

Planejamento de execução de aplicação baseada no tipo de tarefa com o foco na análise de desempenho

Emerson da Silva Borges, Maurício Amaral de Almeida

Pós-Graduação / Strictu Sensu - Programa de Mestrado em Tecnologia

Centro Estadual de Educação Tecnológica Paula Souza - São Paulo – Brasil

Agosto de 2012

borges_emerson@yahoo.com.br, madealmeida@gmail.com

Resumo – Este artigo apresenta uma proposta de método de planejamento de execução de uma aplicação baseado no tipo de tarefa com foco na análise de desempenho e a análise dos resultados obtidos durante a aplicação do método na tarefa de pré-processamento de textos de ementas da jurisprudência da Justiça do Trabalho do Estado de São Paulo.

Palavras chave: *Grids* Computacionais, Computação Distribuída, Planejamento de Tarefas, Análise de Desempenho, Pré-Processamento de Textos.

Introdução

Os *Grids* emergem como uma infraestrutura cibernética e global de aplicações para a próxima geração de *e-science*, integradas em larga escala, distribuídas e de recursos heterogêneos. Comunidades científicas como é o caso de física de alta energia e de ondas gravitacionais, geofísica, astronomia e bioinformática estão utilizando *Grids* para compartilhar, gerenciar e processar grandes conjuntos de dados. Para alcançar este potencial em *Grids Computacionais* o escalonamento de tarefas é uma importante questão a ser considerada [1].

Entre os principais desafios dos ambientes heterogêneos e muitas vezes geograficamente distribuídos, como os *Grids Computacionais*, está o problema de escalonamento de tarefas. Diversos estudos têm sido propostos no sentido de direcionar quais são os tipos de aplicações candidatas à execução em ambientes de *Grids Computacionais* com abordagem de RMS (*Resource Management Systems*) tais como [2,3,4,5,6] com base em políticas e algoritmos de escalonamento, mas qual é o método que pode ser aplicado para fazer o estudo de viabilidade de executar uma aplicação nos ambientes de *Grids Computacionais*?

Uma das técnicas de descobrir se uma aplicação é candidata a ser executada em ambientes de computação distribuída é a classificação da aplicação quanto a sua divisibilidade [7]. Este artigo apresenta uma proposta de metodologia que possibilita o estudo de viabilidade de execução de uma aplicação, de um determinado domínio, tomando por base o estudo da divisão da aplicação em tarefas menores.

A ementa é um resumo de uma decisão (acórdão) tomada por um colegiado de desembargadores. As ementas das decisões mais relevantes compõem a jurisprudência de um Tribunal.

A classificação, em Mineração de Textos, visa a identificar os tópicos principais em um documento e associar este documento a uma ou mais categorias predefinidas [14]. Segundo Konchady [13], o problema da classificação pode ser descrito como a classificação de documentos em múltiplas categorias, onde se tem um conjunto de n categorias $\{C_1, C_2, \dots, C_n\}$ para as quais são associados m documentos $\{D_1, D_2, \dots, D_m\}$. O processo de mineração de texto pode ser dividido em quatro etapas conforme Gonçalves e Rezende [15]: Coleta de Documentos, Pré-processamento, Extração de Conhecimento, Avaliação e Interpretação dos Resultados.

O método de Aprendizado de Classificadores proposto por Ferauche e Almeida [8] tem o objetivo de validar o aprendizado de máquina utilizando as ementas da jurisprudência do Tribunal Regional do Trabalho da 2ª. Região – SP. Para isto foi utilizada a análise estatística seguindo as 4 etapas citadas anteriormente, adaptando a etapa de Avaliação e Interpretação dos Resultados para comparar os resultados de indução dos algoritmos classificadores com a classificação dos especialistas previamente realizada. A etapa de Pré-processamento consiste de um conjunto de ações realizadas sobre o conjunto de textos obtido na etapa anterior, com o objetivo de prepará-los para a extração de conhecimento.

O presente trabalho tem o objetivo de validar a técnica proposta utilizando como aplicação modelo a etapa de Pré-processamento de documentos da Justiça do Trabalho do Estado de São Paulo [8], em ambientes de *Grids Computacionais*.

Metodologia

A pesquisa está baseada no estudo de Borges e Almeida [9] do problema de escalonamento em *Grids Computacionais* com o foco na análise de desempenho. O trabalho propõe um método de análise de viabilidade de execução de uma aplicação em computação distribuída, com base no tipo de tarefa.

Este estudo apresenta um planejamento de execução das tarefas de uma aplicação neste ambiente computacional. O processo é dividido em sete etapas, ordenadas sequencialmente e, tem o objetivo de dar uma contribuição ao processo de tomada de decisão no escopo da utilização de ambientes de computação distribuída para a execução de uma aplicação de um domínio conhecido. As etapas da proposta são:

- 1. Estudo de possibilidade de execução de uma aplicação em computação distribuída:** o foco desta etapa é a investigação com base no estudo da classe da aplicação: *parameter sweep, bag-of-tasks ou workflow*¹, se é possível realizar o processamento da aplicação em estudo em *Grids Computacionais*;
- 2. Estudo das métricas de avaliação:** esta etapa tem a idéia básica de definir quais são as métricas de avaliação a extrair da proposta;
- 3. Divisão da aplicação em tarefas menores a ser executadas:** aqui é realizado o

¹ Aplicações *Bag-of-tasks* são aplicações que podem ser divididas e executadas de forma independente; Aplicações *parameter Sweep* podem ser divididas, independentes mas com o objetivo de resolver um problema comum e Aplicações *Workflow* são compostas por um conjunto de tarefas que devem ser executadas seguindo uma ordem parcial determinada por dependência de controle de dados.

fracionamento, de fato, da aplicação em unidades menores de processamento com o fim específico de distribuí-las nos nós computacionais disponíveis para a sua execução;

4. **Definição dos métodos de transferência dos arquivos de entrada e saída:** esta etapa descreve como deve ser realizada a transferência dos arquivos de entrada, bem como a forma de recebimento dos arquivos de resultados. Aqui podem ser definidas transferências simples aos ambientes de execução da aplicação até complexos *workflows* de operações conhecidas como *File Stage*² no domínio de computação distribuída;
5. **Montagem dos lotes de execução das tarefas:** nesta etapa com base nas políticas de escalonamento adequadas ao problema da aplicação em estudo e aplicação da técnica de divisão da aplicação em tarefas menores [7] é feito o agrupamento de tarefas de execução;
6. **Processamento das tarefas e recebimentos dos resultados:** após a realização da melhor política das tarefas que compõem os *jobs*, nesta etapa eles são submetidos ao sistema de interface de execução da aplicação. Então basta fazer o agendamento da execução destas unidades de processamento, monitorá-las e aguardar os dados recebidos de resultados a serem compilados e estudados na próxima fase;
7. **Análise dos resultados:** esta etapa é responsável pela transformação dos dados gerados pelos diversos tipos de arquivos de *log* (resultados) recebidos pelo ambiente de interface, sendo utilizado na execução do experimento. Em seguida é realizado a criação de planilhas e gráficos de análise baseados nestes arquivos de resultados.

O estudo utilizará o método proposto na pesquisa, composto pelas etapas descritas acima com a finalidade de confirmar a possibilidade de obtenção do referido estudo de viabilidade baseado na utilização do processo a um exemplo do mundo real.

Estudo de possibilidade de execução da aplicação em computação distribuída

Ao iniciar o estudo da natureza da aplicação de pré-processamento de documentos de Ferauche e Almeida [8] é possível verificar que ela tem natureza heterogênea. O pré-processamento de documentos é utilizado com o propósito de extrair termos relevantes dos documentos da coleção em estudo. Essa aplicação pode ser dividida em várias etapas de pré-processamento de categorias. A possibilidade de divisão da aplicação em várias tarefas independentes permite a classificação da mesma como *Bag-of-Tasks*, quando cada tarefa pode ser executada independente das demais e utilizadas para resolver problemas distintos que no caso do estudo o treinamento de classificadores de documentos binários. O resultado desta etapa é afirmativo, a aplicação em estudo é divisível em tarefas menores.

² File Staging é conhecido no domínio de aplicações executadas em ambiente de computação distribuída como o estágio de transferência dos arquivos ou lote de arquivos. Esta técnica se refere a quando todos os arquivos são arranjados como coleções e movidos para onde eles são necessários. Muitos escalonadores de tarefas e recursos contemplam este mecanismo tais como o Condor [10], PBS [11], LSF [12] entre outros.

Estudo das métricas de avaliação

A métrica escolhida para a avaliação deste estudo é o tempo em segundos, sendo possível a obtenção dos índices *makespan* e *Speedup*³. Nesta etapa são consideradas as características de análise de resultados para o apoio a decisão pretendida no estudo. Para o presente artigo o objetivo é obter dados de economia de recursos computacionais, foco na análise de desempenho. Esta é a justificativa pela utilização da grandeza tempo em segundos na definição das métricas. É importante observar também que os experimentos computacionais foram realizados em um ambiente de computação local, portanto sem interconexões de rede distribuídas remotamente, não sendo relevante para o estudo a medição de métricas de consumo de largura de banda.

Divisão da aplicação em tarefas menores

A aplicação desta técnica se dá após o descobrimento da unidade mínima de processamento de tarefas, possibilitada através do estudo do domínio da aplicação que foi necessário, na etapa 1. Esta etapa permitiu a classificação da aplicação em várias etapas distintas e independentes, portanto uma aplicação do tipo *bag-of-tasks*. A unidade mínima de divisão é o pré-processamento de uma coleção de documentos de uma determinada categoria. Um exemplo desta divisão é descrito na Tabela 1.

Tarefa	Exemplos (+)	Exemplos (-)	Total de documentos por categoria
EXECUCAO	5370	5373	10743
MAO DE OBRA	4308	4311	8619
EMBARGOS	4248	4250	8498
PROVA	3689	3691	7380
PRESCRICAO	2834	2836	5670
Total de Documentos	20449	20461	40910

Tabela 1: Exemplo de 5 tarefas do domínio da aplicação

As tarefas descritas na Tabela 1 é composta pela seleção de documentos, seguindo a distribuição de quantidades de documento proposta em Ferauche e Almeida [8]. Cada categoria tem uma coleção de documentos para a realização da etapa de pré-processamento de documentos conforme a distribuição da tabela anterior, deste modo no exemplo temos cinco unidades de processamento computacionais podendo ser distribuídas para o ambiente computacional em utilização.

Definição do método de transferência dos arquivos

O estudo dos arquivos de entrada e saída necessários para a execução de cada tarefa é o escopo desta etapa. No caso em estudo a ferramenta utilizada para o pré-processamento das ementas trabalhistas permite, através de arquivo de

³ A variável *makespan* é utilizada no escopo deste artigo como a medida de tempo de intervalo entre o início da execução de uma tarefa até a sua finalização. Já o *Speedup* é o menor tempo de execução atingido na execução de uma tarefa ou de um lote de tarefas, podendo também ser entendido como a tarefa de tarefa de execução mais rápida

configuração e passagem de parâmetros de linha de comando de sistemas operacional, que seja utilizado um diretório específico para a leitura e gravação destes arquivos textos. Aproveitando este recurso é possível utilizar tanto unidades locais, como também unidades de rede para a leitura e escrita dos dados de entrada e saída.

Nos experimentos deste trabalho foi utilizado uma unidade de compartilhamento de arquivos do tipo NFS (*Network File System*) facilitando a etapa de *File Stage*, já que as coleções de textos foram transferidas para a NFS no início do processo.

Montagem dos lotes de execução de tarefas

Nesta etapa foram montados três tipos de lotes de execução seguindo em ordem decrescente de categorias com o maior número de documentos a serem pré-processados. Como resultado foram obtidos três tipos de lotes conforme a distribuição tabelas 2,3 e 4.

Tipo de Lote 1 – Tarefa	Exemplos (+)	Exemplos (-)	Total de doc. categoria
EXECUCAO	5370	5373	10743
MAO DE OBRA	4308	4311	8619
EMBARGOS	4248	4250	8498
PROVA	3689	3691	7380
PRESCRICAO	2834	2836	5670
Total de Documentos	20449	20461	40910

Tabela 2: Lote com 5 tarefas

Tipo de Lote 3 – Tarefa	Exemplos (+)	Exemplos (-)	Total de documentos por categoria
EXECUCAO	5370	5373	10743
MAO DE OBRA	4308	4311	8619
EMBARGOS	4248	4250	8498
PROVA	3689	3691	7380
PRESCRICAO	2834	2836	5670
DANO MORAL E MATERIAL	2532	2534	5066
COMPETENCIA	2151	2153	4304
SINDICATO OU FEDERAÇÃO	2094	2096	4190
SERVIDOR PÚBLICO EM GERAL	2020	2023	4043
JORNADA	1979	1982	3961
ASSISTENCIA JUDICIÁRIA	1564	1567	3131
HONORARIOS	1559	1561	3120
CONCILIAÇÃO	1377	1379	2756
MULTA	1321	1324	2645
RECURSO	1297	1299	2596
INSALUBRIDADE OU PERICULOSIDADE	1260	1261	2521
APOSENTADORIA	1248	1251	2499
ESTABILIDADE OU GARANTIA DE EMPREGO	1184	1187	2371
NULIDADE PROCESSUAL	1160	1162	2322
Total de Documentos	43195	43240	86435

Tabela 4: Lote com 20 tarefas

Aplicando esta técnica e executando o início dos experimentos foi observado que havia uma categoria de textos denominada “PREVIDÊNCIA”, onde a quantidade de documentos era superior ao dobro da segunda maior categoria da coleção: “EXECUÇÃO”. Esta observação gerou uma situação não desejada na execução dos lotes, ou seja, todas as vezes que esta categoria foi incluída num lote de execução o escalonador de recursos sempre processava esta categoria em prioridade, prejudicando o processamento das demais tarefas do lote por esta razão foi tomada a decisão de não considerá-la para a montagem e execução dos lotes.

Processamento das tarefas e recebimento dos resultados

Após a realização do agrupamento das tarefas em lotes, da etapa anterior, é feito aqui a criação de arquivos de lote para a execução respeitando o modelo de processamento e arquivos de descrição de lotes de execução do ambiente que está sendo utilizado nos experimentos. O resultado desta etapa é a preparação de arquivos de agendamento e execução de tarefas, monitoramento e posterior recebimento dos logs de resultados a ser explorados na próxima etapa. Foram realizados diferentes tipos de testes utilizando os lotes de processamento definidos com a aplicação da técnica. O planejamento dos lotes de execução se deu da seguinte maneira:

1. Definição dos lotes a ser executados;
2. Definição da quantidade de nós para execução com base no item;
3. Escolha a quantidade de tarefas para execução com base no item;
4. Criação de um arquivo de agendamento de tarefas para cada quantidade de nós desejados, de acordo com a distribuição da Tabela 5;
5. Agendamento da execução dos 13 jobs, monitoramento e compilação dos resultados.

Número Lote	Lote(Descrição)	Número nós
1	05 Tarefas	2
2	05 Tarefas	5
3	05 Tarefas	10
4	10 Tarefas	2
5	10 Tarefas	4
6	10 Tarefas	5
7	10 Tarefas	6
8	10 Tarefas	10
9	20 Tarefas	2
10	20 Tarefas	4
11	20 Tarefas	5
12	20 Tarefas	6
13	20 Tarefas	10

Tabela 5: Distribuição dos lotes do experimento

Análise de Resultados

Nesta seção são apresentados os resultados obtidos com a aplicação através do método proposto. A Tabela 6 apresenta o comparativo entre os resultados obtidos através da execução de cada tarefa individualmente com os resultados gerados com o processamento dos experimentos em diversos nós computacionais.

Número Lote	Lote(Descrição)	Tempo(1)	Tempo(2)	Número nós	Desempenho
1	05 Tarefas	4485	1830	2	40,80%
2	05 Tarefas	4485	1623,6	5	36,20%
3	05 Tarefas	4485	1657,8	10	36,96%
4	10 Tarefas	6406	2706	2	42,24%
5	10 Tarefas	6406	2056,2	4	32,10%
6	10 Tarefas	6406	1987,2	5	31,02%
7	10 Tarefas	6406	1816,2	6	28,35%
8	10 Tarefas	6406	1890,6	10	29,51%
9	20 Tarefas	8440	3312	2	39,24%
10	20 Tarefas	8440	2491,8	4	29,52%
11	20 Tarefas	8440	2223	5	26,34%
12	20 Tarefas	8440	2830,8	6	33,54%
13	20 Tarefas	8440	2117,4	10	25,09%

Tabela 6: Comparação dos resultados locais com as execuções distribuídas experimentais

São apresentados na Tabela 6 os resultados de todos os experimentos realizados sendo comparados aos resultados da execução das unidades de tarefas em processamento local. O número de nós é a quantidade de máquinas utilizadas no processamento de cada lote. As colunas Tempo(1) e Tempo(2) são na sequência o tempo gasto para o processamento do lote em uma máquina e o mesmo processamento na quantidade de nós da coluna número de nós. Exemplo: no lote 1 o tempo(2) representa 40,80% do tempo(1). As três linhas em destaque (2, 7 e 13) da tabela representam o *Speedup*, métrica de melhor tempo de processamento, independente do número de nós alocados no processo. Esse *Speedup* é apresentado no gráfico da Figura 1.

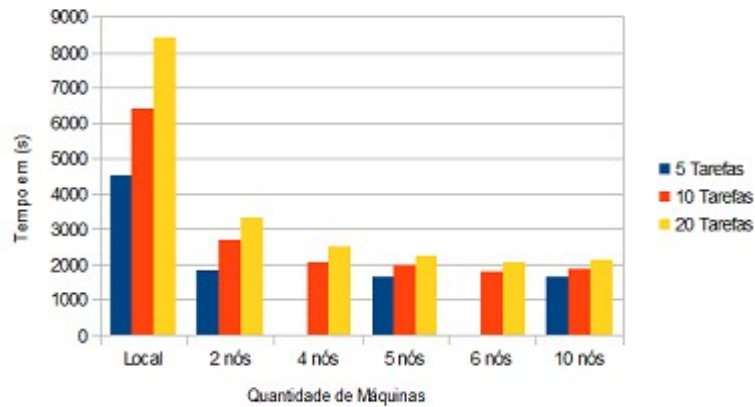


Figura 1: Tempo de execução para os lotes de 5, 10 e 20 tarefas nas configurações: Local, 2nós, 4nós, 5nós, 6nós e 10nós

O gráfico da Figura 1 apresenta a visualização dos pontos ótimos na execução de cada lote de processamento. Analisando cada um dos tempo em segundos obtidos no processamento de cada lote de tarefas nas configurações de nós exibidas pode-se afirmar:

1. Há um ganho da ordem de aproximadamente 60% quando é realizado o mesmo processamento dos lotes: local para 2 nós;
2. Ao migrar os lotes de tarefas de 2 nós para 4 nós observa-se ainda um ganho de processamento de aproximadamente 11% para o lote de 5 tarefas, em torno de 20% para o lotes de 10 tarefas e um ganho de aproximadamente 32% para o lote de 20 tarefas;
3. Os ganhos percentuais ficam bem menos expressivos quando as tarefas são executadas nas configurações de 5 e 6 nós, comparadas as configurações de 4 nós atingindo na maioria dos lotes de tarefas ganhos bem inferiores a 10%;
4. Quando os lotes de tarefas são executados em ambientes computacionais acima de 5 nós verificam-se margens de ganho muito pouco expressivas tendendo a valores próximos de 0 (zero).

Os resultados verificados nesta seção permitem a visualização dos ganhos de tempo de processamento quando executados os lotes de tarefas para este experimento em ambientes de *Grids* Computacionais, onde as tarefas são executadas em paralelo e contam com escalonadores de recursos para a submissão de tarefas e acompanhamento e monitoramento das mesmas. Além da visualização destes ganhos, pela análise do gráfico das figura 1 é possível fazer a verificação dos pontos ótimos de execução de cada lote de tarefas possibilitando o estudo de análise de execução de cada tipo de lote em ambientes de 2 a 10 nós computacionais permitindo fazer uma análise de tendências tal qual a análise feita do gráfico da Figura 1.

Conclusão

Este artigo apresentou a proposta e a aplicação de um método de planejamento de estudo de viabilidade de execução de tarefas, aplicável em ambientes de computação distribuída, especialmente em *Grids* Computacionais.

Os resultados obtidos demonstram que, através da aplicação do método proposto no trabalho, é possível obter uma consistente análise de viabilidade econômica da execução de uma aplicação de um determinado domínio. Conclui-se também que efetivamente pela aplicação da metodologia deste trabalho com as sete etapas propostas aplicadas ao caso de estudo foi possível comprovar que o método é aplicável em sua totalidade.

Referências

- [1] BARUAH, A. **Taxonomy of Workflow Scheduling in Grids**. INTERNACIONAL JOURNAL OF COMPUTER APPLICATION: ISSUE2, V. 1, February 2012.
- [2] BUYYA, R.; ABRAMSON, D.; GIDDY, J. **Nimrod/G: An architecture for a resource management and scheduling system in a global computational Grid**. Proceedings of the International Conference on High Performance Computing in Asia–Pacific Region (HPC Asia 2000), 2000.
- [3] ASSIS, L., Nóbrega-Júnior, N., BRASILEIRO, F., CIRNE, W. **Uma heurística de particionamento de carga divisível para grids computacionais**. XXIV Simpósio Brasileiro de Redes de Computadores. 2006
- [4] CASANOVA, H. et al. **Heuristics for Scheduling Parameter Sweep Applications** in: Grid Environments. 9th Heterogeneous Computing Workshop, Cancun Mexico, p. 349-363. 01 maio 2000.
- [5] CASANOVA, H.; HAYES, J.; YANG, Y. **Algorithms and Software to Schedule and Deploy Independent Tasks in Grid Environments**. Workshop On Distributed Computing, Metacomputing, And Resource Globalization, Aussois France, p. 3-17. dez. 2002 .
- [6] CASANOVA, H.; BERMAN, F. **Parameter Sweeps on the Grid with APST**. In: FOX, G.; HEY, T. (Org). Grid Computing: Making the Global Infrastructure a Reality. West Sussex-England: John Wiley & Sons, 2003. p. 773-787.
- [7] WILKINSON, B. **Grid Computing: Techniques and Application**. USA: Taylor and Francis Group, 2010. (ISBN: 9781420069532).
- [8] FERAUCHE, T.; ALMEIDA, M. A. **Aprendizado de Classificadores**. In: VI WORKSHOP DE PÓS-GRADUAÇÃO E PEQUISA DO CENTRO PAULA SOUZA, São Paulo: CEETEPS, 2011.
- [9] BORGES, E.S.; ALMEIDA, M. **GRIDS COMPUTACIONAIS: UMA PROPOSTA DE MÉTODO DE PLANEJAMENTO DE EXECUÇÃO DE APLICAÇÃO BASEADA NO TIPO**

DE TAREFA COM O FOCO NA ANÁLISE DE DESEMPENHO. 2011. 93 f. Dissertação (Mestrado) - Curso de Tecnologia de Informação Aplicada, Departamento de Programa de Mestrado em Tecnologia: Tecnologia de Informação Aplicada, CEETEPS, São Paulo, 2012.

[10] **CONDOR**, Disponível em: <<http://www.cs.wisc.edu/condor>>. Acesso em: 10 de janeiro de 2012.

[11] **PBS: Portable Batch System**. Disponível em:<[http://www.nas.nasa.gov/hecc/support/kb/Portable-Batch-System-\(PBS\)-Overview_126.html](http://www.nas.nasa.gov/hecc/support/kb/Portable-Batch-System-(PBS)-Overview_126.html)>. Acesso em: 10 de janeiro de 2012.

[12] **LSF: Load Sharing Facility**. Disponível em: <<http://www.platform.com/workload-management/high-performance-computing>>. Acesso em: 10 janeiro de 2012.

[13] KONCHADY, M. **Text Mining Application Programming: Charles River Media**, 2006. ISBN 1-58450-460-9.

[14] EBECKEN, N. F. F.; LOPES, M. C. S.; COSTA, M. C. A. (2003), **Mineração de Textos**. In: REZENDE, S. O. Sistemas Inteligentes: fundamentos e aplicações. Barueri, SP: Manole.

[15] GONÇALVES, L. S. M.; REZENDE, S. O. (2002), **Categorização em Text Mining. Disponível** em: <<http://www.icmc.usp.br/~stded/Artigos/Computacao/IC/LeaSilviaMG.pdf>> - Acesso em: 05/09/2007.

Contatos

Emerson da Silva Borges

Mestre em Tecnologia da Informação pelo CEETEPS

Analista de Sistemas, Consultor de Banco de Dados

borges_emerson@yahoo.com.br - Fone: (11) 99715-4922