

ENSINO DE COMPUTAÇÃO EM GRID PARA O CURSO DE ENGENHARIA DA COMPUTAÇÃO

Charles Boulhosa Rodamilans¹; Edson Toshimi Midorikawa²

USP - Escola Politécnica, Departamento de Engenharia da Computação e Sistemas Digitais

Av Prof Luciano Gualberto – travessa 3 – 158

CEP : 05508-900, São Paulo, SP

¹charles.rodamilans@poli.usp.br

²edson.midorikawa@poli.usp.br

***Resumo:** Este artigo faz uma análise do estado atual do ensino de uma área emergente que está se difundindo rapidamente na Engenharia da Computação: a Computação em Grid. Vários analistas prevêem que o impacto da aplicação da Computação em Grid na sociedade será semelhante a do surgimento da Internet. Isto porque será possível usar a infra-estrutura mundial para processamento, tornando viável uma série de aplicações que ainda hoje não são suportados pelos mais rápidos supercomputadores. Uma análise mostrou que o ensino deste tópico ainda não é coberto por vários cursos de Engenharia no Brasil e no exterior. Este artigo propõe o oferecimento de uma disciplina de “Computação em Grid” nos cursos de Engenharia da Computação, apresentando os tópicos a serem abordados, bibliografia sugerida e uma metodologia de ensino baseada em simuladores de Grid. O uso de simuladores é detalhado para o ensino de escalonamento de tarefas com a proposição de atividades práticas para a consolidação dos conceitos.*

***Palavras-chave:** Computação em Grid, Ensino, Escalonamento de tarefas, Engenharia da Computação*

1. INTRODUÇÃO

Computação em Grid está relacionada à possibilidade de utilização de recursos (processamento, armazenamento, serviços) distribuídos em diferentes domínios computacionais respeitando as suas políticas administrativas locais, conforme definido por Foster (2001).

Problemas de grande porte em diversas áreas – WCG (2008), BOINC(2008) – como a pesquisa da Aids, distrofia muscular, proteoma humano, dengue e malária – na área da Medicina, dinâmica dos fluídos, colisão de partículas, terremotos, climatologia – da Física, maximização da capacidade nutricional dos alimentos – da Biologia, dentre outros, estão sendo beneficiadas pela infraestrutura em Grid, seja para o processamento quanto para o armazenamento dos dados.

Atualmente, diversos projetos estão relacionados com a criação e manutenção de infraestrutura em Grid. No Brasil, têm-se várias iniciativas como o projeto SPRACE (2008) do Instituto de Física Teórica da Unesp, e o desenvolvimento do GridUNESP (2008), dentre outros. No exterior, têm-se o Grid5000 (2008) da França, o TeraGrid (2008) dos EUA, o World-Wide Grid (2008) da Austrália (e outros países) e outros listados no GridCafe (2008).

Apesar dessa demanda eminente por pessoal capacitado na área da Computação em Grid, poucos cursos de Engenharia da Computação abordam este tópico em seus currículos. O

objetivo deste trabalho é analisar o estado atual do ensino de Computação em Grid em cursos da área de computação. Para isto fez-se um levantamento dos currículos, tópicos abordados e metodologia aplicada em vários cursos do Brasil e do exterior e, ao final, após uma análise crítica, é feita uma proposta da inclusão da disciplina “*Computação em Grid*” no currículo de um curso de Engenharia da Computação.

A seção 2 aborda os currículos de referência existentes atualmente na área de computação. Na seção 3 é descrito e analisado o estado atual do ensino de Computação em Grid no Brasil e no exterior. Em seguida, é apresentada a nossa proposta para a disciplina “Computação de Grid” para o curso de Engenharia da Computação. Na seção 5, é detalhada uma proposta para o ensino do tópico escalonamento de tarefas na disciplina proposta. Ao final são apresentadas as considerações finais sobre o trabalho.

2. CURRÍCULOS DE REFERÊNCIA

Se faz necessário haver uma referência para a criação de currículos na área da Computação. No Brasil, a Sociedade Brasileira de Computação (SBC) é responsável pelo Currículo de Referência (CR) da área de computação, seja a computação para fim quanto para meio. Na proposta do Currículo de Referência do SBC (2005), as propostas das matérias são organizadas em seis núcleos, conforme a Tabela 1. As matérias da área de Computação estão organizadas em:

- **Fundamentos da Computação (F):** envolve a parte científica e as técnicas fundamentais para os diversos cursos de computação
- **Tecnologia da Computação (T):** as matérias estão relacionada ao conhecimento que permite ao aluno elaborar soluções dos problemas nos diversos domínios de aplicação.

Das outras áreas:

- **Matemática (M):** propicia a abstração, modelagem e raciocínio lógico, servindo de base para várias matérias da Área de Computação
- **Ciências Básicas (CB):** oferece o conhecimento de ciências básicas e desenvolve no aluno a habilidade para a aplicação do método científico
- **Eletrônica (E):** fornece os conhecimentos básicos necessários para o projeto de circuitos eletrônicos
- **Contexto Social e Profissional (P):** propicia uma visão humanísticas das questões sociais e profissionais.

Tabela 1 - Estrutura das Matérias

Área de Computação	Outras Áreas
Fundamentos da Computação	Matemática
Tecnologia da Computação	Ciências Básicas
	Eletrônica
	Contexto Social e Profissional

A disciplina Computação em Grid, para a Engenharia da Computação, poderia ser classificada como pertencente ao núcleo de Tecnologia da Computação, entretanto ela não faz parte da proposta CR2005. As matérias que mais se assemelham são *T15 - Programação Paralela* e *T19 - Sistemas Distribuídos*.

O IEEE/ACM Computing Curricula (2005) é responsável pelo Currículo de Referência da área de computação dos EUA. Este também não explicita a área de Grid como tópico relevante. Pectu (2008) sugere que algumas seções do currículo, como *Net Centric Principles*

and Design, Net Centric Use and Configuration, Sistemas Distribuídos e Arquitetura e Organização de Computadores, poderiam incorporar alguns tópicos de um curso específico de Computação em Grid. Esta mesma autora cita, como exemplo, que na disciplina de Sistemas Distribuídos o professor poderia apresentar um tipo de *middleware* de Grid que permite o uso de recursos de terceiros.

3. ENSINO DE COMPUTAÇÃO EM GRID

Com o crescimento da Computação em Grid, aumentou-se também a importância do ensino da Computação em Grid nos cursos de Engenharia da Computação.

No Brasil, o ensino da Computação em Grid está normalmente vinculado em disciplinas¹ de pós-graduação. Algumas universidades incluíram tópicos de Computação em Grid, como por exemplo, a Universidade Federal de Campina Grande – UFCG (2008) que oferece a disciplina “*Sistemas Distribuídos*” e a Universidade Federal do Rio Grande do Sul – UFRGS (2008) com a disciplina “*CMP 157 - Programação Distribuída e Paralela*”. Outras criaram disciplinas que abordam esse tema, como é o caso da Universidade Federal Fluminense – UFF (2008), com a disciplina “*Arquiteturas Paralelas III: Computação em Grids e Clusters*” e o Instituto Mauá de Tecnologia – MAUÁ (2008) com a disciplina “*ECM913 - Computação Grid*”. A Tabela 2 apresenta os tópicos e a bibliografia da disciplina ministrada na UFCG. A metodologia adotada são aulas expositivas introduzindo os conceitos básicos, seguida por seminários de artigos da área, apresentados individualmente pelos alunos, com os temas relacionados a distribuição de conteúdo e sistemas entre-pares, E/S paralela, engenharia de software para sistemas distribuídos e computação global. Os alunos são avaliados a partir da pontuação obtida pelo seminário, resenhas e prova.

Tabela 2 – Currículo da UFCG

Tópicos da disciplina “Sistemas Distribuídos” (UFCG)
1. Introdução aos sistemas distribuídos
2. Modelos de sistemas distribuídos
3. Arquiteturas de sistemas distribuídos
4. Comunicação em sistemas distribuídos.
5. Sincronização em sistemas distribuídos
6. Resultados recentes em sistemas distribuídos
Bibliografia
1. <i>Distributed Systems for System Architects</i> (VERISSIMO, et al., 2001)
2. <i>Distributed Systems</i> (MULLENDER, 1993)
3. <i>Building Secure and Reliable Network Applications</i> (BIRMAN, 1996)
4. <i>Fault Tolerance in Distributed Systems</i> (JALOTE, 1994)

Na Tabela 3 são apresentados os tópicos e a bibliografia da disciplina ministrada na UFRGS. A metodologia adotada são aulas expositivas, trabalhos individuais e em grupos e exercícios. Os alunos também devem apresentar individualmente um trabalho final (oral e escrito) relacionados a um dos temas seguintes: programação de um algoritmo relacionado a um problema concorrente ou distribuído ou paralelo; análise/comparação/síntese de dois ou mais artigos/capítulos de livros; ou instalação/resumo/análise-avaliação de uma ferramenta de Programação Distribuída e Paralela (PDP).

¹ As disciplinas estão nas referências bibliográficas de acordo com o nome das universidades.

Tabela 3 – Currículo da UFRGS

Tópicos da disciplina “Programação Distribuída e Paralela” (UFRGS)
1. Revisão: <ol style="list-style-type: none"> a. Sistemas distribuídos e paralelos; b. Introdução às aplicações distribuídas; c. Expressão da concorrência; d. Comunicação e sincronização.
2. Algoritmos Distribuídos
3. Computação em Grade e Pervasiva
4. Escalonamento em Sistemas Distribuídos
5. Tópicos de Projetos em Sistemas Distribuídos e Paralelos do “Grupo de Processamento Paralelo e Distribuído (GPPD)”
Bibliografia
1. <i>An Introduction to Distributed Algorithms</i> (BARBOSA, 1996)
2. <i>Distributed Algorithms</i> (LYNCH, 1996)
3. <i>Distributed algorithms and protocols</i> (RAYNAL, 1992)
4. <i>A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems</i> (CASABANT, KUHL, 1994)
5. <i>Scheduling Theory and its Applications</i> (CHRÉTIENNE, et al., 1997)
6. <i>Task Scheduling in Parallel and Distributed Systems</i> (EL-REWINI, et al., 1994)
7. <i>The Grid 2: Blueprint for a New Computing Infrastructure</i> (FOSTER, KESSELMAN, 2003)
8. <i>Grid Computing: Making The Global Infrastructure a Reality</i> (BERMAN, et al., 2003)
9. <i>Globus Toolkit 4: Programming Java Services</i> (SOTOMAYOR, et al., 2005)
10. <i>Grid Computing For Developers</i> (SILVA, 2005)

No exterior, diversas universidades já possuem uma disciplina de Computação em Grid nos seus cursos de pós-graduação. Na University of Melbourne (2008), Austrália, a disciplina “433-678: *Cluster and Grid Computing*” é ministrada pelo professor Rajkumar Buyya. Os tópicos da disciplina são apresentados na Tabela 4. O objetivo da disciplina é apresentar os conceitos de Clusters e Grids de maneira aprofundada e utiliza a programação paralela para dar um enfoque mais prático. As aulas são expositivas, havendo recomendação de leitura de artigos e/ou textos técnicos para cada aula. São apresentados os seguintes assuntos: as tendências, tecnologias, aplicações, arquiteturas, gerenciamento de recursos, escalonamento, *resource broker*, segurança e serviço de informação no Grid. Os *middlewares* Globus, GridBus e Aneka são apresentados. Os alunos são avaliados através de exercícios de programação (*Threads* e MPI), projeto final, artigo e prova. O projeto é realizado em grupo sendo necessário apresentar a implementação e um artigo. O artigo consiste no estudo de tecnologias relacionadas a escalonamento.

Tabela 4 – Currículo da University of Melbourne

Tópicos da disciplina “433-678: Cluster and Grid Computing” (MELBOURNE)
1. <i>Foundations</i> <ol style="list-style-type: none"> a. <i>Cluster, Distributed Computing and Grid Computing</i> b. <i>Socket and Thread Programming</i> c. <i>Parallel Computing</i>
2. <i>Cluster Computing</i> <ol style="list-style-type: none"> a. <i>Introduction and System Architecture</i> b. <i>Single System Image</i> c. <i>Parallel Programming Models and Paradgms</i>

d. <i>MPI Programming and Executing MPI using PBS</i>
3. <i>Grid Computing</i>
a. <i>Introduction and Overview of Grid Computing: Trends, Challenges, Technologies, and Applications</i>
b. <i>Grid Resource Management and Grid Economy</i>
c. <i>Enterprise Grids and the Aneka Middleware</i>
d. <i>Nimrod-G Grid Resource Broker</i>
e. <i>Economic Scheduling Algorithms</i>
f. <i>Grid Security Infrastructure</i>
g. <i>Gridbus Broker - An Introduction</i>
h. <i>Globus Components: GRAM, GASS, and GIS</i>
i. <i>Molecular Docking on Grid Using Nimrod-G</i>
Bibliografia
1. <i>High Performance Cluster Computing</i> (BUY YA, 1999)
2. <i>The Grid : Blueprint for a New Computing Infrastructure</i> (FOSTER, et al., 1999)
3. <i>Economic-based Distributed Resource Management and Scheduling for Grid Computing</i> (BUY YA, 2002)

Na University of Alabama at Birmingham (2008), Estados Unidos, a metodologia utilizada na disciplina “CS 633/733 *Grid Computing*” é baseada em aulas expositivas. O currículo é apresentado na Tabela 5. O objetivo da disciplina é apresentar conceitos teóricos de Computação em Grid com uma parte prática compreendendo uma implementação de um ambiente de Computação em Grid. O curso inicia fazendo uma revisão de conceitos Computação distribuída e paralela e, em seguida, são apresentados conceitos de Grid como, aplicações, arquiteturas, tecnologias, segurança, alocação de recursos, gerenciamento de dados, serviços de informações de Grid. Algumas aulas são ministradas baseadas em artigos técnicos. Os alunos são avaliados através provas, exercícios, projeto, apresentação do projeto e artigo. O projeto consiste na criação de um “*Grid Testbed*”, utilizando o Globus Toolkit (2008) e NSF *Middleware Initiative* – NMI (2008), e de um ambiente de Grid e/ou aplicações que utilizem o Testbed desenvolvido pelos próprios alunos.

Tabela 5 – Currículo da University of Alabama at Birmingham

Tópicos da disciplina “CS 633/733 Grid Computing” (ALABAMA)
1. Introduction and Overview
2. Distributed Computing Review
3. Parallel Computing Overview
4. The Evolution of the Grid
5. What is the Grid?
6. The Anatomy of the Grid
7. Computational Grids
8. Grid Security Infrastructure
9. Grid Resource Allocation Management
10. Grid Information Services
11. The Physiology of the Grid
12. Grid Programming Models
13. Java Commodity Toolkit
14. Grid Computing Environments
15. Introduction to Grid Services
16. Grid Services Standards
17. Developing Grid Services
18. Data Grids

19. Semantic Grids
Bibliografia
1. <i>The Grid 2: Blueprint for a New Computing Infrastructure</i> (FOSTER, et al., 2003)
2. <i>Grid Computing: Making The Global Infrastructure a Reality</i> (BERMAN, et al., 2003)
3. <i>Globus Toolkit 4: Programming Java Services</i> (SOTOMAYOR, et al., 2005)

A Universidade do Porto (2008), Portugal, possui a disciplina “*Computação em Grid*” para a pós-graduação. Na Tabela 6 são apresentados os tópicos e a bibliografia utilizados na disciplina. Alguns artigos também são utilizados como suporte. O objetivo da disciplina é ensinar Grades Computacionais e Grades de Dados visando sua implementação e utilização. O *middleware* gLite (2008) é utilizado com base para ensinar conceitos relacionados a Computação em Grid. O curso utiliza a infraestrutura do projeto Grid Infn Laboratory for Dissemination Activities – GILDA (2008). GILDA é um grande ambiente de teste para Grid da União Européia, que possui o intuito de demonstrar e disseminar as capacidades da Computação em Grid e também é utilizado como infraestrutura base para eventos do Enabling Grid for E-science – EGEE (2008). Nas aulas práticas, os alunos aprendem como se certificar no GILDA e como utilizar o gLite. Os alunos são avaliados através de uma prova e um trabalho. O trabalho consiste na implementação de uma solução distribuída para o problema de “*floor design*” descrito em Foster (1995), utilizando o *middleware* gLite.

Tabela 6 – Currículo da Universidade de Porto

Tópicos da disciplina “Computação em Grid” (PORTO)
Aulas Teóricas
1. Introdução (Cluster e Grid)
2. gLite e o Projecto EGEE
3. Grid Architecture and Globus
4. Grid Application Description Languages
5. Authentication and Authorization in gLite
6. Grid Systems and scheduling
7. gLite: Workload Management System
8. gLite: Data Management System
Aulas Práticas
1. Obtenção do certificado GILDA, download do GBB
2. Tópicos sobre segurança no gLite
3. Sistema de Informação do gLite
4. Basic job submission and basic JDL
5. Basic Data Management
Bibliografia
1. <i>Grid Computing: Making The Global Infrastructure a Reality</i> (BERMAN, et al., 2003)
2. <i>The Grid 2: Blueprint for a New Computing Infrastructure</i> (FOSTER, KESSELMAN, 2003)

Uma análise do levantamento realizado nas várias universidades estudadas com relação à metodologia aplicada nas disciplinas, contida na , podemos observar que todas as disciplinas possuem aulas expositivas lecionadas pelo professor. Também, a maioria exigem um projeto final prático com sua respectiva apresentação, com exceção UFCG. Todas as universidades exigem dos alunos um trabalho final escrito, seja pela forma de artigo, resenha ou relatório.

Tabela 7 – Metodologia

Metodologia	Brasil		Ásia	Amer. Norte	Europa
	UFCG	UFRGS	MELBOURNE	ALABAMA	PORTO
Tipo de Aula					
Expositiva	X	X	X	X	X
Prática					X
Avaliação dos Alunos					
Seminários	X				
Exercícios		X	X	X	
Prova	X			X	X
Projeto/Apresentação		X	X	X	X
Artigo/Resenha/Relatório	X	X	X	X	X

Além das disciplinas mencionadas, existem as Escolas de Grid - como IWSGC (2008), ISSGC (2008), FIGS (2008), COREGRID (2008) e outras listadas em ICEAGE (2008a) que são cursos de curta duração oferecidas para pesquisadores do mundo inteiro. Podemos citar também a existência de programas de mestrado em Computação em Grid, como o da University of Westminster (2008) e os listados em ICEAGE (2008b).

4. PROPOSTA DE UMA DISCIPLINA

Nesta seção, iremos apresentar uma proposta de uma disciplina “Computação em Grid” para o curso de Engenharia da Computação.

De acordo com o levantamento realizado, verificamos que as disciplinas do exterior demonstram uma motivação de desenvolver e utilizar Computação em Grid, assim como sua evolução. Também apresentam as tecnologias e arquitetura, utilizando *middlewares* para demonstrar os conceitos. Essas disciplinas conseguem oferecer um grande aprofundamento (teórico e prático). No Brasil, as universidades analisadas não possuem uma disciplina específica, e os conceitos são apresentados em alguns tópicos. Assim, os trabalhos práticos realizados no Brasil não estão diretamente relacionados a Computação em Grid, e alguns conceitos podem não ser abordados. Na Tabela 8, destacamos os tópicos que são abordados pela maioria das universidades e que consideramos relevantes para a disciplina proposta.

As disciplinas (no Brasil e no Exterior) possuem aulas expositivas lecionadas pelo professor responsável e trabalhos práticos realizados pelos alunos. A UFCG é única a ter seminários de artigos apresentados pelos alunos. Entretanto, as aulas de todas as disciplinas são baseadas em artigos, possuem uma bibliografia relacionada a Computação em Grid e não adotam um livro-texto. A partir dos artigos utilizados nas disciplinas, brasileiras e estrangeiras, dos tópicos fundamentais, e das nossas experiências, propomos os artigos e bibliografia que estão contidos na Tabela 8.

A metodologia a ser utilizada são aulas expositivas lecionadas pelo professor, e os alunos são avaliados através de pequenos exercícios práticos (p.ex.: implementação de um algoritmo simples de escalonamento de tarefas), do desenvolvimento de um projeto (implementação/implantação) com sua respectiva apresentação oral e monografia ou artigo, do resumo de artigos e da prova conceitual dos assuntos ministrados na disciplina.

Tabela 8 – Disciplina “Computação em Grid”
proposta para o Curso de Engenharia da Computação

#	Tópicos para a disciplina “Computação em Grid”	Referências / Artigos
1.	Revisão sistemas distribuídos e paralelos	<ul style="list-style-type: none"> • Utilizar Bibliografia
2.	Modelos e Paradigmas de Computação Paralela	<ul style="list-style-type: none"> • Utilizar Bibliografia
3.	Introdução sobre Computação em Grid (Conceitos, Tendências e Desafios)	<ul style="list-style-type: none"> • <i>What is the Grid?</i> • <i>The Anatomy of the Grid: Enabling Scalable Virtual Organizations</i> • <i>Computational Grids</i> • <i>The evolution of the Grid</i>
4.	Arquitetura	<ul style="list-style-type: none"> • <i>The Physiology of the Grid</i> • <i>Global Grids and Software Toolkits: A Study of Four Grid Middleware Technologies</i>
5.	Aplicações e Tecnologias	<ul style="list-style-type: none"> • <i>Grids and Grid Technologies for Wide-Area Distributed Computing,</i>
6.	Segurança	<ul style="list-style-type: none"> • <i>A Security Architecture for Computational Grids</i>
7.	Serviços de Informação	<ul style="list-style-type: none"> • <i>Grid Information Services for Distributed Resource Sharing</i>
8.	Gerenciamento de Alocação Recursos	<ul style="list-style-type: none"> • <i>A Taxonomy and Survey of Grid Resource Management Systems</i> • <i>A General Architecture for Scheduling on the Grid</i> • <i>Ten Actions When SuperScheduling</i>
9.	Escalonamento de Tarefas	<ul style="list-style-type: none"> • <i>A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems</i> • <i>Scheduling in the Grid Application Development Software Project</i>
10.	Grid de Dados	<ul style="list-style-type: none"> • <i>The Open Grid Services Architecture, and data Grids</i> • <i>Data Access, Integration, and Management</i>
#	Bibliografia	
1.	The Grid: Blueprint for a New Computing Infrastructure (FOSTER, et al., 1999)	
2.	Grid Computing: Making The Global Infrastructure a Reality (BERMAN, et al., 2003)	
3.	The Grid - Core Technologies	
4.	Grid Computing For Developers (SILVA, 2005)	
5.	An Introduction to Distributed Algorithms (BARBOSA, 1996)	
6.	Distributed Algorithms (LYNCH, 1996)	
7.	Scheduling Theory and its Applications (CHRÉTIENNE, et al., 1997)	
8.	Scheduling Algorithms by Peter Brucker	
9.	Scheduling: Theory, Algorithms, and Systems By Pinedo	
10.	Task Scheduling in Parallel and Distributed Systems (EL-REWINI, et al., 1994)	
#	Metodologia	
	Tipo de Aula	Avaliação dos Alunos
1.	Expositiva	Exercícios
2.		Prova
3.		Resumo
4.		Projeto/Apresentação/Artigo

5. DETALHAMENTO: ENSINO DE ESCALONAMENTO DE TAREFAS

5.1. Conceituação

A Computação em Grid envolve muitos recursos em diversos domínios administrativos (FOSTER, 2001). Um grande desafio na Computação em Grid é a alocação desses recursos para várias aplicações de maneira eficiente. Dessa forma, a atividade de escalonamento pode ser dividido em escalonamento do recursos (*Resource Allocation*) e o escalonamento das tarefas das aplicações (*Scheduling*).

De acordo com Berman (1999), a aplicação está diretamente relacionada a sua performance e, para obter uma boa performance em Grid, o escalonamento de tarefas é fundamental. Schopf (2002) define o escalonamento de tarefas (*Scheduling*) em Grid como o processo de tomada de decisões escalonamento de em múltiplos domínios administrativos. As estratégias de escalonamento estão relacionadas ao tipo de aplicação.

Na literatura técnica temos vários algoritmos publicados, como o WQR, XSufferage, Storage Affinity (SANTOS-NETO, 2004) que são utilizados para escalonamento de tarefas independentes (*Bag-of-Tasks*). O algoritmo *Workqueue with Replication* (WQR) foi desenvolvido para solucionar o problema da obtenção de informações sobre a aplicação e a carga de utilização dos recursos do Grid (PARANHOS, 2003). Inicialmente, as tarefas são submetidas para os recursos de maneira aleatória como o algoritmo WQ original. Depois que todas as tarefas foram submetidas, inicia-se o processo de replicação das tarefas que ainda não terminaram. Assim que a tarefa original ou uma réplica finalizar, as demais são interrompidas.

Assim, se faz necessário o ensino de estratégias de escalonamento de tarefas na disciplina de Computação em Grid. Também é necessário uma infraestrutura de testes para que os alunos possam implementar e testar as estratégias. Murshed, et al., (2002) apresentam alguns problemas relacionados ao ensino de escalonamento: primeiro, nem todos os alunos possuem acesso a uma infraestrutura de testes e nem todas as universidades oferecem esse tipo de infraestrutura. A criação de um ambiente de testes necessita de recursos e despendem muito tempo, e acaba se limitando a um ambiente local. A análise de várias estratégias de escalonamento requer vários testes com diferentes recursos e quantidade de usuários. A execução de tarefas pode consumir bastante tempo. O ambiente de testes real não oferece suporte a repetição e controle do ambiente para a experimentação e validação das estratégias. Por causa desses problemas, Murshed, et al., (2002) propõem a utilização de simuladores no ensino de escalonamento.

5.2. Uso de Simuladores de Grid

O GridSim (MURSHED, et al., 2002; BUYYA, et al., 2002) é um simulador escrito em Java baseado no simulador JavaSim. Permite modelar diferentes tipos de recursos, com a definição de sua capacidade de processamento, e também permite a criação de diferentes tipos de ambiente para teste. É possível a execução de aplicações heterogêneas podendo ser de processamento e/ou Entrada/Saída intensiva. Vários tipos de usuários podem ser criados. A velocidade da rede pode ser especificada. Com o GridSim torna-se viável a criação de um ambiente que pode ser utilizado para experimentação e avaliação dos algoritmos de escalonamento.

Para as atividades práticas, foi desenvolvido o CEGSE, um ambiente de simulação baseado no GridSim para ensino de algoritmo de escalonamento de tarefas em Grid. Este ambiente permite simular vários casos de testes para diferentes algoritmos. Em cada simulação, existe a possibilidade de variar alguns parâmetros, como: número de tarefas, tamanho da tarefa (em Mflops), tamanho dos arquivos de entrada e saída (em Mbytes).

Também permite configurar o ambiente de computação (os recursos disponíveis no Grid) especificando, por exemplo, a capacidade computacional de cada nó (em Mflops) e a velocidade dos *links* de comunicação entre os nós.

A implementação do ambiente de simulação CEGSE é baseada na hierarquia de classes do GridSim. Um resumo das classes do GridSim é apresentado na Figura 1. A classe *GridSimCore* é a classe que abrange todas as outras classes relativas ao escalonamento de tarefas. Ela herda as características de gerenciamento dos eventos e de entidades de thread da classe *Sim_entity* que faz parte do simulador SimJava. A classe *GridSim* herda as características da classe *GridSimCore*, sendo responsável pela inicialização, execução e parada de toda a simulação.

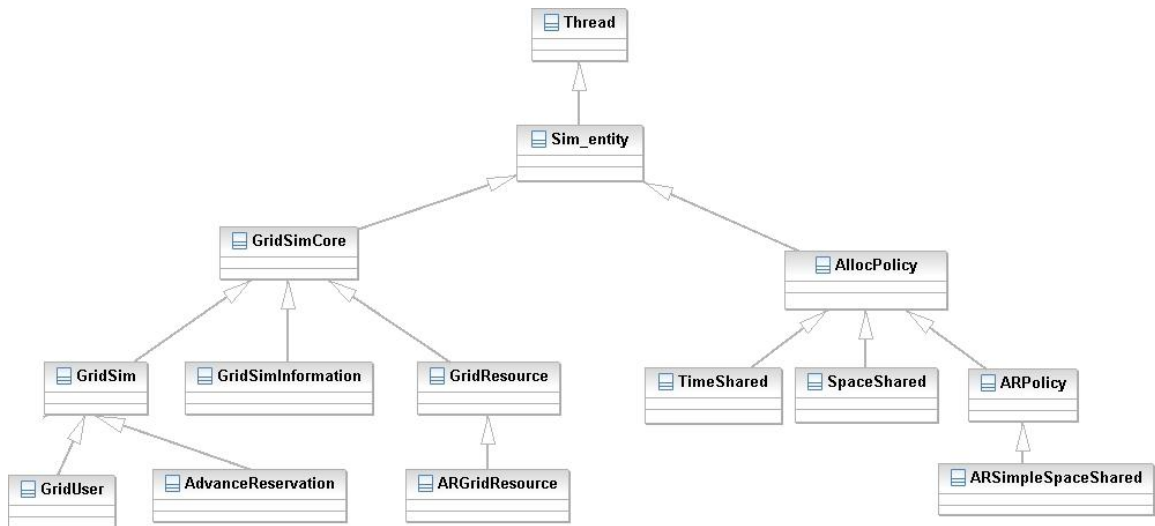


Figura 1 – Resumo da hierarquia de classes do GridSim

No CEGSE, a classe *GridBroker* é responsável por descobrir os possíveis recursos a serem utilizados, escalonar as tarefas e monitorá-las. A classe *Scheduling* é uma subclasse do *GridBroker*, sendo responsável pelo escalonamento de tarefas. A implementação dos vários algoritmos de escalonamento de tarefas é feita através da criação de uma subclasse da classe *Scheduling* (Figura 2).

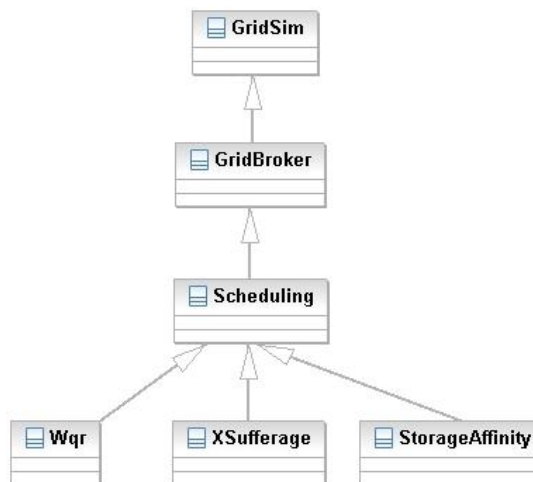


Figura 2 – Extensões da classe *Scheduling*.

5.3. Atividades Programadas

O ensino do t3pico escalonamento de tarefas pode ser desenvolvido com atividades de implementa33o pr3tica e avalia33o experimental dos algoritmos estudados. A implementa33o est3 relacionada a uma atividade pr3tica de projeto com programa33o, onde os alunos implementam os algoritmos estudados estendendo a classe *Scheduling* do CEGSE. Esta parte do trabalho tem o intuito de complementar os conceitos te3ricos estudados. Os alunos s3o divididos em grupos e cada grupo fica respons3vel pela implementa33o de um dos algoritmos estudados. Quando os alunos estiverem implementando o algoritmo, eles v3o entender as dificuldades reais que est3o relacionadas ao problema de escalonamento de tarefas. D3vidas sobre o comportamento do algoritmo e sugest3es de melhorias poss3veis come3am a surgir.

Na atividade de avalia33o dos algoritmos, cada grupo recebe os algoritmos implementados pelos outros grupos, e cada grupo fica respons3vel pela compara33o de todos os algoritmos. Os alunos recebem um arquivo contendo o ambiente de computa33o, para ser utilizado no CEGSE, e executam um conjunto de simula33es variando o n3mero de tarefas, o tamanho dos arquivos de entrada e de sa3da. Em seguida, comparam os resultados experimentais de cada algoritmo. Cada grupo deve entregar um trabalho escrito contendo os gr3ficos das avalia33es das tarefas e suas respectivas an3lises. Os alunos aprendem conceitos relacionados ao desempenho do sistema, e que este est3 diretamente relacionado ao algoritmo utilizado e ao tipo de aplica33o. Aprendem que a utiliza33o de alguns algoritmos se mostram invi3veis quando se aumenta o n3mero de tarefas. Tamb3m aprendem conceitos relacionados aos arquivos de entrada e sa3da, e como estes influenciam no desempenho do ambiente de Grid. Assim, passam a ficar atento a esta quest3o quando forem desenvolver um programa que necessite gerar ou ler arquivos. Na Figura 3, t3m-se um exemplo do resultado experimental com a compara33o de desempenho dos algoritmos de escalonamento de acordo com a quantidade de tarefas (eixo X). 3 utilizado como m3trica de desempenho o tempo m3dio da execu33o das tarefas em segundos (eixo Y). Neste exemplo, os alunos podem concluir que o algoritmo Dynamic-FPLTF possui um melhor desempenho que os demais algoritmos.

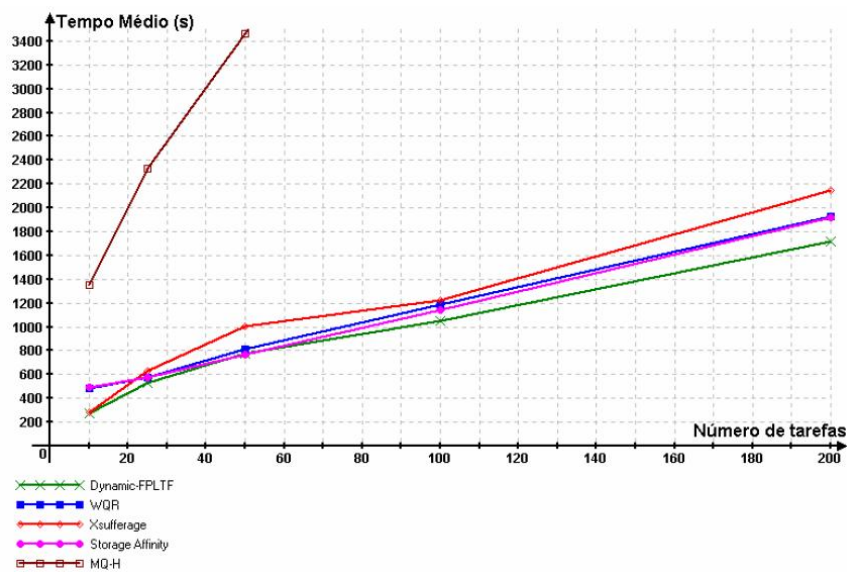


Figura 3 – Exemplo de Compara33o de Algoritmos de Escalonamento
(Fonte: JUNIOR, et al., 2007)

6. CONSIDERAÇÕES FINAIS

A contribuição deste trabalho inclui a análise curricular dos cursos de Engenharia da Computação com relação ao ensino de um tópico que está se difundindo rapidamente: a Computação em Grid. A possibilidade de usar a infra-estrutura mundial para o processamento de aplicações relevantes socialmente, como a pesquisa da cura da Aids e do Câncer, faz com que a Computação em Grid tenha um impacto semelhante ao da Internet.

Uma análise dos currículos de vários cursos no Brasil e no exterior mostrou que o ensino da Computação em Grid ainda não está consolidado, e apenas alguns tópicos são apresentados nos módulos finais de disciplinas como Programação Paralela e Distribuída ou Sistemas Distribuídos. Faz-se necessária então a rápida introdução de uma disciplina que cubra os tópicos relevantes.

Com base na análise efetuada, o artigo propôs uma ementa da disciplina “Computação em Grid” a ser introduzido nos cursos de Engenharia da Computação, contendo os tópicos a serem abordados, sugestão de referências bibliográficas e o uso de simuladores de Grid para a realização de atividades práticas com o intuito de consolidar os conceitos ensinados. Como exemplo, foi detalhado o ensino do tópico escalonamento de tarefas, com a proposição de atividades práticas como o desenvolvimento de algoritmos e uma análise comparativa de desempenho. Como infra-estrutura computacional foi desenvolvido um ambiente de simulação denominado CEGSE.

Agradecimentos

Os autores gostariam de agradecer aos pesquisadores e professores que vem contribuindo para o desenvolvimento da área de Computação em Grid. Agradecimentos também ao CNPq pelo financiamento parcial de um dos autores.

REFERÊNCIAS BIBLIOGRÁFICAS

- BARBOSA, V. C. **An Introduction to Distributed Algorithms**. MIT Press, 1996.
- BERMAN, F. **High-performance Schedulers**. In: (FOSTER, 1999). 1999.
- BERMAN, F; FOX, G.; HEY, A. J. G. **Grid Computing: Making The Global Infrastructure a Reality**. John Wiley & Sons, 2003.
- BIRMAN, K.P. **Building Secure and Reliable Network Applications**. Prentice Hall, 1996.
- BOINC. **Projects in Berkeley Open Infrastructure For Network Computing**. Disponível em: <<http://boinc.berkeley.edu/projects.php>>. Acesso em: 5 jun. 2008.
- BRUCKER, P. **Scheduling Algorithms**. Springer, 2007.
- BUYYA, R. **Economic-based Distributed Resource Management and Scheduling for Grid Computing**, 2002. Thesis (Doctor of Philosophy) Monash University, Melbourne, Australia.
- BUYYA, R. **High Performance Cluster Computing, Vol. 1 e Vol 2**. Prentice Hall, 1999.
- BUYYA, R. MURSHED, M. [GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing](#). **The Journal of Concurrency and Computation: Practice and Experience (CCPE)**, Wiley Press, 2002. V. 14, Issue 13-15.

CASABANT, T.L., KUHL, J.G. A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems. In **Readings in Distributed Computing Systems**, eds. Casavant, T.L. and Singhal, M., IEEE Computer Society Press, Los Alamitos, CA, 1994.

COFFMAN, E. G.; LENSTRA, J. K. LIU; Z. Edward G. **Scheduling Theory and its Applications**. John Wiley and Sons, 1997.

COREGRID. **CoreGRID Summer School 2008**. Disponível em: <<http://www.it.irf.tu-dortmund.de/IT/CoreGRID/>>. Acesso em: 5 jun. 2008.

EGEE. **Enabling Grid for E-science**. Disponível em: <<http://www.eu-egee.org/>> e <<http://www.egee.nesc.ac.uk/schedreg/index.cfm>>. Acesso em: 5 jun. 2008.

EL-REWINI, H.; LEWIS, T. G.; ALI, H.H. **Task Scheduling in Parallel and Distributed Systems**. Prentice-Hall, 1994.

FIGS. **Florida International Grid School 2008**. Disponível em: <<https://twiki.grid.iu.edu/twiki/bin/view/Education/FloridaInternationalGridSchool2008>>. Acesso em: 5 jun. 2008.

FOSTER, I. Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering. **Addison Wesley**. 1995.

FOSTER, I.; KESSELMAN, C. **The Grid : Blueprint for a New Computing Infrastructure**. Morgan Kaufmann Publishers, 1999.

FOSTER, I; KESSELMAN, C. **The Grid 2: Blueprint for a New Computing Infrastructure**. Morgan Kaufmann, 2003).

FOSTER, I.; KESSELMAN, C.; TUECKE, T. [The Anatomy of the Grid: Enabling Scalable Virtual Organizations](#). **International Journal of High Performance Computing Applications**. V. 15, n. 3, 2001.

GILDA. **Grid Infn Laboratory for Dissemination Activities**. Disponível em: <<http://www.nsf-middleware.org/>>. Acesso em: 5 jun. 2008.

GLITE. **Lightweight Middleware for Grid Computing**. Disponível em: <<http://glite.web.cern.ch/glite/>>. Acesso em: 5 jun. 2008.

GLOBUS. **Globus Toolkit**. Disponível em: <<http://www.globus.org/toolkit/>>. Acesso em: 5 jun. 2008.

GRID5000. **Grid5000: Home**. Disponível em: <<https://www.grid5000.fr/>>. Acesso em: 5 jun. 2008.

GRIDCAFE. **Projects in Berkeley Open Infrastructure For Network Computing**. Disponível em: <<http://gridcafe.web.cern.ch/gridcafe/gridprojects/testbed.html>>. Acesso em: 5 jun. 2008.

GRIDUNESP. **Grid da Universidade Estadual Paulista**. Disponível em: <<https://www.unesp.br/grid/>>. Acesso em: 5 jun. 2008.

ICEAGE. **International Summer School on Grid Computing 2008**. Disponível em: <<http://www.iceage-eu.org/v2/other%20summer%20schools.cfm>>. Acesso em: 5 jun. 2008a.

ICEAGE. **Master of Science programmes in Grid Computing or related areas**. Disponível em: <<http://www.iceage-eu.org/v2/msc%20courses.cfm>>. Acesso em: 5 jun. 2008b.

IEEE/ACM. **Computing Currícula 2005: The Overview Report**. Vol. Computing Curricula Series. Disponível em: <<http://www.acm.org/education/curricula-recommendations>>. Acesso em: 5 jun. 2008.

Instituto Mauá de Tecnologia. **Disciplina “ECM913 - Computação Grid”**. Disponível em: <<http://www.maua.br/ceun/index/docente/id/84>>. Acesso em: 5 jun. 2008.

ISSGC. **International Summer School on Grid Computing 2008**. Disponível em: <<http://www.iceage-eu.org/issgc08/index.cfm>>. Acesso em: 5 jun. 2008.

IWSGC. **International Winter School on Grid Computing 2008**. Disponível em: <<http://www.iceage-eu.org/iwsgc08/index.cfm>>. Acesso em: 5 jun. 2008.

JALOTE, P. **Fault Tolerance in Distributed Systems**. Prentice-Hall, 1994.

JUNIOR, J. N. F.; JÚNIOR, A. M.; OLIVEIRA, L. J.; BOCCARDO, D. R. Avaliação de algoritmos de escalonamento em Grids para diferentes configurações de ambiente. **Anais do Wperformance 2007**, Rio de Janeiro, 2007

LI, M.; BAKER, M. **The Grid: Core Technologies**. Wiley, 2005.

LYNCH, N. **Distributed Algorithms**. Morgan Kaufmann, 1996.

MULLENDER, S. **Distributed Systems**. Addison Wesley Publishing Company, 1993.

MURSHED, M. BUYYA, R. Using the GridSim Toolkit for Enabling Grid Computing Education. In: **Proceedings International Conference on Communication Networks and Distributed Systems Modeling and Simulation**, SCS, 2002, pp. 18–24.

NMI. **National Science Foundation Middleware Initiative**. Disponível em: <<http://www.nsf-middleware.org/>>. Acesso em: 5 jun. 2008.

PARANHOS, D; CIRNE, W.; BRASILEIRO, F. V. Trading Cycles for Information: Using Replication to Schedule Bag-of-Tasks Applications on Computational Grids. In: **Proceedings of the Euro-Par 2003: International Conference on Parallel and Distributed Computing**, 2003.

PETCU, D. Grid Technologies to PhD Students, Part 1: Using Best Practices to Build the Course. **IEEE Distributed Systems Online**, vol. 9, no. 3, 2008.

PINEDO, M. **Scheduling: Theory, Algorithms, and Systems**. Prentice Hall, 2001.

RAYNAL, M. **Distributed algorithms and protocols**. John Wiley and Sons, 1992.

SANTOS-NETO, E. L. **Escalonamento de Aplicações que processam grande quantidade de dados em Grids Computacionais**. 2004 Dissertação de mestrado, Universidade Federal da Campina Grande – UFCG.

SBC. **Currículo de Referência da SBC para Cursos de Graduação em Bacharelado em Ciência da Computação e Engenharia de Computação**. Sociedade Brasileira de Computação. Proposta versão 2005. Disponível em: <www.sbc.org.br/index.php?language=1&content=downloads&id=198>. Acesso em: 5 jun. 2008.

SOTOMAYOR, B; CHILDERS, L. **Globus Toolkit 4: Programming Java Services Paperback, First Edition**. Morgan Kaufmann, 2005.

SCHOPF, J. M. **A General Architecture for Scheduling on the Grid**. Argonne National Laboratory preprint, 2002.

SILVA, V. **Grid Computing For Developers**. Charles River Media, 2005.

SPRACE. **São Paulo Regional Analysis Center**. Disponível em: <<http://www.sprace.org.br/>>. Acesso em: 5 jun. 2008.

TERAGRID. **TeraGrid**. Disponível em: <www.teragrid.org/>. Acesso em: 5 jun. 2008.

Universidade do Porto. **Disciplina “Computação em Grid”**. Disponível em: <<http://www.dcc.fc.up.pt/~ines/aulas/0708/CG/CG.html>>. Acesso em: 5 jun. 2008.

Universidade Federal de Campina Grande - UFCG. **Disciplina “Sistemas Distribuídos”**. Disponível em: <<http://fubica.lsd.ufcg.edu.br/wiki/doku.php?id=sdpos:2007.2>>. Acesso em: 5 jun. 2008.

Universidade Federal Fluminense - UFF. **Disciplina “Arquiteturas Paralelas III: Computação em Grids e Clusters”**. Disponível em: <<http://www.ic.uff.br/~vefr/teaching/ArqPar.html>>. Acesso em: 5 jun. 2008.

Universidade Federal do Rio Grande do Sul - UFRGS. **Disciplina “CMP 157 - Programação Distribuída e Paralela”**. Disponível em: <<https://saloon.inf.ufrgs.br/twiki/view/Disciplinas/CMP157/PDP2007>>. Acesso em: 5 jun. 2008.

University of Alabama at Birmingham. **Disciplina “433-678: Cluster and Grid Computing”**. Disponível em: <<http://www.cis.uab.edu/cs633/spring2007/>>. Acesso em: 5 jun. 2008.

University of Melbourne. **Disciplina “433-678: Cluster and Grid Computing”**. Disponível em: <<http://www.cs.mu.oz.au/678/>>. Acesso em: 5 jun. 2008.

University of Westminster. **MSc Cluster and Grid Computing**. Disponível em: <<http://www.wmin.ac.uk/cscs/page-1776>>. Acesso em: 5 jun. 2008.

VERÍSSIMO, P.; RODRIGUES, L. **Distributed Systems for System Architects**. Kluwer Academic Publishers, 2001.

WCG. **Active Research in World Community Grid**. Disponível em: <http://www.worldcommunitygrid.org/projects_showcase/viewResearch.do>. Acesso em: 5 jun. 2008.

WWG, **World-Wide Grid**. Disponível em: <<http://www.gridbus.org/wwg/>>. Acesso em: 5 jun. 2008.

TEACHING GRID COMPUTING FOR THE COURSE OF COMPUTER ENGINEERING

Abstract: *This article states analyses a current education of an emerging area has being spreading quickly on Computer Engineering: Grid Computing. Several analysts predict the impact of grid computing application in the society will be similar of internet arises. It occurs because of possibly utilization of worldwide infrastructure for processing, make viable several applications with no support of fastest supercomputers nowadays. The analyses of this topic it is not covered by various engineering courses in Brazil or abroad. This article propose to offer a discipline of “Grid computing” in the Computer Engineering courses, presenting topics to be approaches, suggested bibliography as well as education methodology based on grid simulator. The simulator utilization is detailed for scheduling tasks statement within practices activities for concept consolidation.*

Key-Word: *Grid Computing, education, tasks scheduling, computer engineering.*