

3 Trabalhos Relacionados

Este capítulo apresenta, em linhas gerais, dois trabalhos relacionados que contribuíram de alguma forma para a concepção e desenvolvimento do método de verificação de sistemas multiagentes apresentado no capítulo 4.

O primeiro deles, denominado “*equation-free macroscopic analysis*” – ou “análise macroscópica livre de equações”, em tradução livre –, foi inicialmente proposto por Kevrekidis em (Kevrekidis, 2003). Tom DeWolf contribuiu de forma significativa para o estado da arte ao realizar uma extensa análise conceitual e prática desse método como parte de sua tese de doutorado (De Wolf, 2007).

O segundo trabalho revisado foi publicado por Maíra Gatti (Gatti, 2006) como monografia em Ciência da Computação, mais especificamente fruto de um trabalho realizado durante a disciplina de “Aprendizagem de Máquina” desta universidade. Esse trabalho procurou efetuar um estudo preliminar da aplicabilidade de planejadores para a verificação autonômica de sistemas multiagentes auto-organizáveis, o que acabou tornando-se o principal fator motivador para a concepção da solução final proposta neste trabalho.

3.1 Análise macroscópica livre de equações

A verificação de sistemas multiagentes é o processo de análise do comportamento macroscópico exibido por esse sistema quando em execução. O seu principal objetivo é garantir que o sistema não exibirá um comportamento emergente indesejado que o faça se comportar de maneira caótica. Sendo assim, esse processo de verificação procura identificar se o sistema é ou não capaz de cumprir os objetivos para os quais foi projetado. Essa verificação é imprescindível para que a aplicação de sistemas multiagentes em contextos que envolvam qualquer tipo de risco – como em indústrias, hospitais, bancos, etc – seja factível.

Entretanto, garantir o comportamento de um sistema auto-organizável em um ambiente altamente dinâmico e adaptativo não é uma tarefa trivial. Nos métodos tradicionais (figura 3.1, lado esquerdo), equações evolucionárias

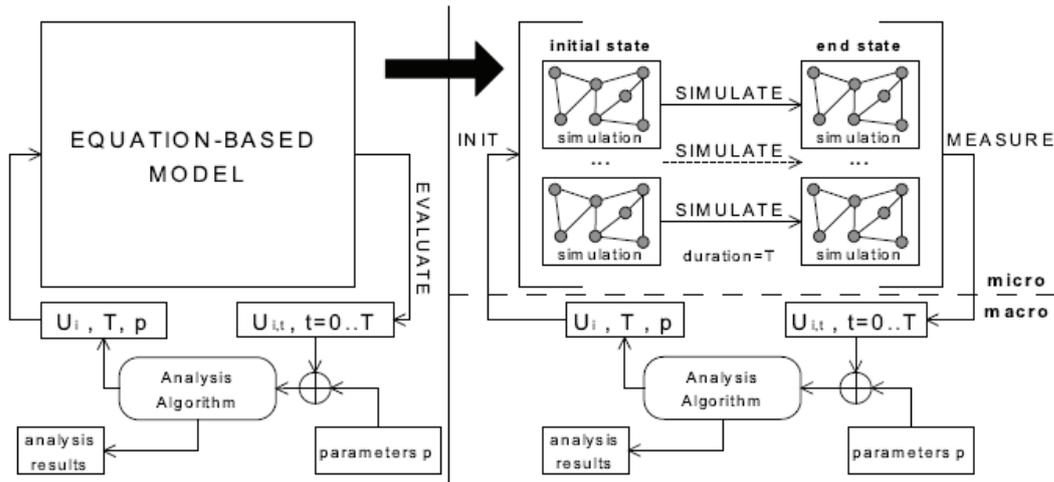


Figura 3.1: Métodos tradicionais vs. análise macroscópica livre de equações (De Wolf, 2007)

representando matematicamente a evolução macroscópica do sistema são derivadas a partir do seu nível microscópico. Considerando algumas condições iniciais U_i , a quantidade de intervalos discretos de tempo T a serem avaliados e os valores de parâmetros p , o modelo matemático é repetidamente avaliado, obtendo resultados para todos os intervalos de tempo pré-determinados. Então, de posse de uma gama de algoritmos de análise numérica, ele verifica o comportamento dessas equações e determina novas condições iniciais para a próxima iteração da verificação. O procedimento é repetido até que sejam obtidas conclusões satisfatórias a respeito do comportamento evolucionário do sistema modelado.

O problema é que derivar equações evolucionárias que representem fielmente um sistema multiagente e, ainda, provar formalmente a sua correteza é algo completamente inviável (De Wolf, 2007). Os modelos de interação desses sistemas podem ser tão complexos que muitas vezes suas representações matemáticas são consideradas incompletas. Segundo (Brueckner, 2000), os sistemas atuais são muito complexos para serem modelados formalmente com o nível de detalhamento necessário. Sendo assim, enumerar todas as possibilidades de comportamento existentes em um sistema multiagente não é apenas difícil, mas geralmente impossível (Wegner, 1997). Essa dificuldade é ainda mais relevante quando consideramos que o especialista em sistemas, na maioria das vezes, não domina técnicas avançadas de programação matemática.

A alternativa para esse problema poderia ser a utilização de métodos empíricos para a verificação do comportamento emergente de um sistema multiagente. Tipicamente, isso seria realizado através da execução de simulações experimentais desse sistema e da consequente observação estatística

das tendências de alteração das suas variáveis macroscópicas. Porém, para que os dados coletados sejam confiáveis, são necessárias várias execuções da simulação, com diferentes parâmetros de entrada, o que representa um alto custo computacional.

Para resolver esse problema, (De Wolf, 2007) avalia um método proposto em (Kevrekidis, 2003) que, ao mesmo tempo em que procura eliminar a necessidade de se modelar matematicamente o sistema, também desobriga a execução de simulações experimentais completas. Simplificando os métodos tradicionais, a abordagem denominada “*equation-free macroscopic analysis*” (figura 3.1, lado direito) – ou simplesmente “análise macroscópica livre de equações”, em tradução livre –, substitui a utilização dos modelos formais por pequenas simulações experimentais no micro-nível do sistema, considerando diversas configurações e parâmetros de entrada. Os resultados das simulações são, então, processados utilizando as mesmas técnicas de análise macroscópica aplicadas nos métodos tradicionais de verificação.

É visível que essa última abordagem simplifica, de certa forma, os métodos tradicionais. Entretanto, uma dificuldade ainda persiste: a avaliação dos resultados das simulações continua sendo realizada através de algoritmos de análise numérica. Além disso, equações matemáticas não são capazes de descrever comportamentos evolucionários que não sejam constantes e suaves (De Wolf, 2007), o que é um problema para a sua utilização em sistemas auto-organizáveis bastante dinâmicos. Como o método proposto por Kevrekidis aproveita os algoritmos de análise numérica das abordagens tradicionais, ele continua sujeito à essa mesma restrição.

3.2 Verificação autônômica via planejador

A revisão da bibliografia recente que aborda as estratégias de verificação de sistemas multiagentes evidencia que o conceito de planejadores não tem sido muito utilizado para verificar a viabilidade de sistemas auto-organizáveis, apesar de parecer adequado para tal. Isso pode ser justificado pelas características intrínsecas à esses sistemas, que muitas vezes possui inúmeras possibilidades de interação entre os seus agentes, o que impede o mapeamento de todos os estados existentes. Além disso, o surgimento de comportamentos não previstos (emergentes) complica ainda mais a tentativa de se enumerar os estados de um sistema auto-organizável.

Em problemas não determinísticos, estocásticos e dinâmicos, onde não se conhece todos os estados alcançáveis, utilizar estratégias de busca *offline* (planejamento clássico) é insuficiente e inadequado (Gatti, 2006). De fato, a

literatura tem demonstrado que é praticamente impossível enumerar todos os estados resultantes da interação entre os agentes de um sistema auto-organizável complexo (Brueckner, 2000, Wegner, 1997).

Por esse motivo, a única forma viável para se aplicar planejadores no processo de verificação de sistemas multiagentes é adotar o conceito de planejadores em tempo-real (*online*), onde não existe a necessidade de se conhecer de antemão todos os estados do sistema. Nesse caso, o planejador intercala a execução de ações com a exploração dos estados, isto é, os estados vão sendo visitados e explorados à medida que as ações executadas modificam o sistema.

Entretanto, ao contrário dos planejadores clássicos onde é possível descobrir com certa facilidade o caminho de ações que levam o sistema ao seu estado final (objetivo), os planejadores em tempo-real precisam estimar o custo de um determinado estado até o objetivo e, então, tentar escolher um caminho para ser explorado. Existem vários algoritmos para desempenhar essa função, dentre os quais destacam-se o A^* e o LRTA* (“*Learning Real-Time A**”, ou “aprendizagem em tempo-real A^* ”, em livre tradução) (Sadikov, 2008). Este último, inclusive, possui algumas variações que prometem uma convergência mais rápida ao estado objetivo. (Furcy, 2004).

Todavia, a eficiência de todos esses algoritmos dependem da elaboração de uma função heurística que seja capaz de estimar com certa precisão o custo para se atingir o estado objetivo a partir de um determinado estado do problema. Quanto mais precisa for essa estimativa, mais rápido o planejador será capaz de encontrar o estado procurado. Na prática, isso faz com que, para cada domínio de problema diferente, seja necessário propor uma nova função, pois dificilmente uma mesma estimativa heurística pode ser aplicada em dois problemas completamente distintos.

Em (Gatti, 2006), a autora avalia a aplicabilidade de planejadores em tempo-real para a verificação de sistemas multiagentes, através de uma abordagem autônoma. A proposta inicial baseou-se no PLANSIM (Liporace, 2005), um *framework* para o desenvolvimento de planejadores especializados implementado em C++. Ele define uma arquitetura para planejadores que utilizem busca heurística direta como mecanismo de planejamento e simuladores de eventos discretos para a modelagem do sistema. Além disso, provê diversas estratégias de busca pré-definidas, inclusive a implementação padrão do algoritmo LRTA*.

O primeiro problema identificado nesse trabalho foi a obrigatoriedade de que cada ação executada possuísse uma ação reversa, para que, caso necessário, o planejador fosse capaz de reverter o estado da simulação. O segundo foi

a observação de que ações externas também devem ser monitoradas pelo planejador, pois também são capazes de modificar o estado da simulação. De fato, ambas as observações são muito importantes – elas, inclusive, são pontos essenciais do método de verificação proposto neste trabalho, como será visto no capítulo 4.

Finalmente, a autora conclui que a tentativa de aplicação do PLANSIM para a verificação de sistemas multiagentes é uma boa ideia, mas inviável devido às características de desenvolvimento do *framework*. Segundo ela, a melhor alternativa seria evoluir a proposta original do PLASIM, introduzindo o conceito de simulador “caixa-branca”, isto é, o planejador deve ser capaz de analisar as estruturas internas do sistema (agentes), e não somente possuir uma visão generalizada da simulação como uma “caixa-preta”.