Lógica Modal para Sistemas Multi-Agentes Abertos

Juliana Carpes Imperial   Edward Hermann Haeusler
Departamento de Informática – PUC-Rio – Rio de Janeiro, RJ
{*juliana, hermann*}@*inf.puc-rio.br*

Resumo

Hoje em dia muitos são os sistemas distribuídos abertos de larga-escala que precisam operar em ambientes com muita incerteza e que se modificam constantemente. Dada essa complexidade, esses componentes, e o sistema resultante, são cada vez mais contextualizados, desenhados e construídos usando técnicas baseadas em agentes. Portanto, é interessante haver uma lógica modal para modelar o fato de um sistema multi-agentes ser aberto, onde os mundos possíveis que modelam um sistema multi-agentes representam quais agentes estão presentes em um dado instante de tempo.

1 Introdução

Muitas aplicações para computadores são sistemas distribuídos nos quais (muitos) componentes constituintes estão espalhados através de uma rede, em um regime de controle descentralizado, e que estão sujeitos a mudanças constantes durante o tempo de vida do sistema. Exemplos incluem computação ponto-a-ponto, web semântica, serviços da web, e-business, m-commerce, computação independente (*autonomic computing*), computação em grade (*grid computing*) e ambientes de computação pervasivos. Em todos esses casos, há a necessidade de se ter componentes autônomos que agem e interagem de maneira flexível de modo a atingir os objetivos para os quais foram criados em ambientes com incertezas e dinâmicos. Dado isso, a computação baseada em agentes tem sido defendida como um modelo computacional natural para tais sistemas [Jen01, Ram04].

Mais especificamente, sistemas distribuídos abertos podem ser modelados como sistemas multi-agentes (MAS, *multi-agent system*) [Woo02] abertos que são compostos de agentes autônomos que interagem uns com os outros. Dado que o sistema é aberto, agentes podem entrar e sair a qualquer momento, não se conhecendo a priori quem são os agentes que irão interagir nem quem foram os desenvolvedores desses agentes.

Portanto, é interessante haver uma lógica modal [Che80, Gol92, Hug96] para modelar o fato de um MAS ser aberto. Com ela, pode-se modelar quais agentes estão presentes em um dado instante de tempo usando-se a semântica de mundos possíveis, onde cada mundo representa quais agentes estão presentes em cada instante de tempo. A partir dessa formalização lógica, é possível se formular propriedades desejáveis para que o MAS aberto funcione adequadamente. Isso permite a descrição do comportamento de entrada e saída dos agentes e de quais agentes estão presentes a cada instante de tempo.

O resto do trabalho é organizado como a seguir. Na sessão 2, explica-se o porquê de se usar lógica modal para modelar o comportamento de um MAS aberto. A seguir, na sessão 3, modela-se o funcionamento de um MAS usando lógica modal. Já na sessão 4 descreve-se um trabalho relacionado. Por fim, a sessão 5 conclui o texto, apresentando um trabalho futuro.

2 O Porquê do Uso de Lógica Modal

Usa-se lógica para formalizar um sistema multi-agentes aberto porque com ela é possível se fixar em uma linguagem artificial bem-definida e estruturada, expressando propriedades de uma maneira rigorosa e matemática. Outra vantagem é que a ambigüidade pode ser removida com o uso de lógica. Tem-se ainda que ao se expressar as propriedades de agentes e de sistemas multi-agentes como axiomas lógicos e teoremas em uma linguagem com uma semântica clara, os pontos principais da teoria são explicitados. Finalmente, a teoria torna-se transparente: as propriedades, os relacionamentos e as inferências são abertas a examinação. Usando-se uma modelagem em linguagem natural, por exemplo, que em geral é mal-estruturada e mal-definida, não se consegue tal clareza [Dal97, End02, Woo00].

Seria possível usar lógica clássica de primeira ordem para formalizar o sistema, já que a mesma é expressiva o suficiente para ser usada para codificar praticamente qualquer forma de conhecimento. Com certa ingenuidade, é possível codificar em lógica clássica de primeira ordem o que for codificável em lógica modal, por exemplo. Portanto, pode-se afirmar que a lógica clássica de primeira ordem é de fato um formalismo genérico. Entretanto, as sentenças ficam muito mais extensas e menos intuitivas quando abordam modalidades ou idéias temporais, algo presente ao se modelar sistemas multi-agentes.

Mais ainda, existem diversos casos de traduções que não se cumprem em lógica clássica de primeira ordem para conceitos definidos de forma computável (leia-se decidível) em lógica modal. Outros exemplos são as classes de *frames* não-elementares, que não podem ser definidos em lógica clássica de primeira ordem, mas que são modalmente especificáveis. Esses são definidos por axiomas que não possuem a classe de *frames* que os validam (a saber *frames* com o reverso da relação de acessibilidade bem-fundados), não podendo ser especficados por nenhum conjunto de sentenças em lógica clássica de primeira ordem.

Mais especificamente, as lógicas modais podem ser entendidas como linguagens especializadas para representar propriedades de estruturas relacionais. Elas são úteis por sua capacidade de formalizar aspectos relacionais (do *frame*) de forma efetiva, em comparação com a versão dos mesmos em primeira ordem. No caso de propriedades temporais, por exemplo, a estrutura relacional em questão é a estrutura temporal do mundo. Se conectivos modais não forem utilizados para representar propriedades da estrutura relacional, parece ser necessário introduzir a estrutura relacional na linguagem de representação explicitamente, tornando-a muito pesada. Ou seja, com lógica modal, o formalismo fica mais próximo da linguagem natural. Assim, o entendimento e a busca por propriedades no mesmo fica bastante facilitada.

Outra vantagem da lógica modal é que, enquanto a maioria das lógicas modais é decidível, a lógica clássica de primeira ordem é semi-decidível, sendo a maior parte dos sistemas de lógica modal PSPACE-completo. Apesar da alta complexidade, na prática, usando-se técnicas de model-checking [Cla99], consegue-se provar a satisfabilidade de uma fórmula em um dado modelo eficientemente no tamanho do mesmo.

3 Modelagem Lógica

Formaliza-se agora o sistema multi-agentes. Como o sistema é aberto, em cada instante de tempo, agentes podem entrar, sair ou permanecer no sistema. Há basicamente duas maneiras diferentes para se formalizar o sistema usando a semântica de mundos possíveis:

- Só há mudança de estado quando um agente sai ou entra no sistema. Ou seja, se o conjunto de agentes permanecer o mesmo, não há transição de um mundo para outro.

- A cada instante de tempo há uma transição de um mundo para outro, que pode ser o mesmo se o conjunto de agentes permanecer o mesmo.

O segundo modelo foi o escolhido por permitir a representação de tempo explicitamente. Adicionalmente, mesmo que o conjunto de agentes não mude, eles estão executando ações e, possivelmente, interagindo entre si, o que não é instantâneo. Portanto, enquanto um agente está sem tomar nenhuma ação, apenas tomando decisões e analisando o ambiente, o tempo está passando. Mais ainda, o tempo é considerado como sendo global. Ou seja, o tempo passa da mesma forma e é o mesmo para todos os agentes do sistema. Para isso, pode-se supor que os agentes podem consultar seus parceiros, sua organização ou o sistema quanto ao tempo. Isso pode ser necessário quando um agente entra no sistema, pois o sistema anterior do agente poderia vir a se comportar de uma maneira diferente quanto ao tempo. Afinal, cada sistema tem as suas convenções, podendo variar quanto à definição de tempo.

O modelo temporal possui as seguintes características [Cla99, Woo00]:

- discreto;
- limitado no passado (possui um estado inicial);
- não-limitado no futuro (não possui um estado final);
- linear no passado (só há um passado), e
- ramificado no futuro (o futuro é não-determinístico; os eventos futuros ainda não são determinados).

O modelo temporal é formalizado usando-se uma estrutura temporal ramificada, que é total, e um grafo direcionado e linear no passado sobre um conjunto de instantes T. Note-se que uma relação R é total se cada nó em R tiver ao menos um sucessor. Ou seja, cada elemento na relação sempre vai ter um par. Deve ficar claro que dizer que a relação é total não significa que ela é uma ordem total, ou seja, dizer que a relação é linear. Logo, a relação binária R sobre T é total se satisfizer a seguinte condição:

$$\forall t \in T \Rightarrow \exists t'[t' \in T \text{ e } (t, t') \in R]$$

Como não há um estado final, essa propriedade é satisfeita. $R \subseteq TXT$ representa a estrutura temporal ramificada que codifica todas as maneiras nas quais o sistema pode evoluir.

O sistema é formalizado através de uma estrutura de mundos possíveis $< W, R >$, onde $W \neq \emptyset$, cada mundo $w \in W$ representa o conjunto de agentes presentes em um dado instante e R representa a relação de transição de um mundo para outro, que representa a passagem de um instante de tempo. Note-se que o R aqui é diferente do da relação de passagem de tempo definido no parágrafo anterior. Se o conjunto de agentes continuar sendo o mesmo no instante seguinte, tem-se que $R(w, w)$. Já se o conjunto de agentes de $w \in W$ se modificar para ser o conjunto de agentes de $w' \in W$, tem-se que $R(w, w')$ [Hug96].

Já a presença dos agentes é representada por variáveis proposicionais a_0, a_1, ..., correspondendo aos agentes 0, 1, ... Ou seja, a variável proposicional a_i representa o agente i. Quando um agente estiver presente no sistema, o valor da variável proposicional correspondente será $\top$; caso contrário, $\bot$. As variáveis de indivíduo a_i têm designação rígida (possuem o mesmo significado em todos os mundos). Ou seja, o significado de a_i é o mesmo em todos os mundos, que é a presença do agente i em um instante de tempo. Então, a_i não vai se referir a um agente j em algum mundo ou a qualquer outra coisa que não seja em relação a presença ou não do agente i no sistema.

Então, tem-se que um modelo para a estrutura do sistema multi-agentes é da forma $< W, R, V >$, onde V é uma atribuição de valores a variáveis proposicionais em um mundo $w \in W$ [Hug96]. Nesse caso, a atribuição de valores para as variáveis vai depender do fato de o agente estar ou não presente no sistema em um dado mundo. Logo, tem-se que: $V(\alpha_i, w) = 1$ se e somente se o agente i estiver presente no mundo $w \in W$. Caso contrário, $V(\alpha_i, w) = 0$.

Quanto às sentenças lógicas, será utilizada a lógica modal proposicional, com $\Box$ representando a modalidade de "necessidade" e $\Diamond$, a de "possibilidade". A noção de tempo está presente implicitamente, passando a todo instante, mude o conjunto de agentes ou não, haja ações de agentes ou não.

Para um sistema multi-agentes genérico, o modelo é reflexivo e um grafo contendo ciclos, onde cada mundo está relacionado com todos os outros. Isso acontece porque a cada instante qualquer agente pode entrar ou sair do sistema, ou o conjunto de agentes presentes permanecer o mesmo. O sistema genérico também é, obviamente, não-determinístico, pois não se sabe quais agentes podem entrar ou sair em um dado instante. Mas na prática isso pode não necessariamente acontecer. Por exemplo, se novos agentes puderem entrar no sistema mas não puderem sair, os mundos futuros não podem ter transição para mundos com o mesmo conjunto de agentes de mundos passados. Portanto, dependendo do comportamento do sistema, determinados axiomas da lógica modal podem valer ou não. Alguns exemplos são dados a seguir [Che80, Gol92, Hug96]:

$K : \Box(a_i \rightarrow a_j) \rightarrow (\Box a_i \rightarrow \Box a_j)$: Se em todo estado futuro, se o agente j estiver presente quando o agente i estiver presente, então, em todo estado futuro, se o agente i estiver presente, então, em todo estado futuro o agente j estará presente. Esse axioma vale para qualquer sistema multi-agente.

$T : \Box a_i \rightarrow a_i$: Se em todo mundo futuro o agente i estiver presente, então o agente i está presente no mundo atual. Esse axioma é válido se o modelo for reflexivo, isto é, se a cada instante for sempre possível que o conjunto de agentes presentes não seja modificado, ou seja, que nenhum agente saia ou entre no sistema. Então, o conjunto de agentes pode ser o mesmo (ou repetido) em um instante futuro. O axioma não será válido se necessariamente um agente entrar ou sair do sistema em um dado instante de tempo. Formalmente:

$$\forall w \in W((w, w) \in R)$$

$D : \Box a_i \rightarrow \Diamond a_i$: Esse axioma diz que o sistema multi-agentes sempre continuará funcionando, nunca tendo um estado final. Na prática a maioria dos sistemas multi-agentes possui essa propriedade, que representa a serialidade. Formalmente:

$$\forall w \in W\ \exists w' \in W((w, w') \in R)$$

$4 : \Box a_i \rightarrow \Box\Box a_i$: Esse axioma diz que se um agente estiver presente em todo instante futuro, ele continuará presente no instante a seguir. Em um sistema onde esse axioma valha, se necessariamente houver agentes após o estado inicial do sistema, eles não poderão mais sair. Ou seja, se novos agentes tiverem que entrar, eles não poderão deixar o sistema. Esse axioma representa a propriedade de transitividade. Formalmente

$$\forall w, w', w'' \in W((w, w') \in R \wedge (w', w'') \in R \rightarrow (w, w'') \in R)$$

Um exemplo de sistema onde isso ocorre é em comércio eletrônico e em sites de leilões virtuais. Uma vez cadastrado no sistema, um agente não pode se descadastrar, ou seja, sair do sistema. Como exemplos, tem-se a Amazon.com e o eBay.

$5 : \Diamond a_i \rightarrow \Box\Diamond a_i$: Esse axioma diz que, se o agente i estiver presente em um mundo futuro, então ele estará presente em todo mundo posterior a esse mundo futuro. Outra forma de descrevê-lo é dizer que, dada a passagem de um instante de tempo, se o conjunto de agentes puder ser modificado de duas maneiras distintas, sendo que em uma delas pode permanecer igual, então, a partir dessas duas configurações futuras pode-se chegar na outra e vice-versa no próximo instante de tempo. Por exemplo, se em um dado instante puderem entrar um agente i ou um agente j, mas não os dois ao mesmo tempo, no instante seguinte deve ser possível o outro agente entrar. Isso significa que o modelo que representa o sistema é euclidiano. Formalmente:

$$\forall w, w', w'' \in W((w, w') \in R \wedge (w, w'') \in R \rightarrow (w', w'') \in R)$$

$B : a_i \rightarrow \Box\Diamond a_i$: Esse axioma vale em sistemas nos quais, se for possível mudar o conjunto de agentes presentes no sistema em um dado instante, é possível voltar a configuração anterior, ou seja, ao mesmo conjunto de agentes que estava presente antes. Ou seja, se passou um instante de tempo e determinados agentes entraram, deve ser possível que no instante seguinte aos agentes que entraram saírem. O mesmo vale para agentes que saíram em um dado instante. Esse axioma representa a propriedade de simetria. Formalmente:

$$\forall w, w' \in W((w, w') \in R \rightarrow (w', w) \in R)$$

$Triv : a_i \leftrightarrow \Box a_i$: Nesse caso, o conjunto de agentes nunca muda. Ou seja, não é um sistema multi-agentes aberto. Porém, ele está funcionando. Formalmente:

$$\forall w, w' \in W((w, w) \in R \wedge w = w')$$

$Ver : \Box a_i$: Nesse caso, o sistema está parado. Também não é uma propriedade de interesse. Formalmente:

$$\forall w, w' \in W((w, w) \notin R \wedge w = w')$$

$T_C : a_i \rightarrow \Box a_i$: Nesse caso, o conjunto de agentes nunca muda. O sistema pode estar ou não parado. Formalmente:

$$\forall w, w' \in W(w = w')$$

$D_C : \Diamond a_i \rightarrow \Box a_i$: Nesse caso, se em um mundo futuro um agente estiver presente, então ele vai estar presente em todos os mundos futuros. O sistema é determinístico:

ou o sistema muda o conjunto de agentes de uma única forma ou permanece para sempre com o mesmo conjunto de agentes, ou então está parado. Pode-se também dizer que a relação de acessibilidade é parcialmente funcional. Formalmente:

$$\forall w, w', w'' \in W((w, w') \in R \wedge (w, w'') \in R \rightarrow w' = w'')$$

$G : \Box(\Box a_i \rightarrow a_i) \rightarrow \Box a_i$: $G + K = GL$, que é a lógica da provabilidade. Pode-se tomar com sendo a que tem por classe de frames os transitivos, finitos e irreflexivos (finito-transitivo quer dizer que a cadeia de transitividades é finita). A modalidade em GL é mais associada a conhecimento como demonstração, que pode ser descrita como: se for possível provar que o que quer que seja que for provável for verdadeiro ($\Box(\Box a_i \rightarrow a_i)$), então o que quer que seja é provável ($\Box a_i$). G é um exemplo de axioma que não possui a classe de frames que o valida.

O axioma G vai ser válido em um sistema onde nunca vai haver um mundo onde o conjunto de agentes vai continuar sendo o mesmo após o instante seguinte, por ser irreflexivo. Ou seja, a cada instante de tempo sempre vai haver mudança no conjunto de agentes. Mais ainda, por ser transitivo, os agentes que entrarem não vão poder mais sair. Adicionalmente, a execução do sistema é finita. A irreflexividade até poderia ocorrer em um sistema muito grande, com um número enorme de agentes querendo entrar a todo momento. Mas, na prática, não é uma propriedade muito razoável. Tem-se ainda a finitude, que não é uma propriedade desejável. A primeira propriedade já foram formalizada em 4. A finitude e a irreflexividade são formalizadas a seguir, respectivamente:

$$\forall w \in W \neg \exists w' \in W R(w, w') \qquad \forall w \in W \neg R(w, w)$$

$\Box(a_i \wedge \Box a_i \rightarrow a_j) \vee \Box(a_j \wedge \Box a_j \rightarrow a_i)$: Nesse caso, o conjunto de mundos possíveis é fracamente conectado, ou seja, a partir de um conjunto de agentes presentes no sistema, é possível se chegar a qualquer outro conjunto possível de agentes no sistema através da entrada, saída ou permanência dos agentes. Formalmente:

$$\forall w \forall w' \forall w''((w, w') \in R \wedge (w, w'') \in R \rightarrow (w', w'') \in R \vee w' = w'' \vee (w'', w') \in R)$$

É importante enfatizar que o axioma K vale em todos os tipos de sistema citados acima. Mais ainda, em sistemas onde valem mais de um axioma além de K, é possível se conseguir mais propriedades. Por exemplo, o sistema multi-agentes genérico, onde a qualquer instante agentes podem entrar ou sair satisfaz T e 5.

Um sistema multi-agentes que satisfaz tais axiomas têm como lógica modal correspondente o sistema $S5$. Exemplos de sistemas onde agentes podem entrar e sair a qualquer momento são aqueles que envolvem computação móvel.

Já um sistema onde valha T e o axioma abaixo

$$(a_i \wedge a_j \wedge \Diamond(a_i \wedge \neg a_j)) \rightarrow \Box a_i$$

significa que de um estado para outro, só há uma maneira de o conjunto de agentes se modificar. Isto é, o conjunto de agentes pode se manter igual ou se modificar de apenas uma maneira no máximo. Na semântica de mundos possíveis, isso significa que só há transição de um mundo para ele mesmo (reflexibilidade) e de, no máximo, para apenas um outro mundo. Formalmente:

$$\forall w, w', w'' \in W((w,w) \in R \wedge ((w,w') \in R \wedge (w,w'') \in R \rightarrow w' = w''))$$

Se em um sistema valerem os axiomas D e D_C, tem-se o seguinte axima:

$$\Diamond a_i \leftrightarrow \Box a_i$$

onde pode-se dizer que a relação de acessibilidade é totalmente funcional. Formalmente:

$$\forall w \exists ! w'(w,w') \in R$$

4 Trabalho Relacionado

Em [Woo00], apresenta-se uma lógica chamada LORA (*Logic of Rational Agents*), contendo uma representação temporal, que permite a representação da dinâmica de como os agentes e os seus ambientes se modificam através do tempo e a representação de ações que os agentes executam e os efeitos dessas ações. LORA pode ser usada para capturar muitos componentes de uma teoria de agentes racionais, incluindo noções de comunicação e cooperação. Esse trabalho lida com sistemas fechados, ou seja, não há a preocupação de se lidar com a entrada e a saída de agentes no sistema.

5 Conclusões e Trabalho Futuro

Este texto apresentou uma modelagem em lógica modal para um MAS aberto, modelando o conjunto de agentes presentes e a entrada e a saída de agentes do sistema. Com isso, é possível formular propriedades para o comportamento dos agentes no sistema, que podem ser desejáveis ou não.

Como trabalho futuro, pretende-se analisar mais axiomas da lógica modal que possam descrever comportamentos de MAS abertos. Deseja-se também implementar uma ferramenta CAV (*Computer Aided Validation*), onde se entraria um modelo de um MAS aberto. Então, o validador analizaria fórmulas descrevendo propriedades de encontro ao modelo verificando se o mesmo satisfaz as propriedades desejadas.

Referências

[Cla99] CLARKE, E. M.; JR., G. O. ; PELED, D. A.. **Model Checking**. MIT Press, 1999.

[Che80] CHELLAS, B.. **Modal Logic: An Introduction**. Cambridge University Press, Cambridge, England, 1980.

[Dal97] VAN DALEN, D.. **Logic and Structure**. Springer, third edition, 1997.

[End02] ENDERTON, H.. **A Mathematical Introduction to Logic**. Academic Press, second edition, 2002.

[Gol92] GOLDBLATT, R.. **Logics of Time and Computation, Lecture Notes No. 7**. CSLI, second edition, 1992.

[Hug96] HUGHES, G. E.; CRESSWELL, M. J.. **A New Introduction to Modal Logic**. Routledge, 1996.

[Jen01] JENNINGS, N. R.. **An agent-based approach for building complex software systems**. Communications of the ACM, 44(4):35–41, 2001.

[Ram04] RAMCHURA, S. D.; HUYNH, T. D. ; JENNINGS, N. R.. **Trust in multi-agent** systems. The Knowledge Engineering Review, 19(1), March 2004.

[Woo00] WOOLDRIDGE, M.. **Reasoning About Rational Agents**. The MIT Press, Cambridge, Massachusetts, 2000.

[Woo02] WOOLDRIDGE, M.. **An Introduction to MultiAgent Systems**. Wiley, Liverpool, UK, 2002.