and a world $w \in W$ such that $\mathcal{M}, w \models \varphi$.

Let us define a super-additive-model $\mathcal{M}^f = (W, R, V^f)$ by:

- the set of worlds W is the same than in $\mathcal{M}$;
- relations $R_\emptyset$, $R_{\{1\}}$, ..., $R_{\{m-1\}}$ and $R_{\{1,\ldots,m-1\}}$ are the same than in $\mathcal{M}$;
- $V^f(u) = V(u) \cup \{p_{[B]\psi} \mid \mathcal{M}, u \models [B]\psi\}$.

Let us define $freeze(\varphi)$ where we have replaced each subformula $[B]\psi$ where B is of type > 0 by the extra proposition $p_{[B]\psi}$. Formally:

Definition 52 ($freeze(\varphi)$)
The formula $freeze(\varphi)$ is defined by induction on φ as follows:

- $freeze(p) = p$;
- $freeze(\psi_1 \vee \psi_2) = freeze(\psi_1) \vee freeze(\psi_2)$;
- $freeze(\neg\psi) = \neg freeze(\psi)$;

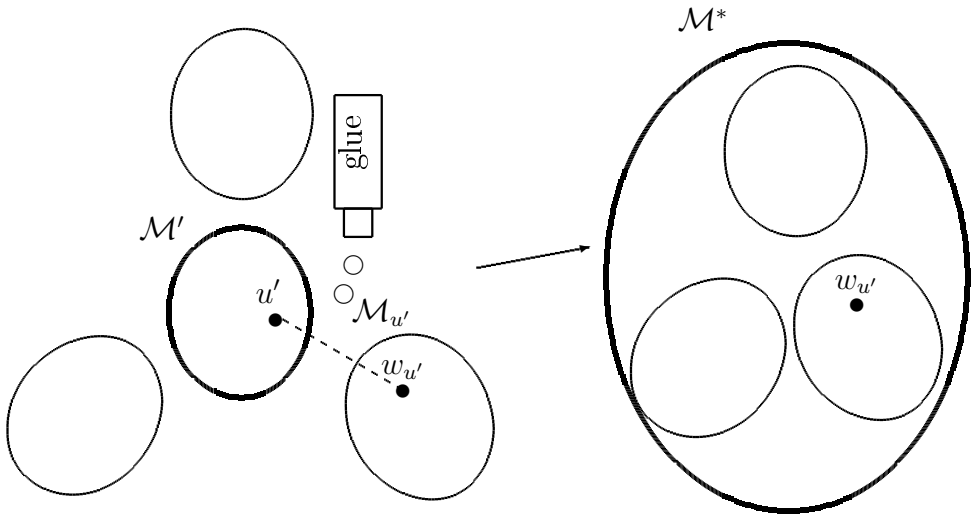


Figure 7.5: Gluing the super-additive-models $\mathcal{M}'_u$ in order to get $\mathcal{M}^*$

- $freeze([A]\psi) = [A]freeze(\psi)$ if A is of type 0;
- $freeze([B]\psi) = p_{[B]\psi}$.

We have that for all formulas ψ, $\mathcal{M}, u \models \psi$ iff $\mathcal{M}^f, u \models freeze(\psi)$.

Now we filter the model $\mathcal{M}^f$ by the set[2] $\{freeze(\psi) \mid \psi \in SF(\varphi)\}$where $SF(\varphi)$ is the set of all subformulas of φ. The result is a super-additive-model $\mathcal{M}' = (W', R', V')$. We have:

- $card(W') \leq 2^{|\varphi|}$;
- there exists $w' \in W'$ such that $\mathcal{M}', w' \models freeze(\varphi)$;
- for $u' \in W'$ there exists $u = c(u') \in W$ such that for all subformulas $\psi \in SF(\varphi)$ we have $\mathcal{M}^f, u \models freeze(\psi)$ iff $\mathcal{M}', u' \models freeze(\psi)$. (as $\mathcal{M}'$ is a filtered model of $\mathcal{M}^f$, we take $c(u')$ as a representative of the equivalence class u')

Let $u' \in W'$. Let $u = c(u') \in W$. Note that

$$\text{for all subformulas } \psi \in SF(\varphi),\ \mathcal{M}, u \models \psi \text{ iff } \mathcal{M}', u' \models freeze(\psi). \tag{7.1}$$

For all $\psi \in SF(\varphi)$ and for all $u' \in W'$ we define $simplify(\psi, u')$ as the formula ψ in which we have replaced all subformulas of the form $[A]\psi$ where $[A]$ is of type 0 (not nested into another operator of type 0):

[2]Notice that the proof is not correct if we filter by the set of all subformulas of $freeze(\varphi)$!

- by $\top$ if $\mathcal{M}', u' \models [A]freeze(\psi)$;
- or by $\bot$ if $\mathcal{M}', u' \not\models [A]freeze(\psi)$.

More formally:

Definition 53 ($simplify(\psi, u')$)
Let $u' \in W'$, let $\psi \in SF(\varphi)$. $simplify(\psi, u')$ is defined by induction on ψ as:

- $simplify(p, u') = p$ for all $p \in ATM$;
- $simplify(\psi_1 \vee \psi_2, u') = simplify(\psi_1, u') \vee simplify(\psi_2, u')$;
- $simplify([B]\psi, u') = [B]simplify(\psi, u')$ if B is of type > 0;
- $simplify([A]\psi, u') = \top$ if $\mathcal{M}', u' \models [A]freeze(\psi)$ and A is of type 0;
- $simplify([A]\psi, u') = \bot$ if $\mathcal{M}', u' \not\models [A]freeze(\psi)$ and A is of type 0.

Lemma 9 *Let $u' \in W'$ and $u = c(u') \in W$. For all $\psi \in SF(\varphi)$, $\mathcal{M}, u \models simplify(\psi, u')$ iff $\mathcal{M}', u' \models freeze(\psi)$.*

PROOF.
First we prove by induction on $\psi \in SF(\varphi)$ the following property:
$P(\psi)$ = "for all $v \in R_{AGT}(u)$, $\mathcal{M}, v \models simplify(\psi, u')$ iff $\mathcal{M}, v \models \psi$."

(Propositions) ok.

$[B]\psi$ Let B be a coalition of type > 0. We have:

$\mathcal{M}, v \models simplify([B]\psi, u')$ iff for $u \in R_B(v)$, $\mathcal{M}, u \models simplify(\psi, u')$
iff for $u \in R_B(v)$, $\mathcal{M}, u \models \psi$ (because $R_B \subseteq R_{AGT}$)
iff $\mathcal{M}, v \models [B]\psi$

$[A]\psi$ Let A be a coalition of type 0. Let us consider the case $\mathcal{M}', u' \models freeze([A]\psi)$. We have $simplify([A]\psi, u') = \top$. Hence $\mathcal{M}, v \models simplify([A]\psi, u')$. But in this case, as $[A]\psi \in SF(\varphi)$, (7.1) implies $\mathcal{M}, u \models [A]\psi$. Similarly when $\mathcal{M}', u' \not\models freeze([A]\psi)$, We have $simplify([A]\psi, u') = \bot$. Hence $\mathcal{M}, v \not\models simplify([A]\psi, u')$. But in this case, as $[A]\psi \in SF(\varphi)$, (7.1) implies $\mathcal{M}, u \not\models [A]\psi$.

In particular we have proven that for all $\psi \in SF(\varphi)$ $\mathcal{M}, u \models \psi$ iff $\mathcal{M}, u \models simplify(\psi, u')$. We conclude with (7.1).■

Definition 54 ()
We define $propagation(\varphi, u')$ as the formula

$$\bigwedge_{p\in ATM\cap\varphi|\mathcal{M}',u'\models p} p \wedge \bigwedge_{p\in ATM\cap\varphi|\mathcal{M}',u'\not\models p} \neg p \wedge$$

$$\bigwedge_{[B]\psi\in SF(\varphi)|\mathcal{M}',u'\models p_{[B]\psi}} [B]simplify(\psi, u') \wedge \bigwedge_{[B]\psi\in SF(\varphi)|\mathcal{M}',u'\not\models p_{[B]\psi}} \neg[B]simplify(\psi, u') \wedge$$

$$\bigwedge_{[A]\psi\in SF(\varphi)|A \text{ of type 0 and } \mathcal{M}',u'\models[A]freeze(\psi)} [\{1,\ldots,m\}]simplify(\psi, u').$$

Lemma 10 *Let $\psi = propagation(\varphi, u')$. $|\psi| \leq 3\alpha_\varphi + 2L_\varphi M_\varphi$ where α_φ is the number of atomic propositions in φ, L_φ is the number of subformulas of the form $[C]\psi$ in φ and M_φ is the maximal size of a subformula $[C]\psi$ of φ. Furthermore:*

- $\alpha_\psi = \alpha_\varphi$;
- $L_\psi \leq L_\varphi$;
- $M_\psi \leq M_\varphi$.

Proof.
Notice that $|simplify(\psi, u')| \leq |\psi|$. ■

Lemma 11 *For all $u' \in W'$, the formula $propagation(\varphi, u')$ is satisfiable in a $\mathcal{L}_{lat}$-super-additive-model.*

Proof.
By Lemma 9.
■

For all $u' \in W'$, as $propagation(\varphi, u')$ is satisfiable in a $\mathcal{L}_{lat}$-super-additive-model, by induction, there exists a super-additive-model $\mathcal{M}_{u'} = (W_{u'}, R_{u'}, V_{u'})$ and a world $w_{u'}$ such that:

- $card(W_{u'}) \leq 2^{(\Delta\varphi-1)(2L_\varphi M_\varphi+3A_\varphi)}$;
- $\mathcal{M}_{u'}, w_{u'} \models propagation(\varphi, u')$.

Furthermore, we can suppose that $R_{u'\{1,\ldots,m\}} = W_{u'} \times W_{u'}$.

We now define the super-additive-model $\mathcal{M}^* = (W^*, R^*, V^*)$ as follows:

- $W^* = \bigcup_{u'\in W'} W_{u'}$;
- $R^*_A = \{(x, y) \mid x \in W_{u'}, y \in W_{v'} \text{ and } u'R'_A v'\}$ for all A of type 0;

- $R^*_B = \bigcup R_{u'B}$ for all B of type > 0;
- $V^*(x) = V_{u'}(x)$ where u' such that $x \in W_{u'}$.

First notice that $card(W^*) \leq 2^{\Delta\varphi \max(2L_\varphi M_\varphi + 3A_\varphi, |\varphi|)}$.

Secondly we claim that $\mathcal{M}^*, w_{w'} \models \varphi$. In order to prove it, we first prove by induction on $\psi \in SF(\varphi)$, the property $P(\psi) =$ for all $u' \in W'$, for all $v \in R^*_{AGT}(w'_u)$, $\mathcal{M}*, v \models \psi$ iff $\mathcal{M}*, v \models simplify(\psi, u')$.

propositions $simplify(p, u') = p$ so $\mathcal{M}*, v \models p$ iff $\mathcal{M}*, v \models simplify(p, u')$.

$[B]\psi$ Let B be a coalition of type > 0.

$$\begin{aligned}
\mathcal{M}*, v \models [B]\psi \quad & \text{iff for all } x \in R_B(v),\ \mathcal{M}*, x \models \psi \\
& \text{iff for all } x \in R_B(v),\ \mathcal{M}*, x \models simplify(\psi, u') \\
& \qquad\qquad \text{(because } x \in R^*_{AGT}(w'_u)) \\
& \text{iff } \mathcal{M}*, v \models [B]simplify(\psi, u') \\
& \text{iff } \mathcal{M}*, v \models simplify([B]\psi, u').
\end{aligned}$$

$[A]\psi$ Let A be a coalition of type 0.

There are two cases:

- Case 1: $\mathcal{M}', u' \models freeze([A]\psi)$. In this case, $simplify([A]\psi, u') = \top$ and of course we have $\mathcal{M}*, v \models simplify([A]\psi, u')$.

  Let us prove that we have $\mathcal{M}^*, v \models [A]\psi$. Better said, we have to prove that for all $t \in R^*_A(v)$, $\mathcal{M}^*, t \models \psi$.

  Let $t \in R^*_A(v)$. Let $t' \in W'$ be such that $u'R'_A v'$. As $u'R'_A v'$, $\mathcal{M}', t' \models freeze([A]\psi)$. By Definition of $propagation(\varphi, t')$, we have that $\mathcal{M}*, w_{t'} \models [\{1, \ldots, m\}]simplify(\psi, t')$. Hence as $tR_{\{1,\ldots,m\}}t'$, we have $\mathcal{M}*, t \models [simplify(\psi, t')$. By induction $(P(\psi))$, $\mathcal{M}^*, t \models \psi$ for all $t \in R^*_A(v)$. So $\mathcal{M}^*, v \models [A]\psi$.
- Case 2: $\mathcal{M}', u' \not\models freeze([A]\psi)$. In this case, $simplify([A]\psi, u') = \bot$ and of course we have $\mathcal{M}*, v \not\models simplify([A]\psi, u')$.

  Let us prove that we have $\mathcal{M}^*, v \not\models [A]\psi$.

  $\mathcal{M}', u' \not\models freeze([A]\psi)$ implies that there exists $t' \in R'_A(u')$ such that $\mathcal{M}', t' \not\models freeze(\psi)$. But then by definition of $propagation(\varphi, t')$ we have $\mathcal{M}^*, w_{t'} \models simplify(\neg\psi, t')$. By induction $(P(\psi))$, we have $\mathcal{M}^*, w_{t'} \models \neg\psi$. So as $w_{t'}R^*_A v$, we have $\mathcal{M}^*, v \not\models [A]\psi$.

Finally we have $\mathcal{M}*, w_{w'} \models \varphi$. Indeed, we prove by induction that $P(\psi) = \mathcal{M}*, w_{u'} \models \psi$ iff $\mathcal{M}', u' \models Freeze(\psi)$.

Propositions $\mathcal{M}*, w_{u'} \models p$ iff $\mathcal{M}', u' \models p$ (because p or $\neg p$ appears in $propagation(\varphi, u')$ according to the truth of p in $\mathcal{M}', u'$)

$[B]\psi$ $\mathcal{M}*, w_{u'} \models [B]\psi$ iff $\mathcal{M}*, w_{u'} \models simplify([B]\psi, u')$
iff $\mathcal{M}_{u'}, w_{u'} \models simplify([B]\psi, u')$
iff $\mathcal{M}, u' \models p_{[B]\psi}$.

$[A]\psi$ $\mathcal{M}*, w_{u'} \models [A]\psi$ iff $\mathcal{M}*, w_{u'} \models simplify([A]\psi, u')$
iff $\mathcal{M}_{u'}, u' \models [A]freeze(\psi))$.

■

Now let us consider the other direction: we have to prove that if a formula is satisfiable in a $\mathcal{L}_{lat}$-super-additive-model, then it is so in a STIT-model. In a super-additive-model we define only R_J for J in the lattice Lat whereas a STIT-model is defined in terms of $R_{\{i\}}$ for all agents $i \in AGT$. In a STIT-model, the relations R_J are defined from $R_{\{i\}}$ by $\bigcap_{i \in J} R_{\{i\}} = R_J$. Our problem is now to see a super-additive-model as a STIT-model (Definition 47). Of course the following system of equations

$$\text{for all } J \in Lat, \bigcap_{i \in J} R_{\{i\}} = R_J \tag{7.2}$$

where the unknown are $R_{\{i\}}$, has not always a solution. In fact, it is the same problem that the difference between $\mathcal{L}_{[\{i\}],[AGT]}$-super-additive-models and STIT-model in Subsection 7.1.2 but here generalized for coalitions from Lat. Fortunately we can always transform the super-additive-model by adding worlds without changing the truth of formulas in order to be able to solve the system of equations and to get a STIT-model as in Theorem 24.

Theorem 29 *Let φ a formula in $\mathcal{L}_{lat}$ that is satisfiable in a super-additive-model. Then φ is satisfiable in STIT-Kripke-model with the same number of $R_{\{i\}}$-classes (at most $2^{(n+2)O(|\varphi|^2)}$) where n is the total number of agents.*

Proof.

Let φ a formula in $\mathcal{L}_{lat}$ satisfiable in a finite super-additive-model. We are going to transform the super-additive-model into a STIT-model satisfying exactly the same formulas, and with the same number of $R_{\{i\}}$-classes. Let $N = 2^{(n+2)O(|\varphi|^2)}$.

The proof is done by induction over the maximal difference of types of the relations appearing in the model. If the maximal difference of types is 0, we already deal with a STIT-Kripke model. If not, the proof is based on the notations of the proof of Theorem 28 and is done as follows:

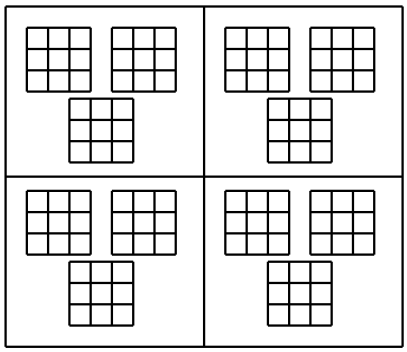


Figure 7.6: The model $\mathcal{M}^*$

- First we consider the filtered model $\mathcal{M}'$ defined from $\mathcal{M}$ as in the proof of Theorem 28. We can suppose that $\mathcal{M}'$ satisfies $R_{\{1,\dots,m-1\}} = \bigcap_{i\in\{1,\dots,m-1\}} R_{\{i\}}$ and also that the number of R_i-classes for all $i \in \{1,\dots,m-1\}$ and the number of points in $R_{\{1,\dots,m-1\}}$ are constant equal to N by Theorem 25;

- By induction, we then transform each super-additive-model $\mathcal{M}_{u'}$ into a "STIT-Kripke model" $\mathcal{M}'_{u'}$ where relations are $R_{\{1,\dots m\}}$, $R_{\{1,\dots,m\}\cup\{m+1\}}$, $\dots$, $R_{\{1,\dots,m\}\cup\{m+i\}}$, $\dots$, $R_{\{1,\dots,m\}\cup\{km+m-1\}}$ and the number of classes $R_{\{1,\dots,m\}\cup\{m+i\}}$ are equal to N;

- Each submodel $\mathcal{M}'_{u'}$ has the same number N of $R_{\{1,\dots,m\}\cup\{m+i\}}$-classes. Classes are numbered from 1 to N. For instance $R_{\{1,\dots,m\}\cup\{m+2\}}$-classes included in a submodel $\mathcal{M}'_{u'}$ are noted $C_1^{\mathcal{M}'_{u'},R_{\{1,\dots,m\}\cup\{m+i\}}},\dots,$ $C_N^{\mathcal{M}'_{u'},R_{\{1,\dots,m\}\cup\{m+i\}}}$.

- Then we define a model $\mathcal{M}^*$ in the same manner as in the proof of Theorem 28. The model $\mathcal{M}^*$ is depicted in Figure 7.6.

- We claim that the model $\mathcal{M}^*$ is such that we can solve the system of equations (7.2). As $R_\emptyset$, $R_{\{1\}}$, $\dots$, $R_{\{m-1\}}$ are already defined, we have to define $R_{\{m\}}$, $R_{\{m+1\}}$, $\dots$, $R_{\{km+m+1\}}$. We define $wR_{\{m\}}v$ iff the number of the submodel in which w belongs in its $R_{\{1,\dots m-1\}}$-class and the number of the submodel in which v belongs in its $R_{\{1,\dots m-1\}}$-class are equal. (Figure 7.7)

  We define $wR_{\{m+i\}}v$ iff the number of the $R_{\{1,\dots m\}\cup\{m+i\}}$-class in which w belongs in its submodel and the number of the $R_{\{1,\dots m\}\cup\{m+i\}}$-class in which v belongs in its submodel are equal. (Figure 7.8)

  We have to check that $R_{\{1,\dots m\}} = \bigcap_{i\in\{1,\dots m\}} R_{\{i\}}$ and $R_{\{1,\dots m\}\cup\{m+j\}} = \bigcap_{i\in\{1,\dots m\}\cup\{m+j\}} R_{\{i\}}$. We also have to ensure the prop-

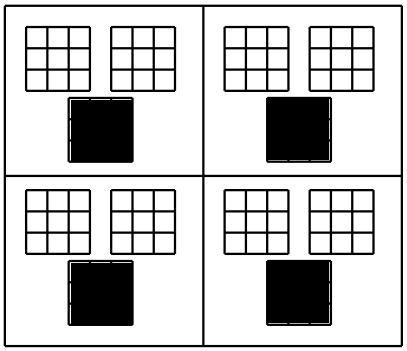


Figure 7.7: A $R_{\{m\}}$-class

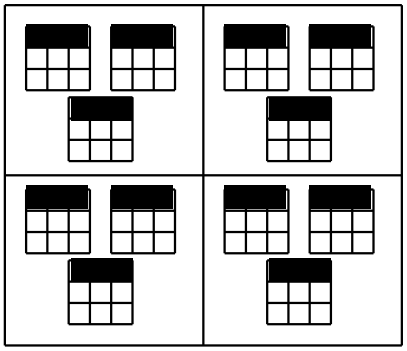


Figure 7.8: A $R_{\{m+i\}}$-class

erty of independence of agents: $\bigcap_{i \in AGT} R_{\{i\}}(w_i) \neq \emptyset$, for all $(w_i)_{i \in AGT}$. We leave the reader with Figures 7.7 and 7.8.

- You can check that that the number of R_i^* classes for all $i \in AGT$ is N.

■

7.1.3.1 Complexity

Suppose that $m > 2$. We can define two satisfiability problems like in [WLWW06]. In the first problem the set AGT is fixed while in the second problem the set AGT is part of the input.

Let $m > 2$ and $k \geq 0$. We define $AGT = \{1, \ldots km + m - 1\}$.

- Input: A formula $\varphi \in \mathcal{L}_{lat}$;
- Output: yes iff φ is STIT-satisfiable.

and

- Input: two integers $k \geq 0$, $m \geq 0$ and formula $\varphi \in \mathcal{L}_{lat}$ where $AGT = \{1, \ldots km + m - 1\}$.
- Output: yes iff φ is STIT-satisfiable.

Theorem 30 *The two decision problems are both NEXPTIME-complete.*

Proof.

They belong to NEXPTIME because of Theorem 28 and Theorem 29. They are NEXPTIME-hard because of they contain the satisfiability problem of a given formula of individual STIT [BHT08] with two agents, which is already NEXPTIME-hard. ■

7.1.3.2 Axiomatization

As all the constraints in a $\mathcal{L}_{lat}$-super-additive-Kripke model corresponds to Salqvist formulas [BDRV02, Th. 4.42], and as we have the same valid formulas of the language $\mathcal{L}_{lat}$ with STIT-Kripke models and $\mathcal{L}_{lat}$-super-additive-Kripke models, we have the following axiomatization:

Theorem 31 *A formula of the language $\mathcal{L}_{lat}$ is valid in STIT-Kripke models iff it is provable from the following axiom schemas with the rule Modus Ponens and the Necessitation rules of all modal operators:*

S5([J]) *the axiom schemas of S5 for all $J \in Lat$;*

($\emptyset \rightarrow i$) $[J]\varphi \rightarrow [J']\varphi$, *for every* $J, J' \in AGT$ *such that* $J \subseteq J'$;

(AIA_n^k)

$$(\langle\{1,\ldots,km\}\rangle[\{1,\ldots,km\}\cup\{km+1\}]\varphi_1 \wedge \ldots \wedge$$
$$\langle\{1,\ldots,km\}\rangle[\{1,\ldots,km\}\cup\{km+m-1\}]\varphi_{m-1})$$
$$\rightarrow \langle\{1,\ldots,km\}\rangle([\{1,\ldots,km\}\cup\{km+1\}]\varphi_1 \wedge \ldots \wedge$$
$$[\{1,\ldots,km\}\cup\{km+m-1\}]\varphi_{m-1})$$

. *for all* k.

7.1.4 The logic of chains of coalitions

In this subsection, we are going to investigate the case where the set of coalitions of the language form a chain:

$$J_1 \subset J_2 \subset J_3 \ldots.$$

This logic has already been investigated in [HS08] when AGT is fixed. Let us begin to recall the results of complexity and axiomatization in this case and then we give the results for the case when AGT is not fixed.

7.1.4.1 The case when AGT is fixed

Let $AGT = \{1 \ldots n\}$ be a finite set. Without losss of generality, we only study the following chain:

$$\emptyset \subset \{1\} \subset \{1,2\} \subset \ldots \{1,\ldots,n\}.$$

More precisely, we are interested in the following fragment of the whole language, denoted as $\mathcal{L}_{[\{1\ldots i\}]\text{fixed}}$, by the following rule:

$$\varphi \ ::= \ \bot \ | \ p \ | \ (\varphi \vee \varphi) \ | \ \neg\varphi \ | \ [J]\varphi$$

where $p \in ATM$ and $J \in \{\emptyset, \{1\}, \{1,2\}, \ldots \{1,\ldots,n\}\}$.

Definition 55 ()
The satisfiability problem in $\mathcal{L}_{[\{1\ldots i\}]\text{fixed}}$ is defined as follows:

- Input: φ in $\mathcal{L}_{[\{1\ldots i\}]\text{fixed}}$ (where coalitions are $\emptyset$, $\{1\}$, $\{1,2\}$, $\ldots \{1\ldots,n\}$);
- Ouput: yes iff the formula φ is satisfiable.

Definition 56 ()
A $\mathcal{L}_{[\{1...i\}]\text{fixed}}$-super-additive model is a structure $\mathcal{M} = (W, R_\emptyset, R_{\{1\}}, \ldots, R_{\{1,\ldots,n\}}, V)$ such that:

- each $R_{\{1,\ldots k\}}$ is an equivalence relation;
- $R_{\{1,\ldots,k+1\}} \subseteq R_{\{1,\ldots,k\}}$.

Proposition 10 *Let φ a formula of $\mathcal{L}_{[\{1...i\}]fixed}$. The formula φ is satisfiable in a STIT-model iff it is satisfiable in a $\mathcal{L}_{[\{1...i\}]fixed}$-super-additive model.*

Proof.
It comes from Theorem 29 where $m = 2$. ■

In the same manner than in Theorem 31, this logic is axiomatizable by the following axiomatics:

S5($[J]$) the axiom schemas of S5 for all $J \in Lat$;

($\emptyset \rightarrow i$) $[\{1, \ldots k\}]\varphi \rightarrow [\{1, \ldots, k+1\}]\varphi$, for all k.

We will see that the satisfiability problem in $\mathcal{L}_{[\{1...i\}]\text{fixed}}$ is NP-complete, see Corollary 6.

7.1.4.2 The case when AGT is variable

In [WLWW06], the authors have proved that if we put the number of agents into the input of the satisfiability problem of the logic ATL, the problem remains in EXPTIME. Here for STIT and chains of coalitions, the result is different. If the number of agents is fixed and is not in the input of the problem, the satisfiability problem is NP-complete (Corollary 6) whereas it is PSPACE-complete if the number of agents is part of the input (Corollary 7 and Theorem 32).

Definition 57 ()
The satisfiability problem in $\mathcal{L}_{[\{1...i\}]\text{variable}}$ is:

- Input:
  - the cardinality n of AGT;
  - a formula φ whose coalitions are $\emptyset$, $\{1\}$, $\{1, 2\}$, $\ldots \{1, \ldots, n\}$;
- Ouput: yes iff the formula φ is satisfiable.

First we establish the upper bound: the satisfiability problem in $\mathcal{L}_{[\{1...i\}]\text{variable}}$ is in PSPACE. Roughly speaking this result comes from two facts:

```
function sat(φ, k)
    if φ only contains only operator [k] then
        S5 − sat_k(φ)
    else
        choose a model M' = (W', R'_{1,...k}, V') where card(W') is bounded
        by the number of modal operators [{1, ... k}] appearing in φ,
        R'_{1,...k} = W' × W', V' is a valuation for atomic propositions and
        also extra propositions [B]ψ.
        choose w' ∈ W';
        if M', w' ⊨ Freeze(φ) then
            for all u' ∈ W'
                call sat(propagation(φ, u'), k + 1)
            endFor
        else
            reject
        endIf
    endIf
endFunction
```

where $freeze(\varphi)$, $propagation(\varphi, u')$ are defined in the proof of Theorem 28.

Figure 7.9: An algorithm for satisfiability problem in $\mathcal{L}_{[\{1...i\}]\text{variable}}$

- We can treat the coalitions one after the other. Contrary to the Theorem 28 where the filtration provides a model $\mathcal{M}'$ with an exponential number of worlds, here we can use a selection-of-points argument as for S5 [Lad77] providing a "partial" model with a linear number of worlds (the B_i's in the algorithm of Figure 7.9);

- In the proof of Theorem 28, we then apply the induction hypothetis to formulas $propagation(\varphi, u')$. The algorithm is based on the same idea but explore the satisfiability of $propagation(\varphi, k, B_i)$ one after the other so that the algorithm only require a polynomial amount of memory although the whole model can be of exponential size.

Theorem 32 *The satisfiability problem in $\mathcal{L}_{[\{1...i\}]variable}$ is in PSPACE.*

Proof.

Let us consider the non-deterministic procedure $sat(\varphi, k)$ of the Figure 7.9. We leave the reader check that this procedure terminates and only uses a polynomial amount of memory.

We can check that for all formulas φ containing operators of type $\geq k$, we have $sat(\varphi, k)$ succeeds iff φ is satisfiable.

Basic case If there is no modal operator or only the operator $\{1,\ldots,k\}$ in φ then we have $sat(\varphi,k)$ succeeds iff φ is satisfiable.

Inductive case

$\Rightarrow$

Suppose that $sat(\varphi,k).$ succeeds. Then there exists a model $\mathcal{M}'=(W',R_{\{1,\ldots k\}},V')$ and $w'\in W'$ such that $\mathcal{M}',w'\models freeze(\varphi)$. Every call $sat(propagation(\varphi,k,u'),k+1)$ has been successful so by induction for all u', $propagation(\varphi,k,B_i)$ is satisfiable: there exists $\mathcal{L}_{[\{1\ldots i\}]\text{fixed}}$-super-additive-model $\mathcal{M}_{u'}=(W_{u'},R_{\{1\ldots k+1\}_{u'}},\ldots,V_{u'})$ and a point $w_{u'}\in W_i$ such that $\mathcal{M}_{u'},w_{u'}\models propagation(\varphi,k,u')$.

We define a model $\mathcal{M}^*$ in the same way as in the proof of Theorem 28 and that $\mathcal{M}^*,w_{w'}\models\varphi$.

$\Leftarrow$

Reciprocally suppose that the formula φ is satisfiable. There exists a model $\mathcal{M}=(W,R_{\{1,\ldots k\}}\ldots,V)$ and a world $w\in W$ such that $\mathcal{M},w\models\varphi$.

Let $\alpha_1=[\{1,\ldots,k\}]\psi_1,\ldots,\alpha_K=[\{1,\ldots,k\}]\psi_K$ an enumeration (with repetition) of all formulas of the form $[\{1,\ldots,k\}]\psi$ such that $\mathcal{M},w\not\models[\{1,\ldots,k\}]\psi$.

For all $i\in\{1,\ldots|\varphi|\}$, there exists a world $u_i\in W$ such that $\mathcal{M},u_i\models\psi_i$.

Now we define the Kripke-model $\mathcal{M}'=(W',R'_{\{1,\ldots k\}},V')$ in the following way:

- $W'=\{u_1,\ldots u_K\}\cup\{w\}$;
- $R'_{\{1,\ldots k\}}=W'\times W'$;
- For all $u'\in W'$, $V'(u')=V(u')\cup\{p_{[B]\psi}\mid\mathcal{M},u'\models[B]\psi\}$.

We have $\mathcal{M}',w'\models\psi$. Furthermore, $\mathcal{M},u_i\models propagation(\varphi,B_i)$. So by induction the calls $sat(propagation(\varphi,B_i),k+1)$ are successful. So the call $sat(\varphi,k)$ succeeds. ■

Theorem 33 *Let φ be of $\mathcal{L}_{[\{1\ldots i\}]fixed}$. If φ is satisfiable in a $\mathcal{L}_{[\{1\ldots i\}]fixed}$-super-additive-model then:*

1. *it is satisfiable in a $\mathcal{L}_{[\{1\ldots i\}]fixed}$-super-additive-model where the number of worlds is at most $(|\varphi|)^{\Delta\varphi+1}$;*
2. *it is satisfiable in a STIT-model where the number of worlds is at most $(|\varphi|)^{\Delta\varphi+1}$.*

Proof.

We focus on the algorithm given in Figure 7.9 and prove Item 1. by induction:

- If there is no operator or only one operator $[1, \ldots, k]$ (i.e. $\Delta\varphi = 0$), the algorithm call S5-sat and a formula φ is satisfiable in S5 iff it is satisfiable in a model with $|\varphi|$ worlds;
- The induction step is as following:
  - we can suppose that each super-additive-model $\mathcal{M}_i$ for $propagation(\varphi, k, B_i)$ has at most $(|\varphi|)^{\Delta\varphi}$ worlds by induction hypothesis;
  - There are at most $|\varphi|$ models $\mathcal{M}_i$;

  The super-additive-model $\mathcal{M}$ has at most $|\varphi| \times |\varphi|^{\Delta\varphi} = (|\varphi|)^{\Delta\varphi+1}$ worlds.

  Concerning Item 2, you can read again the proof of Theorem 29 with $N = |\varphi|$.

■

Corollary 6 *If the number of agents is fixed, then the satisfiability problem is NP-complete.*

PROOF.
Comes from Theorem 33. If $n = card(AGT)$ is fixed, $(|\varphi|)^{\Delta\varphi+1} \leq (|\varphi|)^n$ is X^n a polynomial of degree n. So a non-deterministic algorithm for the satisfiability problem of a given formula φ consists in guessing a super-additive-$\mathcal{L}_{[\{1\ldots i\}]\text{fixed}}$-model of size at most $(|\varphi|)^n$ and then checking if the formula φ is satisfiable. ■

7.2 With the neXt operator

7.2.1 Individual STIT plus the grand coalition plus neXt operator

In this subsection, we suppose $AGT = \{1, \ldots m-1\}$ where $m \geq 1$ and we introduce the language $\mathcal{L}_{[\{i\}],[AGT],X}$ defined by:

$$\varphi \ ::= \ \bot \ | \ p \ | \ \varphi \vee \varphi \ | \ \neg\varphi \ | \ [\{i\}]\varphi \ | \ [AGT]\varphi \ | \ X\varphi$$

where p ranges over ATM and i ranges over AGT.

In order to prove the complexity of the satisfiability problem of a formula of $\mathcal{L}_{[\{i\}],[AGT],X}$, the idea consists in flattening the time like it is depicted in Figure 7.10. In the "flattened" model (a $\mathcal{L}_{lat}$-super-additive-Kripke model), a point corresponds to a history. Proposition p at time 0, 1, etc. of a XCSTIT-Kripke model are represented in the flattened model by p_0, p_1, etc. The relation of choice $R_\emptyset$ at time 0, 1, etc. is simulated by the different relations of the lattice depicted in Figure 7.3. Operators of type 0 of $\mathcal{L}_{lat}$ are used to simulate operators of $\mathcal{L}_{[\{i\}],[AGT],X}$ at the

time 0. Operators of type 1 of $\mathcal{L}_{lat}$ are used to simulate operators of $\mathcal{L}_{[\{i\}],[AGT],X}$ at time 1, and so on. In order to do that, we have to consider the $\mathcal{L}_{lat}$ such that the number of coalitions is not bounded. (k can be arbitrarily big).

Definition 58 ()
For all positive integers k, we define the following translation:

$$tr_k : \quad \mathcal{L}_{[\{i\}],[AGT],X} \quad \rightarrow \quad \mathcal{L}_{lat}$$

by induction:

- $tr_k(p) = p_k$;
- $tr_k(\varphi \wedge \psi) = tr_k(\varphi) \wedge tr_k(\varphi)$;
- $tr_k(X\varphi) = tr_{k+1}(\varphi)$;
- $tr_k([\emptyset]\varphi) = [\{1, \ldots, km\}]tr_k(\varphi)$;
- $tr_k([\{i\}]\varphi) = [\{1, \ldots, km + i\}]tr_k(\varphi)$;
- $tr_k([AGT]\varphi) = [\{1, \ldots, km + m - 1\}]tr_k(\varphi)$;

The translation tr_k translates a formula of $\mathcal{L}_{[\{i\}],[AGT],X}$ into $\mathcal{L}_{lat}$ considering that the current time is k. A proposition p in $\mathcal{L}_{[\{i\}],[AGT],X}$ is translated into p_k in $\mathcal{L}_{lat}$ meaning that "p is true at time k". At time k, $X\varphi$ is translated as "φ will be true at time $k+1$". For $\mathcal{L}_{[\{i\}],[AGT],X}$-operators, at time k, we use operators of the lattice Lat of type k.

Property 2 of $\mathcal{L}_{lat}$-super-additive-models (Definition 51) corresponds to independence of agents in XCSTIT-models for all times k. The property 1. of $\mathcal{L}_{lat}$-super-additive-models corresponds to the property of no choice between undivided histories in XCSTIT-models. Hence, we have a correspondence in terms of satisfiability in $\mathcal{L}_{[\{i\}],[AGT],X}$ and in $\mathcal{L}_{lat}$. Formally:

Theorem 34 *Let φ a formula of $\mathcal{L}_{[\{i\}],[AGT],X}$. φ is satisfiable in a XCSTIT-Kripke frame iff $tr_0(\varphi)$ is satisfiable in a $\mathcal{L}_{lat}$-super-additive Kripke model.*

Proof.

$\boxed{\Rightarrow}$ Suppose there exists a XCSTIT-Kripke frame $\mathcal{M} = (W, R_X, \{R_J\}_{J \subseteq AGT}, V)$ and $w \in W$ such that R_X is injective and $\mathcal{M}, w \models \varphi$. (we can suppose that R_X is injective because of Theorem 20)

We define the $\mathcal{L}_{lat}$-super-additive-model $\mathcal{M}' = (W', \{R'_J\}, V')$ by:

- $W' = R_\emptyset(w)$;

- $R'_{\{1,\ldots,nk\}} = R_X^k \circ R_\emptyset \circ R_X^{-k}$;
- $R'_{\{1,\ldots,nk+1\}\cup\{n+i\}} = R_X^k \circ R_{\{i\}} \circ R_X^{-k}$;
- $R'_{\{1,\ldots,nk+n-1\}} = R_X^k \circ R_{AGT} \circ R_X^{-k}$;
- $V'(u) = \{p_k \mid k \in \mathbb{N} \text{ and } p \in V(R_X(u))\}$.

We can prove by induction on $\psi \in \mathcal{L}_{[\{i\}],[AGT],X}$ that for all $u \in R_\emptyset(w)$, for all $k \in \mathbb{N}$, $\mathcal{M}, R_X^k(u) \models \psi$ iff $\mathcal{M}', u \models tr_k(\psi)$. In particular, we have $\mathcal{M}', w \models tr_0(\varphi)$.

$\boxed{\Leftarrow}$ Suppose there exists a $\mathcal{L}_{lat}$-super-additive-model $\mathcal{M}' = (W', \{R'_J\}, V')$ and $w \in W$ such that $\mathcal{M}', w \models tr_0(\varphi)$.

We define the XCSTIT-Kripke model $\mathcal{M} = (W, R_X, \{R_J\}_{J\subseteq AGT}, V)$ by:

- $W = W' \times \mathbb{N}$;
- $R_\emptyset = \{((w,k),(u,k)) \mid wR'_{1,\ldots,nk}u\}$;
- $R_{\{i\}} = \{((w,k),(u,k)) \mid wR'_{\{1,\ldots,nk+1\}\cup\{n+i\}}u\}$;
- $R_{AGT} = \{((w,k),(u,k)) \mid wR'_{\{1,\ldots,nk+n-1\}}u\}$;
- $R_X((u,k)) = (u,k+1)$;
- $V((u,k)) = \{p \mid p_k \in V'(u)\}$.

We can prove by induction on $\psi \in \mathcal{L}_{[\{i\}],[AGT],X}$ that for all $u \in W'$, for all $k \in \mathbb{N}$, $\mathcal{M}, (u,k) \models \psi$ iff $\mathcal{M}', u \models tr_k(\psi)$. In particular, we have $\mathcal{M}, (w,0) \models \varphi$. ■

As an exercise, we can check that if we restrict group STIT plus "next operator" to coalitions in the lattice of Figure 7.3 we can also define a translation in the same flavour than Definition 58 and obtain the same result of complexity for the satisfiability problem, that is to say in NEXPTIME.

7.2.2 When there is only one agent

Theorem 35 *The satisfiability problem of a given formula in $\mathcal{L}_{X,[\emptyset],\{1\}}$ is PSPACE-hard.*

Proof.

The logic K is the logic of all trees [BDRV02]. Its satisfiability problem is PSPACE-hard. We are going to reduce the satisfiability problem of K to the satisfiability problem of a given formula in $\mathcal{L}_{X,[\emptyset],\{1\}}$. Here is the translation: $tr(\Box\psi) = [\emptyset]Xtr(\psi)$. We prefer to leave it to the reader.

■

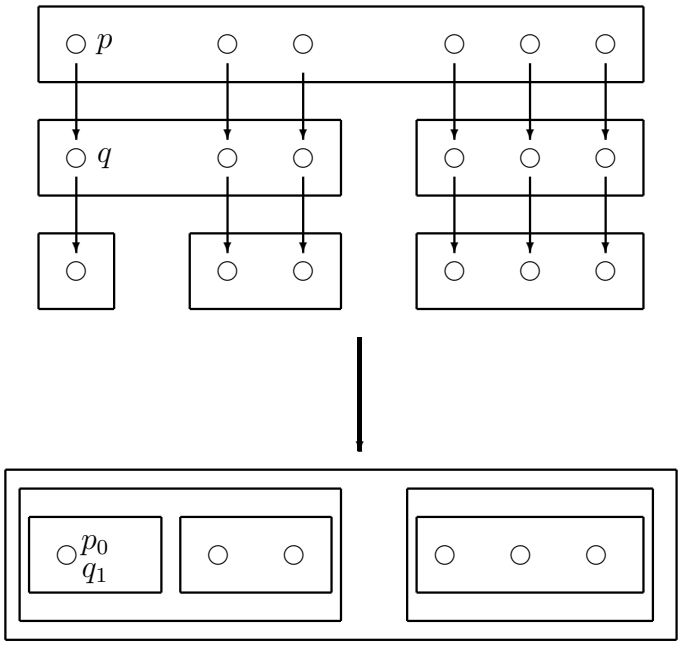


Figure 7.10: Flatten the time

Corollary 7 *The satisfiability problem of* $\mathcal{L}_{[\{1...i\}]variable}$ *(unbounded chain of coalitions) is PSPACE-hard.*

PROOF.
By Theorem 35 and Theorem 34.
■

The PSPACE-ness of the satisfiability problem of a given formula in $\mathcal{L}_{X,[\emptyset],\{1\}}$ is the case because we can embed $\mathcal{L}_{X,[\emptyset],\{1\}}$ with one agent and with the next operator into the logic $\mathcal{L}_{[\{1...i\}]\text{variable}}$. We use the same translation than in the previous subsection but with $m = 2$.

Corollary 8 *The satisfiability problem of* $\mathcal{L}_{X,[\emptyset],\{1\}}$ *is in PSPACE.*

PROOF.
By Theorem 34 and Theorem 32.
■

Let ${\mathcal{L}_{X,[\emptyset],\{1\}}}^n$ is the language of all formulas of $\mathcal{L}_{X,[\emptyset],\{1\}}$ where the X-modal depth is fixed at most n.

Corollary 9 *The satisfiability problem of a formula in* ${\mathcal{L}_{X,[\emptyset],\{1\}}}^n$ *is NP-complete.*

PROOF.
By Theorem 34 and Corollary 6. ■

7.3 Conclusion and perspectives

In this Chapter we have seen that the satisfiability problem of group STIT is undecidable in the general case $card(AGT) \geq 3$. Furthermore, this logic is not finitely axiomatizable in the general case $card(AGT) \geq 3$. We have given complete axiomatizations of group STIT fragments and studied the complexity of their satisfiability problems. In this sense, we have filled the gap between two previous results:

- the satisfiability problem of a given formula of individual STIT is NEXPTIME-complete [BHT08];
- the satisfiability problem of a given formula of group STIT without restriction is undecidable [HS08]

We have broken the myth saying STIT with coalitions is undecidable. In order to get a decidable logic, we can use coalitions in STIT but the coalitions we write into formulas must be part of a specific lattice (see Figure 7.3). You can note that the fragment where coalitions are this lattice does not make the difference between a normal-Kripke-model and a super-additive-model. In some sense it works like for the Coalition Logic [Pau02] which also does not make this difference. We want that the reader pay attention to the following conjecture:

Conjecture 1 *Let us consider a STIT fragment $\mathcal{L}_{fragment}$ with temporal operator. The satisfiability problem of a formula in $\mathcal{L}_{fragment}$ is decidable iff the language $\mathcal{L}_{fragment}$ does not capture the difference between additive and super-additive models.*

We have also extended this work to the satisfiability problem of individual STIT with the "next". Of course, the "next" operator is a weak operator with a poor expressive power. We realize even more this weakness when we have translated individual STIT plus "next" operator into group STIT in Section 7.2 and inherited complexities results of the satisfiability problem. Figure 7.11 sums up the different results.

The ultimate aim is to identify an expressive fragment of strategic STIT with a "next", "in the future" and "until" operator embedding ATL [BHT06a] and to prove its satisfiability problem to be NEXPTIME-complete. Thus we will be able to reason about counterfactual emotions, etc. while having powerful time operators like in ATL.

Language of φ	Complexity	s-a model	STIT-model
$\mathcal{L}_{[\{1...i\}]\text{fixed}}$ $(\mathcal{L}_{X,[\emptyset],\{1\}^n})$	NP	$\lvert\varphi\rvert^n$	$\lvert\varphi\rvert^n$
$\mathcal{L}_{[\{1...i\}]\text{variable}}$ $(\mathcal{L}_{X,[\emptyset],\{1\}})$	PSPACE	$\lvert\varphi\rvert^{\Delta\varphi+1}$	$\lvert\varphi\rvert^{\Delta\varphi+1}$
$\mathcal{L}_{[\{i\}]}$	NEXPTIME	$2^{\lvert\varphi\rvert}$	$2^{\lvert\varphi\rvert}$
$\mathcal{L}_{[\{i\}],[AGT]}$	NEXPTIME	$2^{\lvert\varphi\rvert}$	$2^{(n+2)\lvert\varphi\rvert}$
$\mathcal{L}_{lat}$ $(\mathcal{L}_{[\{i\}],[AGT],X})$	NEXPTIME	$2^{(\Delta\varphi)O(\lvert\varphi\rvert^2)}$	$2^{n(n+2)O(\lvert\varphi\rvert^2)}$
$\mathcal{L}_{[\{J\}]}$ $(\mathcal{L}_{[\{J\},X]})$	undecidable	...	...

where:

- $n = card(AGT)$;
- $\Delta\varphi$ represents the maximal difference of type of operators in φ;
- s-a means super-additive model.

Figure 7.11: Exact complexities of the satisfiability problem of a given formula φ

Open questions

- Assume that ATM is finite. What is the complexity of the satisfiability problem of a formula φ of $S5^2$ (or STIT with two agents)? (thanks to Mikhail Rybakov)
- Study the link between capturing the difference between additivity and super-additivity and decidability of STIT?

Chapter 8

A weak STIT fragment

Unfortunately, in the Chapter 6 (and also in [HS08]) group STIT has been proved to be undecidable and unaxiomatizable (with a finite number of axioms schemas, necessitation rules and modus ponens).

Here we here introduce a decidable and axiomatizable fragment of STIT with agents and groups called *df*STIT which is sufficiently expressive to formalize counterfactual emotions. First, in Subsection 8.1, we recall the syntax of STIT and define the syntactic fragment *df*STIT. In Subsection 8.2, we recall definition of models of the logic STIT but here we suppose that $R_{AGT} = id_W$. Then, in Subsection 8.3, we recall the logic NCL [BGH$^+$08, Tro07, Sch07]. The logic NCL shares the same syntax with STIT and its semantics looks like the semantics of STIT. in fact NCL is the logic STIT where you have replaced the notion of additivity by *super-additivity*. As you may guess, NCL is axiomatizable. The logic NCL will be a key point to prove the decidability of the another STIT fragment *df*STIT and to give a complete axiomatization of *df*STIT (Subsection 8.4) inherited from NCL axiomatization. This work is part of [LS09].

8.1 Syntax

Let us recall the syntax of STIT. Let n be a strictly positive integer. Let ATM be a countable set of atomic propositions and let $AGT = \{1, \ldots, n\}$ be a finite set of agents. The language $\mathcal{L}_{\mathsf{STIT}}$ of the logic STIT with agents and groups proposed by Horty [Hor01a] is defined by the following BNF:

$$\varphi ::= p \mid \varphi \wedge \varphi \mid \neg\varphi \mid [J]\varphi$$

where p ranges over ATM and J over 2^{AGT}. $\langle J\rangle\varphi$ is an abbreviation of $\neg[J]\neg\varphi$. Operators of type $[J]$ are used to describe the effects of the action that has been chosen by J. If J is a singleton we refer to J as an *agent*, whereas if J has more

than one element we refer to J as a *group*. We call *joint actions* the actions chosen by groups. If J has more than one element the construction $[J]\varphi$ means "group J sees to it that φ no matter what the other agents in $AGT \setminus J$ do". If J is a singleton $\{i\}$ the construction $[\{i\}]\varphi$ means "agent i sees to it that φ no matter what the other agents in $AGT \setminus \{i\}$ do". For notational convenience, we write $[i]$ instead of $[\{i\}]$. $[\emptyset]\varphi$ can be shorten to "φ is necessarily true". The operator $[\emptyset]$ is exactly the *historic necessity operator* already present in the individual STIT logic [BPX01]. The dual expression $\langle\emptyset\rangle\varphi$ means "φ is possibly true". Note that the operators $\langle\emptyset\rangle$ and $[J]$ can be combined in order to express what agents and groups can do: $\langle\emptyset\rangle[J]\varphi$ means "J can see to it that φ whatever the other agents in $AGT \setminus J$ do".

Here we are interested in a fragment of $\mathcal{L}_{\mathsf{STIT}}$ we call $\mathcal{L}_{df\mathsf{STIT}}$. It is defined by the following BNF:
$\chi ::= \bot \mid p \mid \chi \wedge \chi \mid \neg\chi$ (propositional formulas)
$\psi ::= [J]\chi \mid \psi \wedge \psi$ (see-to-it formulas)
$\varphi ::= \chi \mid \psi \mid \varphi \wedge \varphi \mid \neg\varphi \mid \langle\emptyset\rangle\psi$ (see-to-it and "can" formulas)
where p ranges over ATM and J over $2^{AGT} \setminus \{\emptyset\}$.

$\mathcal{L}_{df\mathsf{STIT}}$ is a syntactic restriction of $\mathcal{L}_{\mathsf{STIT}}$. We have $\mathcal{L}_{df\mathsf{STIT}} \subseteq \mathcal{L}_{\mathsf{STIT}}$ but $\mathcal{L}_{\mathsf{STIT}} \not\subseteq \mathcal{L}_{df\mathsf{STIT}}$. For instance, $[\{1\}][\{1,2\}]p$ is in $\mathcal{L}_{\mathsf{STIT}}$ but is not in $\mathcal{L}_{df\mathsf{STIT}}$.

8.2 Models

We give two semantics of STIT. It is proved in [HS08] that these two semantics are equivalent. The first one corresponds to the original semantics of STIT with agents and groups proposed by Horty [Hor01a]. The other one is based on the product logic $S5^n$ [GKWZ03] and will be used in Section 8.4 in order to characterize the satisfiability of a *df*STIT-formula. Let us first recall the original semantics of STIT, except that we suppose determism (the actions of agents completely determine the world):

Definition 59 (STIT-model)
A STIT-model is a tuple
$\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ where:

- W is a non-empty set of possible worlds or states;
- For all $J \subseteq AGT$, R_J is an equivalence relation over W such that:
  1. $R_J \subseteq R_\emptyset$;
  2. $R_J = \bigcap_{j \in J} R_{\{j\}}$;
  3. for all $w \in W$, for all $(w_j)_{j \in AGT} \in R_\emptyset(w)^n$, $\bigcap_{j \in AGT} R_{\{j\}}(w_j) \neq \emptyset$;

4. $R_{AGT} = id_W$.

- V is a valuation function, that is, $V : W \to 2^{ATM}$.

As in the previous Constraint 3, it is convenient to view relations on W as functions from W to 2^W, that is, for every $J \in 2^{AGT}$, $R_J(w) = \{v \in W \mid wR_Jv\}$. $R_J(w)$ represents the actual action chosen by J in the world w: if wR_Jv then v is an outcome of the action chosen by J at w. We recall that $R_\emptyset$ is the relation over all possible outcomes: if w is the current world and $wR_\emptyset v$ then v is a possible outcome at w. Thus, Constraint 1 on STIT models just means that all outcomes brought about by J are possible outcomes. Constraint 2 just says that the set of outcomes brought about by J at a given world w is equal to the pointwise intersection of the sets of outcomes brought about by the agents in J at w. Constraint 3 expresses a so-called *assumption of independence of agents*: if $w_1, \ldots, w_n$ are possible outcomes at w then the intersection of the set of outcomes that agent 1 brings about at w_1, and the set of outcomes that agent 2 brings about at w_2,..., and the set of outcomes that agent n brings about at w_n is not empty. More intuitively, this means that agents can never be deprived of choices due to the choices made by other agents. Constraint 4 expresses an assumption of determinism: the set of outcomes brought about by all agents is a singleton that is to say we have $R_{AGT}(w) = \{w\}$ for all $w \in W$.

Truth conditions for atomic formulas and the boolean operators are entirely standard. For every $J \in 2^{AGT}$, the truth conditions of the modal operators $[J]$ are classicaly defined by:

$$\mathcal{M}, w \models [J]\varphi \text{ iff } \mathcal{M}, v \models \varphi \text{ for all } v \in W \text{ such that } wR_Jv.$$

The alternative semantics of STIT is based on the product logic $S5^n$. It is defined as follows:

Definition 60 (product STIT-model)
A product STIT-model is a tuple $\mathcal{M} = (W, V)$ where:

- $W = W_1 \times \cdots \times W_n$ where W_i are non-empty sets of worlds or states;
- V is a valuation function, that is, $V : W \to 2^{ATM}$.

Truth conditions for atomic formulas and the boolean operators are also entirely standard. The truth conditions for the modal operators $[J]$ in product STIT-models are:

$$\mathcal{M}, (w_1, \ldots, w_n) \models [J]\varphi \text{ iff } \mathcal{M}, (v_1, \ldots, v_n) \models \varphi$$
$$\text{for all } (v_1, \ldots, v_n) \in W \text{ such that } v_j = w_j \text{ if } j \in J.$$

Now let us just recall the notion of validity and satisfiability in STIT. As there is an equivalence between a STIT-model and a product STIT-model as proved by [HS08], we can define those notions either with STIT-models or with STIT-models. A formula φ is STIT-valid (noted $\models_{\mathsf{STIT}} \varphi$) if and only if φ is true in every world of every STIT-model. Or, equivalently, a formula φ is STIT-valid if and only if φ is true in every world of every product STIT-model. A formula φ is STIT-satisfiable if and only if there exists a STIT-model $\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ and a point $w \in W$ such that $\mathcal{M}, w \models \varphi$. Or, equivalently, a formula φ is STIT-satisfiable if and only if there exists a product STIT-model $\mathcal{M} = (W, V)$ and a point $w \in W$ such that $\mathcal{M}, w \models \varphi$.

8.3 The NCL logic

Unfortunately, STIT is not axiomatizable. Nevertheless, there exists an axiomatizable logic which is very close to STIT. This logic is the fragment of the Normal Coalition Logic [BGH$^+$08, Tro07, Sch07, BHT07b] in which we do not deal with the *next* operator. The Normal Coalition Logic was originally proposed in order to embed the non-normal Coalition Logic CL [Pau02] into a *normal* modal logic. This embedding uses a general technique developed in [GH93]. The reader can find more details about this specific embedding in [BGH$^+$08, Tro07, BHT07b]. Furthermore, the Normal Coalition Logic is also axiomatizable and decidable as CL. Below we show that the fragment of this logic without time axiomatizes the set of validities in the fragment $\mathcal{L}_{df\mathsf{STIT}}$ of STIT. Moreover, we prove our central characterization theorem of STIT-satisfiable formula of the fragment $\mathcal{L}_{df\mathsf{STIT}}$ by using the Normal Coalition Logic without time. From now on, we define NCL as the fragment of the Normal Coalition Logic with the group operators $[J]$ and without the *next* operator.

8.3.1 Definition

We start by giving the definition of the logic NCL. Concerning the syntax, as here we do not deal with the *next* operator, the language of NCL-formulas is the same as the language of STIT-formulas, that is to say, $\mathcal{L}_{\mathsf{NCL}} = \mathcal{L}_{\mathsf{STIT}}$.

Concerning the semantics, here is the definition of a NCL-model:

Definition 61 (NCL-model)
A NCL-model is a tuple $\mathcal{M} = (W, R, V)$ where:

- W is a nonempty set of worlds or states;
- R is a collection of equivalence relations R_J (one for every coalition $J \subseteq AGT$) such that:

1. $R_{J_1 \cup J_2} \subseteq R_{J_1} \cap R_{J_2}$;
2. $R_{\emptyset} \subseteq R_J \circ R_{AGT \setminus J}$;
3. $R_{AGT} = Id_W$.

- $V : W \to 2^{ATM}$ is a valuation function.

As in Definition 59, $R_J(w)$ represents the set of outcomes of the action chosen by the group J. Constraint 1 says that the outcomes of the action chosen by the group $J_1 \cup J_2$ are outcomes of the action chosen by the group J_1 and outcomes of the action chosen by the group J_2. Constraint 2 is close to the assumption of independence of agents of STIT logic. According to Constraint 2, if v is a possible outcome at w then, there must exists a world u such that u is an outcome of the action chosen by group J at w and v is an outcome of the action chosen by group $AGT \setminus J$ at u. Constraint 3 expresses an assumption of determinism.

As usual truth conditions for atomic formulas and the boolean operators are entirely standard and the truth conditions of the operators $[J]$ are given in a traditional way by:

$$\mathcal{M}, w \models [J]\varphi \text{ iff } \mathcal{M}, v \models \varphi \text{ for all } v \in W \text{ such that } wR_Jv.$$

In the same way, we introduce notions of validity and satisfiability in NCL. A formula φ is NCL-valid (noted $\models_{\mathsf{NCL}} \varphi$) if and only if φ is true in every world of every NCL-model. A formula φ is NCL-satisfiable if and only if there exists a NCL-model $\mathcal{M} = (W, R, V)$ and a point $w \in W$ such that $\mathcal{M}, w \models \varphi$.

8.3.2 Axiomatization of NCL

Constraints 1, 2, 3 presented in the Definition 61 above directly correspond to Sahlqvist axiom schemas [BDRV02]. For instance Constraint 2 ($R_{\emptyset} \subseteq R_J \circ R_{AGT \setminus J}$) corresponds to the axiom schema $\langle\emptyset\rangle\varphi \to \langle J\rangle\langle AGT \setminus J\rangle\varphi$. This is the reason why NCL logic is axiomatizable unlike STIT logic. The following Theorem 36, which has been proved by [BHT07b], sums up this fact.

Theorem 36 *The logic NCL is complete with respect to the following axiomatization:*

(ProTau)	*all tautologies of propositional calculus*
S5([J])	*all S5-theorems, for every* $[J]$
(Mon)	$[J_1]\varphi \vee [J_2]\varphi \to [J_1 \cup J_2]\varphi$
Elim(∅)	$\langle\emptyset\rangle\varphi \to \langle J\rangle\langle AGT \setminus J\rangle\varphi$

Triv(AGT) $\varphi \rightarrow [AGT]\varphi$

plus modus ponens and necessitation for all $[J]$.

PROOF.
Sahlqvist theorem. [BDRV02] ■

As NCL is axiomatizable, we can introduce the symbol $\vdash_{\mathsf{NCL}}$ to deal with proofs. We write $\vdash_{\mathsf{NCL}} \varphi$ to say that φ is a theorem of the axiomatization given in Theorem 36.

8.3.3 Link between STIT and NCL

In the individual case, that is to say when the language only has operators $[\emptyset]$ and $[\{i\}]$ where $i \in AGT$, the notion of satisfiability in STIT and the notion of satisfiability in NCL are equivalent [BGH$^+$08] (Theorem 14). Nevertheless, when we authorize the whole group STIT language, the two notions are different. The following Proposition 11 highlights the relation between satisfiability in STIT and satisfiability in NCL.

Proposition 11 *Let* φ *be a formula of* $\mathcal{L}_{STIT}$.

- *If* $\mathrm{card}(AGT) \leq 2$, φ *is STIT-satisfiable iff* φ *is NCL-satisfiable;*
- *If* $\mathrm{card}(AGT) \geq 3$, φ *is STIT-satisfiable implies* φ *is NCL-satisfiable. (the converse is false: there exists* φ *such that* φ *is NCL-satisfiable and* $\neg\varphi$ *STIT-valid.)*

PROOF.
Let us prove that a STIT-model is a NCL-model. Let $\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ be STIT-model and let us prove it is an NCL model. It suffices to prove that the constraints of a NCL model are true in $\mathcal{M}$. For instance: $R_{J_1 \cup J_2} = \bigcap_{j \in J_1 \cup J_2} R_{\{j\}} = \bigcap_{j \in J_1} R_{\{j\}} \cap \bigcap_{j \in J_2} R_{\{j\}} = R_{J_1} \cap R_{J_2}$. So we have $R_{J_1 \cup J_2} \subseteq R_{J_1} \cap R_{J_2}$. Now let us prove $R_\emptyset \subseteq R_J \circ R_{AGT \setminus J}$. If $wR_\emptyset v$, then the constraints 3 of Definition 59 gives: $\bigcap_{j \in J} R_{\{j\}}(w) \cap \bigcap_{j \in \overline{J}} R_{\{j\}}(v) \neq \emptyset$. That is to say: $R_J(w) \cap R_{\overline{J}}(v) \neq \emptyset$. So $wR_J \circ R_{\overline{J}} v$.

Now given that a STIT-model is a NCL-model, whatever the cardinality of AGT, we have the implication "φ is STIT-satisfiable implies φ is NCL-satisfiable".

If $card(AGT) = 1$, we have φ is STIT-satisfiable φ is NCL-satisfiable. Indeed, both the logic STIT and NCL is fusion of the logic S5 for the operator $[\emptyset]$ and the trivial operator $[\{1\}]$.

If $card(AGT) = 2$, From [HS08] we have that STIT is exactly the logic $S5^2$ where operators are $[\{1\}]$ and $[\{2\}]$. (we do not care about operators $[\{1,2\}]$

and $[\emptyset]$ because we have the two validities $[\{1,2\}]\varphi \leftrightarrow \varphi$ and $[\emptyset]\varphi \leftrightarrow [\{1\}][\{2\}]\varphi$.) Concerning NCL, Directly from the axiomatics of NCL, we have that NCL is exactly $[S5, S5]$ where operators are $[\{1\}]$ and $[\{2\}]$. As $S5^2 = [S5, S5]$ [GKWZ03], we have that STIT and NCL have the same satisfiable formulas.

If $card(AGT) \geq 3$, the problem of satisfiability of NCL is in NEXPTIME (see [BGH$^+$08] or [Sch07]) whereas the problem of satisfiability of STIT is undecidable (see [HS08]). So the two logics do not have the same satisfiable formulas.

To sum up, we have:

- If $card(AGT) = 1$, STIT is the same logic than $S5$ and NCL;
- If $card(AGT) = 2$, STIT is the same logic than $S5^2$ and NCL;
- If $card(AGT) \geq 3$, we have:
  - STIT is the same logic than $S5^{card(AGT)}$;
  - If a formula is STIT-satisfiable then it is NCL-satisfiable. There exists a NCL-satisfiable which is not STIT-satisfiable.

■

Although the satisfiability problem of NCL is NEXPTIME-complete [BGH$^+$08], hence decidable, we can not use NCL to capture reasoning about choices like STIT because there are no philosophical justifications of this logic. NCL was only devoted to embed Coalition Logic into a decidable normal modal logic. Moreover NCL and STIT already differs with a formula of modal depth 3. Indeed, the formula $\varphi = \neg[\langle\{2,3\}\rangle p \wedge \langle\{1,3\}\rangle q \wedge \langle\{1,2\}\rangle r \rightarrow \langle\emptyset\rangle[\langle\{2,3\}\rangle(\langle\{1,3\}\rangle p \wedge \langle\{2,3\}\rangle q) \wedge \langle\{1,3\}\rangle(\langle\{2,3\}\rangle r \wedge \langle\{1,2\}\rangle p) \wedge \langle\{2,3\}\rangle(\langle\{1,2\}\rangle q \wedge \langle\{1,3\}\rangle r)]]$ is NCL-satisfiable and $\neg\varphi$ is STIT-valid. [GKWZ03] We leave to the reader the proof that $\neg\varphi$ is STIT-valid and you can check that φ is indeed satisfiable in the NCL-model depicted in Figure 8.1. This model is an NCL-model for $AGT = \{1,2,3\}$. It contains 25 worlds. The relations $R_{\{1,2\}}$, $R_{\{1,3\}}$, $R_{\{2,3\}}$ are defined by the picture. $R_\emptyset = W \times W$ and $R_{AGT} = id_W$. Then R_1 is defined as $R_{\{1,2\}} \circ R_{\{1,3\}}$, R_2 is defined as $R_{\{1,2\}} \circ R_{\{2,3\}}$ and R_3 is defined as $R_{\{1,3\}} \circ R_{\{2,3\}}$.

It is an open question to know if NCL and STIT differs with a formula of modal depth 2.

Although the two logics NCL and STIT are different, the property of independence of agents holds in NCL. This fact is stated in the following Lemma 12 and illustrated in Figure 8.2. Every NCL-model satisfies the constraint 3 (*assumption of independence of agents*) of Definition 59. This property will be important in the constructive proof of Theorem 37. More precisely, it will be used in the proof of Lemma 13.

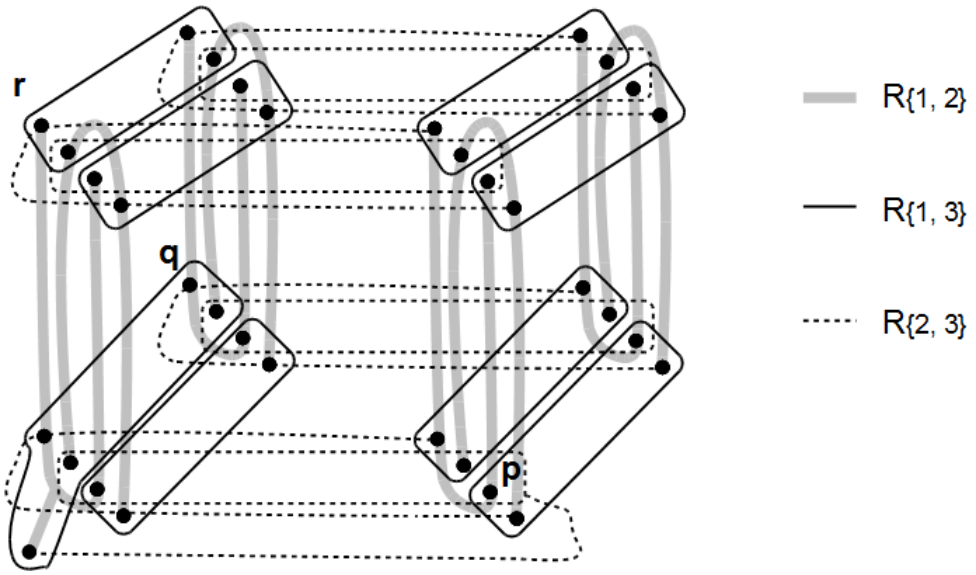


Figure 8.1: A NCL-model for $\neg[\langle\{2,3\}\rangle p \wedge \langle\{1,3\}\rangle q \wedge \langle\{1,2\}\rangle r \rightarrow \langle\emptyset\rangle[\langle\{2,3\}\rangle(\langle\{1,3\}\rangle p \wedge \langle\{2,3\}\rangle q) \wedge \langle\{1,3\}\rangle(\langle\{2,3\}\rangle r \wedge \langle\{1,2\}\rangle p) \wedge \langle\{2,3\}\rangle(\langle\{1,2\}\rangle q \wedge \langle\{1,3\}\rangle r)]]$

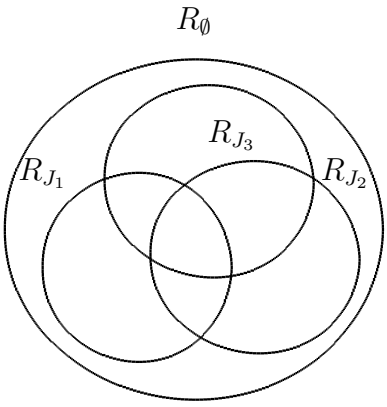


Figure 8.2: Independence of agents in NCL

Lemma 12 *Let $\mathcal{M} = (W, R, V)$ be a NCL-model. Let r be a positive integer*[1]*. Let $w_1, \dots w_r \in W$ be such that for all $i, j \in \{1, \dots r\}$, $w_i R_\emptyset w_j$. Let $J_1, \dots J_r \subseteq AGT$ such that $i \neq j$ implies $J_i \cap J_j = \emptyset$. We have:*

$$\bigcap_{i=1\dots r} R_{J_i}(w_i) \neq \emptyset.$$

PROOF.

Let us prove the lemma by recurrence on $r \in \mathbb{N}^*$. Let us call $\mathcal{P}(r)$ the statement of the lemma.

- $\mathcal{P}(1)$ is true.
- Let us prove $\mathcal{P}(2)$ because we need it in order to prove $\mathcal{P}(r+1)$ from $\mathcal{P}(r)$.

  Let u and w be in W such that $uR_\emptyset w$. Let $J, K \subseteq AGT$ such that $J \cap K = \emptyset$. As $uR_\emptyset w$, we have $uR_J \circ R_{\overline{J}} w$. And then $uR_J \circ R_K w$. This proves $\mathcal{P}(2)$.
- Now, assume that $\mathcal{P}(r)$ is true for a fixed $r \in \mathbb{N}^*$ and let us prove $\mathcal{P}(r+1)$ is true. Let $w_1, \dots, w_r, w_{r+1} \in W$ such that for all $i, j \in \{1, \dots r\}$, $w_i R_\emptyset w_j$. Let $J_1, \dots, J_r, J_{r+1} \subseteq AGT$ such that $i \neq j$ implies $J_i \cap J_j = \emptyset$. As $P(r)$ is assumed, we can apply it on $(w_1, \dots, w_r)$ and $(J_1, \dots, J_r)$ and obtain $\bigcap_{i=1\dots r} R_{J_i}(w_i) \neq \emptyset$. Let $w \in \bigcap_{i=1\dots r} R_{J_i}(w_i)$. Now consider $R_{\bigcup_{i=1\dots r} J_i}(w)$ and $R_{J_{r+1}}(w_{r+1})$. By applying $P(2)$ on (w, w_{r+1}), and $(\bigcup_{i=1\dots r} J_i, J_{r+1})$, we obtain that $R_{\bigcup_{i=1\dots r} J_i}(w) \cap R_{J_{r+1}}(w_{r+1})$ is not empty, i.e. $R_{\bigcup_{i=1\dots r} J_i}(w) \cap R_{J_{r+1}}(w_{r+1})$ contains a point v. Notice that by point 1. of Definition 61 we have $R_{\bigcup_{i=1\dots r} J_i}(w) \subseteq \bigcap_{i=1\dots r} R_{J_i}(w)$. As $\bigcap_{i=1\dots r} R_{J_i}(w) \subseteq \bigcap_{i=1\dots r} R_{J_i}(w_i)$, we have a point v in $\bigcap_{i=1\dots r+1} R_{J_i}(w_i)$. In other words, $\mathcal{P}(r+1)$ is true.

Conclusion: We have proved by recurrence that for all $r \geq 1$, $\mathcal{P}(r)$ is true. ■

Our fragment *df*STIT of STIT logic with agents and groups has interesting computational properties. In the rest of this section, we are going to show that *df*STIT can be axiomatized by the axiomatics of the logic NCL, and that *df*STIT is decidable. To prove this, we are going to study the link between NCL and STIT when we restrict formulas to the fragment *df*STIT. Proposition 11 given above explains that in the general case, if a formula is STIT-satisfiable then it is NCL-satisfiable. The following Theorem 37 explains that the notion of satisfiability in STIT and in NCL is the same if we restrict formulas to the fragment *df*STIT.

Theorem 37 *Let $\varphi \in \mathcal{L}_{dfSTIT}$. Then, the following three propositions are equivalent:*

[1]You can notice that the Lemma 12 in the degenerated case $r = 0$ says that the intersection of zero subset is not empty. Indeed the intersection of zero subset of $W \times W$ is equal to $W \times W$ itself and therefore is not empty.

1. *φ is NCL-satisfiable;*
2. *φ is STIT-satisfiable;*
3. *φ is STIT-satisfiable in a polynomial sized product STIT-model.*

PROOF.

Given that proving that 2. implies 1. is exactly the Proposition 11, we focus here on the proof of 1. implies 3. and we use a selection-of-points argument as in [Lad77]. Let φ a NCL-satisfiable formula: there exists a NCL–model $\mathcal{M} = (W, V)$ and z_0 such that $\mathcal{M}, z_0 \models \varphi$. We first construct from $\mathcal{M}$ a product STIT-model $\mathcal{M}' = (W', V')$. Secondly we ensure that that there exists a point $(Z_0, \ldots Z_0) \in W'$ such that $\mathcal{M}', (Z_0, \ldots Z_0) \models \varphi$. Broadly speaking, we take care in the construction to create a new point in $\mathcal{M}'$ for each subformula $\langle\emptyset\rangle\psi$ true in $\mathcal{M}$. We also take care to construct enough points so that all subformulas $\langle\emptyset\rangle\psi$ and $[J]\chi$ false in $\mathcal{M}, z_0$ can also be false in t $\mathcal{M}'$.

Notations

- Elements of W are noted x, y etc.
- elements of W' are noted $\vec{x}$, $\vec{x}_0$, $\vec{y}$ etc. x_j stands for the j-th coordinate of $\vec{x}$. Given an element $\vec{x}$, we note $\vec{x}_J = (x_j)_{j\in J}$;
- $(P, \ldots, P)$ denotes the vector $\vec{x}$ where for all $j \in AGT$, $x_j = P$;
- $(P, \ldots, P)_J$ denotes $\vec{x}_J$ where for all $j \in J$, $x_j = P$;
- $SF(\varphi)$ denotes the set of all subformulas of φ. $SF_1(\varphi)$ is the set of all subformulas which are not in the scope of a modal operator and which are of the form $[J]\chi$ where χ is propositional. For instance, if $\varphi = [\{1\}]p \wedge \langle\emptyset\rangle[\{2\}]q$, then $SF(\varphi) = \{p, q, \{1\}]p, [\{2\}]q, \langle\emptyset\rangle[\{2\}]q, \varphi\}$ whereas $SF_1(\varphi) = \{[\{1\}]p\}$.

Part 1: we define the model $\mathcal{M}'$

The definition of $\mathcal{M}'$ relies on the two following sets of formulas:

- $Pos = \{\psi \mid \langle\emptyset\rangle\psi \in SF(\varphi) \text{ and } \mathcal{M}, z_0 \models \langle\emptyset\rangle\psi\} \cup \{Z_0\}$
  where $Z_0 = \bigwedge\{[J]\chi \mid [J]\chi \in SF_1(\varphi) \text{ and } \mathcal{M}, z_0 \models [J]\chi\}$. Formulas in Pos are called *positive formulas.*
- $Neg = \{[J]\chi \mid [J]\chi \in \psi \text{ and } \langle\emptyset\rangle\psi \in SF(\varphi) \text{ and } \mathcal{M}, z_0 \not\models \langle\emptyset\rangle\psi\} \cup Neg\_in\_z_0$ where $Neg\_in\_z_0 = \{[J]\chi \mid [J]\chi \in SF_1(\varphi) \text{ and } \mathcal{M}, z_0 \not\models [J]\chi\}$. Formulas in Pos are called *negative formulas.*

Example 19 *Let* $\varphi = \langle\emptyset\rangle([1]\chi_1 \wedge [\{1,3\}]\chi_2) \wedge \neg\langle\emptyset\rangle([2]\chi_3 \wedge [4]\chi_4) \wedge [5]\chi_5 \wedge [6]\chi_6 \wedge \neg[7]\chi_7 \wedge \neg[8]\chi_8$.

We have:

- $Z_0 = [5]\chi_5 \wedge [6]\chi_6$;
- $Pos = \{[1]\chi_1 \wedge [\{1,3\}]\chi_2, [5]\chi_5 \wedge [6]\chi_6\}$;
- $Neg\_in\_z_0 = \{[7]\chi_7, [8]\chi_8\}$;
- $Neg = \{[2]\chi_3, [4]\chi_4, [7]\chi_7, [8]\chi_8\}$.

First we define the cartesian product $W' = C^n = C \times C \times \ldots C$ where $C = Pos \cup \{0, \ldots, card(Neg) - 1\}$.

The vector $(Z_0, \ldots Z_0)$ is the root of the model. For instance, if $\langle\emptyset\rangle P = \langle\emptyset\rangle([1]p \wedge [\{2,3\}]q)$ is true in $\mathcal{M}$, then the point $(P, \ldots, P)$ will be both the root of the hyperplane $\{(P, \alpha_2, \ldots \alpha_n) \in W' \mid \alpha_2, \ldots \alpha_n \in C\}$ where p is true and the root of the space of dimension $n-2$ $\{(\alpha_1, P, P, \alpha_4 \ldots \alpha_n) \in W' \mid \alpha_1, \alpha_4, \ldots \alpha_n \in C\}$ where q is true. Idea for negative formulas will be explained later.

Secondly we will define the valuation V'. Before that, we introduce few notations and prove the following Lemma 13 which is a bridge between W' and W.

- For all $\vec{x} \in W'$, for all $P \in Pos$, we consider the set:

$$Coord^{\vec{x}}_{=P} = \{j \in AGT \mid x_j = P\};$$

  Given a vector $\vec{x}$ and a positive formula $P \in Pos$, the set $Coord^{\vec{x}}_{=P}$ denotes the set of the agents j such that the coordinate j of the vector $\vec{x}$ is equal to P.

- For all $\vec{x} \in W'$, we consider the set:

$$Pos_{\vec{x}} = \{\chi \mid P \in Pos, [J]\chi \in SF(P), J \subseteq Coord^{\vec{x}}_{=P}\};$$

  $Pos_{\vec{x}}$ denotes a set of boolean formulas that must be true in $\vec{x}$ because of positive formulas. Formulas are boolean because of the syntactic restriction over the language (Definition of dfSTIT). For instance let us consider the positive formula $P = [1]p \wedge [\{2,3\}]q$. The model $\mathcal{M}'$ will be designed such that the point $(P, \ldots, P)$ is the world where P must be true. Indeed, as we have defined $Pos_{(P,\alpha_2,\ldots\alpha_n)}$ so that it contains p, the formula p must be true in the hyperplane $\{(P, \alpha_2, \ldots \alpha_n) \in W' \mid \alpha_2, \ldots \alpha_n \in C\}$.

- For all $\vec{x} \in W'$, we consider the formula

$$Boxes_{\vec{x}} = \bigwedge_{\chi \in Pos_{\vec{x}}} \chi$$

  $Boxes_{\vec{x}}$ corresponds to the conjunction of all (boolean) formulas which have to be true in $\vec{x}$ because of $[J]\chi$ positive formulas.

- We fix a bijection $i : \{0, \ldots, card(Neg) - 1\} \to Neg$;

  We need such a bijection between integers in $\{0, \ldots, card(Neg) - 1\}$ and Neg in order to use arithmetic operations $+$ and $\bmod$ (modulo) in the following construction. Those arithmetic operations enables us to define the model in an easier way. The modulo enables us to permute negative formulas in the model.

- We extend i to a function from W' to Neg in the following way:

$$i(\vec{x}) = i\left(\Sigma_{i \in \{1,\ldots,n\} | x_i \in \{0,\ldots,card(Neg)-1\}} x_i \bmod card(Neg)\right).$$

  where $\bmod$ is the operation of modulo.

  Intuitively, $i(\vec{x})$ will correspond to the negative formula $[J]\chi$ which will be false at $\vec{x}$ if there are no contradictions with $Boxes_{\vec{x}}$.

Lemma 13 *For all $\vec{x} \in W'$, there exists $y \in W$ such that $\mathcal{M}, y \models Boxes_{\vec{x}}$.*

PROOF.

We just recall that by definition of Pos, we have that for all $P \in Pos$, $\mathcal{M}, z_0 \models \langle\emptyset\rangle P$. So for all $P \in Pos$, there exists a point $y_P \in W$, such that $\mathcal{M}, y_P \models P$.

Let $\vec{x} \in W'$. In the proof, first we define $y \in W$. Then we prove $\mathcal{M}, y \models Boxes_{\vec{x}}$.

1. First, we define the candidate $y \in W$ of our Lemma 13. As $\mathcal{M}$ is an NCL-model, $\mathcal{M}$ satisfies the *assumption of independence of agents* (Lemma 12). We are simply going to apply Lemma 12 where points are $\{y_P \mid P \in Pos\}$ and sets of agents are $\{Coord^{\vec{x}}_{=P}, P \in Pos\}$. We take care that sets $Coord^{\vec{x}}_{=P}$ are disjoint if P ranges over Pos. Indeed, for all $P, Q \in Pos$, $Coord^{\vec{x}}_{=P} \cap Coord^{\vec{x}}_{=Q} \neq \emptyset$, implies there exists $j \in Coord^{\vec{x}}_{=P} \cap Coord^{\vec{x}}_{=Q}$. By Definition of $Coord^{\vec{x}}_{=P}$, we have $x_j = P$. In the same way $x_j = Q$, so $P = Q$. Briefly, Lemma 12 leads to:

$$\bigcap_{P \in Pos} R_{Coord^{\vec{x}}_{=P}}(y_P) \neq \emptyset.$$

   As this set is not empty, let us take y in it. Let $y \in \bigcap_{P \in Pos} R_{Coord^{\vec{x}}_{=P}}(y_P)$.

2. We have defined $y \in W$. Now let us prove that $\mathcal{M}, y \models Boxes_{\vec{x}}$. Better said, we are going to prove that for all $\chi \in Pos_{\vec{x}}$, $\mathcal{M}, y \models \chi$.

   Let $\chi \in Pos_{\vec{x}}$. By definition of $Pos_{\vec{x}}$, there exists $P \in Pos$ and $[J]\chi \in SF(P)$ such that $J \subseteq Coord^{\vec{x}}_{=P}$. Recall that $\mathcal{M}, \vec{y}_P \models P$ and, consequently, we have $\mathcal{M}, \vec{y}_P \models [J]\chi$. By definition of y, we have $y_P R_{Coord^{\vec{x}}_{=P}} y$. But as $J \subseteq Coord^{\vec{x}}_{=P}$, we have $R_{Coord^{\vec{x}}_{=P}} \subseteq R_J$. So, we have $y_P R_J y$ and, consequently, we have $\mathcal{M}, y \models \chi$. So we have $\mathcal{M}, y \models Boxes_{\vec{x}}$.

■

Finally, we define $V' = f \circ V$ where f is a mapping from W' to W defined by:

- $f(Z_0, \ldots, Z_0) = z_0$;
- For all $\vec{x} \in W'$ such that $\vec{x} \neq (Z_0, \ldots, Z_0)$, $i(\vec{x})$ is of the form $[J]\chi \in Neg$.
  - If there exists $y \in W$, such that $\mathcal{M}, y \models \neg\chi \wedge Boxes_{\vec{x}}$ then we pose $f(\vec{x}) = y$.
  - Else, we choose a point y in W such that $\mathcal{M}, y \models Boxes_{\vec{x}}$. We pose $f(\vec{x}) = y$. We recall that there always exist such a point because of Lemma 13.

Clearly, $\mathcal{M}' = (W', V')$ is a product STIT-model and its size is polynomial. As $V' = f \circ V$, we have immediately the following lemma useful for the Part 2 of the proof.

Lemma 14 *For all $\vec{x} \in W'$, $\mathcal{M}', \vec{x} \models Boxes_{\vec{x}}$.*

PROOF.
Let $\vec{x} \in W'$. By Definition of f, $\mathcal{M}, f(\vec{x}) \models Boxes_{\vec{x}}$. But recall that $V' = f \circ V$: in particular, we have $V'(\vec{x}) = V(f(\vec{x}))$. Recall also that $Boxes_{\vec{x}}$ is a boolean formula. So we obtain $\mathcal{M}, \vec{x} \models Boxes_{\vec{x}}$. ■

Part 2 of the proof : we prove $\mathcal{M}', (Z_0, \ldots, Z_0) \models \varphi$

1. Let $P \in Pos$. Let us prove that $\mathcal{M}', (P, \ldots P) \models P$. P is a conjunction of formula of the form $[J]\chi$ where χ is a Boolean formula. Let $[J]\chi \in SF(P)$. We have to show that for $\vec{x} \in W'$ such that $\vec{x}_J = (P, \ldots, P)_J$, we have $\mathcal{M}, \vec{x} \models \chi$. The situation is drawn in Fig. 8.3. The subspace represents all worlds $\vec{x}$ of W' where J performs the same actions than in $(P, \ldots P)$. But for those $\vec{x}$, we have $J \subseteq Coord^{\vec{x}}_{=P}$. So $\chi \in Pos_{\vec{x}}$ implying that $\models Boxes_{\vec{x}} \rightarrow \chi$. But, by Lemma 14, $\mathcal{M}', \vec{x} \models Boxes_{\vec{x}}$ and this leads to $\mathcal{M}', \vec{x} \models \chi$. Finally, $\mathcal{M}', (P, \ldots, P) \models [J]\chi$. Finally, $P \in Pos$, $\mathcal{M}', (P, \ldots P) \models P$.

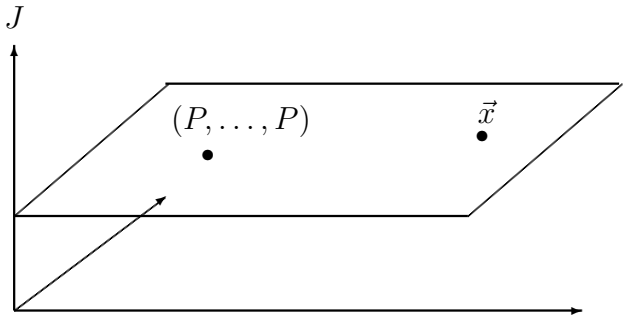


Figure 8.3: Model $\mathcal{M}'$: a point $(P, \ldots, P)$ and the subspace of all points $\vec{x}$ such that $\vec{x}_J = (P, \ldots, P)_J$, that is to say the subspace of worlds of $\mathcal{M}'$ where agents in J all perform action "P".

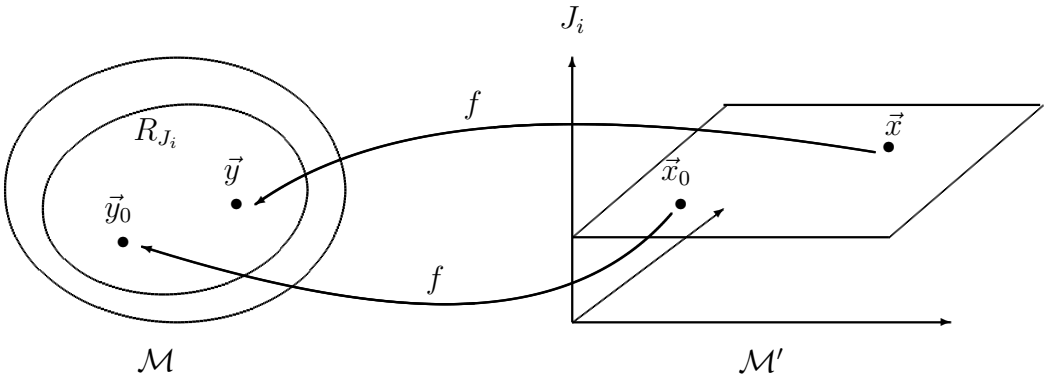


Figure 8.4: Case 2. (a) in the part 2 of the Proof of Theorem 37

2. (a) Let $N = [J_1]\chi_1 \wedge \ldots [J_k]\chi_k$ be such that $\langle\emptyset\rangle N \in SF(\varphi)$ and $\mathcal{M}, \not\models \langle\emptyset\rangle N$. Let us prove that for all $\vec{x}_0 \in W'$, $\mathcal{M}', \vec{x}_0 \models \neg N$. We suggest the reader to look at the Fig. 8.4 during this part.

   Consider $y_0 = f(\vec{x}_0) \in W$. By definition of f, we have $\mathcal{M}, y_0 \models Boxes_{\vec{x}_0}$. We also have $\mathcal{M}, y_0 \models \neg N$. So, there is $i \in \{1, \ldots, k\}$ such that $\mathcal{M}, y_0 \not\models [J_i]\chi_i$. Notice that $[J_i]\chi_i$ belongs to Neg.

   Now we are going to prove that $\mathcal{M}', \vec{x}_0 \not\models [J_i]\chi_i$. We are going to define a vector $\vec{x} \in W'$ such that $\vec{x}_0 R'_{J_i} \vec{x}$ and $\mathcal{M}', \vec{x} \models \neg\chi_i$. As depicted on the Fig. 8.4, we want that J_i performs the same actions in both $\vec{x}_0$ and $\vec{x}$.

   The case where $J_i = AGT$ is trivial: we take $\vec{x} = \vec{x_0}$. Else, let j_0 be an arbitrary agent in $\overline{J_i}$ and $\vec{x} \in W'$ be the candidate vector such that:

   - $\vec{x}_{J_i} = \vec{x_0}_{J_i}$;

- $x_j = 0$ for all $j \in \overline{J_i} \setminus \{j_0\}$;
- $x_{j_0} = i^{-1}([J_i]\neg\chi_i) - \Sigma_{j \in AGT \mid j \neq j_0 \text{ and } x_j \in \{0,\ldots,card(Neg)-1\}} x_j \mod N$

Now we check that $\mathcal{M}', \vec{x} \models \neg\chi_i$. As $\mathcal{M}, y_0 \models \langle J_i\rangle\neg\chi_i$, there exists $y \in W$ such that $y \in R_{J_i} y_0$ and $\mathcal{M}, y \models \neg\chi_i$. Notice that $\mathcal{M}, y \models Boxes_{\vec{x}}$. Indeed, $Boxes_{\vec{x}}$ only contains subformulas χ_1 provided by formulas of the form $[K]\chi_1$ from Pos, where $K \subseteq J_i$. (because only coordinates in J_i of $\vec{x}$ are in Pos; others are integer). Then we have $\models Boxes_{\vec{x_0}} \rightarrow Boxes_{\vec{x}}$. Hence $\mathcal{M}, y \models Boxes_{\vec{x}}$. To sum up, we have $\mathcal{M}, y \models Boxes_{\vec{x}} \wedge \neg\chi_i$. So, as $i(\vec{x}) = [J_i]\neg\chi_i$, by definition of f we have that $f(\vec{x})$ is a such point y where $\mathcal{M}, y \models Boxes_{\vec{x}} \wedge \neg\chi_i$. Finally, by definition of V', $\mathcal{M}', \vec{x} \models \neg\chi_i$.

(b) We prove $\mathcal{M}', (Z_0, \ldots Z_0) \models Neg\_in\_z_0$ in the same way. Let us prove that $\mathcal{M}', (Z_0, \ldots Z_0) \models Neg\_in\_z_0$. More precisely we prove that for all $[J]\chi \in Neg\_in\_z_0$, $\mathcal{M}', (Z_0, \ldots, Z_0) \models \langle J\rangle\neg\chi$. We know that $\mathcal{M}, z_0 \models \langle J\rangle\neg\chi$. So there exists $y \in W$ such that $yR_J z_0$ and $\mathcal{M}, y \models \neg\chi$. The case $J = AGT$ is trivial. Let us consider $j_0 \in \overline{J}$ and let us define the candidate $\vec{x}$:

- $\vec{x}_J = (Z_0, \ldots, Z_0)_J$;
- $x_j = 0$ for all $j \in \overline{J} \setminus \{j_0\}$;
- $\vec{x}_{j_0} = i^{-1}([J]\chi)$;

Let us check that $\mathcal{M}', \vec{x} \models \neg\chi$. Remark that $Boxes_{\vec{x}}$ only contains Boolean formulas χ' where formulas $[J']\chi'$ are subformulas of Z_0, where $J' \subseteq J$. Hence $\mathcal{M}, y \models Boxes_{\vec{x}}$. Furthermore, $\mathcal{M}, y \models \neg\chi$. So by definition of f, as $i(\vec{x}) = [J]\chi$ we have that $f(\vec{x})$ is a point y such that $\mathcal{M}, y \models \neg\chi \wedge Boxes_{\vec{x}}$. By definition of V', $\mathcal{M}', \vec{x} \models \neg\chi$.

The conclusion of the proof is left to the reader.

■

Figure 8.5 highlights the relation between STIT and NCL. If we consider the whole set of formulas $\mathcal{L}_{\mathsf{STIT}}$, then we have that all validities of NCL are validities of STIT but not the converse. But if we restrict formulas to the fragment $\mathcal{L}_{df\mathsf{STIT}}$, then the set of validities of NCL is equal to the set of validities of STIT.

8.4 Decidability and axiomatization

The result of Theorem 37 is close to the result of Pauly in [Pau02]. In [Pau02], Pauly compares strategic form games (like STIT-models) and CL standard models (like NCL-models). Theorem 37 provides two crucial results: one about complexity and another one about axiomatization of *df*STIT.

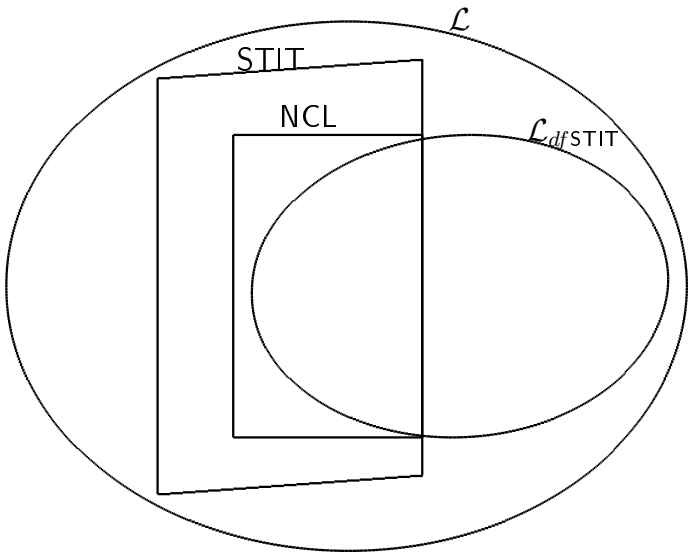


Figure 8.5: Overview over languages $\mathcal{L}$ and $\mathcal{L}_{df\mathsf{STIT}}$ and logics STIT and NCL.

The following corollary follows from the equivalence between point 2 and 3 in the Theorem 37.

Corollary 10 *Deciding if a formula in $\mathcal{L}_{df\mathsf{STIT}}$ is STIT-satisfiable is NP-complete.*

Proof.

SAT is reducible to the STIT-satisfiability problem of a formula in $\mathcal{L}_{df\mathsf{STIT}}$. Thus deciding if a formula in $\mathcal{L}_{df\mathsf{STIT}}$ is STIT-satisfiable is NP-hard. Now let us see that it is in NP.

According to Theorem 37, if a formula φ is STIT-satisfiable, φ is satisfiable in a polynomial-sized STIT-model. So a non-deterministic algorithm to solve the satisfiability can be the following:

- we guess a polynomial-sized model $\mathcal{M}' = (W', V')$ and a world $\vec{x} \in W'$;
- we check whether $\mathcal{M}', \vec{x} \models \varphi$ holds or not.

Note that checking whether $\mathcal{M}', \vec{x} \models \varphi$ or not can be done in polynomial time in the size of $\mathcal{M}'$ and the length of φ. As the size of $\mathcal{M}'$ is polynomial in the length of φ, checking whether $\mathcal{M}', \vec{x} \models \varphi$ or not can be done in polynomial time in the size of φ.

■

The following corollary follows from the equivalence between point 1 and 2 in the Theorem 37.

Corollary 11 *A formula φ in $\mathcal{L}_{df\,\mathsf{STIT}}$ is STIT-valid iff we have $\vdash_{\mathsf{NCL}} \varphi$.*

PROOF.
We have:

- for all formula $\varphi \in \mathcal{L}$, $\models_{\mathsf{NCL}} \varphi$ iff $\vdash_{\mathsf{NCL}} \varphi$ (Theorem 36);
- for all formula $\varphi \in \mathcal{L}_{df\,\mathsf{STIT}}$, $\models_{\mathsf{STIT}} \varphi$ iff $\models_{\mathsf{NCL}} \varphi$. (Theorem 37).

Hence: for all formula $\varphi \in \mathcal{L}_{df\,\mathsf{STIT}}$, $\models_{\mathsf{STIT}} \varphi$ iff $\vdash_{\mathsf{NCL}} \varphi$. ■

8.5 Open questions

- The fragment we have obtained here is NP-complete whereas all the other interesting fragment of the Chapter 7 are generally NEXPTIME-complete. Can we exhibit an interesting fragment of STIT which is PSPACE?
- It is an open question to know if NCL and STIT differs with a formula of modal depth 2. May be for all formula φ of the language of group STIT, we have φ STIT-satisfiable iff φ NCL-satisfiable.

Part III

Knowing, Doing

Chapter 9

Modal logic of epistemic games

9.1 Introduction

We present a multi-modal logic integrating the concepts of joint action, preference and knowledge. Our logic supports reasoning about epistemic games in strategic form in which agents decide what to do according to some general principles of rationality while being uncertain about several aspects of the interaction such as other agents' choices, other agents' preferences, etc.

This logic is not strickly speaking STIT but the philosophy of the models is the same: as you will see, a world is determined by agents are performing. In this sence, this logic differs from Coalition Logic, etc.

While epistemic games have been extensively studied in economics (in the so-called interactive epistemology area, see e.g. [AB95, Aum99, Bon08, BB99, Bra92, Gin09]) and while there have been few analyses of epistemic games in modal logic (see, e.g., [vB07, dB04, Bon08, Roy08]), no modal logic approach to epistemic games has been proposed up to now which addresses all the following issues at the same time:

- to provide a *formal language*, and a corresponding *formal semantics*, which is sufficiently general to express solution concepts like Nash equilibrium or iterated deletion of strictly dominated strategies (IDSDS) and to deduce formally the epistemic and rationality conditions on which such solution concepts are based;
- to prove its *soundness* and *completeness*;
- to study its computational properties like *decidability* and *complexity*.

In this Chapter, we try to fill this gap by proposing a sound and complete modal logic for epistemic games interpreted on a Kripke-style semantics. We also provide complexity results for our logic.

We think that developing modal logic frameworks for the analysis of epistemic aspects of strategic interaction is a promising research avenue which can contribute to the cross-fertilization of economics with other disciplines like computer science, artificial intelligence (AI) and formal philosophy. Modal logics has been extensively studied in the last four decades both in the area of theoretical computer science and in the area of philosophical logic, and enormous progress have been made in recent times especially on model-theoretic aspects and on complexity aspects of epistemic logic and multi-modal logics (see, e.g., [GKWZ03, BDRV02, FHMV95]). Game-theoretic models of strategic interaction can therefore benefit from these advancements in the area of modal logic for several reasons.

First of all, it is typical of multi-modal logics to define a given concept such as the concepts of space, time, knowledge, preference, etc. by a corresponding modal operator and to specify the relationships between different concepts by means of so-called 'interaction' axioms between different modal operators. This is the reason why ,ulti-modal logics are formal frameworks which are well-suited to do conceptual analysis. Epistemic games studied in economics involve several primitive concepts like the concepts of action, knowledge, preference, and time (in the case of extensive games). Therefore, a multi-modal logic of epistemic games can be extremely useful to better understand the properties of these primitive concepts, and the relationships between them. Closely related to the previous point is the fact that a modal logic analysis of the epistemic aspects of strategic interaction can facilitate the task of studying different assumptions on players' knowledge and the task of verifying whether different equilibrium notions such as Nash equilibrium and IDSDS are based on these assumptions. In fact, in a modal logic analysis of epistemic games every assumption on the players' knowledge can be easily formulated by means of a logical axiom on epistemic modal operators. Typical assumptions on players' knowledge are for example the assumption that every player knows what he has decided to do, or the assumption that a player has perfect knowledge about some aspects of the game such as the players' strategy sets (or action repertoires) and the players' preference ordering over strategy profiles.

Another aspect of a modal logic analysis of strategic interaction which could be relevant for a game-theorist is computational complexity. Computational complexity is a fundamental issue in computer science and in modal logic. The study of computational complexity of a modal logic of games can raise many interesting questions closely related to the reasoning aspects involved in strategic interaction. For example, how complex is the problem of deciding whether a certain strategy profile is a Nash equilibrium of a given game? how complex is the problem of deciding whether a given action is a best response to the action of another player? Is a realistic resource-bounded agent able to face with such a complexity? If no, is it plausible to suppose that a human involved in strategic interaction does 'best-

response' reasoning (i.e. he chooses a given action only if he believes that this action is a best response to what he expects the others will do)?

The remainder of the article is organized as follows. In Section 9.2 we present our modal logic of joint actions, preference and knowledge called $\mathcal{MLEG}$ (*Modal Logic of Epistemic Games*). Section 9.3 is devoted to the analysis in $\mathcal{MLEG}$ of the epistemic conditions of Nash equilibrium and IDSDS. In Section 9.4 we make $\mathcal{MLEG}$ dynamic by extending it with constructions of Dynamic Epistemic Logic (DEL) [vDvdHK07, BM04, GG97], and we show that this dynamic version of $\mathcal{MLEG}$ enables to express the notion IDSDS in a more compact way than in the static $\mathcal{MLEG}$. In Section 9.5 we show how our logical framework can be applied to the analysis of strategic interaction with imperfect information about the game structure. In Section 9.6 we discuss several assumptions about different variants of perfect information on a game structure. Finally, in Section 9.7, we compare our approach with some existing approaches to epistemic games in modal logic.

9.2 A logic of joint actions, knowledge and preferences

We present in this section the multi-modal logic $\mathcal{MLEG}$ (*Modal Logic of Epistemic Games*) integrating the concepts of joint action, belief and preference. This logic supports reasoning about epistemic games in strategic form in which an agent might be uncertain about the current choices of the other agents.

9.2.1 Syntax

The syntactic primitives of $\mathcal{MLEG}$ are the finite set of agents Agt, the set of atomic formulas Atm, a nonempty finite set of atomic action names $Act = \{a_1, a_2, \ldots, a_{|Act|}\}$ and a non-empty finite set of n integers $I = \{0, \ldots, n\}$. Non-empty sets of agents are called *coalitions* or *groups*, noted $C_1, C_2, \ldots$. We note $2^{Agt*} = 2^{Agt} \setminus \{\emptyset\}$ the set of coalitions.

To every agent $i \in Agt$ we associate the set Act_i of all possible ordered pairs agent/action $i{:}a$, that is, $Act_i = \{i{:}a \mid a \in Act\}$. Besides, for every coalition C we note Δ_C the set of all joint actions of this coalition, that is, $\Delta_C = \prod_{i \in C} Act_i$. Elements in Δ_C are C-tuples noted α_C, β_C, γ_C, $\delta_C, \ldots$. If $C = Agt$, we write Δ instead of Δ_{Agt}. Elements in Δ are also called strategy profiles. Given $\delta \in \Delta$, we note δ_i the element in δ corresponding to agent i. Moreover, for notational convenience, we write $\delta_{-i} = \delta_{Agt \setminus \{i\}}$.

The language $\mathcal{L}_{\mathcal{MLEG}}$ of the logic $\mathcal{MLEG}$ is given by the following rule:

$$\varphi \ ::= \ p \mid \bot \mid \neg\varphi \mid (\varphi \vee \varphi) \mid [\delta_C]\,\varphi \mid \Box\varphi \mid \mathsf{K}_i\varphi \mid [\mathsf{good}]_i\,\varphi$$

where p ranges over Atm, i ranges over Agt, and δ_C ranges over $\bigcup_{C \in 2^{Agt*}} \Delta_C$. The classical Boolean connectives $\wedge$, $\rightarrow$, $\leftrightarrow$ and $\top$ (tautology) are defined from $\bot$, $\vee$ and $\neg$ in the usual manner. We also follow the standard rules for omission of parentheses.

The formula $[\delta_C]\,\varphi$ reads "if coalition C chooses the joint action δ_C then φ holds". Therefore, $[\delta_C]\,\bot$ reads "coalition C does not choose the joint action δ_C ".

$\Box$ is a necessity operator which enables to quantify over possible joint actions of all agents, that is, over the strategy profiles of the current game (the terms "joint actions of all agents" and "strategy profiles" are supposed here to be synonymous). $\Box\varphi$ reads "φ holds for every alternative strategy profile of the current game", or simply "φ is necessarily true".

Operators K_i are standard epistemic modal operators. Construction $\mathsf{K}_i\varphi$ is read as usual "agent i knows that φ is true", whereas the construction $[\mathsf{good}]_i\,\varphi$ is read "φ is true in all worlds which are for agent i at least as good as the current one concerning the strategy profile that is chosen". We define $\langle\mathsf{good}\rangle_i\varphi$ as an abbreviation of $\neg\,[\mathsf{good}]_i\,\neg\varphi$. Operators $[\mathsf{good}]_i$ are used in $\mathcal{MLEG}$ to define agents' preference orderings over the strategy profiles of the current game. Similar operators are studied by [vBL07].

We use $\mathsf{EK}_C\varphi$ as an abbreviation of $\bigwedge_{i\in C}\mathsf{K}_i\varphi$, i.e. every agent in C knows φ (if $C = \emptyset$ then $\mathsf{EK}_C\varphi$ is equivalent to $\top$). Then we define by induction $\mathsf{EK}^k_C\varphi$ for every natural number $k \in \mathbb{N}$:

$$\mathsf{EK}^0_C\varphi \stackrel{\text{def}}{=} \varphi$$

and for all $k \geq 1$,

$$\mathsf{EK}^k_C\varphi \stackrel{\text{def}}{=} \mathsf{EK}_C(\mathsf{EK}^{k-1}_C\varphi).$$

We define for all natural numbers $n \in \mathbb{N}$, $\mathsf{MK}^n_C\varphi$ as an abbreviation of $\bigwedge_{1\leq k\leq n}\mathsf{EK}^k_C\varphi$. $\mathsf{MK}^n_C\varphi$ expresses C's mutual knowledge that φ up to n iterations, i.e. everyone in C knows φ, everyone in C knows that everyone in C knows φ, and so on until level n.

Finally, $\langle\delta_C\rangle\varphi$ abbreviates $\neg\,[\delta_C]\,\neg\varphi$, $\Diamond\varphi$ abbreviates $\neg\Box\neg\varphi$ and $\widehat{\mathsf{K}}_i\varphi$ abbreviates $\neg\mathsf{K}_i\neg\varphi$. $\Diamond\varphi$ means "φ is possibly true". Therefore $\langle\delta_C\rangle\varphi$ reads "coalition C chooses the joint action δ_C and φ holds", and $\langle\delta_C\rangle\top$ simply reads "coalition C chooses the joint action δ_C".

The operator $\Diamond$ and the operators $\langle\delta_C\rangle$ can be combined in order to express what a coalition of agents can do. In particular, $\Diamond\langle\delta_C\rangle\top$ has to be read "coalition C can choose the joint action δ_C". For the individual case, $\Diamond\langle i{:}a\rangle\top$ has to be read "agent i can choose action a" or also "action a is in the strategy set (action repertoire) of agent i". Furthermore, $\Diamond\langle\delta\rangle\top$ is read "coalition Agt can choose the joint action (strategy profile) δ" or also "δ is a strategy profile of the current game".

9.2.2 Semantics

In this subsection, we introduce a Kripke-style possible world semantics of our logic $\mathcal{MLEG}$.

Definition 62 ($\mathcal{MLEG}$-frames)
$\mathcal{MLEG}$-frames are tuples $F = \langle W, \sim, R, E, \preceq \rangle$ where:

- W is a nonempty set of possible worlds or states;
- $\sim$ is an equivalence relation on W;
- R is a collection of total functions $R_C : W \longrightarrow \Delta_C$ one for every coalition $C \in 2^{Agt*}$, mapping every world in W to a joint action of the coalition such that:
  - **C1** $\delta_C = R_C(w)$ if and only if for every $i \in C$, $\delta_i = R_i(w)$,[1]
  - **C2** if for every $i \in Agt$ there is v_i such that $w \sim v_i$ and $\delta_i = R_i(v_i)$ then there is a v such that $w \sim v$ and $\delta = R_{Agt}(v)$,
  - **C3** if $w \sim v$ and $\delta = R_{Agt}(w)$ and $\delta = R_{Agt}(v)$, then $w = v$;
- $E : Agt \longrightarrow W \times W$ maps every agent i to an equivalence relation E_i on W such that:
  - **C4** if wE_iv, then $i{:}a = R_i(w)$ if and only if $i{:}a = R_i(v)$,
  - **C5** if wE_iv then $w \sim v$;
- $\preceq$: $Agt \longrightarrow W \times W$ maps every agent i to a reflexive, transitive relation $\preceq_i$ on W such that:
  - **C6** if $w \preceq_i v$ then $w \sim v$,
  - **C7** if $w \sim v$ and $w \sim u$ then $v \preceq_i u$ or $u \preceq_i v$.

$\delta_C = R_C(w)$ means that coalition C performs the joint action δ_C at world w.

If $w \sim v$ then w and v correspond to alternative strategy profiles of the same game. For short, we say that v is alternative to w. Given a world w, we use the notation $\sim(w) = \{v \mid w \sim v\}$ to denote the equivalence class made up of those worlds corresponding to alternative strategy profiles of the game of which w is one of the strategy profile. Consider e.g. $Agt = \{1, 2\}$ and $Act = \{c, d, skip\}$. In the frame in Figure 9.1 we have $w_1 \sim w_2$. This means that the joint action performed at w_1 (viz. $\langle 1{:}c, 1{:}c \rangle$) and the one performed at w_2 (viz. $\langle 1{:}c, 1{:}d \rangle$) are

[1] Note that for notational convenience we write R_i instead of $R_{\{i\}}$.

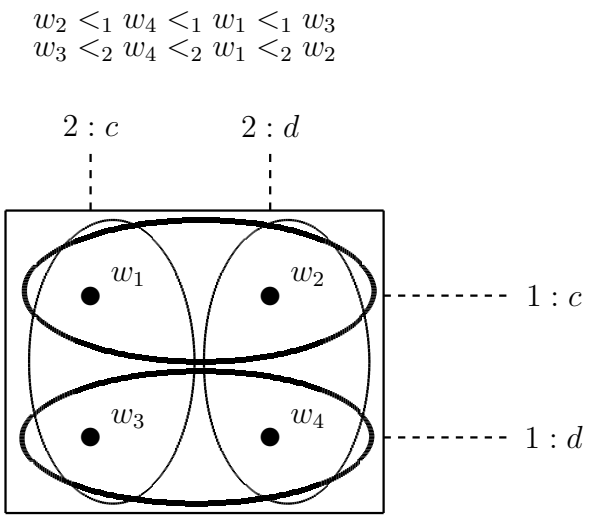


Figure 9.1: The equivalence class $\{w_1, w_2, w_3, w_4\}$ represents the Prisoner's Dilemma game [OR94] between two players 1 and 2 (action c stands for 'cooperate' and action d stands for 'defect'). Thick ellipses are epistemic relations for 1, thin ellipses are epistemic relations for 2 (both 1 and 2 are uncertain about the other's action).

alternative strategy profiles of the same game defined by the equivalence class $\sim(w_1) = \{w_1, w_2, w_3, w_4\}$.

For every $C \subseteq Agt$, if there exists $v \in \sim(w)$ such that C performs δ_C at v then we say that δ_C is *possible* at w (or δ_C *can be performed* at w).

wE_iv means that, for agent i, world v is (epistemically) possible at w, whilst $w \preceq_i v$ means that for agent i, world v is at least as good as world w. We write $w =_i v$ iff $w \preceq_i v$ and $v \preceq_i w$, and $w <_i v$ iff $w \preceq_i v$ and not $v \preceq_i w$.

Let us discuss the seven semantic constraints **C1-C7** in Definition 62.

According to Constraint **C1**, at world w coalition C chooses the joint action δ_C if and only if, every agent i in C chooses the action δ_i at w. In other words, a certain joint action is performed by a coalition if and only if every agent in the coalition does his part of the joint action. According to the Constraint **C2**, if every individual action in a joint action δ is possible at world w, then their simultaneous occurrence is also possible at world w. We moreover suppose determinism for the joint actions of all agents: different worlds in an equivalence class $\sim(w)$ correspond to the occurrences of *different* strategy profiles (Constraint **C3**).

Constraint **C4** just says that an agent knows what he has decided to do. This is a standard assumption in interactive epistemology and epistemic analysis of games (see [Bon08] for instance).

We also suppose perfect information about the specification of the game, including the players' strategy sets (or action repertoires) and the players' preference ordering over strategy profiles. This assumption is formally expressed by the Constraint **C5**: if world v is epistemically possible for agent i at w, then w and v correspond to alternative strategy profiles of the same game. Perfect information about the structure of the game is a standard assumption in game theory. In Section 9.5, this assumption will be relaxed in order to deal with realistic situations in which an agent might be uncertain about his own utility and other agents' utilities associated to a certain strategy profile, as well as about his own action repertoire and other agents' action repertoires.

Finally, we have two constraints over the relations $\preceq_i$. We suppose that a world v is for agent i at least as good as w only if v is a world which is possible at w, i.e. only if v and w correspond to alternative strategy profiles of the same game (Constraint **C6**). Furthermore, we suppose that every agent has a complete preference ordering over the strategy profiles of the current game (Constraint **C7**).

Definition 63 ($\mathcal{MLEG}$-models)
$\mathcal{MLEG}$-models are couples $F = \langle F, \pi \rangle$ where:

- F is a $\mathcal{MLEG}$-frame;
- $\pi : Atm \longrightarrow 2^W$ is a valuation function.

The truth conditions for Boolean operators and for operators $[\delta_C]$, $\Box$, K_i and $[\mathsf{good}]_i$ are:

- $M, w \models p$ iff $w \in \pi(p)$;
- $M, w \models \neg\varphi$ iff not $M, w \models \varphi$;
- $M, w \models \varphi \vee \psi$ iff $M, w \models \varphi$ or $M, w \models \psi$;
- $M, w \models [\delta_C]\,\varphi$ iff if $R_C(w) = \delta_C$ then $M, w \models \varphi$;
- $M, w \models \Box\varphi$ iff $M, v \models \varphi$ for all v such that $w \sim v$;
- $M, w \models \mathsf{K}_i\varphi$ iff $M, v \models \varphi$ for all v such that wE_iv;
- $M, w \models [\mathsf{good}]_i\,\varphi$ iff $M, v \models \varphi$ for all v such that $w \preceq_i v$.

A formula φ is *true in an $\mathcal{MLEG}$-model* M iff $M, w \models \varphi$ for every world w in M. A formula φ is *$\mathcal{MLEG}$-valid* (noted $\models \varphi$) iff φ is true in all $\mathcal{MLEG}$-models. A formula φ is *$\mathcal{MLEG}$-satisfiable* iff $\neg\varphi$ is not $\mathcal{MLEG}$-valid.

All principles of classical propositional logic	(**CPL**)
All principles of modal logic S5 for $\Box$	($\mathbf{S5}_{\Box}$)
All principles of modal logic S5 for every K_i	($\mathbf{S5}_{\mathsf{K}_i}$)
All principles of modal logic S4 for every $[\mathsf{good}]_i$	($\mathbf{S4}_{[\mathsf{good}]_i}$)
$[\delta_C]\,\varphi \leftrightarrow (\langle\delta_C\rangle\top \rightarrow \varphi)$	($\mathbf{Def}_{[\delta_C]}$)
$\langle\delta_C\rangle\top \leftrightarrow \bigwedge_{i\in C}\langle\delta_i\rangle\top$	(**JointAct**)
$\bigvee_{\delta_C\in\Delta_C}\langle\delta_C\rangle\top$	(**Active**)
$\langle\delta_C\rangle\top \rightarrow [\delta'_C]\,\bot$ if $\delta_C \neq \delta'_C$	(**Single**)
$\left(\bigwedge_{i\in Agt}\Diamond\langle\delta_i\rangle\top\right) \rightarrow \Diamond\langle\delta\rangle\top$	(**Indep**)
$(\langle\delta\rangle\top \wedge \varphi) \rightarrow \Box(\langle\delta\rangle\top \rightarrow \varphi)$	(**JointDet**)
$\langle i{:}a\rangle\top \rightarrow \mathsf{K}_i\langle i{:}a\rangle\top$	(**Aware**)
$\Box\varphi \rightarrow [\mathsf{good}]_i\,\varphi$	($\mathbf{Incl}_{[\mathsf{good}]_i,\Box}$)
$(\Diamond\varphi \wedge \Diamond\psi) \rightarrow (\Diamond(\varphi \wedge \langle\mathsf{good}\rangle_i\psi) \vee \Diamond(\psi \wedge \langle\mathsf{good}\rangle_i\varphi))$	(**PrefConnect**)
$\Box\varphi \rightarrow \mathsf{K}_i\varphi$	(**PerfectInfo**)
$\dfrac{\varphi,\ \varphi \rightarrow \psi}{\psi}$	(**ModusPonens**)

Figure 9.2: Axiomatization of $\mathcal{MLEG}$

9.2.3 Axiomatization

We call $\mathcal{MLEG}$ the logic that is axiomatized by the principles given in Figure 9.2.

Note that the principles of modal logic S5 for the operator $\Box$ are: the four axiom schemas (K) $(\Box\varphi \wedge \Box(\varphi \rightarrow \psi)) \rightarrow \Box\psi$, (T) $\Box\varphi \rightarrow \varphi$, (4) $\Box\varphi \rightarrow \Box\Box\varphi$, (B) $\varphi \rightarrow \Box\Diamond\varphi$, and the rule of inference (Necessitation) $\frac{\varphi}{\Box\varphi}$. The principles of modal logic S5 for the operators K_i are: the four axiom schemas (K) $(\mathsf{K}_i\varphi \wedge \mathsf{K}_i(\varphi \rightarrow \psi)) \rightarrow \mathsf{K}_i\psi$, (T) $\mathsf{K}_i\varphi \rightarrow \varphi$, (4) $\mathsf{K}_i\varphi \rightarrow \mathsf{K}_i\mathsf{K}_i\varphi$, (B) $\varphi \rightarrow \mathsf{K}_i\widehat{\mathsf{K}}_i\varphi$, and the rule of inference (Necessitation) $\frac{\varphi}{\mathsf{K}_i\varphi}$. The principles of modal logic S4 for the operators $[\mathsf{good}]_i$ are: the three axiom schemas (K) $([\mathsf{good}]_i\,\varphi \wedge [\mathsf{good}]_i\,(\varphi \rightarrow \psi)) \rightarrow [\mathsf{good}]_i\,\psi$, (T) $[\mathsf{good}]_i\,\varphi \rightarrow \varphi$, (4) $[\mathsf{good}]_i\,\varphi \rightarrow [\mathsf{good}]_i\,[\mathsf{good}]_i\,\varphi$, and the rule of inference (Necessitation) $\frac{\varphi}{[\mathsf{good}]_i\varphi}$.

Note also that Axiom **Indep** is the $\mathcal{MLEG}$ counterpart of the so-called *axiom*

of independence of agents of STIT logic (the logic of *Seeing to it that*) [BPX01]. This axiom enables to express the basic game theoretic assumption that the set of strategy profiles of a game in strategic form is the cartesian product of the sets of individual actions for the agents in Agt.

We write $\vdash_{\mathcal{MLEG}} \varphi$ if φ is a theorem of $\mathcal{MLEG}$, that is, if φ can be deduced by applying the axioms and the rules of inference of the logic $\mathcal{MLEG}$.

As the following theorem 38 highlights, we can prove that the logic $\mathcal{MLEG}$ is sound and complete with respect to the class of $\mathcal{MLEG}$-models.

Theorem 38 *$\mathcal{MLEG}$ is determined by the class of $\mathcal{MLEG}$-models.*

PROOF.

We here provide a sketch of the proof.

It is straightforward to show that all our axioms are valid and that the rules of inference preserve validity in the class of $\mathcal{MLEG}$-models. The other part of the proof is shown using two major steps.

Step 1. We provide an alternative semantics for $\mathcal{MLEG}$ in terms of standard Kripke models whose semantic conditions correspond one-to-one to the axioms in Table 9.2. The definition of Kripke $\mathcal{MLEG}$-models is the following one.

Definition 64 (Kripke $\mathcal{MLEG}$-model)
Kripke $\mathcal{MLEG}$-models are tuples $M = \langle W, \sim, R, E, \preceq, \pi \rangle$ where:

- W is a nonempty set of possible worlds or states;
- $\sim$ is an equivalence relation on W;
- $R : \bigcup_{C \in 2^{Agt*}} \Delta_C \longrightarrow 2^{W \times W}$ maps every joint action δ_C to a transition relation $R_{\delta_C} \subseteq W \times W$ between possible worlds such that:
  - S1 $R_{\delta_C}(w) \neq \emptyset$ if and only if, for every $i \in C$ $R_{\delta_i}(w) \neq \emptyset$,
  - S2 if $R_{\delta_C}(w) \neq \emptyset$ then $R_{\delta_C}(w) = \{w\}$,
  - S3 $\bigcup_{\delta_C \in \Delta_C} R_{\delta_C}(w) \neq \emptyset$,
  - S4 if $\delta_C \neq \delta'_C$ then $R_{\delta_C}(w) = \emptyset$ or $R_{\delta'_C}(w) = \emptyset$,
  - S5 if for every $i \in Agt$ there is v_i such that $w \sim v_i$ and $R_{\delta_i}(v_i) \neq \emptyset$ then there is a v such that $w \sim v$ and $R_\delta(v) \neq \emptyset$,
  - S6 if $w \sim v$ and $R_\delta(w) \neq \emptyset$ and $R_\delta(v) \neq \emptyset$, then $w = v$;
- $E : Agt \longrightarrow W \times W$ maps every agent i to an equivalence relation E_i on W such that:
  - S7 if $(w, v) \in E_i$, then $i{:}a = R_i(w)$ if and only if $i{:}a = R_i(v)$,

S8 if wE_iv then $w \sim v$;

- $\preceq$: $Agt \longrightarrow W \times W$ maps every agent i to a reflexive, transitive relation $\preceq_i$ on W such that:

  S9 if $w \preceq_i v$ then $w \sim v$,

  S10 if $w \sim v$ and $w \sim v'$ then $v \preceq_i v'$ or $v' \preceq_i v$;

- $\pi : Atm \longrightarrow 2^W$ is a valuation function.

Truth conditions of $\mathcal{MLEG}$ formulas in Kripke $\mathcal{MLEG}$-models are again standard for atomic formulas and the Boolean operators. The truth conditions for Boolean operators and for operators $\Box$, K_i and $[\mathsf{good}]_i$ are the ones of Section 9.2.2. The truth condition for operators $[\delta_C]$ are:

- $M, w \models [\delta_C]\,\varphi$ iff $M, v \models \varphi$ for all $v \in R_{\delta_C}(w)$.

It is a routine task to prove that the axiomatic system of the logic $\mathcal{MLEG}$ given in Table 9.2 is sound and complete with respect to this class of Kripke $\mathcal{MLEG}$-models via the Sahlqvist theorem, cf. [BDRV02, Th. 2.42]. Indeed all axioms in Table 9.2 are in the so-called Sahlqvist class [Sah75]. Thus, they are all expressible as first-order conditions on Kripke models and are complete with respect to the defined model classes.

Step 2. The second step shows that the semantics in terms of $\mathcal{MLEG}$-models of Definition 63 and the semantics in terms of Kripke $\mathcal{MLEG}$-models of Definition 64 are equivalent. As the logic $\mathcal{MLEG}$ is sound and complete for the class of Kripke $\mathcal{MLEG}$-models and is sound for the class of $\mathcal{MLEG}$-models, we have that for every $\mathcal{MLEG}$ formula φ, if φ is valid in the class of Kripke $\mathcal{MLEG}$-models then φ is valid in the class of $\mathcal{MLEG}$-models. Consequently, for every $\mathcal{MLEG}$ formula φ, if φ is satisfiable in the class of $\mathcal{MLEG}$-models then φ is satisfiable in the class of Kripke $\mathcal{MLEG}$-models. Therefore, in this second step we just need to show that for every $\mathcal{MLEG}$ formula φ, if φ is satisfiable in the class of Kripke $\mathcal{MLEG}$-models then φ is satisfiable in the class of $\mathcal{MLEG}$-models.

Suppose φ is satisfiable in the class of Kripke $\mathcal{MLEG}$-models. This means that there is a Kripke $\mathcal{MLEG}$-model $M = \langle W, \sim, R, E, \preceq, \pi\rangle$ and world w such that $M, w \models \varphi$. We can now build a $\mathcal{MLEG}$-model $M' = \langle W', R', E', \preceq', \pi'\rangle$ which satisfies φ. The model M' is defined as follows:

- $W' = W$;
- for every $C \in 2^{Agt*}$ and $v \in W'$, $R'_C(v) = \delta_C$ if and only if $R_{\delta_C}(v) \neq \emptyset$;
- for every $i \in Agt$, $E'_i = E_i$;

- for every $i \in Agt$, $\preceq'_i = \preceq_i$;
- $\pi' = \pi$.

By induction on the structure of φ, it is just a trivial exercise to show that we have $M', w \models \varphi$.

■

Moreover we can prove a result about complexity of the satisfiability problem of the logic $\mathcal{MLEG}$, that is, the complexity of the problem of deciding whether a given $\mathcal{MLEG}$ formula φ is $\mathcal{MLEG}$-satisfiable or not. This question is highly related to automated reasoning.

Theorem 39 *The satisfiability problem of $\mathcal{MLEG}$ is NP-complete.*

PROOF.

The satisfiability problem of $\mathcal{MLEG}$ is clearly NP-hard because it is a conservative extension of the classical propositional logic whose satisfiability problem in NP-complete (Cook's Theorem [Pap03]).

Now let us prove it is in NP. Clearly if a formula φ is $\mathcal{MLEG}$-satisfiable, there exists a $\mathcal{MLEG}$-model $F = \langle F, \pi \rangle$ whose size is bounded by $card(Act)^{card(Agt)}$. Here is an non-deterministic algorithm to check if a given formula φ is satisfiable:

- Guess non-deterministically a $\mathcal{MLEG}$-model $M = \langle F, \pi \rangle$ whose size is bounded by $card(Act)^{card(Agt)}$ where π only gives truthness of propositions occuring in φ;
- Guess non-deterministically a world w of M;
- Check if $M, w \models \varphi$.

This algorithm non-deterministically runs in polynomial time. So the satisfiability problem of $\mathcal{MLEG}$ is in NP.

■

9.3 A logical account of epistemic games

This section is devoted to the analysis in the modal logic $\mathcal{MLEG}$ of the epistemic aspects of strategic games. We first consider the basic game-theoretic concepts of best response and Nash equilibrium, and their relationships with the notion of epistemic rationality assumed in classical game theory. Finally, we provide an analysis of *iterated deletion of strictly dominated strategies* (IDSDS).

9.3.1 Best response and Nash equilibrium

The modal operators $[\mathsf{good}]_i$ and $\Box$ enable to capture in $\mathcal{MLEG}$ a notion of comparative goodness over formulas of the kind "φ is for agent i at least as good as ψ", noted $\psi \leq_i \varphi$:

$$\psi \leq_i \varphi \stackrel{\text{def}}{=} \Box(\psi \rightarrow \langle \mathsf{good} \rangle_i \varphi).$$

According to the previous definition, φ is for agent i at least as good as ψ if and only if, for every world v corresponding to a strategy profile of the current game in which ψ is true, there is a world u corresponding to a strategy profile of the current game in which φ is true and which is for agent i at least as good as world v. We can prove that $\psi \leq_i \varphi$ is a total preorder. Indeed, the formulas $\psi \leq_i \psi$ (reflexivity), $(\varphi_1 \leq_i \varphi_2) \wedge (\varphi_2 \leq_i \varphi_3) \rightarrow (\varphi_1 \leq_i \varphi_3)$ (transitivity) and $(\varphi_1 \leq_i \varphi_2) \vee (\varphi_2 \leq_i \varphi_1)$ (connectedness) are valid in $\mathcal{MLEG}$. We define the corresponding strict ordering over formulas:

$$\psi <_i \varphi \stackrel{\text{def}}{=} (\psi \leq_i \varphi) \wedge \neg(\varphi \leq_i \psi).$$

Formula $\psi <_i \varphi$ has to read "φ is for agent i strictly better than ψ". Finally, we define a notion of comparative goodness over strategy profiles and the corresponding strict ordering over strategy profiles:

$$\delta \leq_i \delta' \stackrel{\text{def}}{=} \langle \delta \rangle \top \leq_i \langle \delta' \rangle \top \text{ and } \delta <_i \delta' \stackrel{\text{def}}{=} (\delta \leq_i \delta') \wedge \neg(\delta' \leq_i \delta).$$

Formula $\delta \leq_i \delta'$ has to be read "strategy profile δ' is for agent i at least as good as strategy profile δ" and formula $\delta <_i \delta'$ has to be read "strategy profile δ' is for agent i strictly better than strategy profile δ".

Some basic concepts of game theory can be expressed in $\mathcal{MLEG}$ in terms of comparative goodness. We first consider *best response*. Agent i's action a is said to be a best response to the other agents' joint action δ_{-i}, noted $\mathsf{BR}(i{:}a,\delta_{-i})$, if and only if i cannot improve his utility by deciding to do something different from a while the others choose the joint action δ_{-i}, that is:

$$\mathsf{BR}(i{:}a,\delta_{-i}) \stackrel{\text{def}}{=} \bigwedge_{b \in Act} ((\langle i{:}b \rangle \top \wedge \langle \delta_{-i} \rangle \top) \leq_i (\langle i{:}a \rangle \top \wedge \langle \delta_{-i} \rangle \top)).$$

Given a certain strategic game, the strategy profile (or joint action) δ is said to be a *Nash equilibrium* if and only if for every agent $i \in Agt$, i's action δ_i is a best response to the other agents' joint action δ_{-i}:

$$\mathsf{Nash}(\delta) \stackrel{\text{def}}{=} \bigwedge_{i \in Agt} \mathsf{BR}(\delta_i,\delta_{-i}).$$

From Axiom **PerfectInfo** and S5 for $\Box$, the following theorems are provable expressing perfect information about the players' preferences ordering over strategy

profiles, perfect information about the existence of a Nash equilibrium, and perfect information about the players' repertoires: $(\psi \leq_i \varphi) \leftrightarrow \mathsf{MK}^n_{Agt}(\psi \leq_i \varphi)$, $\mathsf{Nash}(\delta) \leftrightarrow \mathsf{MK}^n_{Agt}\mathsf{Nash}(\delta)$ and $\Diamond\langle\delta_i\rangle\top \leftrightarrow \mathsf{MK}^n_{Agt}\Diamond\langle\delta_i\rangle\top$, for every $n \in \mathbf{N}$.

9.3.2 Epistemic rationality

The following $\mathcal{MLEG}$ formula characterizes a notion of rationality which is commonly supposed in the epistemic analysis of games (see, e.g., [BB99, vB07]):

$$\bigwedge_{a,b\in Act}\Big(\langle i{:}a\rangle\top \rightarrow \bigvee_{\delta\in\Delta}\Big(\widehat{\mathsf{K}}_i\langle\delta_{-i}\rangle\top \wedge (\langle\delta_{-i}, i{:}b\rangle \leq_i \langle\delta_{-i}, i{:}a\rangle)\Big)\Big).$$

This means that an agent i is rational if and only if, if he chooses a particular action a then for every alternative action b, there exists a joint action δ_{-i} of the other agents that he considers possible such that, playing a while the others play δ_{-i} is for i at least as good as playing b while the others play δ_{-i}. As in $\mathcal{MLEG}$ formula $\delta \leq_i \delta'$ and formula $\mathsf{K}_i(\delta \leq_i \delta')$ are equivalent, the previous definition of rationality can be rewritten in the following equivalent form:

$$\mathsf{Rat}_i \stackrel{\text{def}}{=} \bigwedge_{a,b\in Act}\left(\langle i{:}a\rangle\top \rightarrow \bigvee_{\delta\in\Delta}\left(\widehat{\mathsf{K}}_i\langle\delta_{-i}\rangle\top \wedge \mathsf{K}_i\left(\langle\delta_{-i}, i{:}b\rangle \leq_i \langle\delta_{-i}, i{:}a\rangle\right)\right)\right).$$

Theorem 40 *For all $i \in Agt$:*

$$\vdash_{\mathcal{MLEG}} \mathsf{Rat}_i \leftrightarrow \mathsf{K}_i\mathsf{Rat}_i \tag{9.40a}$$

$$\vdash_{\mathcal{MLEG}} \neg\mathsf{Rat}_i \leftrightarrow \mathsf{K}_i\neg\mathsf{Rat}_i \tag{9.40b}$$

Proof.

We only give the proof of 9.40a. Rat_i is equivalent to

$$\textstyle\bigwedge_{a,b\in Act}(\langle i{:}a\rangle\top \rightarrow \bigvee_{\beta\in\Delta}(\hat{K}_i\langle\beta_{-i}\rangle\top \wedge (\langle\beta_{-i}, i{:}b\rangle \leq_i \langle\beta_{-i}, i{:}a\rangle)))$$

which is equivalent to

$$\textstyle\bigwedge_{a,b\in Act}(\mathsf{K}_i\,[i{:}a]\,\bot \vee \bigvee_{\beta\in\Delta}\mathsf{K}_i(\hat{K}_i\langle\beta_{-i}\rangle\top \wedge (\langle\beta_{-i}, i{:}b\rangle \leq_i \langle\beta_{-i}, i{:}a\rangle))),$$

by classical principles of propositional logic, Axiom 5 for K_i, and the two $\mathcal{MLEG}$ theorems $(\beta' \leq_i \beta) \leftrightarrow \mathsf{K}_i(\beta' \leq_i \beta)$ and $[i{:}a]\,\bot \leftrightarrow \mathsf{K}_i\,[i{:}a]\,\bot$.

The latter implies

$$\textstyle\bigwedge_{a,b\in Act}\mathsf{K}_i([i{:}a]\,\bot \vee \bigvee_{\beta\in\Delta}(\hat{K}_i\langle\beta_{-i}\rangle\top \wedge (\langle\beta_{-i}, i{:}b\rangle \leq_i \langle\beta_{-i}, i{:}a\rangle))),$$

by standard principles of normal modal logic. By standard principles of normal modal logic, the latter is equivalent to

$$\textstyle\mathsf{K}_i\bigwedge_{a,b\in Act}([i{:}a]\,\bot \vee \bigvee_{\beta\in\Delta}(\hat{K}_i\langle\beta_{-i}\rangle\top \wedge (\langle\beta_{-i}, i{:}b\rangle \leq_i \langle\beta_{-i}, i{:}a\rangle))).$$

This is equivalent to $\mathsf{K}_i\mathsf{Rat}_i$. ■

Theorem 40 highlights that the concepts of rationality and irrationality are introspective. That is, an agent i is (resp. is not) epistemically epistemically rational if and only if he knows this. The following theorem specifies some sufficient

epistemic conditions for guaranteeing that the chosen strategy profile is a Nash equilibrium: if all agents are rational and every agent knows the choices of the other agents, then the selected strategy profile is a Nash equilibrium. A similar theorem has been stated for the first time in [AB95, Bra92].

Theorem 41 *For all $n \in \mathbb{N}$, for all $\delta \in \Delta$:*
$\vdash_{\mathcal{MLEG}} \left(\left(\bigwedge_{i \in Agt} \mathsf{Rat}_i\right) \wedge \bigwedge_{i \in Agt} \mathsf{K}_i \langle \delta_{-i} \rangle \top\right) \rightarrow \mathsf{Nash}(\delta)$

PROOF.

Let us take a $\mathcal{MLEG}$-model M and a world w such that $M, w \models (\bigwedge_{i \in Agt} \mathsf{Rat}_i \wedge \bigwedge_{i \in Agt} \mathsf{K}_i \langle \delta_{-i} \rangle \top)$. Now, let us prove that $M, w \models \mathsf{Nash}(\delta)$.

Let $i \in Agt$ and let us prove that $M, w \models \mathsf{BR}(\delta_i, \delta_{-i})$.

More precisely, we have to prove that $M, w \models ([\delta_i] \bot \wedge \langle \delta_{-i} \rangle \top) \leq_i (\langle \delta_i \rangle \top \wedge \langle \delta_{-i} \rangle \top)$.

Let $j \neq i$. We have $\mathsf{K}_j \langle -j \rangle \top$, so $M, w \models \langle \delta_i \rangle \top$ by Axiom T for K_j. As $M, w \models \mathsf{Rat}_i$, we then have $M, w \models \bigwedge_{b \in Act} \bigvee_{\beta \in \Delta} (\hat{K}_i \langle \beta_{-i} \rangle \top \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, \delta_i \rangle)))$. That is to say for all $b \in Act$, there exists $\beta \in \Delta$ such that $M, w \models \hat{K}_i \langle \beta_{-i} \rangle \top \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, \delta_i \rangle)$.

But, we have $\mathsf{K}_i \langle \delta_{-i} \rangle \top$. So for all $b \in Act$, $\beta_{-i} = \delta_{-i}$.

So we have $M, w \models \bigwedge_{b \in Act} (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, \delta_i \rangle)$. This is equivalent to $M, w \models ([\delta_i] \bot \wedge \langle \delta_{-i} \rangle \top) \leq_i (\langle \delta_i \rangle \top \wedge \langle \delta_{-i} \rangle \top)$.

■

9.3.3 Iterated deletion of strictly dominated strategies

A strategy a for agent i is a *strictly dominated strategy*, noted $\mathsf{SD}^{\leq 0}(i{:}a)$, if and only if, if a can be performed then there is another strategy b such that, no matter what joint action δ_{-i} the other agents choose, playing b is for i strictly better than playing a:

$$\mathsf{SD}^{\leq 0}(i{:}a) \stackrel{\text{def}}{=} \langle i{:}a \rangle \top \rightarrow$$

$$\bigvee_{b \in Act} \left(\Diamond \langle i{:}b \rangle \top \wedge \bigwedge_{\delta \in \Delta} (\Diamond \langle \delta_{-i} \rangle \rightarrow (\langle \delta_{-i}, i{:}a \rangle <_i \langle \delta_{-i}, i{:}b \rangle)) \right).$$

An example of strictly dominated strategy is cooperation in the Prisoner Dilemma (PD) game: whether one's opponent chooses to cooperate or defect, defection yields a higher payoff than cooperation. Therefore, a rational player will never play a dominated strategy. So when trying to predict the behavior of rational players, we can rule out all strictly dominated strategies. The so-called iterated deletion of strictly dominated strategies (IDSDS) (or iterated strict dominance)

[OR94] is a procedure that starts with the original game and, at each step, for every player i removes from the game all i's strictly dominated strategies, thereby generating a subgame of the original game, and that repeats this process again and again. IDSDS can be inductively characterized in our logic $\mathcal{MLEG}$ by defining a concept of strict dominance in the subgame of depth at most n, noted $\mathsf{SD}^{\leq n}(i{:}a)$. For every $n \geq 1$:

$$\mathsf{SD}^{\leq n}(i{:}a) \stackrel{\text{def}}{=} \neg\mathsf{SD}^{\leq n-1}(i{:}a) \rightarrow$$

$$\bigvee_{b \in Act} \left(\neg\mathsf{SD}^{\leq n-1}(i{:}b) \wedge \bigwedge_{\delta \in \Delta} \left(\neg\mathsf{SD}^{\leq n-1}(\delta_{-i}) \rightarrow (\langle \delta_{-i}, i{:}a \rangle <_i \langle \delta_{-i}, i{:}b \rangle) \right) \right).$$

where $\mathsf{SD}^{\leq k}(\delta_C)$ is defined as follows

$$\mathsf{SD}^{\leq k}(\delta_C) \stackrel{\text{def}}{=} \bigvee_{i \in C} \mathsf{SD}^{\leq k}(\delta_i)$$

for every $k \geq 0$ and for every δ_C. According to this definition, a is a strictly dominated strategy for agent i in a subgame of depth at most n, noted $\mathsf{SD}^{\leq n}(i{:}a)$, if and only if, if a is not strictly dominated for i in all subgames of depth $k < n$ then there is another strategy b such that b is not strictly dominated for i in all subgames of depth $k < n$ and, no matter what joint action δ_{-i} the other agents choose, if the elements in δ_{-i} are not dominated in all subgames of depth $k < n$ then playing b is for i strictly better than playing a. In other terms $\mathsf{SD}^{\leq n}(i{:}a)$ means that strategy $i{:}a$ does not survive after n rounds of IDSDS.

It has been shown that common knowledge of rationality implies that players choose strategies which survive IDSDS ([Bon08, BB99, Bra92]). This latter principle can be derived in our logic $\mathcal{MLEG}$. According to the following Theorem 42, if there is mutual knowledge of rationality among the players to n levels and the agents play the strategy profile δ then, for every agent i, δ_i survives IDSDS until the subgame of depth $n{+}1$.

Theorem 42 *For all* $\delta \in \Delta$, $\vdash_{\mathcal{MLEG}} \left(\left(\mathsf{MK}^n_{Agt} \bigwedge_{i \in Agt} \mathsf{Rat}_i \right) \wedge \langle \delta \rangle \top \right) \rightarrow \neg\mathsf{SD}^{\leq n}(\delta)$

(note that $\neg\mathsf{SD}^{\leq n}(\delta)$ is just the abbreviation of $\bigwedge_{i \in Agt} \neg\mathsf{SD}^{\leq k}(\delta_i)$).

Proof.

We are going to prove the theorem by induction on n.

- Let us begin to prove the theorem for $n = 0$. Let us take a $\mathcal{MLEG}$-model M and a world w such that $M, w \models \bigwedge_{i \in Agt} \mathsf{Rat}_i \wedge \langle \delta \rangle \top$. By definition of Rat_i, we have:

  $M, w \models \bigwedge_{i \in Agt} \bigwedge_{b \in Act} \left(\bigvee_{\beta \in \Delta} \left(\hat{K}_i \langle \beta_{-i} \rangle \top \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, i : \delta_i \rangle) \right) \right)$

This implies:

$M, w \models \bigwedge_{i \in Agt} \bigwedge_{b \in Act} \Big(\bigvee_{\beta \in \Delta} (\Diamond \langle \beta_{-i} \rangle \top \wedge \langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, i : \delta_i \rangle) \Big)$, by Axiom **PerfectInfo**.

Furthermore, we have $M, w \models \bigwedge_{i \in Agt} \langle \delta_i \rangle \top$. So, $M, w \models \bigwedge_{i \in Agt} \neg \mathsf{SD}^{\leq 0}(\delta_i)$ and $M, w \models \neg \mathsf{SD}^{\leq 0}(\delta)$.

So

$$\vdash_{\mathcal{MLEG}} \left(\bigwedge_{i \in Agt} \mathsf{Rat}_i \wedge \langle \delta \rangle \top \right) \rightarrow \neg \mathsf{SD}^{\leq 0}(\delta).$$

- Now, let $n \in \mathbb{N}$ and let us prove that if the theorem 42 is true for all $k \leq n$ then it is true for $n+1$. Let us take a $\mathcal{MLEG}$-model M and a world w such that $M, w \models \Big(\mathsf{MK}^{n+1}_{Agt} \bigwedge_{i \in Agt} \mathsf{Rat}_i \Big) \wedge \langle \delta \rangle \top$. We have to prove $M, w \models \neg \mathsf{SD}^{\leq n+1}(\delta)$. That is to say, we have to prove that for all $i \in Agt$, $M, w \models \neg \mathsf{SD}^{\leq n+1}(\delta_i)$.

$$\neg \mathsf{SD}^{\leq n+1}(\delta_i) = \neg \mathsf{SD}^{\leq n}(\delta_i) \wedge$$

$$\bigwedge_{b \in Act} \left(\neg \mathsf{SD}^{\leq n}(i{:}b) \rightarrow \bigvee_{\beta \in \Delta} \left(\neg \mathsf{SD}^{\leq n}(\beta_{-i}) \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, \delta_i \rangle) \right) \right)$$

.

First, as $M, w \models (\mathsf{MK}^{n+1}_{Agt} \bigwedge_{i \in Agt} \mathsf{Rat}_i) \wedge \langle \delta \rangle \top$ we also have $M, w \models (\mathsf{MK}^{n}_{Agt} \bigwedge_{i \in Agt} \mathsf{Rat}_i) \wedge \langle \delta \rangle \top$. So by applying Theorem 42 for n we have $M, w \models \neg \mathsf{SD}^{\leq n}(\delta_i)$.

It remains to be proven $M, w \models \bigwedge_{b \in Act} (\neg \mathsf{SD}^{\leq n}(i{:}b) \rightarrow \bigvee_{\beta \in \Delta} (\neg \mathsf{SD}^{\leq n}(\beta_{-i}) \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, \delta_i \rangle)))$.

In fact, we are going to prove something less strong:

$M, w \models \bigwedge_{b \in Act} \bigvee_{\beta \in \Delta} (\neg \mathsf{SD}^{\leq n}(\beta_{-i}) \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, i{:}\delta_i \rangle))$.

But, we have $M, w \models \mathsf{Rat}_i \wedge \langle \delta \rangle \top$. So, $M, w \models \bigwedge_{b \in Act} \bigvee_{\beta \in \Delta} (\hat{K}_i \langle \beta_{-i} \rangle \top \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, \delta_i \rangle))$.

The only thing which remains to proven is that we have '$\neg \mathsf{SD}^{\leq n}(\beta_{-i})$'.

But for all $b \in Act$, there exists $\beta \in \Delta$ such that $M, w \models \hat{K}_i \langle \beta_{-i} \rangle \top \wedge (\langle \beta_{-i}, i{:}b \rangle \leq_i \langle \beta_{-i}, \delta_i \rangle))$.

For all $b \in Act$, there exists a world u such that $w E_i u$ and $M, u \models \langle \beta_{-i} \rangle \top$. As $M, w \models \mathsf{MK}^{n+1}_{Agt} \bigwedge_{i \in Agt} \mathsf{Rat}_i$, we have for all $k \leq n$, $M, u \models \mathsf{MK}^{k}_{Agt} \bigwedge_{i \in Agt} \mathsf{Rat}_i$.

The Theorem 42 is supposed to be true by induction for n so $M, u \models \neg\mathsf{SD}^{\leq n}(\beta_{-i})$. But as $\models \neg\mathsf{SD}^{\leq n}(\beta_{-i}) \leftrightarrow \Box\neg\mathsf{SD}^{\leq n}(\beta_{-i})$, and as $E_i \subseteq\sim$ (constraint **C5** on $\mathcal{MLEG}$-frames), we have $M, w \models \neg\mathsf{SD}^{\leq n}(\beta_{-i})$, this for all $b \in Agt$.

■

Unfortunately, we can prove by recurrence on n that the length of the formula $\mathsf{SD}^{\leq n}(\delta)$ is

$$O(|Act||Agt|)^{2n+1})$$

where $O(...)$ is the "Big Oh Notation" [Pap03], $|Act|$ is the number of action and $|Agt|$ is the number of agent and n is the number of rounds of IDSDS. That is, the length of the formula $\mathsf{SD}^{\leq n}(\delta)$ is exponential in n. In the next section, we are going to extend the language in order to capture the concept of IDSDS with a compact formula.

9.4 Game transformation

We provide in this section an alternative and more compact characterization of the procedure IDSDS in our logic $\mathcal{MLEG}$. To this aim, we introduce special events whose effect is to transform the current game by removing certain strategies from it. In particular, these special events can used to delete a strictly dominated strategy from the current game. These special events are similar to the notion of announcement in Dynamic Epistemic Logic (DEL) [vDvdHK07, BM04, GG97].

$\mathcal{L}_{\mathcal{GT}}$ is the set of *game transformation formulas* and is defined by the following rule:

$$\chi ::= \Box\psi \rightarrow [i{:}a]\bot \mid \chi \wedge \chi$$

where $\psi \in \mathcal{L}_{\mathcal{MLEG}}$, $i \in Agt$ and $a \in Act$. Thus, game transformation formulas are of the form 'if property ψ necessarily holds in the current game, then action a should not be performed by agent i'.

$\mathcal{GT}$ is the set of *game transformation events* and is defined as $\mathcal{GT} = \{\chi! \mid \chi \in \mathcal{L}_{\mathcal{GT}}\}$.

We extend the $\mathcal{MLEG}$ language with dynamic operators of the form $[\chi!]$ with $\chi! \in \mathcal{GT}$. The formula $[\chi!]\varphi$ has to be read 'φ holds, after the occurrence of the game transformation event $\chi!$'. We call $\mathcal{MLEG}^{\mathcal{GT}}$ the extended logic. The truth condition for $[\chi!]\varphi$ is:

$$M, w \models [\chi!]\varphi \text{ iff } \text{if } M, w \models \chi \text{ then } M^{\chi}, w \models \varphi$$

with $M^\chi = \langle W^\chi, \sim^\chi, R^\chi, E^\chi, \preceq^\chi, \pi^\chi \rangle$ and:

$$
\begin{array}{l}
W^\chi = \{w \mid w \in W \text{ and } M, w \models \chi\};\\
\sim^\chi = \sim \cap (W^\chi \times W^\chi);\\
\text{for every } C \in 2^{Agt*},\ R^\chi_{\delta_C} = R_{\delta_C}|_{W^\chi};\\
\text{for every } i \in Agt,\ E^\chi_i = E_i \cap (W^\chi \times W^\chi);\\
\text{for every } i \in Agt,\ \preceq^\chi_i = \preceq_i \cap (W^\chi \times W^\chi);\\
\text{for every } p \in Atm,\ \pi^\chi(p) = \pi(p) \cap W^\chi.
\end{array}
$$

Thus, an event $\chi!$ removes from the model M all worlds in which χ is false. Every epistemic relations E_i, every preference orderings $\preceq_i$, every function R_{δ_C}, and the valuation π are restricted to the worlds in which χ is true.

In the resulting structure M^χ, the relations $\sim^\chi$, $R^\chi_{\delta_C}$, E^χ_i, $\preceq^\chi_i$ verify the constraints **C1** to **C7** because of the syntactic restriction $\chi \in \mathcal{L}_{\mathcal{GT}}$. This result is summed up in the following theorem:

Theorem 43 *Let $\chi \in \mathcal{L}_{\mathcal{GT}}$. If M is a $\mathcal{MLEG}$ model then M^χ is a $\mathcal{MLEG}$ model.*

Proof.

It is just a routine to verify that $\sim^\chi$ and every E^χ_i are equivalence relations, every $\preceq^\chi_i$ is reflexive and transitive, and the model M^χ satisfies the semantic constraints **C1**, **C3**, **C5**, **C6** and **C7**.

Let us prove that M^χ satisfies constraints **C2** and **C4**.

We first prove that M^χ satisfies constraint **C2**. We introduce the following useful notation. Suppose $\chi_1, \chi_2 \in \mathcal{L}_{\mathcal{GT}}$. Then, $\chi_2 \rightsquigarrow \chi_3$ iff there is $\chi_3 \in \mathcal{L}_{\mathcal{GT}}$ such that $\chi_1 = \chi_2 \wedge \chi_3$.

Now, suppose for every $i \in Agt$ there is v_i such that $v_i \sim^\chi w$ and $R^\chi_i(v_i) = \delta_i$. It follows that for every $i \in Agt$ there is v_i such that $v_i \sim w$ and $R_i(v_i) = \delta_i$. The latter implies that there is v such that $v \sim w$ and $R_\delta(v) \neq \emptyset$ (by the semantic constraint **C2**). Now, suppose for all v' if $v' \sim^\chi w$ then $R^\chi_{Agt}(v') = \delta$. It follows that: there is $i \in Agt$ and $\psi \in \mathcal{L}_{\mathcal{MLEG}}$ such that $\Box\psi \rightarrow [\delta_i]\bot \rightsquigarrow \chi$ and $M, v \models \Box\psi$. The latter implies that there is $i \in Agt$ and $\psi \in \mathcal{L}_{\mathcal{MLEG}}$ such that $\Box\psi \rightarrow [\delta_i]\bot \rightsquigarrow \chi$ and for all $v' \sim w$, $M, v' \models \Box\psi$. We conclude that there is no $v_i \sim^\chi w$ such that $R^\chi_i(v_i) = \delta_i$ which leads to a contradiction.

We now consider constraint **C4**. Suppose $wE^\chi_i v$ and $R^\chi_i(w) = i{:}a$. It follows that $wE_i v$ and $R_i(w) = i{:}a$ which implies $R_i(v) = i{:}a$, because M satisfies constraint **C4**. The latter implies $R^\chi_i(v) = i{:}a$. Now, suppose $wE^\chi_i v$ and $R^\chi_i(v) = i{:}a$. It follows that $wE_i v$ and $R_i(v) = i{:}a$ which implies $R_i(w) = i{:}a$, because M satisfies constraint **C4**. The latter implies $R^\chi_i(w) = i{:}a$.

■

Note that in the general case where $\chi \in \mathcal{L}_{\mathcal{MLEG}}$, the theorem above is false.

We have reduction axioms for $\chi!$ which guarantee the completeness of the logic $\mathcal{MLEG}^{\mathcal{GT}}$ explaining how a dynamic operator $[\chi!]$ interacts with the Boolean operators and modal logic operators of $\mathcal{MLEG}$.

Theorem 44 *The following schemata are valid in the logic* $\mathcal{MLEG}^{\mathcal{GT}}$.

R1. $[\chi!]p \leftrightarrow (\chi \rightarrow p)$
R2. $[\chi!]\neg\varphi \leftrightarrow (\chi \rightarrow \neg[\chi!]\varphi)$
R3. $[\chi!](\varphi_1 \wedge \varphi_2) \leftrightarrow ([\chi!]\varphi_1 \wedge [\chi!]\varphi_2)$
R4. $[\chi!]\Box\varphi \leftrightarrow (\chi \rightarrow \Box[\chi!]\varphi)$
R5. $[\chi!]\mathsf{K}_i\varphi \leftrightarrow (\chi \rightarrow \mathsf{K}_i[\chi!]\varphi)$
R6. $[\chi!]\,[\mathsf{good}]_i\,\varphi \leftrightarrow (\chi \rightarrow [\mathsf{good}]_i\,[\chi!]\varphi)$
R7. $[\chi!]\,[\delta_C]\,\varphi \leftrightarrow (\langle\delta_C\rangle\top \rightarrow [\chi!]\varphi)$

PROOF.

The proofs of **R1-R6** go as in Dynamic Epistemic Logic (DEL) (see [vDvdHK07]). We here prove **R7**.
$M, w \models [\chi!]\,[\delta_C]\,\varphi$,
IFF if $M, w \models \chi$ then $M^\chi, w \models [\delta_C]\,\varphi$,
IFF if $M, w \models \chi$ then $M^\chi, w \models \langle\delta_C\rangle\top \rightarrow \varphi$ (by Axiom $\mathbf{Def}_{[\delta_C]}$),
IFF if $M, w \models \chi$ then $M^\chi, w \models [\delta_C]\,\bot$ or $M^\chi, w \models \varphi$,
IFF if $M^\chi, w \models \langle\delta_C\rangle\top$ then, if $M, w \models \chi$ then $M^\chi, w \models \varphi$,
IFF if $M^\chi, w \models \langle\delta_C\rangle\top$ then, $M, w \models [\chi!]\varphi$,
IFF if $M, w \models \langle\delta_C\rangle\top$ then, $M, w \models [\chi!]\varphi$,
IFF if $M, w \models \langle\delta_C\rangle\top \rightarrow [\chi!]\varphi$. ■

The principles **R1.-R7.** are called reduction axioms because, read from left to right, they reduce the complexity of those operators in a formula. In particular the principles **R1.-R7.** explains how to transform any formula φ of the language with dynamic operators in a formula without dynamic operators. More generally, we have an axiomatization result:

Theorem 45 *The logic* $\mathcal{MLEG}^{\mathcal{GT}}$ *is completely axiomatized by the axioms and inference rules of* $\mathcal{MLEG}$ *together with the schemata of Theorem 44 together with the following rule of replacement of proved equivalence:*

$$\frac{\psi_1 \leftrightarrow \psi_2}{\varphi \leftrightarrow \varphi[\psi_1 := \psi_2]}$$

where $\varphi[\psi_1 := \psi_2]$ *is the formula* φ *in which we have replaced all occurrences of* ψ_1 *by* ψ_2.

PROOF.

By means of the principles **R1-R7** in Theorem 44, it is straightforward to prove that for every $\mathcal{MLEG}^{GT}$ formula there is an equivalent $\mathcal{MLEG}$ formula. In fact, each reduction axiom **R2-R7**, when applied from the left to the right by means of the rule of replacement of proved equivalence, yields a simpler formula, where 'simpler' roughly speaking means that the dynamic operator is pushed inwards. Once the dynamic operator attains an atom it is eliminated by the equivalence **R1**. Hence, the completeness of $\mathcal{MLEG}^{GT}$ is a straightforward consequence of Theorem 38. ■

Now, consider the following formula:

$$\chi_{\mathsf{SD}} \stackrel{\text{def}}{=} \bigwedge_{i \in Agt, a \in Act} (\Box \mathsf{SD}^{\leq 0}(i{:}a) \rightarrow [i{:}a]\, \bot).$$

where $\mathsf{SD}^{\leq 0}(i{:}a)$ has been defined in Subsection 9.3.3. The effect of the game transformation event $\chi_{\mathsf{SD}}!$ is to delete from every game $\sim(w)$ in the model M all worlds in which a strictly dominated strategy is played by some agent.

As the following Theorem 46 highlights, the procedure IDSDS that we have characterized in Section 9.3.3 in the static $\mathcal{MLEG}$ can be characterized in a more compact way in $\mathcal{MLEG}^{\mathcal{GT}}$. Suppose δ is the selected strategy profile. Then, for every agent i, δ_i survives IDSDS until the subgame of depth $n{+}1$ if and only if, the event $\chi_{\mathsf{SD}}!$ can occur $n{+}1$ times in sequence.

Theorem 46 *For all $\delta \in \Delta$, for all $n \geq 0$,*
$\vdash_{\mathcal{MLEG}^{\mathcal{GT}}} \langle\delta\rangle\top \rightarrow \big(\neg \mathsf{SD}^{\leq n}(\delta) \leftrightarrow \langle \chi_{\mathsf{SD}}! \rangle^{n+1} \top\big)$.

PROOF.

We are going first to prove the theorem by induction.

Let us begin to prove the case $n = 0$. Let M, w be a $\mathcal{MLEG}$-pointed-model such that $M, w \models \langle\delta\rangle\top$. $M, w \models \neg \mathsf{SD}^{\leq 0}(\delta)$ means that for all $i \in Agt$, we have $M, w \models \neg \mathsf{SD}^{\leq 0}(\delta_i)$. It is equivalent to: for all $i \in Agt$, for all $a \in Act$, $M, w \models \mathsf{SD}^{\leq 0}(i{:}a) \rightarrow [i{:}a]\bot$ (indeed, if $a = \delta_i$, we have $M, w \models \neg \mathsf{SD}^{\leq 0}(\delta_i)$ and if $a \neq \delta_i$, we have $M, w \models [i{:}a]\bot$). So it is equivalent to $M, w \models \chi_{\mathsf{SD}}$ which is equivalent to $M, w \models \langle \chi_{\mathsf{SD}}! \rangle \top$.

Now, we suppose the theorem true for $n - 1$. We suppose that

$$\vdash_{\mathcal{MLEG}^{GT}} \langle\delta\rangle\top \rightarrow \big(\neg \mathsf{SD}^{\leq n-1}(\delta) \leftrightarrow \langle \chi_{\mathsf{SD}}! \rangle^{n} \top\big) \qquad (*)$$

.

Lemma 15 *Let M, w a $\mathcal{MLEG}$-pointed model. Let $i \in Agt$. There exists $a \in Act$ such that $M \models \neg \mathsf{SD}^{\leq 0}(i{:}a)$.*

PROOF.

By contradiction. Suppose for all $a \in Act$, we have $M, w \models \mathsf{SD}^{\leq 0}(i{:}a)$. Let $\beta \in \Delta$. Let $a_1 \in Act$ such that $M, w \models \Diamond\langle i{:}a_1\rangle\top$. We have $M, w \models \mathsf{SD}^{\leq 0}(i{:}a_1)$. We recall the definition of $\mathsf{SD}^{\leq 0}(i{:}a_1)$:

$$\mathsf{SD}^{\leq 0}(i{:}a_1) \stackrel{\text{def}}{=} \langle i{:}a_1\rangle\top \rightarrow$$

$$\bigvee_{a_2 \in Act} (\Diamond\langle i{:}a_2\rangle\top \wedge \bigwedge_{\delta \in \Delta} (\Diamond\langle \delta_{-i}\rangle \rightarrow (\langle \delta_{-i}, i{:}a_1\rangle <_i \langle \delta_{-i}, i{:}a_2\rangle))).$$

By definition of $\mathsf{SD}^{\leq 0}(i{:}a_1)$, there exists $a_2 \in Act$ such that $\langle \delta_{-i}, i{:}a_1\rangle <_i \langle \delta_{-i}, i{:}a_2\rangle$.

We have $M, w \models \mathsf{SD}^{\leq 0}(i{:}a_2)$. So we can find a_3 such that $M, w \models \langle \delta_{-i}, i{:}a_2\rangle <_i \langle \delta_{-i}, i{:}a_3\rangle$. We continue the process and we define a sequence of actions $a_1, a_2, a_3, \ldots$ such that for all $j \geq 1$, $M, w \models \langle \delta_{-i}, i{:}a_j\rangle <_i \langle \delta_{-i}, i{:}a_{j+1}\rangle$. But Act is finite, so there exists $k > 1$ such that $a_1 = a_k$. By transitivity of $<_i$, we have $M, w \models \langle \delta_{-i}, i{:}a_1\rangle <_i \langle \delta_{-i}, i{:}a_k\rangle$. This is not possible.

■

Lemma 16 *For all $i \in Agt$, for all $a \in Act$,*

- $$M, w \models \neg\mathsf{SD}^{\leq 0}(i{:}a) \text{ iff } M^{\chi_{\mathsf{SD}}}, w \models \Diamond\langle i{:}a\rangle\top.$$
- *for all $n \geq 0$, we have:*
$$M, w \models \neg\mathsf{SD}^{\leq n+1}(i{:}a) \text{ iff } M^{\chi_{\mathsf{SD}}}, w \models \neg\mathsf{SD}^{\leq n}(i{:}a).$$

PROOF.

- Consider a pointed-model M, w such that $M, w \models \neg\mathsf{SD}^{\leq 0}(i{:}a)$. Thus, by definition of $\neg\mathsf{SD}^{\leq 0}(i{:}a)$, we have $M, w \models \Diamond\langle i{:}a\rangle\top$. According to Lemma 15 we have for all $j \neq i$ the existence of β_j such that $M, w \models \neg\mathsf{SD}^{\leq 0}(\beta_j)$. We define δ as $\delta_i = i{:}a$ and $\delta_j = \beta_j$ for all $j \neq i$. So, by the semantic constraint **C2**, there exists a point u such that $w \sim u$ and $M, u \models \langle\delta\rangle\top$. The world u is not removed by the event $\chi_{\mathsf{SD}}!$. Thus, we have $M^{\chi_{\mathsf{SD}}}, u \models \langle i{:}a\rangle\top$.

  If $M, w \models \mathsf{SD}^{\leq 0}(i{:}a)$, then all worlds w in which $M, w \models \langle i{:}a\rangle\top$ are removed because χ_{SD} is false in w. So $M^{\chi_{\mathsf{SD}}}, w \not\models \Diamond\langle i{:}a\rangle\top$.

- The second point is the induction case. You can read the first case as the initial case of induction by defining $\neg\mathsf{SD}^{\leq -1}(i{:}a) \stackrel{\text{def}}{=} \Diamond\langle i{:}a\rangle\top$.

  Let $n \in \mathbb{N}$. Suppose that we have $M \models \neg\mathsf{SD}^{\leq n}(i{:}a)$ iff $M^{\chi_{\mathsf{SD}}} \models \neg\mathsf{SD}^{\leq n-1}(i{:}a)$.

  We leave to the reader checking that the latter implies $M \models \neg\mathsf{SD}^{\leq n+1}(i{:}a)$ iff $M^{\chi_{\mathsf{SD}}} \models \neg\mathsf{SD}^{\leq n}(i{:}a)$.

■

Let M, w be a $\mathcal{MLEG}$-pointed-model such that $M, w \models \langle\delta\rangle\top$. If $M, w \models \neg\mathsf{SD}^{\leq n}(\delta)$ then $M, w \models \neg\mathsf{SD}^{\leq 0}(\delta)$. So, $M, w \models \langle\chi_{\mathsf{SD}}!\rangle\top$, and w remains in $M^{\chi_{\mathsf{SD}}}$.

As $M, w \models \neg\mathsf{SD}^{\leq n}(\delta)$, the lemma gives us $M^{\chi_{\mathsf{SD}}}, w \models \neg\mathsf{SD}^{\leq n-1}(\delta)$. Now we are going to apply the induction hypothesis (*). We obtain $M^{\chi_{\mathsf{SD}}}, w \models \langle\chi_{\mathsf{SD}}!\rangle^{n}\top$. So $M, w \models \langle\chi_{\mathsf{SD}}!\rangle^{n+1}\top$.

If $M, w \models \langle\chi_{\mathsf{SD}}!\rangle^{n+1}\top$, we have also $M, w \models \langle\chi_{\mathsf{SD}}!\rangle\top$. So w remains in $M^{\chi_{\mathsf{SD}}}$. By applying Lemma 16 and induction hypothesis (*), we obtain $M, w \models \neg\mathsf{SD}^{\leq n-1}(\delta)$. Finally, $M, w \models \neg\mathsf{SD}^{\leq n}(\delta)$.

■

The above theorem says that if δ is performed, then the formula $\neg\mathsf{SD}^{\leq n}(\delta)$, defined in Subsection 9.3.3, whose length is exponential in n and the more compact formula $\langle\chi_{\mathsf{SD}}!\rangle^{n+1}\top$ are equivalent. Indeed the length of the formula $\langle\chi_{\mathsf{SD}}!\rangle^{n+1}\top$ is $O(n(|Agt||Act|)^2)$ where n is the number of IDSDS rounds, $|Agt|$ is the number of agents and $|Act|$ is the maximal number of actions. Finally, here is a compact reformulation of Theorem 42 in $\mathcal{MLEG}^{\mathcal{GT}}$.

Theorem 47 *For all* $n \geq 0$, $\vdash_{\mathcal{MLEG}^{\mathcal{GT}}} \left(\mathsf{MK}^{n}_{Agt} \bigwedge_{i\in Agt} \mathsf{Rat}_i\right) \rightarrow \langle\chi_{\mathsf{SD}}!\rangle^{n+1}\top$.

Proof.

By Theorem 42 and 46. Indeed, let M, w a $\mathcal{MLEG}$-pointed-model such that $M, w \models \mathsf{MK}^{n}_{Agt} \bigwedge_{i\in Agt} \mathsf{Rat}_i$. There exists $\delta \in \Delta$ such that $M, w \models \langle\delta\rangle\top$. Theorem 42 gives $M, w \models \neg\mathsf{SD}^{\leq n}(\delta)$. Theorem 46 gives $M, w \models \neg\mathsf{SD}^{\leq n}(\delta) \leftrightarrow \langle\chi_{\mathsf{SD}}!\rangle^{n+1}\top$. So, $M, w \models \langle\chi_{\mathsf{SD}}!\rangle^{n+1}\top$. ■

9.5 Imperfect information

We here consider a more general class of games which includes strategic games with imperfect information about the game structure including the players' strategy sets (or action repertoires) and the players' preference ordering over strategy profiles. This kind of games have been explored in the past by Harsanyi [Har67]. A more recent analysis is given by [HR07]).

We are interested here in verifying whether the results obtained in Sections 9.3.2 and 9.3.3 can be generalized to this kind of games, that is:

1. Are rationality of every player and every agent's knowledge about other agents' choices still sufficient to ensure that the selected strategy profile is a Nash equilibrium in a strategic game with imperfect information about the game structure?

2. Is mutual knowledge of rationality among the players still sufficient to ensure that the selected strategy profile survives iterated deletion of dominated strategies in a strategic game with imperfect information about the game structure?

To answer these questions, we have to remove Axiom **PerfectInfo** of the form $\Box\varphi \rightarrow \mathsf{K}_i\varphi$ from $\mathcal{MLEG}$ and the corresponding semantic constraint **C5** from the definition of $\mathcal{MLEG}$ frames expressing the hypothesis of perfect information about the game structure. We call $\mathcal{MLEG}^*$ the resulting logic and $\mathcal{MLEG}^*$-models the resulting class of models. Then we have to check whether Theorems 41 and 42 given in Sections 9.3.2 and 9.3.3 are still derivable in $\mathcal{MLEG}^*$.

We have a positive answer to the previous first question. Indeed, the formula

$$\Big(\Big(\bigwedge_{i\in Agt}\mathsf{Rat}_i\Big) \wedge \bigwedge_{i\in Agt}\mathsf{K}_i\langle\delta_{-i}\rangle\top\Big) \rightarrow \mathsf{Nash}(\delta)$$

is derivable in $\mathcal{MLEG}^*$. But we have a negative answer to the second question. Indeed, the following formula is invalid in $\mathcal{MLEG}^*$ for every $\delta \in \Delta$ and for every $n \in \mathbb{N}$ such that $n > 0$:

$$\Big(\Big(\mathsf{MK}^n_{Agt}\bigwedge_{i\in Agt}\mathsf{Rat}_i\Big) \wedge \langle\delta\rangle\top\Big) \rightarrow \neg\mathsf{SD}^{\leq n}(\delta).$$

This can be proved as follows. We suppose $Agt = \{1, 2\}$ and we exhibit in Figure 9.3 a $\mathcal{MLEG}^*$-model M and a world w_1 in M in which for all n, $(\mathsf{MK}^n_{\{1,2\}}\bigwedge_{i\in\{1,2\}}\mathsf{Rat}_i) \wedge \langle 1{:}main\rangle\top \wedge \mathsf{SD}^{\leq 1}(1{:}main)$ is true. We call Alarm Game the scenario corresponding to this model.

Scenario description. We call Alarm Game the scenario represented by the model in Figure 9.3. Agent 1 is a thief who intends to burgle agent 2's apartment. Agent 1 can enter the apartment either by the main door or by the back door (action 1:*main* or action 1:*back*). Agent 2 has two actions available. Either he does nothing (action 2:*skip*) or he follows a security procedure (action 2:*proc*) which consists in locking the two doors and in activating a surveillance camera on the main door. Entering the apartment by the main door when agent 2 does nothing (i.e. the strategy profile $\langle 1{:}main, 2{:}skip\rangle$ executed at world w_2) and entering by the back door when agent 2 does nothing (i.e. the strategy profile $\langle 1{:}back, 2{:}skip\rangle$ executed at world w_4) are for agent 1 the best situations and are for him equally preferable. Indeed, in both cases agent 1 will successfully enter and burgle the apartment. On the contrary, trying to enter the apartment by the back door when 2 follows the security procedure (i.e. the strategy profile $\langle 1{:}back, 2{:}proc\rangle$ executed at world w_3) is for 1 strictly better than trying to enter by the main door when 2 follows the security procedure (i.e. the strategy profile $\langle 1{:}main, 2{:}proc\rangle$ executed at world w_1). Indeed, in the former case agent 1 will be simply unable to burgle the apartment, in the latter case not only he will be unable to burgle the apartment but also he will disclose his identity. The two possible situations in which agent 1 does not succeed in burgling the apartment (worlds w_1 and w_3) are equally

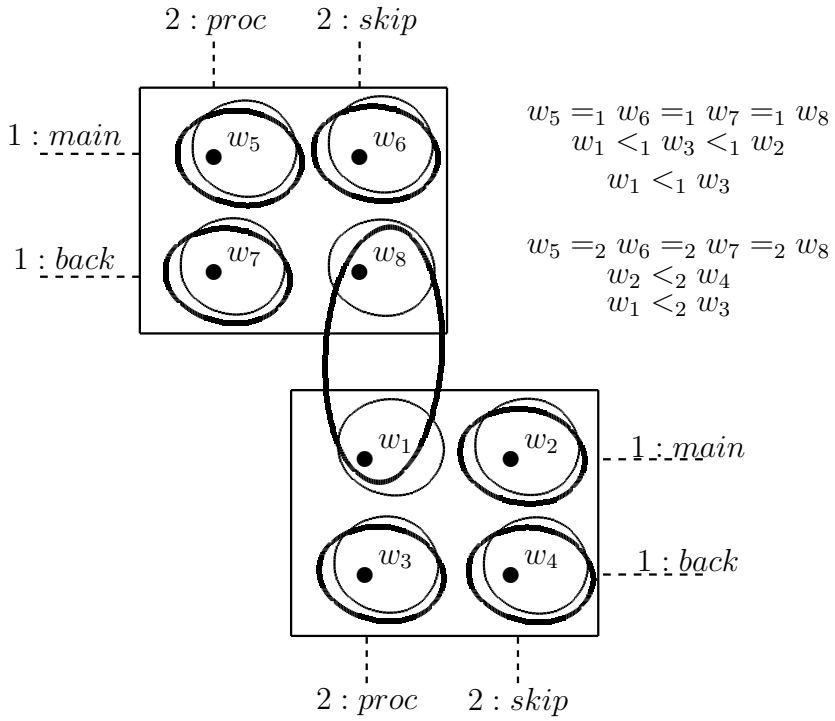


Figure 9.3: Alarm Game. Again thick circles represent epistemic possibility relations for agent 1 whereas thin circles represent epistemic possibility relations for agent 2. The two equivalence classes $\sim(w_1) = \{w_1, w_2, w_3, w_4\}$ and $\sim(w_5) = \{w_5, w_6, w_7, w_8\}$ correspond to two different games where agents have different preference ordering over strategy profiles.

preferable for agent 2 and are for 2 strictly better than the situations in which agent 1 successfully burgles the apartment (worlds w_2 and w_4).

At world w_1 agent 1 enters by the main door while agent 2 follows the security procedure. This is the only world in the model M in which agent 1 has some uncertainty. Indeed, in this world agent 1 can imagine the alternative game defined by the equivalence class $\sim(w_5) = \{w_5, w_6, w_7, w_8\}$ in which he enters by the back door while agent 2 does nothing (world w_8). We suppose that in such a game, even if agent 2 follows the security procedure, agent 1 will succeed in burgling his apartment. This is the reason why the four strategy profiles $\langle 1{:}main, 2{:}skip\rangle$, $\langle 1{:}back, 2{:}skip\rangle$, $\langle 1{:}main, 2{:}proc\rangle$ and $\langle 1{:}back, 2{:}proc\rangle$ are equally preferable for the two agents.

Concerning the automated reasoning aspects of the logic $\mathcal{MLEG}^*$, we can prove that the complexity of the satisfiability problem increases and reaches the complexity of the satisfiability problem of epistemic modal logic.

Theorem 48 • *If* $\text{card}(Agt) = 1$ *and* $\text{card}(Act) = 1$ *then the satisfiability problem of a given formula* φ *in a* $\mathcal{MLEG}^*$*-model is NP-complete.*

• *If* $\text{card}(Agt) \geq 2$ *or* $\text{card}(Act) \geq 2$ *the satisfiability problem of a given formula* φ *in a* $\mathcal{MLEG}^*$*-model is PSPACE-complete.*

PROOF.

We give here some hint for the proof. When there is only one agent and $card(Act) = 1$ then the games are trivial and reduced to singletons. In these settings, a $\mathcal{MLEG}^*$-frame $F = \langle W, \sim, R, E, \preceq \rangle$ is such that $\sim$ and $\preceq_i$ for each agent i are equal to the relation $\{(w, w) \mid w \in W\}$. So the modal operators $[\mathsf{good}]_i$ and $\Box$ are superfluous. The operator $[\delta_C]$ can be treated as a proposition. Hence the logic is similar to the logic S5 which is NP. This is the main argument why when there is only one agent and $card(Act) = 1$ the logic $\mathcal{MLEG}^*$ is NP. NP-hardness is granted because $\mathcal{MLEG}^*$ is a conservative extension of Classical Propositional Logic.

Now let us prove that the satisfiability problem of a given formula φ in a $\mathcal{MLEG}^*$-model is PSPACE-hard in other cases. First let us consider the case where $card(Agt) \geq 2$. Let us consider two distinct agents $a, b \in Agt$. Let φ be a formula written only with atomic propositions and with epistemic modal operators K_a and K_b. We have equivalence between:

1. φ is satisfiable in a $\mathcal{MLEG}^*$-model;
2. φ is satisfiable in the logic $\text{S5}_2(K_a, K_b)$ (i.e. the fusion of the logic S5 for K_a and S5 for K_b).

The direction 1. $\rightarrow$ 2. is straightforward and is already true with the assumption of the Axiom **PerfectInfo**. The direction 2. $\rightarrow$ 1. is true because Axiom **PerfectInfo** has now disappeared. So we can easily transform a model of the epistemic modal logic into a $\mathcal{MLEG}^*$-model. Note that in the case of the logic $\mathcal{MLEG}$, the direction 2. $\rightarrow$ 1. is not true anymore. Indeed, it is not possible to transform a model of $\text{S5}_2(K_a, K_b)$ with more than $card(Act)^{card(Agt)}$ worlds into a $\mathcal{MLEG}$-model. Hence, we have reduced the satisfiability problem of a given formula φ in a $\mathcal{MLEG}^*$-model into the satisfiability problem of a given formula ψ of $\text{S5}_2(K_a, K_b)$ which is PSPACE-hard. So the satisfiability problem of a given formula φ in a $\mathcal{MLEG}^*$-model is PSPACE-hard.

Now let us the consider the case where $Agt = \{a\}$ and $card(Act) \geq 2$. Let a and b be two distinct actions. We prove that we can reduce the satisfiability problem of a given formula φ in a $\mathcal{MLEG}^*$-model to the satisfiability problem of K. Here is a possible translation:

• $tr_0(\blacksquare\psi) = i{:}a \wedge \Diamond K_a tr_1(\psi)$ where $\blacksquare$ is the K-operator;

- $tr_1(\blacksquare\psi) = i{:}b \wedge \Diamond K_a tr_0(\psi)$ where $\blacksquare$ is the K-operator;
- $tr_0(p) = i{:}a \wedge p$ for all propositions p;
- $tr_1(p) = i{:}b \wedge p$ for all propositions p.

And φ is satisfiable in K iff $tr_0(\varphi)$ is satisfiable in $\mathcal{MLEG}^*$. Hence, the logic $\mathcal{MLEG}^*$ is also PSPACE-hard in this case.

Now we are going to prove that the satisfiability problem of $\mathcal{MLEG}^*$ is PSPACE. We do not give all the details but we give the idea for a tableau method [?] for the logic $\mathcal{MLEG}^*$. The tableau method is a non-deterministic procedure. The creation of a model proceeds as follows:

- We start the procedure by guessing a "grid", that is to say an equivalence class for the relation $\sim$ of maximal size $card(Act)^{card(Agt)}$ and also its preference relation as in the algorithm of Theorem 39. We also choose non-deterministically a world w in this class.
- We adapt the classical tableau method rules for the epistemic modal logic [?], that is to say:
  - Suppose that a world w contains a formula of the form $\mathsf{K}_i\psi$. Then we propagate the formula ψ in all nodes v such that wE_iv.
  - Suppose that a world w contains a formula of the form $\hat{K}_i\psi$. Then we create an equivalence class for $\sim$, we choose a point v such that $R_i(v) = R_i(w)$ in this equivalence class and we propagate ψ in v.
- Suppose that a node w contains a formula $\Box\psi$. Then we propagate the formula ψ in all nodes v such that $v \sim w$;
- Suppose that a node w contains a formula $\Diamond\psi$. Then we choose non-determiniscally a world v such that $v \sim w$ and we propagate ψ in v.
- Suppose that a node w contains a formula $[\mathsf{good}]_i\,\varphi\psi$. Then we propagate the formula ψ in all nodes v such that $v \preceq_i w$;
- Suppose that a node w contains a formula $\langle\mathsf{good}\rangle_i\psi$. Then we choose non-determiniscally a world v such that $v \preceq_i w$ and we propagate ψ in v.

During the construction, we explore the structure in depth first so that we only need to have one branch in memory at each step. Thus, the algorithm is a non-deterministic procedure that uses only a polynomial amount of memory. So the satisfiability problem of $\mathcal{MLEG}^*$ is in NPSPACE. According the Savitch's theorem [Sav70], it is in PSPACE.

■

9.6 Weaker forms of perfect information

In the previous section, we have removed Axiom **PerfectInfo** of the form $\Box\varphi \rightarrow \mathsf{K}_i\varphi$ from the logic $\mathcal{MLEG}$ to obtain a new logic $\mathcal{MLEG}^*$ in which agents may have imperfect information about all aspects of the game they play, including the players' strategy sets (or action repertoires) and the players' preference ordering over strategy profiles.

Nevertheless, in some cases we would like to suppose that agents have perfect information about some specific aspects of the game they play. For example, we would like to suppose that:

1. an agent has perfect information about his strategy sets even though he may have imperfect information about other agents' strategy sets or,

2. that an agent has perfect information about the strategy set of every agent even though he may have imperfect information about agents' preference ordering over strategy profiles.

The former assumption applies to the scenario in which a robber enters a bank, approaches the bank teller and demands money waving a gun. In this situation the bank teller has perfect information about his strategy set: he knows that he can either sound the alarm or do nothing. But the bank teller does not know the robber's strategy set, as he is not sure whether the robber's gun is loaded or not (i.e. the bank teller does not know whether the robber is able to kill him by shooting). The latter assumption applies to a card game like Poker. In Poker a player has perfect information about every player's strategy set, as he knows that a given point in the game a player has the option to check (if no bet is in front of him), bet, or fold. However, a Poker player has imperfect information about other players' preference ordering over strategy profiles, as he cannot see other players' cards.

In this section, we are going to show how to relax the axiom **PerfectInfo** in order to be able to model the previous assumptions. If we replace Axiom **PerfectInfo** by the following axiom schemas:

$$\Diamond\langle i{:}a\rangle\top \rightarrow \mathsf{K}_i\Diamond\langle i{:}a\rangle\top \qquad (\textbf{PerfectInfoStrategy}_i)$$

for all $i \in Agt$ and $a \in Act$, then every agent i has perfect information about his strategy set. That is, if an agent i can perform an action a then agent i knows that he can perform action a. Axiom $\textbf{PerfectInfoStrategy}_i$ corresponds to the following semantic constraint on models. For every $i \in Agt$ and $a \in Act$:

- if wE_iu and there is v such that $w \sim v$ and $i{:}a = R_i(v)$ then, there is z such that $u \sim z$ and $i{:}a = R_i(z)$.

If we replace Axiom **PerfectInfo** by the following axiom schemas:

$$\Diamond\langle j{:}a\rangle\top \rightarrow \mathsf{K}_i\Diamond\langle j{:}a\rangle\top \qquad (\mathbf{PerfectInfoStrategy}_{i,j})$$

for all $i, j \in Agt$ and $a \in Act$, then an agent i has perfect information about the strategy sets of every agent. That is, if an agent j can perform an action a then every agent i knows that agent j can perform action a. Axiom $\mathbf{PerfectInfoStrategy}_{i,j}$ corresponds to the following semantic constraint on models. For every $i, j \in Agt$ and $a \in Act$:

- if wE_iu and there is v such that $w \sim v$ and $j{:}a = R_j(v)$ then, there is z such that $u \sim z$ and $j{:}a = R_j(z)$.

Obviously Axiom $\mathbf{PerfectInfoStrategy}_{i,j}$ is more general than Axiom $\mathbf{PerfectInfoStrategy}_i$, that is, $\mathbf{PerfectInfoStrategy}_{i,j}$ implies $\mathbf{PerfectInfoStrategy}_i$. It is also worth noting that the previous Axiom $\mathbf{PerfectInfoStrategy}_{i,j}$ together with Axiom **Indep** and Axiom **JointAct** imply $\Diamond\langle\delta\rangle\top \rightarrow \mathsf{K}_i\Diamond\langle\delta\rangle\top$. The latter means that if δ is a strategy profile of the current game then every agent knows this.

9.7 Related works

Although several modal logics of games in strategic forms have been proposed (see, e.g., [vdHJW05, Lor10a]), few modal logics exist which support reasoning about epistemic (strategic) games. Among them we should mention [dB04, Roy08, Bon08].

De Bruin [dB04] has developed a logical framework which enables to reason about the epistemic aspects of strategic games and of extensive games. His system deals with several game-theoretic concepts like the concepts of knowledge, rationality, Nash equilibrium, iterated strict dominance, backward induction. Nevertheless, de Bruin's approach differs from ours in several respects. First of all, our logical approach to epistemic games is *minimalistic* since it relies on few primitive concepts: knowledge, action, historical necessity and preference. All other notions such Nash equilibrium, rationality, iterated strict dominance are defined by means of these four primitive concepts. On the contrary, in de Bruin's logic all those notions are atomic propositions managed by a *ad hoc* axiomatization (see, e.g., [dB04, pp. 61,65] where special propositions for rationality and iterated strict dominance are introduced). Secondly, we provide a semantics and a complete axiomatics for our logic of epistemic games. De Bruin's approach is purely syntactic: no model-theoretic analysis of games is proposed nor completeness result for the proposed logic is given. Finally, de Bruin does not provide any complexity results

about his logic while we prove that the satisfiability problem of a formula in our logic is PSPACE-complete.

Roy [Roy08] has recently proposed a modal logic integrating preference, knowledge and intention. In his approach every world in a model is associated to a nominal which directly refers to a strategy profile in a strategic game. This approach is however limited in expressing formally the structure of a strategic game. In particular, in Roy's logic there is no principle like the $\mathcal{MLEG}$ Axiom **Indep** explaining how possible actions δ_i of individual agents are combined to form a strategy profile δ of the current game. Another limitation of Roy's approach is that it does not allow to express the concept of (weak) rationality that we have been able to define in Section 9.3.2 (see [Roy08, pp. 101]). As discussed in the previous sections this is a crucial concept in interactive epistemology since it is used for giving epistemic justifications of several solution concepts like Nash equilibrium and IDSDS (see Theorems 41 and 42).

Bonanno [Bon08] integrates modal operators for belief, common belief with constructions expressing agents' preferences over individual actions and strategy profiles, and applies them to the semantic characterization of solution concepts like iterated deletion of strictly dominated strategies (IDSDS) and iterated deletion of inferior profiles (IDIP). As in [Roy08], in Bonanno's logic every world in a model corresponds to a strategy profile of the current game. Although this logic allows to express the concept of weak rationality, it is not sufficiently general to enable to express in the object language solution concepts like Nash equilibrium and IDSDS (note that the latter is defined by Bonanno only in the metalanguage).

It is to be noted that, differently from $\mathcal{MLEG}$, most modal logics of epistemic games in strategic form (including Roy's logic and Bonanno's logic) postulate a one-to-one correspondence between models and games (i.e. every model of the logic corresponds to a unique strategic game, and worlds in the model are all strategy profiles of this game). Such an assumption is quite restrictive since it prevents from analyzing in the logic games with imperfect information about the game structure in which an agent can imagine alternative games. As shown in Section 9.5, this is something we can do in our logical framework by removing Axiom **PerfectInfo** from $\mathcal{MLEG}$.

Before concluding this section about related works it is to be noted that the approach to game dynamics based on Dynamic Epistemic Logic (DEL) we proposed in Section 9.4 is inspired by [vB07] in which strategic equilibrium is defined by fixed-points of operations of repeated announcement of suitable epistemic statements and rationality assertions. However, the analysis of epistemic games proposed in [vB07] is mainly semantical and the author does not provide a full-fledged modal language for epistemic games which allows to express in the object language solution concepts like Nash Equilibrium or IDSDS, and the concept of

rationality. Moreover, van Benthem's analysis does not include any completeness result for the proposed framework and there is no proposal of reduction axioms for a combination of DEL with a static logic of epistemic games. On the contrary, these two aspects are central in our analysis.

9.8 Conclusion

We have presented a multi-modal logic that enables to reason about epistemic games in strategic form. This logic, called $\mathcal{MLEG}$ (*Modal Logic of Epistemic Games*), integrates the concepts of joint action, preference and knowledge. We have shown that $\mathcal{MLEG}$ provides a highly flexible formal framework for the analysis of the epistemic aspects of strategic interaction. Indeed, $\mathcal{MLEG}$ can be easily adapted in order to integrate different assumptions on players' knowledge about the structure of a game.

Directions for future research are manifold. In this article (Section 9.3.2) we only considered the notion of individualistic rationality assumed in classical game theory: an agent decides to perform a certain action only if the agent believes that this action is a best response to what he expects the others will do. Our plan is to extend the present modal logic analysis of epistemic games to other forms of rationality such as fairness and reciprocity [FS03]. According to these notions of rationality, rational agents are not necessarily self-interested but they also consider the benefits of their choices for the group. Moreover, their decisions can be affected by their beliefs about other agents' willingness to act for the well-being of the group. In [Lor10b] we did some first steps into this direction.

Another aspect we intend to investigate in the future is a generalization of our approach to mixed strategies. Indeed, at the current stage the multi-modal logic $\mathcal{MLEG}$ only enables to reason about pure strategies. To this aim, we will have to extend $\mathcal{MLEG}$ by modal operators of probabilistic beliefs as the ones studied by [Hal03, FH94].

Chapter 10

Counterfactual emotions

In this Chapter we exploit the decidable fragment of STIT studied in Chapter 8 in order to express counterfactual emotions like regret and rejoicing. This study is part of [LS09]. This Chapter is organized as follows:

- First we see how to express counterfactual statements in STIT;
- Secondly we add an epistemic modal operator to the decidable fragment of STIT seen in Chapter 8;
- Finally we show how to express counterfactual emotions.

10.1 Counterfactual statements in STIT

In this section we exploit the STIT fragment *df*STIT studied in Section **??** in order to formalize counterfactual statements of the form "group J (or agent i) could have prevented a certain state of affairs χ to be true now". Such statements are indeed basic constituents of the appraisal patters of counterfactual emotions such as regret. In particular, counterfactual emotions such as regret originate from reasoning about this kind of statements highlighting the connection between the actual state of the world and a counterfactual state of the world that might have been had one chosen a different action. The counterfactual statements formalized in this section will be fundamental in the formalization of counterfactual emotions we will give in Section 10.3.

10.1.1 J could have prevented χ

The following counterfactual statement is a fundamental constituent of an analysis of counterfactual emotions:

(*) *J could have prevented* a certain state of affairs χ to be true now.

The statement just means that there is a *counterfactual dependence* between the state of affairs χ and group J (i.e. χ counterfactually depends on J's choice). The STIT fragment studied in Section **??** enables a formal translation of it. We denote this translation by $\texttt{CHP}_J\chi$, defined as follows:

$$\texttt{CHP}_J\chi \stackrel{\text{def}}{=} \chi \wedge \neg[AGT \setminus J]\chi.$$

The expression $\neg[AGT\setminus J]\chi$ just means that: the complement of J with respect to AGT (i.e. $AGT\setminus J$) does not see to it that χ (no matter what the agents in J have chosen to do). This is the same thing as saying that: given what the agents in $AGT\setminus J$ have chosen, there exists an alternative joint action of the agents in J such that, if the agents in J did choose this action, χ would be false now. Thus, χ and $\neg[AGT \setminus J]\chi$ together correctly translate the previous counterfactual statement (*). If J is a singleton $\{i\}$, we write $\texttt{CHP}_i\chi$ instead of $\texttt{CHP}_{\{i\}}\chi$ which means "agent i could have prevented χ to be true".

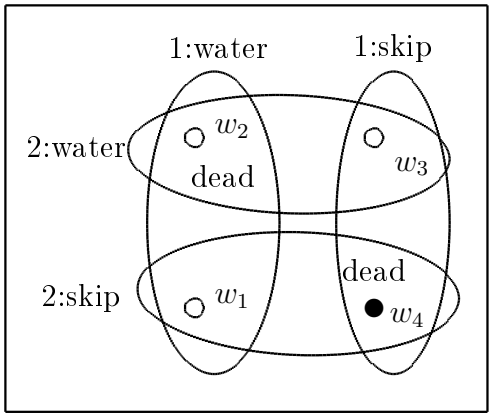


Figure 10.1: The four worlds w_1, w_2, w_3 and w_4 are in the equivalence class determined by $R_\emptyset$. Vertical circles represent the actions that agent 1 can choose, whereas horizontal circles represent the actions that agent 2 can choose. For example, w_1 is the world that results from agent 1 choosing the action *water* and agent 2 choosing the action *skip*.

Example 20 *Imagine a typical coordination scenario with two agents* $AGT = \{1, 2\}$. *Agents* 1 *and* 2 *have to take care of a plant. Each agent has only two actions available: water the plant (water) or do nothing (skip). If either both agents water the plant or both agents do nothing, the plant will die (dead). In the former case the plant will die since it does not tolerate too much water. In the*

latter case it will die since it lacks water. If one agent waters the plant and the other does nothing, the plant will survive (¬dead). The scenario is represented in the STIT model in Fig. 10.1. For instance both at world w_2 and w_4, formulas $\mathtt{CHP}_1$*dead and* $\mathtt{CHP}_2$*dead are true: each agent could have prevented the plant to be dead. Indeed, at world w_2, dead and* $\neg[2]$*dead are true: given what agent 2 has chosen (i.e. water), there exists an alternative action of agent 1 (i.e. skip) such that, if 1 did choose this action, dead would be false now. At world w_4, dead and* $\neg[2]$*dead are also true: given what agent 2 has chosen (i.e. skip), there exists an alternative action of agent 1 (i.e. water) such that, if 1 did choose this action, dead would be false now. The case for agent 2 is completely symmetrical.*

The following are some interesting properties of the operator $\mathtt{CHP}_J$. For every J and for every J_1, J_2 such that $J_1 \subseteq J_2$:

$$\models_{\mathsf{STIT}} \mathtt{CHP}_{J_1}(\chi_1 \vee \chi_2) \rightarrow (\mathtt{CHP}_{J_1}\chi_1 \vee \mathtt{CHP}_{J_1}\chi_2) \tag{10.1}$$

$$\models_{\mathsf{STIT}} \mathtt{CHP}_{J_1}\chi \rightarrow \mathtt{CHP}_{J_2}\chi \tag{10.2}$$

$$\models_{\mathsf{STIT}} (\mathtt{CHP}_{J_1}\chi_1 \wedge \mathtt{CHP}_{J_1}\chi_2) \rightarrow \mathtt{CHP}_{J_1}(\chi_1 \wedge \chi_2) \tag{10.3}$$

$$\models_{\mathsf{STIT}} \neg\mathtt{CHP}_J\top \tag{10.4}$$

$$\models_{\mathsf{STIT}} \neg\mathtt{CHP}_J\bot \tag{10.5}$$

Proof.

We give the proof of Validity 10.2 as an example. Let $\mathcal{M}$ be a STIT-model and $w \in W$ such that $\mathcal{M}, w \models \mathtt{CHP}_{J_1}\chi$. We have $\mathcal{M}, w \models \chi$ and $\mathcal{M}, w \models \neg[AGT \setminus J_1]\chi$. As $R_{AGT\setminus J_1} \subseteq R_{AGT\setminus J_2}$, it implies that $\mathcal{M}, w \models \neg[AGT \setminus J_2]\chi$. That is why we have $\mathcal{M}, w \models \mathtt{CHP}_{J_2}\chi$. ■

According to Validity 10.1, J_1 could have prevented χ_1 or could have prevented χ_2 implies J_1 could have prevented χ_1 or χ_2 to be true. Validity 10.2 expresses a monotonicity property: if J_1 is a subset of J_2 and J_1 could have prevented χ then, J_2 could have prevented χ as well. Validity 10.3 shows how the operator $\mathtt{CHP}_J$ behaves over conjunction: if J_1 could have prevented χ_1 to be true and could have prevented χ_2 to be true separately then J_1 could have prevented χ_1 and χ_2 to be true. Finally, according to the Validities 10.4 and 10.5, tautologies and contradictions cannot counterfactually depend on the choice of a group: it is never the case that a coalition J could have prevented a tautology (resp. a contradiction).

Remark 6 *It is worth noting that counterfactual statements of the form "group J (or agent i)* could have prevented *χ to be true", which are expressible in STIT, are not expressible in other well-known logics of multi-agent interaction such as Alternating-time temporal logic (ATL) [AHK02], Coalition Logic (CL) [Pau02], and some existing approaches based on ATL and CL (see, e.g., [ÅvdHW07, vdHJW05]).*

Admittedly, CL can express the statement "the group of agents $AGT \setminus J$ has not a joint strategy that force χ (in the next state)". In CL we can speak about agents abilities. But we can not express the statement "the current chosen actions of the group $AGT \setminus J$ does not force χ". In CL we can speak about the actions that agents have chosen.

It has been proved formally in [BHT06a] that STIT is more expressive than CL, and STIT extended with strategies (strategic STIT) is even more expressive than ATL. For instance there are STIT formulas such as $[J]\chi$ and $\neg[J]\chi$ that cannot be translated into ATL and CL.

10.1.2 Discussion

We have given above a logical translation of the statement "agent i could have prevented χ to be true" noted $\mathtt{CHP}_i\chi$ and expressing a counterfactual dependence between the state of affairs χ and agent i's choice. We said that the latter statement is true if and only if χ is true and, given what the other agents have chosen, there exists an alternative action of agent i such that, if i did choose this action, χ would be false now.

It is worth noting that $\mathtt{CHP}_i\chi$ does not cover situations in which agent i is partially responsible for χ *up to a certain degree* without being fully responsible for χ. The following voting example illustrates the difference between full responsibility and partial responsibility.

Example 21 *A and B are the two candidates for an election and* 1, 2, 3 *are the three voters. Suppose w_7 in the STIT model in Fig. 10.2 is the actual world. In this world, voter* 1 *and voter* 2 *vote for candidate A while voter* 3 *votes for candidate B so that A wins the election against B by a vote of 2-1. Formulas $\mathtt{CHP}_1$Awin and $\mathtt{CHP}_2$Awin are true at w_7. In fact, at w_7 candidate A wins the elections and, given what the other voters have chosen, there exists an alternative action of voter* 1 *(i.e. voting for candidate B) such that, if voter* 1 *did choose this action, candidate A would not win the elections. In other words, at w_7 the result of the election counterfactually depends on* 1*'s vote. The same is true for voter* 2*: at w_7 the result of the election counterfactually depends on* 2*'s vote. In this case, voter* 1 *and voter* 2 *can be said to be* ***fully*** *responsible for candidate A's win.*

Suppose now w_5 in the STIT model in Fig. 10.2 is the actual world. At w_5 candidate A wins the election against candidate B by a vote of 3-0. In this case, $\mathtt{CHP}_i$Awin is false for every voter, that is, for every voter the result of the election does not counterfactually depend on his vote. Nevertheless, we would like to say that each of the three voters is partially responsible for candidate A's win up to a certain degree. Indeed, voter 1 *is is a cause of A winning even if the vote is 3-0 because, under the contingency that one of the other voters had voted for candidate*

B instead, voter 1*'s vote would have become critical; if he had then changed his vote, candidate A would not have won. The same is true for voter* 2 *and for voter* 3.

It is not the objective of this paper to provide a logical account of the notion of partial responsibility and of the corresponding notion of degree of responsibility. These notions have been studied for instance in [CH04] in which the degree of responsibility of an event A for an event B is supposed to be $\frac{1}{N+1}$, where N is the minimal number of changes that have to be made to the actual situation before B counterfactually depends on A. For instance, in the case of the 3-0 vote in the previous example, the degree of responsibility of any voter for the victory of candidate A is $\frac{1}{2}$, since one change has to be made to the actual situation before a vote is critical. In the case of the 2-1 vote, the degree of responsibility of any voter for the victory is 1, since no change has to be made to the actual situation before a vote is critical.

10.2 A STIT extension with knowledge

10.2.1 knowledge

In order to capture the subjective dimension of emotions, this section presents an extension of the fragment *df*STIT of STIT logic presented in section **??** with standard operators for knowledge of the form K_i, where i is an agent. The formula $K_i\varphi$ means "agent i knows that φ is true". This is a necessary step for the formalization of counterfactual emotions that will be presented in section 10.3.

10.2.2 Definition

First we present the language $\mathcal{L}_{\mathsf{STIT}}$ of the Subsection 8.1 extended with epistemic constructions $K_i\varphi$. We give the language of all formulas we can construct with STIT operators and knowledge operators. The language $\mathcal{L}_{K\mathsf{STIT}}$ of the logic KSTIT is defined by the following BNF:

$$\varphi ::= p \mid \varphi \wedge \varphi \mid \neg\varphi \mid [J]\varphi \mid K_i\varphi$$

where p ranges over ATM, i ranges over AGT and J over 2^{AGT}.

For the same reasons that in Subsection 8.1, we are here interested in a fragment of $\mathcal{L}_{K\mathsf{STIT}}$. Indeed, the satisfiability problem of the logic KSTIT will be undecidable if the number of agents is more than 3 (because the logic KSTIT will be a conservative extension of the logic STIT which is already undecidable). So we focus into a syntactic fragment we call *df*KSTIT.

The language $\mathcal{L}_{dfK\mathsf{STIT}}$ of logic $dfK\mathsf{STIT}$ is defined by the following BNF:
$\chi ::= \bot \mid p \mid \chi \wedge \chi \mid \neg\chi$ (propositional formulas)
$\psi ::= [J]\chi \mid \psi \wedge \psi$ (see-to-it formulas)
$\varphi ::= \chi \mid \psi \mid \varphi \wedge \varphi \mid \neg\varphi \mid \langle\emptyset\rangle\psi \mid K_i\varphi$
(see-to-it, "can", knowledge formulas)
where p ranges over ATM, i ranges over AGT and J over $2^{AGT} \setminus \{\emptyset\}$.

For instance, $K_1\langle\emptyset\rangle[\{1,2\}]p \in \mathcal{L}_{dfK\mathsf{STIT}}$. But $\langle\emptyset\rangle K_1[\{1,2\}]p \notin \mathcal{L}_{dfK\mathsf{STIT}}$.

Let us give the semantics of the logic $dfK\mathsf{STIT}$. We start with the definition of model.

Definition 65 ($K\mathsf{STIT}$ -model)
A $K\mathsf{STIT}$ -model is a tuple $\mathcal{M} = (W, \{R_J\}_{J\subseteq AGT}, \{E_i\}_{i\in AGT}, V)$ where:

- $(W, \{R_J\}_{J\subseteq AGT}, V)$ is a STIT-model (see Definition 59);
- For all $i \in AGT$, E_i is an equivalence relation.

As usual truth conditions for atomic formulas and the boolean operators are entirely standard. Truth conditions for the STIT operators $[J]$ are given in Section ??. Truth conditions for knowledge operators are defined in the standard way:

$$\mathcal{M}, w \models K_i\varphi \text{ iff } \mathcal{M}, v \models \varphi \text{ for all } v \in W \text{ such that } wE_iv.$$

That is, agent i knows that φ at world w in model $\mathcal{M}$ if and only if φ is true at all worlds that are indistinguishable for agent i at world w.

As usual, a formula φ is $K\mathsf{STIT}$ -valid (noted $\models_{K\mathsf{STIT}} \varphi$) iff φ is true in every world of every $K\mathsf{STIT}$ -model. A formula φ is $K\mathsf{STIT}$ -satisfiable iff there exists a $K\mathsf{STIT}$ -model $\mathcal{M} = (W, \{R_J\}_{J\subseteq AGT}, \{E_i\}_{i\in AGT}, V)$ and a world $w \in W$ such that $\mathcal{M}, w \models \varphi$.

10.2.3 Decidability

The following is a extension of Corollary 10 given in Section 8.4.

Theorem 49 *The satisfiability problem of $dfK\mathsf{STIT}$ is NP-complete if* $\mathrm{card}(AGT) = 1$ *and PSPACE-complete if* $\mathrm{card}(AGT) \geq 2$.

10.2.4 Axiomatization

The study of an axiomatization for $\mathrm{df}K\mathsf{STIT}$ relies on an epistemic extension of the logic NCL presented in Section 8.3 which will also be axiomatizable. We call $K\mathsf{NCL}$ this epistemic extension of NCL. The syntax of the logic $K\mathsf{NCL}$ is the same as the logic $K\mathsf{STIT}$, that is to say $\mathcal{L}_{K\mathsf{NCL}} = \mathcal{L}_{K\mathsf{STIT}}$.

Let us now give the definition of model for the logic $K\mathsf{NCL}$.

Definition 66 (KNCL -model)
A KNCL -model is a tuple
$\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, \{E_i\}_{i \in AGT}, V)$ where:

- $(W, \{R_J\}_{J \subseteq AGT}, V)$ is a NCL-model (see Definition 61);
- For all $i \in AGT$, E_i is an equivalence relation.

Truth conditions, validity and satisfiability in KNCL are defined as usual.

We can now prove an extension of Theorem 37, stating the equivalence between the satisfiability in KNCL and the satisfiability in KSTIT if we restrict the formula to the syntactic fragment $\mathcal{L}_{df K\mathsf{STIT}}$.

Theorem 50 *Let φ be a formula of $\mathcal{L}_{df KSTIT}$. We have equivalence between:*

- *φ is satisfiable in KNCL ;*
- *φ is satisfiable in KSTIT .*

In the same way, we have an extension of the Corollary 11 about a complete axiomatization of the logic $df K$STIT.

Corollary 12 *A formula φ in $\mathcal{L}_{df KSTIT}$ is KSTIT -valid iff we have $\vdash_{KNCL} \varphi$ where $\vdash_{KNCL} \varphi$ means that there exists a proof of φ using all principles of the logic NCL, all principles of modal logic S5 for every K_i.*

10.3 A formalization of counterfactual emotions

In the following sections, we will use the STIT fragment extended with epistemic modalities studied in Section 10.2 and called $df K$STIT, in order to provide a logical formalization of this class of emotions. We consider four types of counterfactual emotions: regret and its positive counterpart rejoicing, disappointment and its positive counterpart elation.

10.3.1 Regret and rejoicing

In order to provide a logical characterization of counterfactual emotions such as regret, we need to introduce a concept of agent's preference. Modal operators for desires and goals have been widely studied (see e.g. [CL90, MvdHvL99]). The disadvantage of such approaches is that they complicate the underlying logical framework. An alternative, which we adopt in this paper is to label states with

atoms that capture the "goodness" of these states for an agent. Our approach supposes a binary relation of preference between worlds.

Let us introduce a special atom $good_i$ for every agent $i \in AGT$. These atoms are used to specify those worlds which are good for an agent.

We say that χ is good for agent i if and only if χ is true in all good/pleasant states. Formally:

$$\mathtt{GOOD}_i\chi \stackrel{\text{def}}{=} [\emptyset](good_i \rightarrow \chi).$$

Now, we are in a position to define the concept of desirable state of affairs. We say that χ is desirable for agent i if and only if, i knows that χ is something good for him:

$$\mathtt{DES}_i\chi \stackrel{\text{def}}{=} K_i\mathtt{GOOD}_i\chi.$$

As the following valid formulas highlight, every operator $\mathtt{DES}_i$ satisfies the principle K of normal modal logic, and the properties of positive and negative introspection: χ is (resp. is not) desirable for i if and only if i knows this.

$$\models_{K\mathsf{STIT}} (\mathtt{DES}_i\chi_1 \wedge \mathtt{DES}_i(\chi_1 \rightarrow \chi_2)) \rightarrow \mathtt{DES}_i\chi_2 \tag{10.6}$$

$$\models_{K\mathsf{STIT}} \mathtt{DES}_i\chi \leftrightarrow K_i\mathtt{DES}_i\chi \tag{10.7}$$

$$\models_{K\mathsf{STIT}} \neg\mathtt{DES}_i\chi \leftrightarrow K_i\neg\mathtt{DES}_i\chi \tag{10.8}$$

We have now all necessary and sufficient ingredients to define the cognitive structure of regret and to specify its counterfactual dimension. Such a dimension has been widely studied in the psychological literature where several authors (see, e.g., [LS82, Sug85, Roe97, KM86, Kah95, ZvDM98]) agree in considering regret as the emotion originating from an agent's comparison between the actual bad outcome and a *counterfactual* good outcome that might have been had the agent chosen a different action.

We say that an agent i regrets for χ if and only if $\neg\chi$ is desirable for i and i knows that it could have prevented χ to be true now. Formally:

$$\mathtt{REGRET}_i\chi \stackrel{\text{def}}{=} \mathtt{DES}_i\neg\chi \wedge K_i\mathtt{CHP}_i\chi.$$

The following example is given in order to better clarify this definition.

Example 22 *Consider the popular two-person hand game "Rock-paper-scissors". Each of the two players $AGT = \{1, 2\}$ has three available actions: play rock, play paper, play scissors. The goal of each player is to select an action which defeats that of the opponent. Combinations of actions are resolved as follows: rock wins against scissors, paper wins against rock; scissors wins against paper. If both players choose the same action, they both lose. The scenario is represented in the STIT model in Fig. 10.3. It is supposed winning is something good for each agent*

and each agent has the desire to win the game: $\mathtt{GOOD}_1 1Win$, $\mathtt{GOOD}_2 2Win$, $\mathtt{DES}_1 1Win$ *and* $\mathtt{DES}_2 2Win$ *are true at worlds* w_1-w_9. *Suppose world* w_1 *is the actual world in which* 1 *plays rock and* 2 *plays paper. In this world* 1 *loses the game* ($\neg 1Win$), *and* 1 *knows that (by playing scissors) it could have prevented* $\neg 1Win$ *to be true (i.e.* $K_1\mathtt{CHP}_1\neg 1Win$ *is true at* w_1*). It follows that at* w_1 *player* 1 *regrets for having lost the game, that is,* $\mathtt{REGRET}_1\neg 1Win$ *is true at* w_1.

As the following validity highlights, regret implies the frustration of an agent's desire:

$$\models_{K\mathsf{STIT}} \mathtt{REGRET}_i\chi \rightarrow (K_i\chi \wedge \mathtt{DES}_i\neg\chi) \tag{10.9}$$

More precisely, if agent i regrets for χ then, i knows that χ holds and $\neg\chi$ is something desirable for i (in this sense i feels frustrated for not having achieved $\neg\chi$). Moreover, regret satisfies the properties of positive and negative introspection:

$$\models_{K\mathsf{STIT}} \mathtt{REGRET}_i\chi \leftrightarrow K_i\mathtt{REGRET}_i\chi \tag{10.10}$$

$$\models_{K\mathsf{STIT}} \neg\mathtt{REGRET}_i\chi \leftrightarrow K_i\neg\mathtt{REGRET}_i\chi \tag{10.11}$$

As emphasized by some psychological theories of counterfactual emotions (see, e.g., [ZBvdPdV96, ZvDMvdP00]), the positive counterpart of regret is rejoicing: while regret has a *negative valence* (i.e. it is associated with the frustration of an agent's desire), rejoicing has a *positive valence* (i.e. it is associated with the satisfaction of an agent's desire). According to these theories, a person experiences regret when believing that the foregone outcome would have been better if she did a different action, whilst she rejoices when believing that the foregone outcome would have been worse if she did a different action. More precisely, an agent i rejoices for χ if and only if, χ is desirable for i and, i knows that it could have prevented χ to be true now by doing a different action:

$$\mathtt{REJOICE}_i\chi \stackrel{\text{def}}{=} \mathtt{DES}_i\chi \wedge K_i\mathtt{CHP}_i\chi.$$

Example 23 *Consider again the game "Rock-paper-scissors" represented by the STIT-model in Fig. 10.3. Suppose world* w_2 *is the actual world in which player* 1 *plays rock and player* 2 *plays scissors. In this world player* 1 *is the winner* ($1Win$) *and it knows that (by playing paper or scissors) it could have prevented 1Win to be true (i.e.* $K_1\mathtt{CHP}_1 1Win$ *is true at* w_2*). Since* $\mathtt{DES}_1 1Win$ *holds at* w_2*, it follows that at* w_2 *player* 1 *rejoices for having won the game, that is,* $\mathtt{REJOICE}_1 1Win$ *is true at* w_2.

The following validity highlights that rejoicing implies desire satisfaction:

$$\models_{K\mathsf{STIT}} \mathtt{REJOICE}_i\chi \rightarrow (K_i\chi \wedge \mathtt{DES}_i\chi) \tag{10.12}$$

More precisely, if agent i rejoices for χ then, i knows that χ and χ is something desirable for i (in this sense i feels satisfied for having achieved χ). Like regret, rejoicing satisfies the properties of positive and negative introspection:

$$\models_{KSTIT} \texttt{REJOICE}_i\chi \leftrightarrow K_i\texttt{REJOICE}_i\chi \tag{10.13}$$

$$\models_{KSTIT} \neg\texttt{REJOICE}_i\chi \leftrightarrow K_i\neg\texttt{REJOICE}_i\chi \tag{10.14}$$

That is, agent i rejoices (resp. does not rejoice) for χ if and only if it knows this.

10.3.2 Disappointment and elation

According to some authors [LS87, DZ02, ZvDvdP+98], disappointment too is part of the family of counterfactual emotions: like regret, disappointment originates from the comparison between the actual outcome and a counterfactual outcome that might have occurred. However, there is an important difference between regret and disappointment. If an agent feels regret he considers himself to be responsible for the actual outcome, whereas if he feels disappointed he considers external events and other agents' actions to be responsible for the actual outcome.

Thus, we can say that an agent i feels disappointed for χ if and only if $\neg\chi$ is desirable for i and i knows that the others could have prevented χ to be true now. Formally:

$$\texttt{DISAPPOINTMENT}_i\chi \stackrel{\text{def}}{=} \texttt{DES}_i\neg\chi \wedge K_i\texttt{CHP}_{AGT\setminus\{i\}}\chi.$$

Example 24 *In the "Rock-paper-scissors" game represented in Fig. 10.3, regret is always joined with disappointment. For instance, at world w_1 player 1 not only regrets for having lost the game (i.e. $\texttt{REGRET}_1\neg 1Win$), but also he feels disappointed for this (i.e. $\texttt{DISAPPOINTMENT}_1\neg 1Win$). In fact, at w_1, 1 knows that (by playing scissors) the others (i.e. player 2) could have prevented $\neg 1Win$ to be true (i.e. $K_1\texttt{CHP}_{AGT\setminus\{1\}}\neg 1Win$ is true at w_1).*

Like regret and rejoicing, disappointment satisfies the properties of positive and negative introspection:

$$\models_{KSTIT} \texttt{DISAPPOINTMENT}_i\chi \leftrightarrow K_i\texttt{DISAPPOINTMENT}_i\chi \tag{10.15}$$

$$\models_{KSTIT} \neg\texttt{DISAPPOINTMENT}_i\chi \leftrightarrow K_i\neg\texttt{DISAPPOINTMENT}_i\chi \tag{10.16}$$

Moreover, like regret, disappointment implies desire frustration:

$$\models_{KSTIT} \texttt{DISAPPOINTMENT}_i\chi \rightarrow (K_i\chi \wedge \texttt{DES}_i\neg\chi) \tag{10.17}$$

It is worth noting that regret and disappointment do not necessarily occur in parallel, i.e. the formulas $\texttt{REGRET}_i\chi \wedge \neg\texttt{DISAPPOINTMENT}_i\chi$ and $\neg\texttt{REGRET}_i\chi \wedge \texttt{DISAPPOINTMENT}_i\chi$ are satisfiable. The following examples illustrate the situation in which an agent feels disappointed without feeling regret.

Example 25 *Two agents* $AGT = \{1, 2\}$ *have made an appointment to dine together at a restaurant. When the time of the appointment comes near, each of the two agents can either go to the restaurant in order to meet the other or decide to stay home. The two agents will have dinner together only if each of them decides to go to restaurant to meet the other. The scenario is represented in the* STIT *model in Fig. 10.4. It is supposed that having dinner with agent* 2 *is something good for agent* 1 *and agent* 1 *desires to have dinner with agent* 2*:* $\texttt{GOOD}_1$*dinnerTogether and* $\texttt{DES}_1$*dinnerTogether are true at worlds* w_1*-*w_4*. Suppose world* w_1 *is the actual world in which* 1 *goes to the restaurant, while* 2 *does not go and breaks his appointment with* 1*. In this world* 1 *does not have dinner with* 2 *(*$\neg$*dinnerTogether), and* 1 *knows that (by going to the restaurant) the others (i.e. agent* 2*) could have prevented* $\neg$*dinnerTogether to be true (i.e.* $K_1\texttt{CHP}_{AGT\setminus\{1\}}\neg$*dinnerTogether is true at* w_1*). It follows that at* w_1 *agent* 1 *feels disappointed for not having dinner with* 2*, that is,* $\texttt{DISAPPOINTMENT}_1\neg$*dinnerTogether is true at* w_1*. Note that at* w_1 *agent* 1 *does not feel regret for not having dinner with agent* 2 *(i.e.* $\texttt{REGRET}_1\neg$*dinnerTogether is false at* w_1*). In fact, at* w_1*,* 1 *knows that* $\neg$*dinnerTogether only depends on what* 2 *has decided to do. Therefore, at* w_1*,* 1 *does not think that he could have prevented* $\neg$*dinnerTogether to be true*
(i.e. $\neg K_1\texttt{CHP}_1\neg$*dinnerTogether is true at* w_1*).*

We conclude with a formalization of the positive counterpart of disappointement, that is commonly called *elation* [ZBvdPdV96, ZvDMvdP00]. We say that agent i elates for χ if and only if, χ is desirable for i and i knows that the others could have prevented χ to be true now:

$$\texttt{ELATION}_i\chi \stackrel{\text{def}}{=} \texttt{DES}_i\chi \wedge K_i\texttt{CHP}_{AGT\setminus\{i\}}\chi.$$

Like regret, rejoicing and disappointment, elation satisfies the properties of positive and negative introspection:

$$\models_{K\texttt{STIT}} \texttt{ELATION}_i\chi \leftrightarrow K_i\texttt{ELATION}_i\chi \tag{10.18}$$

$$\models_{K\texttt{STIT}} \neg\texttt{ELATION}_i\chi \leftrightarrow K_i\neg\texttt{ELATION}_i\chi \tag{10.19}$$

Moreover, like rejoicing, elation implies desire satisfaction:

$$\models_{K\texttt{STIT}} \texttt{ELATION}_i\chi \rightarrow (K_i\chi \wedge \texttt{DES}_i\chi) \tag{10.20}$$

Note also that elation and rejoicing do not necessarily occur in parallel, i.e. the formulas $\texttt{REJOICE}_i\chi \wedge \neg\texttt{ELATION}_i\chi$ and $\neg\texttt{REJOICE}_i\chi \wedge \texttt{ELATION}_i\chi$ are satisfiable. In fact, an agent might consider the others to be responsible for the actual good situation, without considering himself to be responsible for the actual good situation.

Notice that constructions $\texttt{REGRET}_i\chi$ and $\texttt{REJOICE}_i\chi$ requires to have group STIT. This justifies the study of the syntactic restriction of the decidable fragment of group STIT introduced in this paper. On the contrary, the individual STIT is sufficient to express constructions $\texttt{DISAPPOINTMENT}_i\chi$ and $\texttt{ELATION}_i\chi$. Let us recall that the fusion of two decidable modal logics is decidable [GKWZ03]. As the satisfiability problem of the individual STIT is decidable [BHT08], the fusion of the individual STIT and the epistemic logic is also decidable. So we do not have to introduce syntactic restrictions in order to automatically reason about $\texttt{DISAPPOINTMENT}_i\chi$ and $\texttt{ELATION}_i\chi$. In particular we can reason about *nested emotions* like "agent i feels disappointed because agent j elates for χ" ($\texttt{DISAPPOINTMENT}_i\texttt{ELATION}_j\chi$).

10.3.3 Discussion

Let us discuss some aspects we did not consider in the previous formalization of counterfactual emotions.

According to [Cas05], disappointment entails invalidation of an agent's positive expectation. That is, an agent feels disappointed for χ, only if $\neg\chi$ is desirable for the agent and the agent believes that χ, and in the previous state he believed $\neg\chi$ to be true in the next state. In other words, an agent feels disappointed for χ because he would like χ to be false now and he just learnt that χ is true and, before learning that χ is true, he believed $\neg\chi$ to be true in the next state. In the formalization of disappointment proposed in Section 10.3.2, this relationship between disappointment and expectations was not considered. We included in the definition of disappointment only the agent's mental states at the moment in which the emotion arises.

Another aspect we did not consider in our formalization of counterfactual emotions is the distinction between regret due to a *choice to act* (i.e. action) and regret due to a *choice not to act* (i.e. inaction). A classical example which clarifies this distinction is the one given by [KT82] in which an agent i owned shares in company A, and he considered switching to stock in company B but he decided against it. He now finds out that he would have been better off if he had switched to the stock of company B (regret due to inaction). Another agent j owned shares in company B, and he switched to stock in company A. He now finds out that he would have been better off if he had kept his stock in Company B (regret due to action). The logic STIT is not sufficiently expressive to make this distinction between regret due to action and regret due to inaction. Indeed, in STIT logic it is supposed that at a given state w every agent has made a choice. Moreover, STIT allows to reason about the effects of the agents' choices at a given state. Nevertheless, STIT does not allow to distinguish the situation in which, at a given state, an agent has made the choice to act from the situation in which the agent

has made the choice not to act.

10.4 Related works

As emphasized in the introduction emotion is a very active field in AI. Several computational architectures of affective agents have been proposed in the last few years (see, e.g., [RB92, Ell92, dR99]). The cognitive architecture EMA (Emotion and Adaption) [GM04] is one of the best example of research in this area. EMA defines a domain independent taxonomy of appraisal variables stressing the many different relations between emotions and cognition, by enabling a wide range of internal appraisal and coping processes used for reinterpretation, shift of motivations, goal reconsideration etc. EMA also deals with complex social emotions based on attributions of responsibility such as guilt and shame.

There are also several researchers who have developed formal languages which allow to reason about emotions and to model affective agents. We discuss here some of the most important formal approaches to emotions and compare them with our approach.

Meyer et al.'s logic of emotions One of the most prominent formal analysis of emotions is the one proposed by Meyer et al. [Mey06, SDM07, TMC09]. In order to formalize emotions, Meyer et al. exploit the logical framework KARO [MvdHvL99]: a framework based on a blend of dynamic logic with epistemic logic, enriched with modal operators for motivational attitudes such as desires and goals.

In Meyer et al.'s approach each instance of emotion is represented with a special predicate, or fluent, in the jargon of reasoning about action and change, to indicate that these predicates change over time. For every fluent a set of effects of the corresponding emotions on the agent's planning strategies are specified, as well as the preconditions for triggering the emotion. The latter correspond to generation rules for emotions. For instance, in [Mey06] generation rules for four basic emotions are given: joy, sadness, anger and fear, depending on the agent's plans. More recently [TMC09], generation rules for social emotions such as guilt and shame have been proposed.

Contrarily to Meyer et al.'s approach, in our logic there are no specific formal constructs, like special predicates or fluents, which are used to denote that a certain emotion arises at a certain time. We just *define* the appraisal pattern of a given emotion in terms of some cognitive constituents such as desire and knowledge. For instance, according to our definition of regret, an agent regrets for χ if and only if, he *desires* $\neg\chi$ and, *i knows* that it could have prevented χ to be true now. In other words, following the so-called appraisal theories in psychology, in our approach an

emotion is reduced to its appraisal variables which can be defined through the basic concepts of a BDI logic (e.g. knowledge, belief, desires, intentions).

It has to be noted that, although Meyer et al. provide a very detailed formal analysis of emotions, they do not take into account counterfactual emotions. This is also due to some intrinsic limitations of the KARO framework in expressing counterfactual reasoning and statements of the form "agent i *could have prevented* χ to be true" which are fundamental constituents of this kind of emotions. Indeed, standard dynamic logic on the top of which KARO is built, is not suited to express such statements. In contrast to that, our STIT-based approach overcomes this limitation.

Note also that while Meyer et al. do not prove completeness and do not study complexity of their logic of emotions, these are central issues in our work. As emphasized in the introduction of the article, our aim is to develop a logic which is sufficiently expressive to capture the fundamental constituents of counterfactual emotions and, at the same time, with good mathematical properties in terms of decidability and complexity.

Other logical approaches to emotions Adam et al. [AHL09] have recently exploited a BDI logic in order to provide a logical formalization of the emotion types defined in Ortony, Clore and Collins's model (OCC model) [OCC88] Similar to our approach, in Adam et al.'s approach emotion types are defined in terms of some primitive concepts (and corresponding modalities) such as the concepts of belief, desire, and action which allow to capture the different appraisal variables of emotions proposed in the OCC model such as the desirability of an event, probability of an event, and degree of responsibility of the author of an action. However, Adam et al. do not consider counterfactual emotions. In fact, the logic proposed by Adam et al. is not sufficiently expressive to capture counterfactual thinking about agents' choices and actions on which emotions like regret, rejoicing, disappointment and elation are based. Moreover, this is due to some limitations of the OCC typology which does not contain definitions of emotions based on counterfactual thinking such as regret and rejoicing.

In [ENYI00] a formal approach to emotions based on fuzzy logic is proposed. The main contribution of this work is a quantification of emotional intensity based on appraisal variables like desirability of an event and its likelihood. For example, following [OCC88], in FLAME the variables affecting the intensity of hope with respect to the occurrence of a certain event are the degree to which the expected event is desirable, and the likelihood of the event. However, in FLAME only basic emotions like joy, sadness, fear and hope are considered and there is no formal analysis of counterfactual emotions as the ones analyzed in our work.

10.5 Conclusion

A logical framework which allows to formalize and to reason about counterfactual emotions has been proposed in this paper. This framework is based on a decidable and finitely axiomatizable fragment of STIT logic called *df*STIT. We have shown that an epistemic extension of *df*STIT called *df K*STIT is sufficiently expressive to capture the fundamental constituents of counterfactual emotions and, at the same time, it has good mathematical properties in terms of complexity and axiomatizability. We have proved that the satisfiability problem of df*K*STIT is NP-complete if $card(AGT) = 1$ and PSPACE-complete if $card(AGT) \geq 2$. This first result is fundamental in order to claim that we can write down algorithms in df*K*STIT to reason about counterfactual emotions such as regret, rejoicing, disappointment and elation. Moreover, we have provided a complete axiomatization of df*K*STIT . This second result is also important because it shows that we can perform syntactic reasoning in df*K*STIT about counterfactual emotions. We hope that the analysis developed in this paper will be useful for improving understanding of affective phenomena and will offer an interesting perspective on computational modeling of affective agents and systems.

Directions for our future research are manifold. The reader may remark there is a gap between the complexity of the satisfiability problem of a formula in *df*STIT (NP-complete) and the complexity of the satisfiability problem of a formula in *df K*STIT (PSPACE-complete). Of course, the complexity for *df K*STIT can not be improved because the satisfiability problem of $S5_n$ is already PSPACE-complete. An interesting open question is to identify a more expressive fragment of STIT such that its satisfiability problem is PSPACE-complete and such that adding knowledge will not increase the complexity of its satisfiability problem. An analysis of intensity of counterfactual emotions was beyond the objectives of the present work. However, we intend to investigate this issue in the future in order to complement our qualitative analysis of affective phenomena with a quantitative analysis. Moreover, we have focused in this paper on the logical characterization of four counterfactual emotions: regret, rejoicing, disappointment and elation. We intend to extend our analysis in the future by studying the counterfactual dimension of "moral" emotions such as guilt and shame. Indeed, as several psychologists have shown (see, e.g., [Laz91]), guilt involves the conviction of having injured someone or of having violated some norm or imperative, and the belief that this *could have been avoided*.

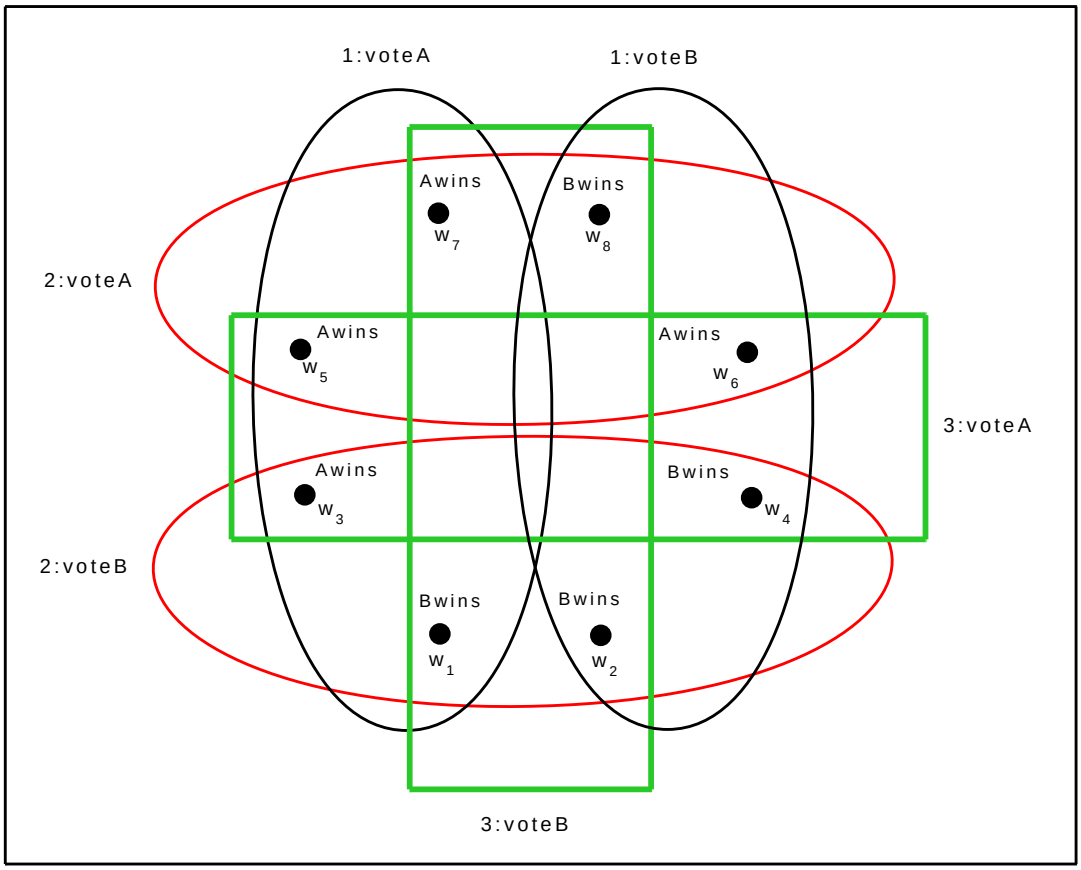


Figure 10.2: Vertical circles represent the actions that voter 1 can choose, horizontal circles represent the actions that voter 2 can choose, and rectangles represent the actions that voter 3 can choose. For example, w_1 is the world in which candidate B wins the election and that results from agent 1 voting for candidate A, and agents 2 and 3 voting for candidate B.

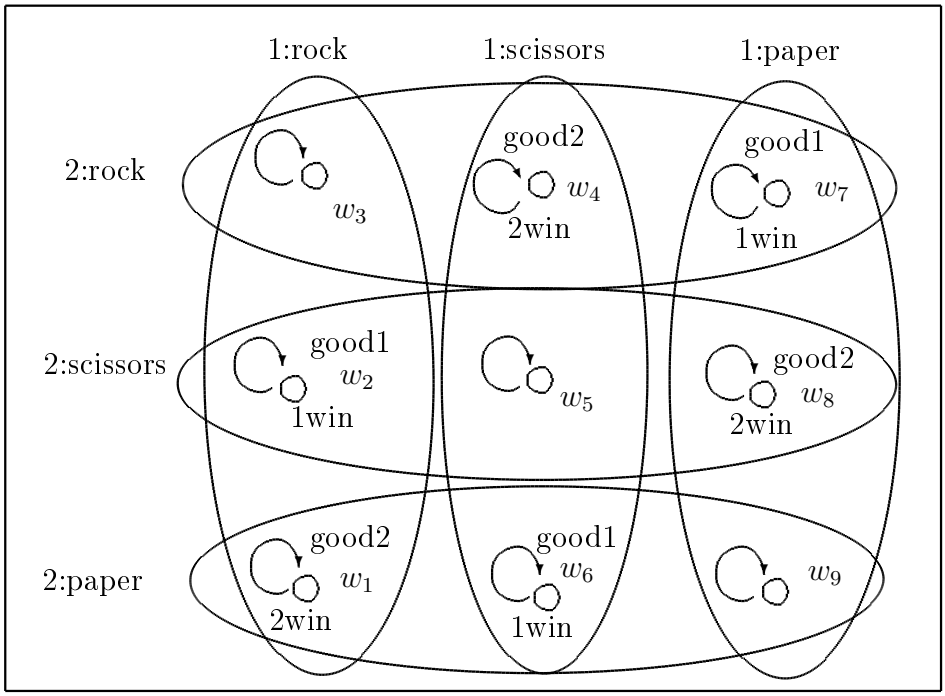


Figure 10.3: Vertical circles represent the actions that player 1 can choose, whereas horizontal circles represent the actions that player 2 can choose. For the sake of simplicity, we suppose that players 1 and 2 do not have uncertainty: everywhere in the model players 1 and 2 only consider possible the world in which they are (reflexive arrows represent indistinguishability relations for the two players).

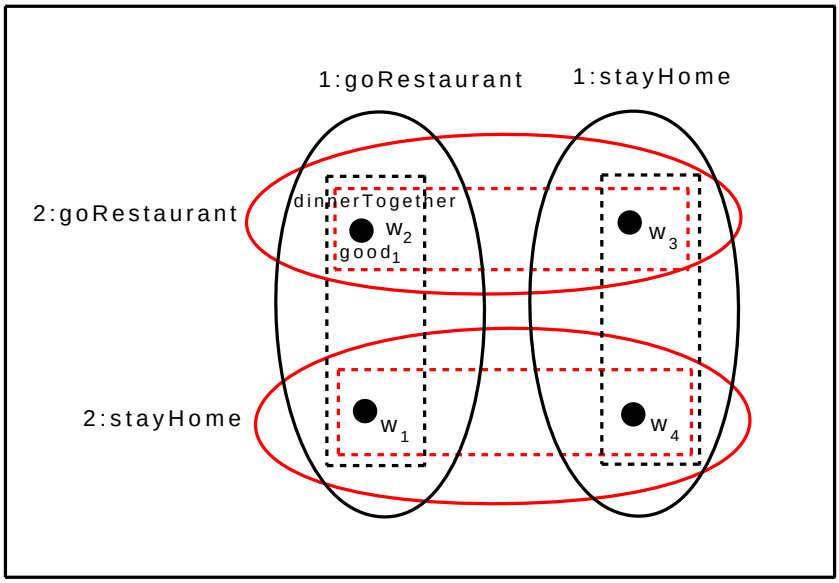


Figure 10.4: Again, vertical circles represent the actions that agent 1 can choose, whereas horizontal circles represent the actions that agent 2 can choose. In this example, we suppose that agents 1 and 2 only have uncertainty about the choice of the other (vertical dotted rectangles represent indistinguishability relations for agent 1, whereas horizontal dotted rectangles represent indistinguishability relations for agent 2).

Conclusion and perspectives

Conclusion

In this thesis, we have investigated knowledge reasoning with two types of constraints: constraints on the geometry of the worlds (Part I) and constraints about the structure of a game or of choices of actions by agents (part III). Part II has been devoted to the study of the structure of a game or of choices of actions by agents via the logic STIT. In other words, we have studied the satisfiability problem of a modal formula in each of those different systems. More precisely:

- In part I, we have studied how to reason about what an agent sees and what an agent knows about what the other agents see when the world is a line (Chapter 4) and when the world is a plane Chapter 5). We have seen that the satisfiability problem in Lineland is PSPACE-complete and that the satisfiability problem in Flatland is decidable but the complexity may be quite high (the lower-bound is still unknown). Nevertheless we have exhibited a weak version of Flatland whose satisfiability problem is PSPACE-complete.

- In part II, we have studied the logic STIT which is a logic which deals with what agents *do*, contrary to logics like CL [Pau02] and ATL [AHK99] which only deals with what agents *can do*. Broesen et al. [BHT05] proved that CL can be embedded into a version of logic STIT. In the same way, they [BHT06b] proved that ATL can be embedded into another version of logic STIT.

  The complexity of the satisfiability problem of STIT does not arise from the treatment of time but from the expressivity of the choice operator of STIT. In Chapter 7, we have focused on the satisfiability problem and on the axiomatization of group STIT. In [BHT08], the authors proved that the satisfiability problem of STIT logic, without time operator and only with individual operators is NEXPTIME-complete. We have proved that the satisfiability problem of STIT logic, without time operator but with general group operators is in general undecidable. Then we have been interested in the quest of searching decidable fragments of STIT, more expressive than the individual STIT and less expressive than group STIT. We have exhibited a fragment of STIT where operators $[J]$ are such that the coalitions J are in a given lattice and

we proved its satisfiability problem is NEXPTIME-complete. Exploring this fragment have enabled us to give results for satisfiability in the individual STIT with "next" operator settings.

In Chapter 8, we have focused on a weak fragment of STIT with a strong syntactic restriction of the STIT-formulas. We proved the satisfiability problem of a formula in this fragment to be NP-complete.

- In Part III, we have given two applications of STIT into two different fields: game theory and formalization of emotions. In Chapter 9, we focused on a logic inspired by STIT and have tried to formalize the notion of Nash Equilibrium and the algorithm of Iterated Deletion of Strictly Dominated Strategies. Since we need more expressivity, we have used a new version of the logic STIT by adding explicit actions names in the language. We proved the satisfiability problem of a formula to be NP-complete in the case of complete information and PSPACE-complete in the case of incomplete information.

  In Chapter 10, we have extended the fragment of STIT studied in Chapter 8 with a knowledge operator and we have proved that we can capture the notions of regret, rejoice, disappointment and elation. We have also proved that the satisfiability problem is PSPACE-complete.

Analogy between Part I and Part II

In this thesis, we have discovered a curious analogy between the study of logics of perception and knowledge in Part I and STIT theory in Part II:

- In both formalisms, when the dimension is low (Lineland in Part I or STIT with one or two agents in Part II), the satisfiability problem in "easy": respectively PSPACE-complete for Lineland and NEXPTIME-complete for STIT;
- In both formalisms, when the dimension is high (Flatland in Part I or STIT with three agents or more in Part II), the complexity of the satisfiability problem increases: respectively in EXPSPACE (maybe less?) for Flatland and undecidable for STIT;
- Moreover when the dimension is high, we were able to weaken the semantics of both formalisms so that the complexity of the satisfiability problem becomes equal to the complexity of the satisfiability problem when the dimension is low, in each case:
  - In Flatland, we have weakened the semantics (see Definition 36) so that the truth of a formula only depends of the truth of literals $a \rhd b$. The

satisfiability problem in this weak Flatland version is PSPACE-complete as in Lineland;

– Concerning STIT, we have weakened the additivity property $R_{J_1 \cup J_2} = R_{J_1} \cap R_{J_2}$ (see Definitions 43, 45) into the super-additivity property (see Definitions 50, 61). In those weaker configurations, the satisfiability problem is NEXPTIME-complete as for STIT with two agents.

Perspectives

Theoretical perspectives

Dynamism

We aim at extending the results of this thesis by adding public announcements to the language. For the moment, the complexity of Lineland, Flatland with public announcements is unknown. We may also add the framework of dynamic epistemic logic to Lineland and Flatland in order to model private announcements.

Concerning the logic STIT extended with the knowledge operator of the Chapter 10, we aim at extending the language with dynamic operators in order to capture emotion changes.

Variant of Lineland/Flatland

In the future, we will compare different semantics of Lineland/Flatland (myopic agents, agents with multiple angles of view, etc.) and we shall study the satisfiability problem and axiomatization of each of these variants.

In addition we aim at combining the epistemic modal constructions $K_a\varphi$ with modal constructions $\Diamond_a\varphi$ of [MP92] meaning "if agent a can make an effort (widen her vision cone) such that φ is true".

Combining Lineland/Flatland and STIT

In Part I, we have studied the combination of constructions $a \rhd b$ (agent a sees agent b) and the epistemic construction $K_a\varphi$ (agent a knows that φ is true). In Part II, we have studied the logic of agency STIT ("see-to-it-that" logic) where modal constructions are of the form $[\{a\}]\varphi$ and stands for "agent a sees to it that φ is true. The logic STIT also provides constructions for group of agents: $[J]\varphi$ stands for "the coalition J sees to it that φ is true". In Part III, we have combined the construction $[J]\varphi$ of logic STIT with epistemic modal logic in order to model notion of rationality and counterfactual emotions.

Another perspective consists in modeling all aspects of "Seeing, knowing and doing" altogether in one framework, that is to say to study a logic containing constructions of the form $a \rhd b$, $K_a\varphi$ and $[J]\varphi$. We may consider a traditional fusion of modal logics. However, we will be rather interested in considering axioms of interaction between the three constructions:

- $a \rhd b \rightarrow ([\{b\}]\varphi \rightarrow K_a[\{b\}]\varphi)$: if an agent a sees an agent b then agent a knows what agent b does;
- $\bigwedge_{b \in J} a \rhd b \rightarrow ([J]\varphi \rightarrow K_a[J]\varphi)$: if an agent a sees all agents belonging to a coalition J then agent a knows what the coalition J is performing.

In other terms, positions of agents in Lineland/Flatland may provide constraints over what agents know about the actions that are performed.

Implementation

A model-checker for Lineland has been already implemented (see Section 4.6). Nevertheless no model-checker have been implemented for Flatland yet. The main problem concerns the link between Flatland and the real number theory:

- Concerning the initial version of Flatland (Definition 30), we only have a translation from our epistemic language into the language of real number theory which preserves the satisfiability (Subsection 5.5.2). Unfortunately, this will not lead to a suitable implementation.
- Concerning the "stupid" version of Flatland (see Definition 36), we have an optimal PSPACE-algorithm (see Figure 5.7). This algorithm requires to check if formulas of the first order theory of real numbers with only existential quantifiers are satisfiable. For the moment, we do not know how to implement this;
- Concerning STIT, the high complexities (NEXPTIME-complete and undecidable) discourage us to implement a satisfiability prover for this logic. However, inspite of this high-complexity, tableau methods could be good candidates for implementation as it is done in [Wan06] for another version of STIT. Furthermore, it would be possible to implement a satisfiability checker for the weak STIT fragment of the Chapter 8. A solution to do this may consist in translating a formula into a formula of the classical propositional logic for which we have efficient satisfiability prover [LBP].

A project

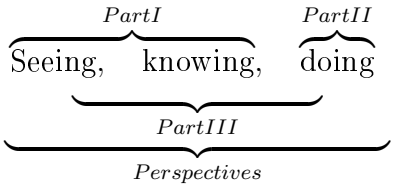
I would like to conclude this thesis with the road-map of a project: I want this thesis to be useful to create a pedagogical video game devoted to teach epistemic modal logic, reasoning about emotions in modal logic, epistemic planning, public announcements, private announcements, etc. This future piece of software may represent artificial agents in Flatland. The user may interact with the agents:

- Asking questions to agents:
  - Do you regret that φ?
  - Do you know that agent a sees agent b?
- Make public announcements;

Of course, this project shall mix the Lineland/Flatland approach and logic STIT. We shall also consider concrete actions for STIT such as:

- agent a moves one step;
- agent a turns;
- agent a widen her vision cone.

When an agent speaks, she may only make private announcements to agents which are near from her.



Bibliography

[AB95] R. Aumann and A. Brandenburger. Epistemic conditions for Nash equilibrium. *Econometrica: Journal of the Econometric Society*, 63(5):1161–1180, 1995.

[Abb84] Edwin Abbott Abbott. *Flatland*. Basil Blackwell Oxford, 1884.

[Ada09] Patrick Adam, Carole & Ye. Reasoning about emotions in an engaging interactive toy. In *Proceedings of EAW @ AAMAS'09*, 2009.

[AHK99] R. Alur, T. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Compositionality: The Significant Difference*, pages 23–60, 1999.

[AHK02] R. Alur, T. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49:672–713, 2002.

[AHL09] C. Adam, A. Herzig, and D. Longin. A logical formalization of the OCC theory of emotions. *Synthese*, 168(2):201–248, 2009.

[AHU83] A.V. Aho, J.E. Hopcroft, and J. Ullman. *Data structures and algorithms*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1983.

[APHvB07] Marco Aiello, Ian Pratt-Hartmann, and Johan van Benthem, editors. *Handbook of Spatial Logics*. Springer, 2007.

[Aum99] R. Aumann. Interactive epistemology I: Knowledge. *International Journal of Game Theory*, 28(3):263–300, 1999.

[ÅvdHW07] T. Ågotnes, W. van der Hoek, and M. Wooldridge. Quantified coalition logic. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI'07)*, pages 1181–1186. AAAI Press, 2007.

[BB99] P. Battigalli and G. Bonanno. Recent results on belief, knowledge and the epistemic foundations of game theory. *Research in Economics*, 53:149–225, 1999.

[BBF+01] B. Bérard, M. Bidoit, A. Finkel, F. Laroussinie, A. Petit, L. Petrucci, P. Schnoebelen, B. Berard, M. Bidoit, and A. Finkel. *Systems and software verification.* Springer, 2001.

[BDRV02] P. Blackburn, M. De Rijke, and Y. Venema. *Modal logic.* Cambridge Univ Pr, 2002.

[BE93] Jon Barwise and John Etchemendy. *Tarski's World: Version 4.0 for Macintosh (Center for the Study of Language and Information - Lecture Notes).* Center for the Study of Language and Information/SRI, 1993.

[BG02] P. Balbiani and V. Goranko. Modal logics for parallelism, orthogonality, and affine geometries. *Journal of Applied Non-Classical Logics*, 12(3-4):365–398, 2002.

[BGH+08] Philippe Balbiani, Olivier Gasquet, Andreas Herzig, François Schwarzentruber, and Nicolas Troquard. Coalition games over Kripke semantics. In Cédric Dégremont, Laurent Keiff, and Helge Rückert, editors, *Festschrift in Honour of Shahid Rahman*, pages 1–12. College Publications, 2008.

[BGS10] Philippe Balbiani, Olivier Gasquet, and François Schwarzentruber. Représentation des connaissances dans flatland. 2010.

[BHT05] Jan Broersen, Andreas Herzig, and Nicolas Troquard. From Coalition Logic to STIT . Workshop at IJCAI 2005, August 2005.

[BHT06a] J. Broersen, A. Herzig, and N. Troquard. Embedding alternating-time temporal logic in strategic STIT logic of agency. *Journal of Logic and Computation*, 16(5):559–578, 2006.

[BHT06b] J. Broersen, A. Herzig, and N. Troquard. A stit-extension of atl. *Logics in Artificial Intelligence*, pages 69–81, 2006.

[BHT07a] J. Broersen, A. Herzig, and N. Troquard. A Reading Companion to the ESSLLI Course "Logics for Agency and Multi-Agent Systems". 2007.

[BHT07b] Jan Broersen, Andreas Herzig, and Nicolas Troquard. Normal Coalition Logic and its conformant extension. In Dov Samet, editor, *Theoretical Aspects of Rationality and Knowledge (TARK), Brussels, 25/06/2007-27/06/2007*, pages 91–101. Presses universitaires de Louvain, 2007.

[BHT08] Philippe Balbiani, Andreas Herzig, and Nicolas Troquard. Alternative axiomatics and complexity of deliberative STIT theories. *Journal of Philosophical Logic*, 2008. to appear.

[BM04] A. Baltag and L. S. Moss. Logics for epistemic programs. *Synthese*, 139(2):165–224, 2004.

[Bon08] G. Bonanno. A syntactic approach to rationality in games with ordinal payoffs. In *Proceedings of the 8th Conference on Logic and the Foundations of Game and Decision Theory (LOFT 2008)*, Texts in Logic and Games Series, pages 59–86. Amsterdam University Press, 2008.

[BPX01] N. Belnap, M. Perloff, and M. Xu. *Facing the future: agents and choices in our indeterminist world*. Oxford University Press, 2001.

[Bra92] A. Brandenburger. Knowledge and equilibrium in games. *Journal of Economic Perspectives*, 6:83–101, 1992.

[Can88] J. Canny. Some algebraic and geometric computations in PSPACE. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 460–469. ACM, 1988.

[Cas05] C. Castelfranchi. Mind as an anticipatory device: For a theory of expectations. In *Proc. of the First International Symposium on Brain, Vision, and Artificial Intelligence (BVAI 2005)*, pages 258–276. Springer-Verlag, 2005.

[CE82] E. Clarke and E. Emerson. Design and synthesis of synchronization skeletons using branching time temporal logic. *Logics of Programs*, pages 52–71, 1982.

[CH01] A. G. Cohn and S. M. Hazarika. Qualitative spatial representation and reasoning: an overview. *Fundamenta Informaticae*, 46(1-2):1–29, 2001.

[CH04] H. Chockler and J. Y. Halpern. Responsibility and blame: A structural-model approach. *Journal of Artificial Intelligence Research*, 22:93–115, 2004.

[Che80] B.F. Chellas. *Modal logic: an introduction*. Cambridge Univ Pr, 1980.

[Che92] Brian F. Chellas. Time and modality in the logic of agency. *Studia Logica*, 51(3/4):485–518, 1992.

[Chu36] A. Church. A note on the Entscheidungsproblem. *Journal of Symbolic Logic*, 1(1):40–41, 1936.

[CKS81] A.K. Chandra, D.C. Kozen, and L.J. Stockmeyer. Alternation. *Journal of the ACM (JACM)*, 28(1):114–133, 1981.

[CL90] P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42(2–3):213–261, 1990.

[CLR92] TH Cormen, CE Leiserson, and RL Rivest. Introduction to algorithms, 1992.

[Coo71] S.A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, page 158. ACM, 1971.

[Coo06] S. Cook. The P versus NP problem. *The millennium prize problems*, pages 87–106, 2006.

[dB04] B. de Bruin. *Explaining games: on the logic of game theoretic explanations*. PhD thesis, University of Amsterdam, The Netherlands, 2004.

[dR99] F. de Rosis, F. & Grasso. Affective natural language generation. In *International Workshop on Affective Interactions (IWAI)*, pages 204–218, Siena, Italie, 1999. Springer.

[DZ02] W. W. Dijk and M. Zeelenberg. Investigating the appraisal patterns of regret and disappointment. *Motivation and Emotion*, 26(4):321–331, 2002.

[EA93] M.B. Eisenberg and H. Abelson. *Programming in SCHEME*. Course Technology Press Boston, MA, United States, 1993.

[EH85] E.A. Emerson and J.Y. Halpern. Decision procedures and expressiveness in the temporal logic of branching time* 1. *Journal of computer and system sciences*, 30(1):1–24, 1985.

[EH86] E.A. Emerson and J.Y. Halpern. "Sometimes" and "not never" revisited: on branching versus linear time temporal logic. *Journal of the ACM (JACM)*, 33(1):151–178, 1986.

[Ell92] C. Elliot. *The Affective reasoner: A process model for emotions in a multi-agent system*. PhD thesis, Northwestern University, Institute for Learning Sciences, 1992.

[ELW+98] R. Eckstein, M. Loy, D. Wood, et al. *Java swing*. O'Reilly Sebastopol, CA, 1998.

[Eng83] E. Engeler. Foundations of Mathematics: Questions of Analysis, Geometry & Algorithmics. Engl. Transl, 1983.

[ENYI00] M. S. El-Nasr, J. Yen, and T. R. Ioerger. FLAME: Fuzzy logic adaptive model of emotions. *Autonomous Agents and Multi-Agent Systems*, 3(3):219–257, 2000.

[FH94] R. Fagin and J. Halpern. Reasoning about knowledge and probability. *Journal of the Association for Computing Machinery*, pages 340–367, 1994.

[FHMV95] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.

[FL79] M.J. Fischer and R.E. Ladner. Propositional dynamic logic of regular programs* 1. *Journal of Computer and System Sciences*, 18(2):194–211, 1979.

[FS03] E. Fehr and K. M. Schmidt. Theories of fairness and reciprocity: Evidence and economic applications. In *Advances in Economics and Econometrics*. Cambridge University Press, 2003.

[GC97] Charles-Michel Marle Gilles Christol, Anne Cot. *Topologie*. 1997.

[GG97] J. Gerbrandy and W. Groeneveld. Reasoning about information change. *Journal of Logic, Language, and Information*, 6:147–196, 1997.

[GH93] O. Gasquet and A. Herzig. Translating non-normal modal logics into normal modal logics. In A.I.J Jones and M. Sergot, editors, *Proceedings International Workshop on Deontic Logic in Computer Science (DEON'94)*, TANO, Oslo, 1993.

[Gin09] H. Gintis. *The Bounds of Reason: Game Theory and the Unification of the Behavioral Sciences*. Princeton University Press, 2009.

[GKWZ03] D. Gabbay, Á. Kurucz, F. Wolter, and M. Zakharyaschev. *Many-Dimensional Modal Logics: Theory and Applications, volume 148 of Studies in Logic*. Elsevier, 2003.

[GM04] J. Gratch and S. Marsella. A Domain-independent Framework for modelling Emotions. *Journal of Cognitive Systems Research*, 5 (4):269–306, 2004.

[GO06] V. Goranko and M. Otto. *Handbook of Modal Logic*, chapter Model Theory of Modal Logic, pages 255–325. Elsevier, 2006.

[Gol82] R. Goldblatt. Axiomatizing the logic of computer programming, volume 130 of Lecture Notes in Computer Science, 1982.

[GS10] Olivier Gasquet and François Schwarzentruber. Knowledge in Lineland (Extended Abstract) (short paper). In *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), Toronto (Canada), 10/05/2010-14/05/2010*, 2010.

[Hal03] J. Y. Halpern. *Reasoning about uncertainty*. MIT Press, Cambridge, 2003.

[Har67] J. C. Harsanyi. Games with incomplete information played by 'bayesian' players. *Management Science*, 14:159–182, 1967.

[HB95] John Horty and Nuel Belnap. The deliberative stit: a study of action, omission, ability and obligation. *Journal of Philosophical Logic*, 24(6):583–644, 1995.

[HC72] G.E. Hughes and MJ Cresswell. *An introduction to modal logic*. Routledge, 1972.

[Hei06] Bernhard Heinemann. Regarding overlaps in 'topologic'. In *Advances in Modal Logic*, pages 259–277, 2006.

[Hin62] J. Hintikka. *Knowledge and belief: an introduction to the logic of the two notions*. Cornell University Press, 1962.

[HL10] A. Herzig and E. Lorini. A Dynamic Logic of Agency I: STIT, Capabilities and Powers. *Journal of Logic, Language and Information*, 19(1):89–121, 2010.

[Hor01a] J. F. Horty. *Agency and Deontic Logic*. Oxford University Press, 2001.

[Hor01b] John F. Horty. *Agency and Deontic Logic*. Oxford University Press, 2001.

[HR07] J. Y. Halpern and L. Rego. Generalized solution concepts in games with possibly unaware players. In *Proceedings of the Eleventh Conference on Theoretical Aspects of Rationality and Knowledge (TARK 2007)*, pages 253–262, 2007.

[HS08] A. Herzig and F. Schwarzentruber. Properties of logics of individual and group agency. In *Proceedings of Advances in Modal Logic 2008*, pages 133–149. College Publ., 2008.

[HT06] A. Herzig and N. Troquard. Knowing how to play: uniform choices in logics of agency. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, page 216. ACM, 2006.

[HV88] J.Y. Halpern and M.Y. Vardi. The complexity of reasoning about knowledge and time: synchronous systems. In *Proc. 20th ACM Symp. on Theory of Computing*, pages 53–65, 1988.

[JÅ07] W. Jamroga and T. Ågotnes. Constructive knowledge: What agents can achieve under incomplete information. *Journal of Applied Non-Classical Logics*, 17(4):423–475, 2007.

[Jag09] M. Jago. Epistemic logic for rule-based agents. *Journal of Logic, Language and Information*, 18(1):131–158, 2009.

[Jon03] G. Jonker. Feasible strategies in alternating-time temporal epistemic logic. *Universiteit Utrecht Master Thesis*, 2003.

[JvdH04] W. Jamroga and W. van der Hoek. Agents that know how to play. *Fundamenta Informaticae*, 63(2):185–219, 2004.

[JYH96] Yoram Moses Joseph Y. Halpern. A guide to completeness and complexity for modal logics of knowledge and belief. 1996.

[Kah95] D. Kahneman. Varieties of counterfactual thinking. In N. J. Roese and J. M. Olson, editors, *What might have been: the social psychology of counterfactual thinking*. Erlbaum, 1995.

[KM86] D. Kahneman and D. T. Miller. Norm theory: comparing reality to its alternatives. *Psychological Review*, 93(2):136–153, 1986.

[Knu73] D.E. Knuth. *The art of computer programming.. vol. 1,. fundamental algorithms*. Addison-Wesley Pub. Co, 1973.

[Kri63] S. Kripke. Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16(1963):83–94, 1963.

[KT82] D. Kahneman and A. Tversky. The psychology of preferences. *Scientific American*, 246:160–173, 1982.

[Lad77] R. E. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM Journal on Computing*, 6(3):467–480, 1977.

[Laz91] R. S. Lazarus. *Emotion and adaptation*. Oxford University Press, New York, 1991.

[LBP] D. Le Berre and A. Parrain. La bibliothèque Sat4j 2.2.

[L.E12] Brouwer L.E.J. Beweis der invarianz des n-dimensionalen gebeits. *Mathematische Annalen*, pages 305–315, 1912.

[Lor10a] E. Lorini. A dynamic logic of agency II: deterministic DLA, coalition logic, and game theory. Technical Report 3, 2010.

[Lor10b] E. Lorini. A logical account of social rationality in strategic games. In *Proceedings of the 9th Conference on Logic and the Foundations of Game and Decision Theory (LOFT 2008)*, 2010. to appear.

[LS82] G. Loomes and R. Sugden. Regret theory: an alternative theory of rational choice under uncertainty. *Economic Journal*, 92(4):805–824, 1982.

[LS87] G. Loomes and R. Sugden. Testing for regret and disappointment in choice under uncertainty. *Economic Journal*, 97:118–129, 1987.

[LS09] Emiliano Lorini and François Schwarzentruber. A logic for reasoning about counterfactual emotions. In C. Boutilier, editor, *International Joint Conference on Artificial Intelligence (IJCAI), Pasadena (USA), 11/07/2009-17/07/2009*, pages 867–872, http://www.aaai.org/Press/press.php, 2009. AAAI Press.

[LW06] C. Lutz and F. Wolter. Modal logics of topological relations. *Arxiv preprint cs/0605064*, 2006.

[LY99] T. Lindholm and F. Yellin. *Java virtual machine specification*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1999.

[Mey06] J.-J. Ch. Meyer. Reasoning about emotional agents. *International Journal of Intelligent Systems*, 21(6):601–619, 2006.

[Min98] G. Mints. A completeness proof for propositional S4 in cantor space. *Logic at work: Essays dedicated to the memory of Helena Rasiowa. Physica-Verlag, Heidelberg*, 1998.

[MP92] L.S. Moss and R. Parikh. Topological reasoning and the logic of knowledge. *Theoretical Aspects of Reasoning about Knowledge (TARK 1992)*, pages 95–105, 1992.

[MT44] JCC McKinsey and A. Tarski. The algebra of topology. *Annals of mathematics*, 45(1):141–191, 1944.

[MvdHvL99] J. J. Ch. Meyer, W. van der Hoek, and B. van Linder. A logical approach to the dynamics of commitments. *Artificial Intelligence*, 113(1-2):1–40, 1999.

[OCC88] A. Ortony, G. L. Clore, and A. Collins. *The cognitive structure of emotions*. Cambridge University Press, 1988.

[ON80] H. Ono and A. Nakamura. On the size of refutation Kripke models for some linear modal and tense logics. *Studia Logica*, 39(4):325–333, 1980.

[OR94] M. J. Osborne and A. Rubinstein. *A course in game theory*. MIT Press, 1994.

[Pap03] C.H. Papadimitriou. *Computational complexity*. John Wiley and Sons Ltd., 2003.

[Pau02] Marc Pauly. A modal logic for coalitional power in games. *Journal of Logic and Computation*, 12(1):149–166, 2002.

[Pla07] Jan Plaza. Logics of public communications. *Synthese*, 158(2):165–179, 2007.

[PMS07] Rohit Parikh, Lawrence S. Moss, and Chris Steinsvold. Topology and epistemic logic. In Aiello et al. [APHvB07], pages 299–341.

[Pnu77] A. Pnueli. The temporal logic of programs. In *18th Annual Symposium on Foundations of Computer Science*, pages 46–57. IEEE, 1977.

[RB92] W. S. Reilly and J. Bates. Building emotional agents. Technical report, CMUCS -92-143, School of Computer science, Canergie Mellon University, 1992.

[RCC92] D.A. Randell, Z. Cui, and A.G. Cohn. A spatial logic based on regions and connection. *KR*, 92:165–176, 1992.

[Ren88] J. Renegar. A faster PSPACE algorithm for deciding the existential theory of the reals. /, pages 291–295, 1988.

[Rey05] Mark Reynolds. An axiomatization of pctl. *Inf. Comput.*, 201(1):72–119, 2005.

[RN99] J. Renz and B. Nebel. On the complexity of qualitative spatial reasoning: A maximal tractable fragment of the region connection calculus. *Artificial Intelligence*, 108(1):69–123, 1999.

[RN07] Jochen Renz and Bernhard Nebel. Qualitative spatial reasoning using constraint calculi. In Aiello et al. [APHvB07], pages 161–215.

[Roe97] N. J. Roese. Counterfactual thinking. *Psychological Bulletin*, 121(1):133–148, 1997.

[Roy08] O. Roy. *Thinking before acting: intentions, logic, rational choice.* PhD thesis, University of Amsterdam, The Netherlands, 2008.

[Sah75] H. Sahlqvist. Completeness and correspondence in the first and second order semantics for modal logics. In *Proceedings 3rd Scandinavian Logic Symposium 1973*, volume number 82 in Studies in Logic, 1975.

[Sai10] Bilal Said. *Réécriture de graphes pour la construction de modèles en logique modale.* Thèse de doctorat, Université de Toulouse, Toulouse, France, janvier 2010.

[Sav70] W.J. Savitch. Relationships between nondeterministic and deterministic tape complexities*. *Journal of computer and system sciences*, 4(2):177–192, 1970.

[SC85] AP Sistla and EM Clarke. The complexity of propositional linear temporal logics. *Journal of the ACM (JACM)*, 32(3):749, 1985.

[Sch04] P.Y. Schobbens. Alternating-time logic with imperfect recall. *Electronic Notes in Theoretical Computer Science*, 85(2):82–93, 2004.

[Sch07] François Schwarzentruber. Décidabilité et complexité de la logique normale des coalitions. Master's thesis, Univ. Paul Sabatier Toulouse III, 2007.

[Sch09] François Schwarzentruber. Knowledge about lights along a line. 2009.

[SDM07] B. R. Steunebrink, M. Dastani, and J.-J. Ch. Meyer. A logic of emotions for intelligent agents. In *Proceedings of AAAI'07*, pages 142–147. AAAI Press, 2007.

[Seg70] K. Segerberg. Modal logics with linear alternative relations. *Theoria*, 36(3):301–322, 1970.

[SM73] LJ Stockmeyer and AR Meyer. Word problems requiring exponential time (Preliminary Report). In *Proceedings of the fifth annual ACM symposium on Theory of computing*, pages 1–9. ACM, 1973.

[Sta06] R. Stalnaker. On logics of knowledge and belief. *Philosophical studies*, 128(1):169–199, 2006.

[Stu] P.C. Sturm. Mémoire sur la résolution des équations numériques. *Collected Works of Charles François Sturm*, pages 345–390.

[Sug85] R. Sugden. Regret, recrimination and rationality. *Theory and Decision*, 19(1):77–99, 1985.

[Tar38] A. Tarski. Der aussagenkalk
"ul und die topologie. *Fundamenta Mathematicae*, 31:103–134, 1938.

[Tar51] Alfred Tarski. A decision method for elementary algebra and geometry. 1951.

[TMC09] P. Turrini, J.-J. Ch. Meyer, and C. Castelfranchi. Coping with shame and sense of guilt: a dynamic logic account. *Journal of Autonomous Agents and Multi-Agent Systems*, 2009. forthcoming.

[Tro07] Nicolas Troquard. *Independent agents in branching time.* PhD thesis, Univ. Paul Sabatier Toulouse III & Univ. degli studi di Trento, 2007.

[TSBH95] J.R.R. Tolkien, B. Sibley, M. Bakewell, and I. Holm. *The lord of the rings.* HarperCollins, 1995.

[Tur37] AM Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 2(1):230, 1937.

[Vak] D. Vakarelov. A modal theory of arrows. Arrow logics I. *Logics in AI*, pages 1–24.

[vB07] J. van Benthem. Rational dynamics and epistemic logic in games. *International Game Theory Review*, 9(1):13–45, 2007.

[vBB07] J. van Benthem and G. Bezhanishvili. Modal logics of space. *Handbook of spatial logics*, pages 217–298, 2007.

[vBL07] J. van Benthem and F. Liu. Dynamic logic of preference upgrade. *Journal of Applied Non-Classical Logics*, 17(2):157–182, 2007.

[vdHJW05] W. van der Hoek, W. Jamroga, and M. Wooldridge. A logic for strategic reasoning. In *Proceedings of Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'05)*, ACM Press, 2005. New York.

[vDvdHK07] H. P. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic.* Kluwer Academic Publishers, 2007.

[Ven98] Yde Venema. Rectangular games. *The Journal of Symbolic Logic*, 63(4), December 1998.

[VOJ05] S. Van Otterloo and G. Jonker. On epistemic temporal strategic logic. *Electronic Notes in Theoretical Computer Science*, 126:77–92, 2005.

[Wan06] H. Wansing. Tableaux for multi-agent deliberative-STIT logic. *Advances in Modal Logic*, 6:503–520, 2006.

[Wei93] Volker Weispfenning. Quantifier elimination for real algebra - the quadratic case and beyond. *AAECC*, 8:85–101, 1993.

[Wil02] T. Williamson. *Knowledge and its Limits*. Oxford University Press, USA, 2002.

[WLWW06] Dirk Walther, Carsten Lutz, Frank Wolter, and Michael Wooldridge. ATL satisfiability is indeed exptime-complete. *Journal of Logic and Computation*, 16:765–787, 2006.

[Xu98] Ming Xu. Axioms for deliberative STIT. *Journal of Philosophical Logic*, 27:505–552, 1998.

[ZBvdPdV96] M. Zeelenberg, J. Beattie, J. van der Pligt, and N. K. de Vries. Consequences of regret aversion: effects of expected feedback on risky decision making. *Organizational behavior and human decision processes*, 65(2):148–158, 1996.

[ZvDM98] M. Zeelenberg, W. W. van Dijk, and A. S. R. Manstead. Reconsidering the relation between regret and responsibility. *Organizational Behavior and Human Decision Processes*, 74(3):254–272, 1998.

[ZvDMvdP00] M. Zeelenberg, W.W. van Dijk, A. S. R. Manstead, and J. van der Pligt. On bad decisions and disconfirmed expetancies: the psychology of regret and disappointement. *Cognition and Emotion*, 14(4):521–541, 2000.

[ZvDvdP+98] M. Zeelenberg, W.W. van Dijk, J. van der Pligt, A. S. R. Manstead, P. van Empelen, and D. Reinderman. Emotional reactions to the outcomes of decisions: the role of counterfactual thought in the experience of regret and disappointment. *Organizational Behavior and Human Decision Processes*, 75(2):117–141, 1998.

Index