



Une gestion simple des ordres complexes : application au dialogue de commande

Olivier Grisvard, Bertrand Gaiffe, Jean-Marie Pierrel

► **To cite this version:**

Olivier Grisvard, Bertrand Gaiffe, Jean-Marie Pierrel. Une gestion simple des ordres complexes : application au dialogue de commande. [Interne] 99-R-025 || grisvard99a, 1999, 10 p. <inria-00107831>

HAL Id: inria-00107831

<https://hal.inria.fr/inria-00107831>

Submitted on 19 Oct 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Une gestion simple des ordres complexes : application au dialogue de commande

Olivier Grisvard, Bertrand Gaiffe et Jean-Marie Pierrel

LORIA
Campus Scientifique - B.P. 239
54506 Vandœuvre-lès-Nancy Cedex
France

Résumé

Nous nous intéressons dans cet article à la représentation des énoncés de type ordre pour des systèmes de dialogue homme-machine. Dans une première étape, nous montrons qu'il est nécessaire de disposer d'une représentation du contenu des ordres plus abstraite et plus proche de l'énoncé que ne le sont les actions de l'application commandée via le système de dialogue. Nous proposons alors d'utiliser des événements pour représenter le contenu des ordres. Nous étudions ensuite des ordres plus complexes, c'est-à-dire contenant des négations, des conditions, des disjonctions et des conjonctions, pour déterminer la portée de la force illocutoire vis à vis du contenu propositionnel de l'ordre. En posant que la force illocutoire porte sur la totalité du contenu, on perd au premier abord la nécessité que ce contenu dénote un événement. Pourtant, le contenu d'un ordre n'est pas une proposition quelconque. Nous montrons donc que seules les propositions permettant de déduire la possibilité d'un événement futur dont l'interlocuteur est l'agent sont candidates à être le contenu d'un ordre. La représentation que nous proposons présente deux avantages :

- les énoncés de type ordre, aussi complexes soient-ils, peuvent être traités de façon homogène en termes de gestion du dialogue ;
- on peut utiliser les acquis des sémantiques formelles pour résoudre les co-références présentes dans ce type d'énoncé.

1. Introduction

Lorsque l'on s'intéresse à la gestion de dialogues de commande¹, il convient de distinguer deux types de systèmes. Certains systèmes de dialogue sont conçus pour des utilisateurs experts d'un domaine. Dans un tel cas, la gestion du dialogue peut être fortement simplifiée :

- l'utilisateur connaît parfaitement les fonctionnalités de l'application, le système n'a alors

1. C'est-à-dire de dialogues permettant à un utilisateur d'agir sur une application informatique.

que peu de sous-dialogues à gérer dans la mesure où l'utilisateur fournit tous les paramètres nécessaires à l'exécution directe des actions qu'il ordonne ;

- dans certaines phases, le système peut prendre complètement l'initiative du dialogue, et l'utilisateur se plie à cette forme de gestion du dialogue.

Le second cas de figure est celui de systèmes plus « grand public » ou moins fréquemment utilisés par un individu donné. Dans ce cadre, on doit faire face à :

- des ordres qui nécessitent d'effectuer plus d'une action pour être satisfaits ;
- des contestations plus fréquentes de la part de l'utilisateur ;
- une gestion de sous-dialogues plus complexe.

L'utilisateur ne dominant pas forcément parfaitement les fonctionnalités spécifiques à l'application, il faut lui permettre de se focaliser sur son but ou sa tâche et non sur leur mise en œuvre. Il doit penser à « que faire » et non à « comment l'exprimer ». Nous allons illustrer ce dernier point dans le cadre de l'application d'édition vidéo pour laquelle nous développons actuellement un système de dialogue. Dans cette application, on peut en particulier commander des caméras vidéo, enregistrer le flux vidéo qui en provient et rejouer des séquences enregistrées. Un énoncé de l'utilisateur tel que (1) débouche ainsi sur l'exécution de deux actions², à savoir `enregistrement(caméra1)` et `enregistrement(caméra2)`.

(1) *Enregistre avec les deux premières caméras.*

Nous présentons par la suite le choix que nous avons fait de recourir à des référents abstraits tels que des événements pour représenter les ordres. Nous montrons en particulier en quoi cela simplifie la gestion du dialogue. Nous nous intéressons ensuite à des ordres plus complexes dont l'argument n'est pas directement un événement, comme (2).

(2) *S'il ne reste plus de place sur le serveur, efface la deuxième séquence puis enregistre avec la caméra numéro deux.*

Nous présentons la solution proposée par la logique illocutoire de Searle et Vanderveken (Searle & Vanderveken, 1985), (Vanderveken, 1991) pour le traitement de ce type d'énoncé. Nous critiquons ensuite cette solution et proposons une forme plus simple de représentation. Enfin, nous montrons en quoi cette forme plus simple correspond à nos besoins en matière de gestion du dialogue et de calcul de la référence, sans pour autant perdre en généralité vis à vis du type de dialogue qu'elle permet de traiter.

2. Représentation des ordres

Un système de dialogue de commande sert d'interface entre un utilisateur et une application informatique. Une telle application correspond à une bibliothèque d'actions (de fonctions) élémentaires. Identifier un ordre ne se limite pas à détecter un verbe à l'impératif dans l'énoncé, d'autres formes sont en effet possibles :

(3) *Maintenant, tu mets la caméra en veille.*

(4) *Mettre la caméra en veille.*

2. Une autre solution serait de décrire des « macro-actions » dans l'application, leur nombre risque cependant de devenir prohibitif (Sabah *et al.*, 1998).

Par ailleurs, tous les énoncés d'un dialogue de commande ne sont pas des ordres, le système peut poser des questions auxquelles l'utilisateur peut répondre par de simples assertions ou refuser de répondre en redonnant un autre ordre :

- (5) U : *Allume la caméra numéro un.*
 S : *Elle est déjà allumée, veux-tu que j'allume la caméra numéro deux?*
 U : *Affiche une fenêtre vidéo.*

Si tout se passe bien, interpréter un ordre revient cependant à identifier une action élémentaire de l'application. Certaines approches du dialogue assimilent donc directement l'action demandée par l'utilisateur à une action élémentaire (Perrault *et al.*, 1978), (Allen & Perrault, 1980), (Cohen & Levesque, 1990). Ainsi dans ces approches, récemment reprises en termes de propositions de standards pour le codage de dialogues (Carletta *et al.*, 1997), la reconnaissance d'un ordre s'appuie sur l'identification de l'action associée. Cependant, il se peut qu'une action associée à un ordre donné ne soit pas directement exécutable, par exemple si l'une de ses pré-conditions n'est pas vérifiée, comme en (6). Un sous-dialogue s'engage alors entre l'utilisateur et le système pour aboutir *in fine* à l'exécution de l'action.

- (6) U : *Filme avec la première caméra.*
 S : *Celle-ci n'est pas allumée, veux-tu que je l'allume ?*
 U : *Oui.*

Pour être en mesure de gérer ce sous-dialogue, le système doit disposer d'une description abstraite de l'action demandée, cette description étant alors partiellement instanciée au cours du sous-dialogue. Il se peut même qu'un énoncé ne débouche pas directement sur une action élémentaire de l'application, comme dans l'exemple (7).

- (7) *Joue la troisième séquence en boucle.*

Auquel cas, le système doit disposer d'une représentation encore plus abstraite de l'action demandée et surtout beaucoup plus proche de la forme utilisée dans l'énoncé, quelque chose comme `jeu_en_boucle(séquence3)` pour l'exemple précédent. C'est sur la base d'une telle représentation qu'on peut identifier, par planification, la suite d'actions élémentaires (Litman & Allen, 1984) :

1. sélection d'une séquence sur le serveur vidéo ;
2. mise du serveur vidéo en mode cyclique ;
3. affichage d'une fenêtre vidéo ;
4. assignation du flux de sortie du serveur à cette fenêtre.

Un objet abstrait de type événement est un bon candidat pour cette représentation abstraite proche de l'énoncé. Cela permet d'être en conformité avec ce que proposent les sémantiques formelles (Kamp & Reyle, 1993), (Asher, 1993) dans le cas du discours. L'action mentionnée dans l'énoncé vérifie bien les propriétés des événements puisque :

- elle est située dans le temps (futur dans notre cas) ;
- ses participants participent à son identification ;
- elle modifie un état de fait concernant ses participants.

L'action ou la séquence d'actions élémentaires déclenchée dans l'application suite à un ordre, est alors le référent de l'événement mentionné dans l'énoncé. On a donc une forme du type

ordre($\exists e, e \succ \text{now} \wedge \text{jeu_en_boucle}(e) \wedge \dots$) où e est un événement et où \succ correspond à la succession temporelle, pour représenter un énoncé comme (7).

Une telle représentation est tout à fait en accord avec la théorie des actes de langage (Austin, 1970), (Searle, 1972), qui représente les énoncés en termes de force illocutoire (FI) (ordre, assertion, promesse, ...) et de contenu propositionnel (CP) sous la forme $FI(CP)$. En termes de mise en œuvre, cependant, il est alors nécessaire de différencier proprement les présuppositions des autres conditions préparatoires aux ordres. Pour la théorie classique des actes de langage, un locuteur donne un ordre si et seulement si :

1. il souhaite voir réaliser le contenu propositionnel par l'interlocuteur ;
2. il pense que l'interlocuteur est en mesure de réaliser ce contenu propositionnel ;
3. il pense que le contenu propositionnel n'est pas déjà réalisé.

La dernière condition en particulier ne doit pas être utilisée pour identifier les objets impliqués dans l'ordre. En effet, on risquerait alors de voir échouer l'interprétation d'un exemple tel que (8), dans le cas où la caméra est déjà allumée, et ce parce qu'on chercherait un objet qui satisfasse $\text{caméra}(x) \wedge \text{numéro}(x, 1) \wedge \text{éteint}(x)$.

(8) *Allume la caméra un.*

Là où Searle et Vanderveken considèrent un acte futur de l'interlocuteur dont les pré-conditions entrent dans les conditions préparatoires de l'ordre, il nous semble par conséquent plus explicite de considérer un événement. La satisfaction de l'ordre passera alors simplement par la réalisation de cet événement.

Il n'est cependant pas évident qu'un ordre soit toujours de la forme **ordre**(événement)³. Une façon de vérifier si tel est le cas est de s'intéresser à des ordres plus complexes contenant des connecteurs logiques dont on sait qu'ils ne s'appliquent pas directement aux événements (Asher, 1993).

3. Ordres complexes

En examinant le cas des ordres simples, nous avons justifié le fait que leur contenu dénote un événement. Cependant il existe de nombreux cas d'ordres plus complexes (négatifs, disjonctifs, multiples, etc) pour lesquels étendre la contrainte que nous venons de mettre en évidence n'est pas trivial. En effet, il n'est par exemple pas évident que la négation d'un événement soit un événement, que la disjonction de deux événements soit un événement, etc (Asher, 1993). Par conséquent nous devons étudier plus en détail les principaux cas d'ordres complexes, à savoir :

- les ordres sous condition, comme « *Si tu veux filmer, allume la caméra.* » ;
- les ordres négatifs, comme « *N'éteins pas la caméra.* » ;
- les ordres disjonctifs, comme « *Eteins la caméra ou mets la en veille.* » ;
- les ordres multiples, comme « *Arrête de filmer et éteins la caméra.* ».

3.1. Ordres et conditionnelles

Le premier problème qui se pose si l'on veut traiter un énoncé comme (9), qui correspond à un ordre associé à une conditionnelle, est la nature de la force illocutoire de l'acte associé, ainsi

3. Ou, pour être plus précis, **ordre**($\exists e, \dots$).

que la portée de celle-ci vis à vis du contenu de l'acte.

(9) *S'il existe une séquence Essai, efface la.*

Les travaux dans la lignée de la théorie classique des actes de langage (Searle & Vanderveken, 1985), (Vanderveken, 1991) proposent une logique illocutoire pour décrire, en particulier, de tels énoncés. Dans cette logique, la forme associée à (9) est $P \rightarrow F(Q)$, P et Q étant les propositions « *Il existe une séquence Essai.* » et « *Efface la séquence Essai.* » respectivement et F la force illocutoire de l'acte, **ordre** dans le cas qui nous intéresse. Ceci suppose de redéfinir l'implication entre P et $F(Q)$ puisque il ne peut s'agir de l'implication logique classique, étant donnée qu'elle connecte un objet de type proposition (P) et un objet de type acte de langage ($F(Q)$). Pour traiter ainsi de tels exemples, il est donc nécessaire de définir entièrement une logique illocutoire complexe avec des connecteurs entre actes de langage.

Une autre difficulté tient à la gestion du dialogue. Même si la proposition P est fautive, un ordre a été donné. Si on suppose que non, on ne saura tout simplement pas quoi faire d'un tel énoncé⁴. Enfin, en ce qui concerne le calcul de la référence, puisque le contenu propositionnel est séparé en deux composantes, P et Q , il faut rebâtir explicitement toute une théorie de l'anaphore. Par exemple, en (9), on a séparé les propositions contenant respectivement l'antécédent « *séquence Essai* » et le pronom « *la* ». Il faut donc redéfinir les conditions d'accès aux antécédents pour les pronoms.

Par ailleurs, dans les cas où la proposition P est manifestement fautive ou présupposée fautive, comme dans l'exemple (10), il semble que l'énoncé doive être interprété comme **ordre**($\neg Q$), c'est-à-dire comme (11). Or la forme $P \rightarrow F(Q)$ ne permet pas de déduire quoi que ce soit concernant la force illocutoire de l'énoncé quand P est fautive, à moins d'ajouter une contrainte supplémentaire sur l'implication.

(10) *Si tu veux perdre tout l'enregistrement, sors sans sauver.*

(11) *Ne sors pas sans sauver.*

Une telle logique illocutoire préserve, au moins sur ce type d'exemple, la contrainte selon laquelle un événement (futur) est sous le champ de l'ordre. Le coût de la préservation de cette contrainte est cependant lourd, il faut en effet :

- définir entièrement la logique illocutoire ;
- expliciter complètement la résolution des anaphores, c'est-à-dire expliciter la combinaison des propositions en termes d'accessibilité par dessus les connecteurs illocutoires ;
- bâtir des modèles du dialogue intégrant cette logique.

Une autre façon de traiter ce type d'ordres consiste tout simplement à intégrer la proposition telle quelle sous le champ de la force illocutoire, et c'est ce que nous préconisons. Dans ce qui suit, nous noterons **dire de**⁵ les ordres. Cette formulation, qui provient de la théorie de la pertinence (Sperber & Wilson, 1989) a le mérite d'imposer, ne serait-ce que syntaxiquement, la présence d'une proposition. Nous sommes donc obligés d'affaiblir l'hypothèse selon laquelle les ordres sont de la forme **dire de**(événement)⁶ et de revenir à la forme « standard » **dire de**(proposition). Le contenu propositionnel d'un acte de langage de type ordre sous condition a alors pour forme logique $P \rightarrow Q$, et la force illocutoire s'applique à l'ensemble du

4. On risque en particulier de le traiter comme une assertion.

5. Au sens de « *dire de faire quelque chose* ».

6. Pour être plus précis, nous ne pouvons pas imposer à la proposition d'être de la forme $\exists e, \dots$

contenu, ce qui nous donne une représentation de la forme *dire de*($P \rightarrow Q$).

Exprimé sous cette forme, on a bien un ordre, globalement. L'interlocuteur doit maintenir la proposition $P \rightarrow Q$, avec Q qui dénote un événement, vraie. Ainsi nous préservons les propriétés véri-conditionnelles du contenu de l'acte. Si P est vraie, alors Q ne reste vraie que si l'événement a lieu. Si P est fausse, on ne préjuge pas de la vérité de Q et l'interlocuteur peut ou non réaliser l'action. En termes de dialogue homme-machine, dans le cas où P est vraie, il faut donc réaliser un événement vérifiant les conditions imposées (en particulier le futur) de telle façon qu'effectivement cet événement existe. Si au contraire P est fausse, le système n'a aucune raison de donner un référent à l'événement. Nous ne reviendrons sur le cas où P est manifestement fausse, comme en (10), que dans la conclusion, dans la mesure où le traitement de tels exemples n'est pas une priorité pour le type de dialogue qui nous intéresse.

3.2. Ordres et négation

Contrairement à la logique illocutoire, nous jugeons inutile de définir la négation d'un ordre, c'est-à-dire un opérateur de négation de la force illocutoire ($\neg F(P)$). A l'origine, ce type de négation a été introduit pour faire la différence entre des énoncés tels que (12) et (13).

(12) *Je ne t'ordonne pas d'effacer cette séquence.*

(13) *N'efface pas cette séquence.*

Cependant, il nous semble que (12) est tout simplement une assertion (*dire que* pour la théorie de la pertinence). La reconnaissance de cette force illocutoire ne nécessite aucunement de passer par une forme comme $\neg \text{dire de}(P)$. Dans le cas d'ordres effectivement négatifs, la négation intervient au niveau du contenu propositionnel. Dans un cas comme (13), pour maintenir la proposition $\neg \exists e, \text{effacement}(e) \wedge \dots$ vraie, le système doit précisément ne pas réaliser d'événement d'effacement de la séquence.

3.3. Ordres et disjonction

Pour les mêmes raisons que précédemment, à savoir, nécessité de préserver les propriétés véri-conditionnelles du contenu propositionnel et d'éviter la définition d'une logique illocutoire complexe, nous considérons que dans la représentation d'un ordre disjonctif, la disjonction reste au niveau du contenu propositionnel: *dire de*($P \vee Q$) plutôt que *dire de*(P) \vee *dire de*(Q). Un exemple tel que (14), donne donc lieu à la forme *dire de*(($\exists e_1, \text{extinction}(e_1) \wedge \dots$) \vee ($\exists e_2, \text{mise_en_veille}(e_2) \wedge \dots$)).

(14) *Eteins la caméra ou mets la en veille.*

Le traitement de cette formule amène le système à donner un référent à e_1 ou e_2 . Le choix de l'événement réalisé pouvant se faire selon des contraintes indépendantes de l'énoncé⁷.

3.4. Ordres et conjonction

Le cas des conjonctions comme (15) est évidemment moins clair, la différence entre *dire de*($P \wedge Q$) et *dire de*(P) \wedge *dire de*(Q) étant peu sensible.

(15) *Arrête de filmer et éteins la caméra.*

7. Un tirage au sort pourrait convenir !

L'interlocuteur, toujours de la même façon, doit maintenir la proposition $P \wedge Q$ vraie. Il doit donc réaliser l'événement dénoté par P et l'événement dénoté par Q . La différence est donc peu sensible dans la mesure où pour la suite d'ordres (16) il devra également réaliser l'événement dénoté par P et celui dénoté par Q .

- (16) U : *Arrête de filmer.*
 U : *Eteins la caméra.*

La seule différence possible, mais elle vaut également pour la logique illocutoire, est que suite à (15), l'interlocuteur pourrait éventuellement réaliser d'abord l'événement dénoté par Q , puis celui dénoté par P , alors que c'est plus difficile dans le cas de (16). Cependant, considérer dans le cas de (15) que la conjonction est interne au contenu propositionnel permet :

- d'unifier le cas de la conjonction et les autres cas ;
- de raisonner sur le contenu pour ordonner les événements selon des critères extérieurs à l'énoncé. Dans l'exemple ci-dessus, il est nécessaire d'arrêter de filmer avant d'éteindre la caméra. Dans un autre système, l'extinction de la caméra pourrait impliquer de fait un événement correspondant à l'arrêt de filmer.

3.5. *Quelle contrainte quant au contenu d'un ordre ?*

Nous avons justifié la possibilité de faire porter la force illocutoire sur la totalité de l'énoncé. Ce faisant, nous avons perdu la contrainte selon laquelle le contenu d'un ordre dénote un événement. Dans la mesure où l'entrée d'un système de dialogue de commande est en général un système de reconnaissance de parole, on aimerait rétablir, autant que faire se peut, des contraintes prédictives pour guider la reconnaissance, ou au moins filtrer parmi les hypothèses d'énoncés reconnues. Intuitivement, n'importe quelle proposition ne peut pas entrer sous le champ d'un ordre.

Or nous pouvons établir que le contenu d'un ordre doit permettre de démontrer la possibilité d'un événement. La grammaire du système mis en œuvre permet ensuite de déduire les formes syntaxiques associées⁸. Nous avons mentionné précédemment les conditions préparatoires aux ordres, à savoir :

- la proposition à rendre vraie par l'interlocuteur ne l'est pas déjà ;
- il est possible de rendre cette proposition vraie.

Autrement dit, la proposition P sous le champ d'un *dire de* est telle que $\neg \Box P$ (i.e. $\Diamond \neg P$) et $\Diamond P$ ⁹. En raisonnant sur l'ensemble des ordres complexes que nous avons considérés, nous allons montrer que ces propriétés demeurent pour les sous-propositions incluses.

- Conditionnelles : on a $\Diamond(A \rightarrow P)$ et $\Diamond \neg(A \rightarrow P)$, donc chaque fois que A est vraie, on a $\Diamond P$ et $\Diamond \neg P$. Le seul cas difficile serait celui où l'on aurait $\Box \neg A$, cas que nous renvoyons en conclusion comme mentionné précédemment.
- Négation : on a $\Diamond \neg P$ et $\Diamond \neg(\neg P)$ c'est-à-dire $\Diamond P$.
- Disjonction : on a $\Diamond(P \vee Q)$ et $\Diamond \neg(P \vee Q)$ c'est-à-dire $(\Diamond P) \vee (\Diamond Q)$ et $(\Diamond \neg P) \wedge (\Diamond \neg Q)$ ce qui garantit que l'une au moins des propositions est telle que $\Diamond P$ et $\Diamond \neg P$.

8. Un cas simple est celui dans lequel les ordres ne peuvent être donnés qu'à l'impératif, on aura alors établi que tout ordre, simple ou non, devra contenir un impératif.

9. Nous reprenons ici les notations habituelles de la logique modale : $\Box P$ pour « P nécessaire » et $\Diamond P$ pour « P possible ».

- Conjonction : on a $\diamond(P \wedge Q)$ et $\diamond\neg(P \wedge Q)$ c'est-à-dire $(\diamond P) \wedge (\diamond Q)$ et $(\diamond\neg P) \vee (\diamond\neg Q)$ ce qui garantit également que l'une au moins des propositions est telle que $\diamond P$ et $\diamond\neg P$.

Chacun de ces cas permet de déduire pour au moins une proposition incluse P_i à la fois $\diamond P_i$ et $\diamond\neg P_i$. L'ensemble de la formule permet donc de déduire par récurrence $\diamond P$. L'interlocuteur doit donc pouvoir rendre la proposition vraie, ce qui suppose qu'il puisse *contrôler* la vérité de cette proposition. La démonstration précédente vaut donc pour des propositions qui dénotent des événements dont l'interlocuteur est agent¹⁰.

4. Conséquences sur le calcul de la référence

Du fait que pour nous l'ensemble du contenu propositionnel reste sous le champ de la force illocutoire, des solutions telles que celle proposées par la DRT (Kamp & Reyle, 1993) sont directement applicables pour le calcul de la co-référence. En particulier, on peut traiter des « donkey-ordres » comme par exemple (17), qui sera alors équivalent à (18).

(17) *Si une caméra se met en veille, rallume la.*

(18) *Rallume toute caméra qui se mettrait en veille.*

Bien entendu, on a alors envie d'étendre le mécanisme, en particulier à des suites d'ordres. La remarque que nous avons faite quant à la faible différence entre *dire de*($P \wedge Q$) et *dire de*(P) suivi de *dire de*(Q) joue dans les deux sens. Autrement dit, on peut envisager de traiter des séquences d'ordres comme raisonnablement équivalentes à un unique ordre sur leur combinaison. En d'autres termes, si on note \oplus l'opération, proposée par la DRT, d'incorporation d'une nouvelle phrase dans la représentation courante d'un discours, on aura : (*dire de*(P) suivi de *dire de*(Q)) = *dire de*($P \oplus Q$). On constate en effet les mêmes impossibilités de reprise par des pronoms que dans le cas des phrases d'un discours, comme en (19) et (20)¹¹, les référents de discours indéfinis introduits sous le champ d'une condition étant inaccessibles pour des reprises pronominales.

(19) *Ne bouge aucune caméra. Eteins la. (?)*

(20) *Si un conducteur possède une voiture bleue, arrête la. Emmène le au poste de police. (?)*

5. Conclusion

Nous avons proposé dans cet article une modélisation des énoncés de type ordre, simples et complexes, dans le cadre du dialogue de commande. Cette approche nous permet de conserver intacte la notion de contenu propositionnel. Nous sommes donc compatibles avec les sémantiques formelles de type DRT/S-DRT pour tout ce qui concerne le traitement de la référence. De plus, nous avons proposé un ensemble de contraintes sur le contenu des ordres qui rend notre approche prédictive, d'une part vis à vis de la reconnaissance de la force illocutoire directive, et d'autre part en ce qui concerne les contestations possibles suite à un ordre.

On remarque que pour un ordre disjonctif, il n'est pas pertinent pour l'interlocuteur de réaliser l'ensemble des événements dénotés par les propositions liées par la disjonction. Il lui suffit de réaliser l'un de ces événements pour satisfaire l'ordre. Il est possible d'expliquer ceci grâce

10. Le fait que la possibilité d'établir la vérité de la proposition soit une croyance éventuellement erronée du locuteur ne change rien quant à la façon d'exprimer un ordre.

11. Nous notons par un (?) les énoncés qui nous semblent douteux.

à la maxime de quantité de Grice (Gazdar, 1979). En effet, étant donné que la disjonction de deux propositions est impliquée par la conjonction de ces mêmes propositions, on peut définir une échelle quantitative entre conjonction et disjonction. Appliquée à cet échelle quantitative, la maxime de quantité de Grice stipule que si le locuteur utilise une disjonction de deux propositions, il implique que la conjonction de ces mêmes propositions n'est pas pertinente. Par conséquent, un ordre disjonctif contient donc une disjonction implicitement exclusive.

On peut appliquer le même raisonnement à un ordre sous condition. En effet, le contenu propositionnel d'un tel ordre est de la forme $P \rightarrow Q$, c'est-à-dire $\neg P \vee Q$, cette disjonction étant pour les mêmes raisons également implicitement exclusive. Par conséquent, si P est fausse ($\neg P$ vraie), on a implicitement $\neg Q$. On voit alors que dans le cas d'un ordre sous condition dont la condition est manifestement fausse ou présupposée fausse, soit *dire de*($\neg P \vee Q$) avec $\neg P$ manifestement vraie ou présupposée vraie, on a implicitement *dire de*($\neg Q$). Ceci est conforme à l'intuition que l'on a face à un exemple comme (10) qui semble devoir être traité de façon similaire à (11).

(10) *Si tu veux perdre tout l'enregistrement, sors sans sauver.*

(11) *Ne sors pas sans sauver.*

Dans le cadre du dialogue de commande, nous avons principalement affaire à des ordres. Cependant, en présence de contestations nous avons vu que le système doit être en mesure de mener un sous-dialogue de négociation avec l'utilisateur. De plus, même pour du dialogue de commande, on peut envisager d'autoriser un utilisateur à poser des questions au système, notamment si cet utilisateur n'est pas expert du domaine d'application et souhaite s'informer sur telle ou telle fonctionnalité avant de produire un ordre. Par conséquent, nous souhaitons étendre par la suite le traitement que nous proposons ici aux autres types d'actes de langage. Ceci afin d'exhiber des contraintes sur le contenu de ces types d'énoncés similaires à celles que nous proposons pour les ordres, et pouvoir ainsi insérer des énoncés de type questions/réponses dans notre modèle de traitement du dialogue.

Références

- ALLEN J. F. & PERRAULT C. R. (1980). Analysing Intentions in Utterances. *Artificial Intelligence*, **15**, 143–178.
- ASHER N. (1993). *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers.
- AUSTIN J. L. (1970). *Quand dire c'est faire*. Editions du Seuil.
- CARLETTA J., DAHLBÄCK N., REITHINGER N. & WALKER M. A. (1997). *Standards for Dialogue Coding in Natural Language Processing*. Seminar Report 167 (9706), Dagstuhl.
- COHEN P. R. & LEVESQUE H. J. (1990). Rational Interaction as the Basis for Communication. In COHEN, MORGAN & POLLACK, Eds., *Intentions in Communication*, p. 221–255. M.I.T. Press.
- GAZDAR G. (1979). *Pragmatics: Implicature, Presupposition, and Logical Form*. Academic Press.
- KAMP H. & REYLE U. (1993). *From Discourse to Logic : Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer Academic Publishers.
- LITMAN D. J. & ALLEN J. F. (1984). *A Plan Recognition Model for Subdialogues in Conversations*. Technical Report 141, Department of Computer Science, University of Rochester.
- PERRAULT C. R., ALLEN J. F. & COHEN P. R. (1978). Speech Acts as a Basis for Understanding Dialogue Coherence. In *2nd Conference on Theoretical Issues in Natural Language Processing*, p. 125–132, Urbana-Champaign, Illinois.
- SABAH G., VIVIER J., VILNAT A., PIERREL J.-M. & ROMARY L. (1998). *Machine, langage et dialogue*. L'Harmattan.

SEARLE J. R. (1972). *Les actes de langage*. Hermann.

SEARLE J. R. & VANDERVEKEN D. (1985). *Foundations of illocutionary logic*. Cambridge University Press.

SPERBER D. & WILSON D. (1989). *La pertinence : Communication et cognition*. Les Editions de Minuit.

VANDERVEKEN D. (1991). *Meaning and speech acts*. Cambridge University Press.