Universidade Federal de Ouro Preto - UFOP
Instituto de Ciências Exatas e Biológicas - ICEB
Departamento de Computação - DECOM

JSensor: Uma plataforma paralela e distribuía para
simulações de redes de sensores

Aluno: Danniell Hugo Ribeiro
Matricula: 08.1.4135

Orientador: Joubert de Castro Lima

Ouro Preto
29 de março de 2012

Universidade Federal de Ouro Preto - UFOP
Instituto de Ciências Exatas e Biológicas - ICEB
Departamento de Computação - DECOM

JSensor: Uma plataforma paralela e distribuída para
simulações de redes de sensores

Proposta de monografia apresentada ao curso de Bacharelado em Ciência da Computação, Universidade Federal de Ouro Preto, como requisito parcial para a conclusão da disciplina Monografia II (BCC391).

Aluno: Danniel Hugo Ribeiro
Matricula: 08.1.4135

Orientador: Joubert de Castro Lima

Ouro Preto
29 de março de 2012

Resumo

Apresentamos uma proposta de desenvolvimento de um simulador algoritmos de redes de sensores que opere de forma paralela e distribuída. Esta terá por finalidade auxiliar na validação de modelos de pesquisas na área de redes de sensores diminuindo o tempo necessário para simular, por exemplo, cenários onde tem-se milhares de nós.

Palavras-chave: Simuladores de Redes. Redes de Sensores. Algoritmos Paralelos. Algoritmos Distribuídos.

Sumário

1	Introdução	1
2	Justificativa	2
3	Objetivos	3
3.1	Objetivo geral	3
3.2	Objetivos específicos	3
4	Metodologia	4
4.1	Tecnologias Utilizadas	4
4.2	Organização do Sistema	4
4.3	Testes	4
5	Cronograma de atividades	5

Lista de Figuras

Lista de Tabelas

1	Cronograma de Atividades primeiro semestre de 2011.	5
2	Cronograma de Atividades primeiro semestre de 2012.	6

1 Introdução

O mundo ao nosso redor possui uma variedade de fenômenos que podem ser descritos por algumas grandezas, como temperatura, pressão e umidade, que podem ser monitorados por dispositivos com poder de sensoriamento, processamento e comunicação. O conjunto desses dispositivos é conhecido na literatura como Rede de Sensores [6]. Pesquisas nesta área utilizam de simulações síncronas e assíncronas para avaliar o impacto de novos modelos a serem propostos de distribuição, roteamento, comunicação ou redução de dados, nos mais variados tipos de redes. Desta forma, Simuladores como o Network Simulator[2] ou o Sinalgo[4] são geralmente utilizados para esses fins.

No entanto, algumas aplicações podem utilizar milhares de nós e tais simuladores podem não suportar tamanha quantidade, o que inviabiliza a validação de algumas soluções propostas e, conseqüentemente, sua concepção em ambientes reais.

Nesta direção, faz-se necessário a utilização de simuladores que sejam capazes de executar com o menor tempo possível, de forma paralela e distribuída[8][3], simulações em larga escala.

2 Justificativa

A execução deste trabalho fomentará o interesse por questões gerais na área de redes de sensores, sistemas embarcados e automação e algoritmos paralelos. Além da área de redes de sensores, esse projeto abrange as áreas de sistemas distribuídos, computação móvel e todas as áreas relacionadas com redes de computadores. Para o desenvolvimento do mesmo podemos considerar algumas áreas de atuação diretamente correlacionadas:

- *Algoritmos paralelos*: Esta subárea engloba a etapa de soluções para simulação de redes de sensores em paralelo, especificamente o projeto e análise dos algoritmos de simulação existentes no mercado.
- *Algoritmos distribuídos*: Essa subárea engloba a etapa de soluções para simulação de forma distribuída.
- *Otimização*: Essa subárea engloba a etapa de soluções para executar simulações em tempo hábil utilizando ao máximo os recursos disponíveis.

3 Objetivos

3.1 Objetivo geral

- Desenvolver um simulador para redes de sensores que possa executar simulações de forma paralela e distribuída.

3.2 Objetivos específicos

- Elaborar modelos de simulação síncrona e assíncrona que possam ser executados em paralelo e de forma distribuída.
- Desenvolver estratégias que nos permitam executar simulações do tipo síncronas e assíncronas.
- Criar uma interface amigável que permita a visualização dos experimentos em tempo real.
- Executar simulações em larga escala para validar a eficiência do simulador.

4 Metodologia

A metodologia adotada no desenvolvimento do JSensor será:

- Estudar implementações já existentes do funcionamento de uma rede de sensores.
- Levantar requisitos e criar diagramas de classe para a construção do sistema.
- Implementar o sistema utilizando a linguagem de programação Java[5] juntamente com a linguagem de marcação XML[1]
- Realização de testes para validação do simulador.

4.1 Tecnologias Utilizadas

O sistema será desenvolvido utilizando as tecnologias Java para montar o núcleo do sistema. Além disso, será utilizada também a linguagem de marcação XML para auxiliar na manipulação de configurações a serem definidas pelo usuário.

4.2 Organização do Sistema

O sistema será dividido em camadas como descrito a seguir:

- *Núcleo*: O núcleo do sistema é o componente central do simulador. Nele serão implementadas as funções que permitirão que as simulações sejam executadas de forma paralela e distribuída. Esse módulo é totalmente transparente ao usuário final.
- *Interface Gráfica*: A interface gráfica, ou GUI[7], é o componente onde o usuário do sistema pode interagir com os elementos da rede, bem como controlar o fluxo da simulação (parar, criar nós, controlar o tempo de atualização da GUI, etc.).
- *Projetos*: Este módulo permite que o usuário crie seu próprio modelo de simulação, especificando por exemplo o tipo de nó, temporizadores, forma de comunicação e detecção, etc.

4.3 Testes

Após o desenvolvimento terá início a etapa de testes. Nesta, o sistema será submetido a vários tipos de simulações, procurando explorar tanto a parte síncrona quando assíncrona do simulador nos módulos paralelo e distribuídos. Além disso, serão feitos testes variando o número de processadores e de máquinas envolvidos para testar a eficiência do mesmo.

5 Cronograma de atividades

As atividades a serem desenvolvidas são:

1. Estudo dos simuladores presentes no mercado.
2. Estudo de algoritmos paralelos.
3. Implementação da GUI.
4. Implementação do módulo síncrono paralelo do JSensor.
5. Implementação do módulo assíncrono paralelo do JSensor.
6. Implementação do módulo distribuído do JSensor.
7. Realização de testes.
8. Elaboração do relatório de atividades.
9. Apresentação do relatório de atividades.
10. Elaboração da monografia.
11. Apresentação da monografia para a banca examinadora.
12. Escrita de artigos científicos.
13. Disponibilização do JSensor para a comunidade por meio do SourceForge.

Na Tabela 1, apresenta-se o cronograma de atividades para o primeiro semestre de 2011.

Atv	Mar-2011	Abr-2011	Mai-2011	Jun-2011	Jul-2011
1	X	X			
2	X	X			
3	X	X	X		
4		X	X	X	X
5		X	X	X	X
6					
7			X	X	X
8					
9					
10					
11					
12					

Tabela 1: Cronograma de Atividades primeiro semestre de 2011.

Na Tabela 2, apresenta-se o cronograma de atividades para o primeiro semestre de 2012.

Atv	Mar-2012	Abr-2012	Mai-2012	Jun-2012	Jul-2012
1					
2					
3					
4					
5					
6	X	X	X	X	
7	X	X	X	X	
8	X				
9	X				
10			X	X	
11				X	X
12	X	X	X	X	X

Tabela 2: Cronograma de Atividades primeiro semestre de 2012.

Referências

- [1] Extensible markup language (xml). <http://www.w3.org/XML/>, 2011. Visitado em 29 de março de 2012.
- [2] The ns-3 network simulator. <http://www.nsnam.org/index.html>, 2011. Visitado em 29 de março de 2012.
- [3] Producer-consumer problem. http://en.wikipedia.org/wiki/Producer-consumer_problem, 2011. Visitado em 29 de março de 2012.
- [4] Sinalgo - simulator for network algorithms. <http://www.disco.ethz.ch/projects/sinalgo/>, 2011. Visitado em 29 de março de 2012.
- [5] What is java? http://www.java.com/en/download/whatis_java.jsp, 2011. Visitado em 29 de março de 2012.
- [6] Ian F. Akyildiz, WellJan Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, August 2002.
- [7] Wilbert O. Galitz. *The Essential Guide to User Interface Design*. Wiley, 2 edition, 2002.
- [8] Andrew S. Tanenbaum and Maarte Van Steen. *Distributed Systems: Principles and Paradigms*. Prentice Hall, 2 edition, 2007.