

7 Conclusões e Trabalhos Futuros

A dificuldade em projetar e implementar requisitos referentes à tolerância a falhas em sistemas multiagentes abertos governados por leis aumenta à medida que a complexidade desse tipo de sistemas aumenta. Considerando ainda sistemas multiagentes de larga escala, este problema se agrava ainda mais. E isto ocorre, principalmente por não ser possível obter nenhum ponto de controle sobre tais sistemas.

Para solucionar o problema de tolerância a falhas em sistemas multiagentes, a técnica de Replicação de Agentes é usada. Esta técnica replica os agentes de acordo com informações que identifiquem suas criticalidades.

Desta forma, buscou-se com este trabalho a extensão de um modelo conceitual que considere a variável criticalidade dos agentes na especificação de sistemas multiagentes abertos governados por leis e uma forma de documentar o *rationale* em torno do projeto e especificação desta variável. Além disso, foi proposta a implementação de uma arquitetura para desenvolvimento de SMAs abertos governados por leis tolerantes a falhas e uma forma de implementar tais mecanismos de forma simples, modular e com reuso.

Dentre os objetivos deste trabalho, um deles foi prover uma forma de documentar e derivar as leis utilizando o modelo conceitual provido pela linguagem declarativa de especificação de leis, chamada XMLaw, através dos requisitos do sistema apoiando a fase de especificação e projeto das leis. E o outro foi prover uma plataforma integrada entre o DimaX, que é um *framework* dinâmico e adaptativo para replicação de agentes, e o M-Law, que é um *framework* para imposição de regras de interação entre agentes, provendo um novo *framework* de monitoramento de criticalidade de agentes baseado em leis.

Com o novo modelo conceitual definido, obteve-se uma análise da criticalidade dos agentes feita pelo DimaX mais refinada através dos elementos presentes no monitoramento de regras de interação entre agentes feita pelo

XMLaw. Para isto foi necessário estender o XMLaw com o objetivo de acrescentar o conceito de criticalidade de agente utilizado pelo DimaX.

O elemento de análise de criticalidade integrou os mecanismos de tolerância a falhas (DimaX) com a abordagem de leis para imposição de regras de interação entre agentes (XMLaw), relacionando um evento do *framework* de imposição com um peso utilizado para o cálculo da criticalidade. Para a especificação da análise de criticalidade, em particular do peso em si, a técnica de Casos de Fidedignidade foi adaptada para o contexto das leis, derivando uma nova abordagem chamada Casos de Leis.

Casos de leis conjugam a aplicação de casos de uso e casos de fidedignidade. A abordagem de casos de uso é útil para mantermos uma visão geral sobre os requisitos que são identificados e casos de fidedignidade são úteis para registrar o detalhamento da análise.

Em nossos experimentos, o uso de análise de risco foi utilizado para determinar valores referentes ao nível de variação no monitoramento da importância de agentes.

De forma sintetizada este trabalho apresentou três contribuições principais para alcançar seus objetivos: as extensões realizadas no M-Law e DimaX a fim de integrá-los, a extensão da linguagem XMLaw na qual o M-Law se baseia com os novos elementos para monitoramento da criticalidade dos agentes, e uma abordagem de documentação e modelagem das leis baseada em um modelo conceitual adaptado da proposta original de Casos de Fidedignidade.

Para validar a arquitetura e abordagem propostas, foram desenvolvidos dois estudos de casos. Como resultado, foram apresentadas planilhas de avaliação das especificações derivadas para o monitoramento da criticalidade, ilustrando o comportamento resultante do monitoramento com uma análise do reflexo da especificação sobre o resultado.

É importante ressaltar que a abordagem proposta neste trabalho não garante que o sistema será fidedigno caso seja gerada uma especificação inadequada para o monitoramento. Assim, ainda que o processo de análise e projeto da especificação tenha sido realizado conforme a abordagem de Casos de Leis proposta, e se o analista não pensou em todos os requisitos não funcionais do sistema necessários para a tolerância a falhas do mesmo, a abordagem não garante a completude da tolerância.

7.1. Trabalhos Futuros

Considerando os trabalhos apresentados no capítulo 2, um trabalho futuro identificado seria utilizar as estratégias relacionadas nos estudos de casos apresentados nesta dissertação com o objetivo de comparar os resultados obtidos para cada estratégia e verificar o quanto o grau de granularidade obtido com as leis de interação contribuíram para as estratégias de estimativa de criticalidade dos agentes.

Um segundo trabalho futuro refere-se ao desempenho da abordagem proposta. Poderia-se calcular o impacto da performance na execução de sistemas multiagentes abertos governados por leis com ou sem o mecanismo de tolerância a falhas em execução. Este trabalho teria como objetivo avaliar em quais situações o módulo de análise de criticalidade deveria ser incluído e em quais não deveria. Ou ainda, quais eventos do sistema deveriam ser monitorados sem impactar a performance do sistema e quais não deveriam ser.

Além disso, tendo como base a linha de monitoramento de sistemas críticos e evolução em tempo de execução, um outro trabalho futuro seria prover uma ferramenta de monitoramento visual da variação da criticalidade dos agentes e, a partir de uma análise deste monitoramento, alterar em tempo de execução a especificação do módulo de análise de criticalidade para que, sistemas críticos que não podem parar, por exemplo, pudessem ter seu mecanismo de monitoramento da criticalidade evoluídos. Esta necessidade foi verificada a partir dos estudos do resultado da variação durante e após a execução do sistema, quando se pode perceber pontos da especificação que não alcançavam os resultados esperados (isto é, número de réplicas esperado) considerando o não determinismo da execução dos protocolos.

Por último, foi percebido que para tornar o mecanismo mais adaptativo e dinâmico, um estudo sobre a possibilidade de armazenar um histórico com os eventos que impactariam mais o sistema do ponto de vista de falhas poderia ser relevante para uma posterior consulta. Esse histórico poderia ser consultado em tempo de execução para determinar *a priori* os eventos antes que os mesmos ocorressem em cenários determinísticos (através da especificação das cenas).

Como um exemplo para esta situação, poderíamos notar que um determinado evento relacionado a um agente não foi considerado um potencial impacto para o sistema, porém, ao longo da execução do mesmo, diversas falhas ocorreram. A cada falha de agente relacionada a este evento, o sistema armazenaria e na n -ésima instância do protocolo que lança este evento (onde n é detectado no momento em que o sistema “percebe” que o evento é importante para a análise da criticalidade), antes mesmo que o protocolo fosse executado, existiria uma adaptação e evolução dinâmica da análise de criticalidade especificada *a priori*, incluindo uma variação da criticalidade para o evento em particular de um agente desempenhando um determinado papel. Porém, para validar essa adaptação, uma análise de performance deverá ser feita a fim de avaliar o trabalho resultante como viável.