6
Conclusões

Este trabalho apresentou um método para a verificação experimental de sistemas multiagentes auto–organizáveis. A principal motivação para a concepção e elaboração desse método foi a necessidade de uma abordagem simples e prática para a verificação do comportamento emergente de sistemas multiagentes, uma vez que o estado da arte apresenta principalmente técnicas tradicionais (formais) de difícil aplicação, baseadas em equações evolucionárias e programação matemática.

O grande problema das abordagens formais é a aplicação das mesmas em sistemas altamente dinâmicos e complexos. Os modelos de interatividade desses sistemas podem ser tão complexos que muitas vezes suas representações matemáticas são consideradas incompletas. Sendo assim, enumerar todas as possibilidades de comportamento existentes em um sistema multiagente não é apenas difícil, mas geralmente impossível (Wegner, 1997).

O método proposto foi elaborado como uma derivação da técnica de verificação proposta por Kevrekidis (Kevrekidis, 2003) e avaliada por De Wolf (De Wolf, 2007). Entretanto, ao invés de se utilizar programação matemática para a verificação dos resultados das simulações do sistema, optou–se pela incorporação de um gerenciador autonômico – para possibilitar o ajuste autonômico dos parâmetros de entrada da simulação – e planejadores em tempo real. A decisão de se incorporar planejadores à este processo vem da constatação de que eles são excelentes ferramentas para verificar as propriedades globais de sistemas auto–organizáveis e indicar como decisões locais perturbam o sistema afetando o seu estado global (Gatti, 2006).

No decorrer dos capítulos deste trabalho, foram apresentados todos os passos envolvidos na elaboração desse método de verificação, desde a sua fundamentação teórica (capítulo 2) e concepção (capítulo 4) até a sua aplicação prática em um estudo de caso que modelou um sistema de transporte automatizado de cargas (capítulo 5).

Um dos objetivos estabelecidos no inicio do desenvolvimento deste trabalho foi que a utilização deste método de verificação deveria ser simples. De fato, a sua aplicação prática não exigiu, em momento algum, a elaboração

PUC-Rio - Certificação Digital Nº 0711263/CA

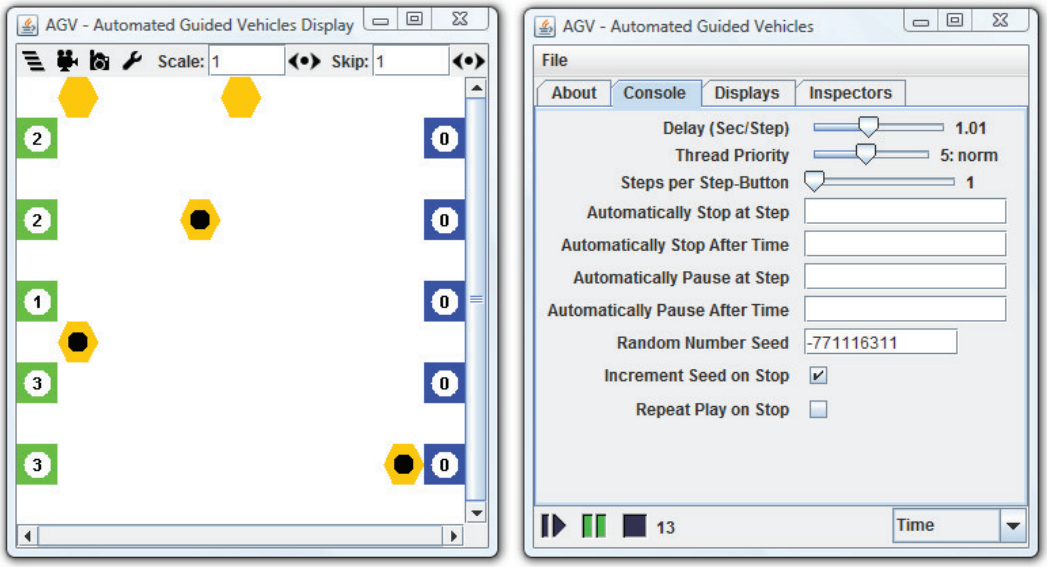

Figura 6.1: Interface gráfica para observação e controle da simulação

de equações evolucionárias complexas ou noções de programação matemática. Toda a instanciação do *framework* foi realizada de forma bastante intuitiva, através de explicações textuais, elaboração de algoritmos em alto–nível, construção de diagramas e, finalmente, codificação do sistema.

A execução das simulações também ocorreu de maneira simples. Para facilitar a análise subjetiva inicial do comportamento do sistema – que precedeu a sua verificação experimental –, utilizou–se uma interface gráfica construída especificamente para observar as interações entre os agentes do sistema (figura 6.1). Essa interface também oferece recursos avançados para o controle do fluxo de execução. Um exemplo disso é a possibilidade de se pausar a simulação e avançar apenas um passo de cada vez, conforme desejado. Esse recurso é extremamente interessante para que o especialista possa validar informalmente o comportamento aparente do sistema proposto.

Já com relação aos resultados obtidos através das simulações realizadas em série – com o objetivo de coletar dados para uma análise posterior – eles demonstraram a aplicabilidade do *framework* proposto para a modelagem e verificação de um sistema que busca solucionar um domínio de problema real. Além disso, foi possível identificar os aspectos de auto–organização exibidos pelo sistema modelado, bem como inspecionar as interferências realizadas pelo verificador com o objetivo de induzir a simulação para um caminho específico.

Esses resultados são um grande motivador para a continuidade deste trabalho. Entretanto, ainda é necessário que se desenvolva um estudo completo e extenso sobre a aplicabilidade deste método de verificação em outros domínios de problema, para formalizar a afirmação de que ele não se trata de uma solução

PUC-Rio - Certificação Digital Nº 0711263/CA

específica, e sim uma abordagem genérica.

Do ponto de vista da implementação realizada, o que se percebeu através das execuções do estudo de caso foi que, em alguns casos específicos, o subsistema central do *framework* consumiu muita memória. Um primeiro palpite antes de uma investigação mais profunda, leva a crer que esse comportamento foi causado pela instanciação exagerada e repetida de objetos em algumas partes do código. Isso poderia ser resolvido com a adoção de estratégias distribuídas para a execução das simulações, como já se verificou em alguns estudos e testes preliminares. Entretanto, ainda se faz necessário um trabalho futuro que realize um estudo mais aprofundado sobre o desempenho do *framework* implementado e proponha possíveis alternativas para melhorá–lo.

PUC-Rio - Certificação Digital Nº 0711263/CA