

Plano de Estudos

MAC-5701 - Tópicos em Ciência da Computação

Título: Um estudo sobre refinamentos formais de agentes móveis

Aluno: David Paulo Pereira

Orientadora: Ana Cristina Vieira de Melo

1 - Motivação

Atualmente, existem inúmeras ferramentas no mercado com o objetivo de minimizar o esforço no desenvolvimento de software, otimizando o trabalho nas diversas etapas do processo tais como o design, a codificação e o teste. Algumas destas ferramentas visam especificamente diminuir o número de linhas que um programador deve escrever, através de geração automática ou semi-automática (assistida) de código com base em parâmetros, configurações ou modelos previamente definidos. Infelizmente, estas ferramentas ainda possuem uma série de limitações que dizem respeito à qualidade e abrangência do código gerado. Geralmente, o resultado final consiste apenas de uma estrutura básica construída sob os rígidos padrões da arquitetura tecnológica escolhida, ficando pendente a definição do comportamento interno de cada componente do sistema, ainda que a estrutura e os relacionamentos entre eles estejam perfeitamente definidos. Um ponto crucial onde as ferramentas de geração automática de código falham é na garantia da corretude do código gerado que deve atender ao comportamento originalmente especificado.

É nesse contexto de aprimoramento dos modelos e ferramentas de engenharia de software, visando sempre uma melhor qualidade e agilidade, que um domínio particularmente complexo vem sendo alvo de estudos: sistemas constituídos por agentes autônomos que trabalham de forma concorrente, comunicando-se entre si através de canais de comunicação que são criados e removidos dinamicamente. Especificações formais, denominadas na literatura como as álgebras de processos procuram caracterizar de forma matemática a estrutura e o comportamento deste tipo de sistema, definindo de forma abstrata e formal as entidades, as operações e as regras de equivalência permitidas. Uma das especificações mais estudadas na caracterização da comunicação entre os agentes móveis é chamada de *pi-calculus*, que acrescenta a mobilidade às características de concorrência e de comunicação existentes na álgebra de processos concorrentes.

Um dos benefícios em se entender e trabalhar com esse tipo de especificação é a garantia de que o sistema possui um comportamento que pode ser previsto, observado e provado por meio de verificações formais, garantindo a correta execução nos mais diversos contextos ou situações. Desta forma, algumas ferramentas de geração automática de código estão sendo desenvolvidas no qual sistemas constituídos por agentes móveis são especificados em *pi-calculus* e posteriormente são traduzidos para uma linguagem de programação de alto nível (por exemplo, o Java). O código correspondente à comunicação externa de cada agente no sistema é gerado automaticamente, restando ao programador a codificação manual do comportamento interno de cada um. Como a teoria do *pi-calculus* apenas garante a corretude em termos de comunicação entre os agentes, o fornecimento adicional e necessário de código para a caracterização de comportamento

interno pelo programador pode eliminar esta garantia de conformidade entre a especificação e a implementação.

Por outro lado, a transformação de uma especificação abstrata em uma implementação correspondente é algo bastante complicado. Por exemplo, na teoria de *pi-calculus*, existe um conceito puramente lógico de canal de comunicação. Entretanto, sabemos que na construção dos sistemas, estes canais de comunicação podem ser implementados das mais distintas formas, como por exemplo, mensagens assíncronas sobre um determinado protocolo ou acesso de informação sobre uma base de dados compartilhada. Existem teorias baseadas em lógica que permitem descrever o comportamento interno de forma bastante abstrata e descer na escala de abstração para níveis mais concretos através de transformações. Estas transformações são chamadas de *refinamentos* e são baseadas em relações de equivalência, garantindo formalmente que em cada etapa de transformação, o comportamento esperado é mantido.

2 - Plano de Estudos

O objetivo deste projeto é o estudo da forma com que as teorias de refinamento e álgebra de processos podem ser utilizadas de modo integrado, complementando as vantagens proporcionadas por cada uma no intuito de elaborar ferramentas de geração semi-automática de código mais robustas no domínio dos agentes móveis. É claro que o intuito não é a criação de ferramentas para a geração perfeita de código e sim para a minimização do esforço nas atividades de teste, pois algumas propriedades do sistema são facilmente provadas com base nas teorias da álgebra de processos e no refinamento. Os passos seguintes estão organizados de forma a se estudar o extenso material bibliográfico existente tanto da teoria de *pi-calculus* como das diversas teorias existentes de refinamento tentando encontrar pontos de possível intersecção, onde regras de equivalência e congruência sejam respeitadas com a aplicação de conceitos de uma teoria em outra.

3 – Referências

[1] R. Milner. Communicating and Mobile Systems: the π -Calculus. Cambridge University Press, Maio 1999.

[2] R. Milner, J. Parrow e D. Walker. A calculus of mobile processes. Parts I/II. Journal of Information and Computation, 100:1-77, Set. 1992.

[3] R. Milner, Functions as Processes. Mathematical Structure in Computer Science 2(2), pp.119--146, 1992.

[4] R. Milner, J. Parrow, e D. Walker. Modal Logics for Mobile Processes. Theoretical Computer Science, Vol. 114, pp. 149-171, 1993.

[5] J. Woodcock, A. Cavalcanti. A Concurrent Language for Refinement. IWFWM 2001.

[6] A. Cavalcanti, J. Woodcock. ZRC - A Refinement Calculus for Z. *Formal Asp. Comput.* 10(3): 267-289 (1998).

[7] A. Sampaio, J. Woodcock, A. Cavalcanti: Refinement in Circus. *FME 2002*: 451-470.

[8] A. Cavalcanti, A. Sampaio. From CSP-OZ to Java with Processes. *IPDPS 2002*.