

Computação Distribuída e Novos Paradigmas em Física Computacional

Cesar R. S. da Silva

*Faculdade de Ciências Integradas do Pontal,
Universidade Federal de Uberlândia*

Abstract

Outstanding problems in computer physics are increasingly complex. Their solutions involve extensive sampling of points in the parameter space. Particularly, studies of structural and thermodynamic properties of materials span increasingly broader samplings, resulting in a rapidly growing number of tasks to be executed, threatening to make these studies unfeasible. A proposal to overcome these difficulties is to use cyberinfrastructure systems (CI) to support concurrent computations of parameter sampling applications in scientific computing, promoting ease of use and aggregation of capacity of a large number of high performance computational resources. These systems are usually implemented as Grid/Portal systems, Service Oriented Architectures (SOA), and clouds. This scenario imposes a new paradigm in computational science. Problem solution is now represented in three levels: *i)* Self contained components of the solution implemented as traditional programs; *ii)* Complete solution implemented as distributed workflows involving a number of instances of those programs; *iii)* Web portals as user interface.

Resumo

Os problemas proeminentes na física computacional são cada vez mais complexos. Suas soluções envolvem amostragem extensiva de pontos no espaço de parâmetros. Particularmente, os estudos de propriedades estruturais e termodinâmicas de materiais geram amostragens cada vez maiores, resultando em rápido crescimento do número de tarefas a serem executadas, ameaçando tornar estes estudos inviáveis. Propõe-se, para superar estas dificuldades, o emprego de sistemas de infraestrutura cibernética (CI) para executar aplicações de amostragem do parâmetro em computação concorrente, promovendo a facilidade de uso e a agregação da capacidade de um grande número recursos computacionais de elevado desempenho. As CI são implementados geralmente como sistemas grid/portal, arquiteturas orientadas serviço (SOA), ou como nuvens. Isso impõe um novo paradigma na ciência computacional. As soluções dos problemas são representada agora em três níveis: *i)* Componentes independentes da solução executada como programas tradicionais; *ii)* Solução completa executada como workflows distribuídos que envolvem várias instâncias desses programas. *iii)* Portais web como interface de usuário.

Key words: Ab-initio molecular dynamics, Materials computations, Distributed computing, Grids, SOA.

Palavras chave: Dinâmica molecular ab-initio, Computação de materiais, Computação distribuída, Grids, SOA.

PACS: 91.40.Ac, 91.30.Ab, 91.40.Ac, 61.43.Bn

1 Introdução

O avanço do uso de métodos numéricos nos diversos ramos da física teórica, bem como em outras ciências, tem sido uma tendência constante nas últimas seis décadas. Durante esse tempo, tanto os sistemas de computação quanto os métodos teórico-numéricos evoluíram tremendamente, e com estes, os seus requerimentos computacionais.

Uma característica comum a todas as áreas é que, com o inexorável avanço da pesquisa científica, os problemas mais simples vão sendo resolvidos e somente problemas cada vez mais complicados permanecem pendentes de solução. Considere-se, por exemplo, estudos ab-initio de propriedades elásticas de materiais. Baseado na experiência pessoal, o autor estima que o número de tarefas requeridas para concluir um estudo estado da arte nessa área evoluiu da ordem de 10^0 no início dos anos noventa [1] para 10^3 atualmente [2]. No início dos anos noventa um estudo estado da arte consistia da determinação de apenas algumas das constantes elásticas, considerando-se apenas a pressão e temperatura zero. Atualmente, o estado da arte exige que determinemos todas as constantes elásticas em um número substancial temperaturas e pressões. Em poucos anos, será imprescindível abordarmos os problemas referentes a ligas e soluções sólidas de maneira formal, o que acarretará um aumento por um fator (≈ 50) do número de tarefas a serem executadas.

O grande número de tarefas decorre de que muitos tipos de cálculos em computação científica exigem que um mesmo procedimento seja repetido para diversos valores dos seus parâmetros (varredura de parâmetros). O conjunto de resultados correspondentes a uma amostra significativa do espaço de parâmetros nos proporciona então uma visão da resposta do sistema sob estudo a uma ampla variedade de possibilidades. Dentre as áreas da ciência que fazem amplo uso de varredura do espaço de parâmetros, podemos citar como significativas, além da área de materiais, os estudos do clima, prospecção de petróleo, testes de estresse de estratégias de investimentos, sismologia, e geodinâmica. Em todos esses casos, um problema que demanda varredura de parâmetros exige a preparação dos arquivos com os dados de entrada para um grande número de tarefas, execução destas tarefas, análise dos resultados e tomada de decisões quanto a possível iteração de todo o processo.

O aprimoramento contínuo de modelos exige, usualmente, um crescente número de parâmetros a serem considerados, bem como grades numéricas mais finas. Como consequência, tais cálculos exigem crescente poder computacional e um envolvimento humano cada vez maior no manejo do processo como um todo. Entretanto, enquanto a natureza da abordagem computacional está mudando dramaticamente, o desenvolvimento em física computacional continua seguindo paradigmas estabelecidos nos primórdios da era dos computadores. O presente artigo visa discutir a adoção de novos paradigmas de desenvolvimento em física computacional de forma a atender as demandas da ciência do século 21.

2 Antecedentes

A visão até hoje predominante é que um cálculo em física computacional se define pela representação de uma instância de um problema em um computador de programa armazenado de Von Neumann. Neste modelo, o computador é dotado de uma unidade central de processamento (CPU) responsável pela execução das operações e de uma unidade separada de armazenamento (memória) diretamente acessível pela CPU, que armazena instruções e dados.

O problema computacional é então visto como uma coleção de um número indeterminado, presumivelmente infinito, de instâncias do problema físico que admite uma solução para cada instância. Cada instância é definida por um conjunto válido de valores dos parâmetros do problema físico, e representada em computador por um particular conjunto de dados armazenados na memória (contexto). A solução do problema é obtida sob a forma de um algoritmo, que é implementado sob a forma de um programa. Este, por sua vez, é entendido como uma sequência de instruções que devem ser executados sobre algum contexto.

Com a evolução dos sistemas de computador, esse paradigma foi adequadamente adaptado para incorporar os avanços em computação paralela. Do ponto de vista do programador, paralelismo de instrução e de dados são aproximadamente transparentes, pois um modelo de equivalência ao modelo de execução sequencial é implementado pelo compilador e pelo hardware. Paralelismo de tarefa, entretanto, nem sempre pode ser eficientemente implementado através de equivalência ao modelo sequencial. Este é usualmente tratado como uma coleção de instâncias de sequência de instruções (threads) que executam concorrentemente e podem trocar dados e sinais de sincronismo entre si. A troca de dados pode ocorrer tanto mediante compartilhamento de acesso memória quanto mediante troca de mensagens. A execução paralela de tarefas é portanto vista como um caso particular de execução concorrente em que: A) Todas as threads cooperam na solução da mesma instância do problema físico; B) E-

xiste acoplamento do ciclo de execução das threads, quer seja por necessidades de troca de dados ou de sincronismo.

Aplicações de varredura de parâmetros não se encaixam nos paradigmas descritos acima. Enquanto inequivocamente existe uma instância de um problema abrangente caracterizada por uma grade de pontos no espaço de parâmetros, sua solução é uma sequência de etapas, algumas das quais se decompõem em uma grande coleção de instâncias de um problema menor. Dentro de uma mesma etapa essas instâncias são desacopladas e não requerem compartilhamento de contexto. Na maioria das vezes, até mesmo o requerimento de compartilhamento de acesso ao ambiente de arquivos pode ser removido pela simples replicação destes. Embora essas etapas possam ser implementadas sob a forma de um programa paralelo mediante a agregação de pontos do espaço de parâmetros, isso não seria economicamente efetivo. Suporte a paralelismo em larga escala é um recurso caro em um sistema de computação. Deve portanto ser reservado para os casos em que esse suporte é essencial, ou casos em que de outra maneira o sistema permaneceria ocioso (uso oportunístico). Essa agregação de pontos deve, portanto, se limitar ao necessário para garantir uma granularidade ótima à tarefa.

Aplicações tipo varredura de parâmetros são usualmente executadas sob a forma de workflows científicos, que podem ser conceituado como Coleção de ações, dados e seus inter-relacionamentos que implementam a automatização de um procedimento de trabalho no qual os dados são passados de ação para ação. Aqui, "ação" significa qualquer porção auto-contida de trabalho, espacial e temporalmente finita, que produz modificação nos dados e é implementada pela execução de uma tarefa. Em muitas aplicações, esses procedimentos de trabalho são executados manualmente ou com auxílio de pequenos scripts. O aumento substancial do número de tarefas torna o trabalho manual ineficaz e propenso a erros. Esses fatores requerem que o workflow seja formalmente incluído na solução formal dos problemas em física computacional. Assim, a solução passa a ser estruturada em três níveis: No nível mais baixo, componentes auto-contidas da solução são implementadas como programas tradicionais. No nível intermediário, a solução completa é implementada como um workflow envolvendo um número indeterminado de instâncias desses programas. O nível superior é a interface com o usuário. Essa estrutura é representada na figura 1 abaixo.

Como visto antes, a maioria dos workflows provenientes de aplicações tipo varredura de parâmetros permitem a execução de um número substancial de tarefas desacopladas, as quais podem atingir alto grau de concorrência. Essa característica permite explorar a capacidade de ambientes distribuídos. Ambientes distribuídos modernos, como grids [3], arquiteturas orientadas a serviços (SOA) [4], e sistemas de computação em nuvens (CC, de cloud computing) [5], são genericamente denominados "sistemas em infraestrutura cibernética"

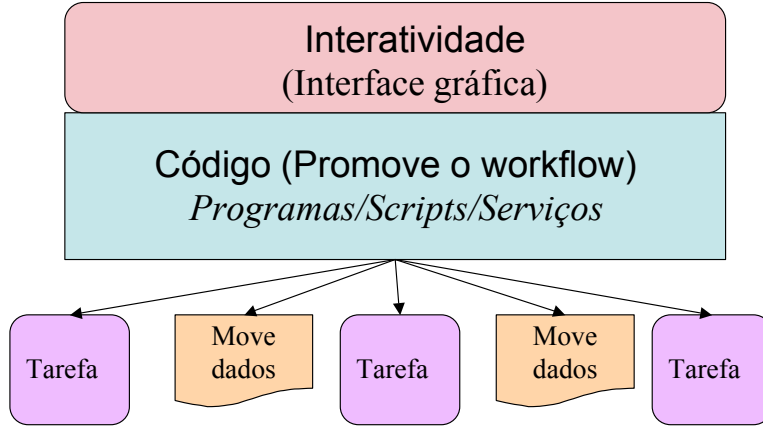


Fig. 1. Diagrama representado a estrutura de camadas de suporte à execução distribuída de workflows, como descrito no texto.

(CI) [6]. A execução das aplicações de varredura de parâmetros em CI é um dos principais objetivos da pesquisa corrente em sistemas distribuídos [6].

3 Caso: Propriedades termo-elásticas de ligas e soluções sólidas

Determinação das propriedades termodinâmicas e elásticas de materiais ao longo do domínio pressão x temperatura (P, T) também resulta em um cálculo de varredura de parâmetros. Cálculos de propriedades termodinâmicas de materiais exigem a determinação da equação fundamental, $G = G(P, T)$, onde G é a energia livre de Gibbs. Uma abordagem direta seria discretizar o domínio (P, T) e calcular $G = G(P_i, T_j)$ diretamente usando a técnica de dinâmica molecular. Isto iria requerer uma amostragem de parâmetros relativamente modesta, $\text{card}\{P_i\} \times \{T_j\} \approx 10^2$. Essa escolha seria, entretanto, altamente ineficiente porque a computação de cada ponto $G(P_i, T_j)$ por dinâmica molecular é de custo extremamente elevado. Uma abordagem alternativa consiste em utilizar a aproximação quase-harmônica (AQH) [7,8] que proporciona a energia livre de Helmholtz, $F(V, T)$, em função do conjunto de frequências da rede, $\{\omega_{qj}\}$:

$$F(V_i, T) = E(V_i) + \sum_{qj} \frac{\hbar\omega_{qj}(V_i)}{2} + k_B T \sum_{qj} \ln \left(1 - \exp \left[-\frac{\hbar\omega_{qj}(V_i)}{k_B T} \right] \right) \quad (1)$$

Neste caso, a temperatura não precisa ser discretizada, mas precisamos encontrar o volume de célula de equilíbrio, $\{V_i\}$, do material para cada pressão, $\{P_i\}$. Subseqüentemente, para cada volume $\{V_i\}$, as frequências $\{\omega_j\}$ precisam ser calculadas em um grade fina de pontos $\{\mathbf{q}\}$ da zona de Brillouin. Para gerar essa grade, a matriz dinâmica do sistema é calculada para um conjunto menor

de pontos $\{\mathbf{q}\}$ pertencentes a parte irreduzível da zona. Este conjunto de pontos é usado para calcular a matriz constante de força, que é, então, usada para gerar a grade fina. Essa abordagem requer uma amostragem de card $\{V_i\} \times \{\mathbf{q}\} \approx 10^{2-3}$ pontos. Os cálculos devem ser executados de forma altamente precisa, normalmente por meio da teoria funcional da densidade (DFT) [9,10] (ab-initio). A célula de equilíbrio para cada pressão é obtida mediante o uso de dinâmica molecular de célula variável [11,12], usando forças e estresses calculados a partir da solução da equação de Kohn-Sham [10]. A dinâmica de rede é então calculada por teoria de perturbação em DFT [13,14]. Este método requer, para células grandes, um número expressivo de operações aritméticas possivelmente superior a 10^{14} por ponto (V_i, \mathbf{q}) do espaço de parâmetros.

Na maioria dos casos, o cálculo correspondente a cada ponto da amostra do espaço de parâmetros exige a execução de uma tarefa, sendo cada tarefa completamente desacoplada das demais.

Estudos como o caso das propriedades termodinâmicas, descrito acima, envolvendo algumas centenas de tarefas, já é suficiente para justificar o desenvolvimento de um sistema em infraestrutura cibernética. Além da óbvia necessidade de se agregar capacidade computacional, existe também o aspecto humano: Preparar centenas de arquivos de entrada, submeter uma multitude de jobs, re-coletar centenas de arquivos de resultados, extrair destes a informação desejada e executar os procedimentos de análise é uma tarefa que consome grande quantidade de recursos humanos. Adicionalmente, a execução manual dos procedimentos acima é extremamente propensa a erros.

Extensões desse tipo de estudo à determinação das constantes elásticas de materiais, ligas desordenadas, e interação de impurezas com defeitos, por exemplo, podem requerer amostragens superiores a 10^4 pontos. Para calcular as constantes elásticas é necessário considerar um conjunto de possíveis deformações do cristal ϵ_{ij} , adicionando uma dimensão extra ao espaço de parâmetros. O número de direções das deformações depende da simetria do cristal, e o número de amplitudes por direção depende do problema particular sob estudo. O número total de deformações a serem calculadas pode ser tão alto quanto 10^2 [15].

Ligas desordenadas e soluções sólidas requerem uma dimensão extra do espaço de parâmetros correspondendo a uma amostragem de configurações. O método mais frequentemente utilizado é a expansão em agregados (CE) [16]. Este método permite o cálculo da energia livre de uma configuração arbitrária σ do sistema a partir de um hamiltoniano análogo ao modelo de Ising:

$$E(\sigma) = \sum_{\alpha} m_{\alpha} J_{\alpha} \left\langle \prod_{i \in \alpha^*} \sigma_i \right\rangle \quad (2)$$

Onde α é o índice dos agregados, σ_i é a ocupação (pseudo-spin) do sítio i . O produtório é calculado sobre as configurações equivalentes mediante as transformações de simetria do cristal puro e m_α é número de agregados equivalentes por simetria. Os coeficientes de expansão J_α são obtidos mediante ajuste da energia proporcionada pelo hamiltoniano de Ising a um conjunto de configurações representativas $\{x_i\}$ ($card \{x_i\} \approx 30 - 50$) calculadas por primeiros princípios. O processo de escolha das configurações usadas no ajuste pode ainda ser automatizado [17].

Um sistema capaz de habilitar cientistas a desenvolver essa classe de estudos tem de preencher alguns requerimentos: i) Agregar capacidade computacional; ii) Prover escalabilidade dinâmica, descobrindo e agregando mais capacidade na medida da necessidade; iii) Confiabilidade; iv) Disponibilidade; v) Usabilidade, tanto no tocante facilidade de uso, quanto na capacidade de potencialização do trabalho do usuário.

4 Visão do papel dos sistemas em infraestruturas cibernéticas em problemas de amostragem de parâmetros

O desenvolvimento de CI para problemas de amostragem de parâmetros é motivado pelo papel que estes podem desempenhar como habilitadores de ciência. A tendência mundial, entretanto, é no sentido de prestigiar projetos multidisciplinares, colaborativos e de amplo impacto transformativo nas comunidades que o cercam. O papel de habilitador de ciência, por si só, não é suficiente para garantir os três requisitos adicionais listados acima. Felizmente, tais CIs podem proporcionar benefícios muito mais abrangentes: Para atuar como habilitadora de ciência, uma infraestrutura básica precisa ser implementada, sendo esta a parte mais onerosa do desenvolvimento. Uma vez implementada a infraestrutura básica abre-se a oportunidade de, a um custo marginal, definir e implementar novos papéis para esta.

4.1 Habilitador de ciência

Para atuar como habilitador de ciência, um sistema de CI deve ser capaz de solucionar dois grandes problemas: a) Em uma amostragem de parâmetros, com centenas ou milhares de pontos, o procedimento de preparar arquivos de entrada, executar as tarefas em diferentes recursos, coletar os resultados dos arquivos de saída, codificá-los na forma adequada para análise, e eventualmente repetir todo o processo, é inviável se tiver de ser executado manualmente; b) A execução de estudos ambiciosos requer um número formidável de operações aritméticas, eventualmente superior a 10^{19} . O uso de base de ondas planas

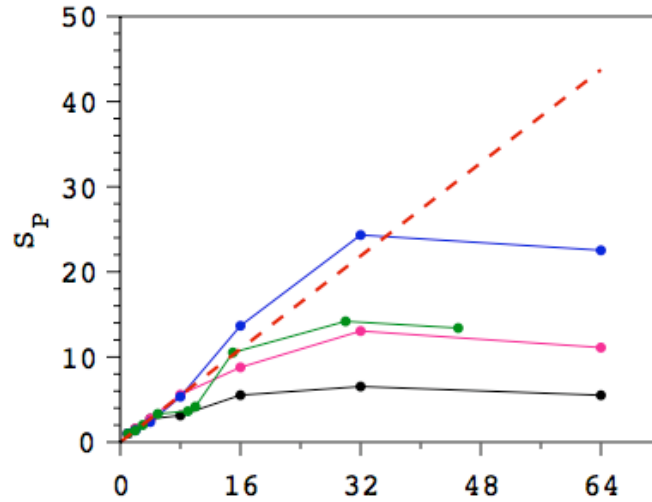


Fig. 2. Speed-up paralelo em um Altix XE 1300. Linha azul: Grade FFT=128x128x128; Vermelha: Performance projetada para até 16 tarefas rodando concorrentemente em 4 processadores cada. Outras curvas: FFT=90x90x90 para diferentes colocações de tarefas.

em métodos de primeiros princípios proporciona cálculos altamente precisos, porém cria um problema adicional: a escalabilidade paralela dos algoritmos usados nesses cálculos é limitada, como visto na figura 2. Isso decorre principalmente da alta parcela de operações executadas dentro de transformações rápidas de Fourier (FFT) [18].

Para solucionar o problema (a), o sistema deve satisfazer aos seguintes requerimentos: i) Proporcionar a execução automática de todo o procedimento de cálculo de varredura de parâmetros a partir de um conjunto mínimo de dados de entrada. Este conjunto deve ser aproximadamente equivalente à entrada típica de uma das tarefa responsável pelo cálculo de um único ponto. Isso inclui gerar as entradas para todas as tarefas, submeter as tarefas para execução, preferencialmente em ambiente distribuído, re-coletar os resultados, executar análise preliminar e iterar todo o processo se necessário. ii) Proporcionar ao usuário uma interface fácil de usar baseada em uma metáfora de fácil assimilação. Essa metáfora é baseada no conceito de projetos, que são abstrações em nível de usuário para instâncias de workflows. Eles criam na mente do usuário a ilusão de que o workflow é uma sequência de operações executadas sobre um conjunto de dados, a qual se desdobra em um número de tarefas executadas concorrentemente de forma distribuída. Esta é uma visão muito mais familiar maioria dos cientistas. Uma discussão muito mais detalhada sobre como o conceito de projetos é implementado pode ser encontrada nas referências [19] e [20].

Workflows podem ser implementados em diversos ambientes distintos. A princípio, grids parecem a opção mais natural. Proporcionam sólidos conjuntos de comandos e interfaces de programação (API). Um portal web de segunda

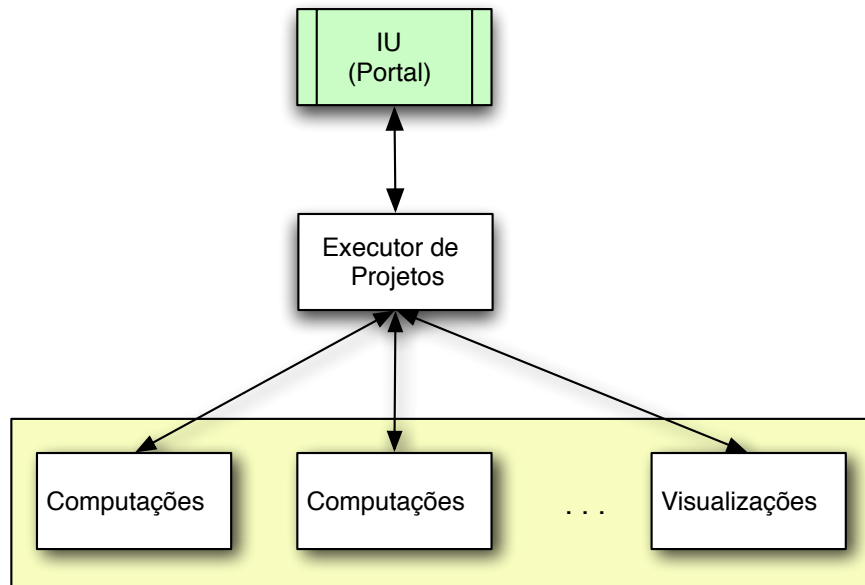


Fig. 3. Diagrama de uso das componentes principais, implementadas como web services, que foram usadas no VLab.

geração pode ser interfaceado com o grid para prover interface com o usuário. Transferências de arquivos e execução remota dos programas podem ser feitas usando-se linguagens de script avançadas. Contudo, APIs de grids são complicadas, o que entrava o desenvolvimento. A instalação, manutenção e administração de grids onera muito as equipes de suporte. Por essa razão, o uso de grids foi preterido em favor de uma arquitetura orientada a serviços para implementar o sistema descrito nas referências [19] e [20] (VLab). Como as operações de controle do workflow representam uma parcela insignificante da carga de trabalho, estas podem ser centralizadas em um único web server. Um código base para este web service já conta com uma biblioteca de classes para manejar as execuções remotas e as transferências de arquivos implicadas. Dessa forma, desenvolver um novo workflow importa em uma quantidade de trabalho semelhante a desenvolver uma classe em java para as operações de controle.

A implementação do conceito de projetos é viabilizada pela adoção do diagrama de uso para a SOA visto na figura 3, no qual aparecem apenas as componentes principais. As referências [19] e [20] mostram uma versão mais detalhada deste diagrama. O web service rotulado “executor de projetos” é responsável pelo controle da execução e monitoramento do workflow. Um conjunto de portlets, referido como “administrador de projetos” é responsável pelo manejo da interface com o usuário. Adicionalmente, cada um dos recursos computacionais que aparece na porção inferior da figura 3 roda um web service chamado de “executor de tarefas”, responsável pela interface do recurso com o resto do sistema.

iii) Prover suporte à atividade colaborativa, para que grupos de usuários possam compartilhar as tarefas de monitoramento e análise de resultados. iv) Prover uma diversidade de ferramentas de análise, visualização, etc..., dentro de uma abstração consistente de um ambiente integrado.

O problema (b) exige que o sistema satisfaça aos requerimentos: i) Suporte para computações distribuídas. Como visto na figura 2, a combinação de uma abordagem de computação paralela com computação concorrente em ambiente distribuído (linha vermelha) é economicamente muito mais efetiva do que uma abordagem puramente paralela. O uso de computação distribuída, entretanto, deve ser feito de forma transparente, provendo ao usuário a abstração de um sistema centralizado para atender ao requerimento (ii) do problema (a). ii) O sistema deve ser tolerante à falhas, de forma que ocorrências pro-saicas, como um lapso no fornecimento de energia provocando a queda de um servidor importante, por exemplo, não coloquem a perder um estudo de vários meses. iii) Scheduling otimizado: O scheduling de tarefas tem de ser otimizado [21] de forma que o desempenho real do sistema se aproxime o máximo possível da linha vermelha da figura 2, mantendo-se acima da melhor estratégia de colocação de tarefas em ambiente paralelo. O sistema deve ainda ser capaz de facultar ao usuário a escolha entre um scheduling orientado para o menor tempo de retorno dos resultados finais, menor custo, ou uma combinação desses objetivos.

4.2 Facilidade comunitária

Uma vez que o sistema satisfaça aos requisitos como habilitador de ciência, é necessário acrescentar muito pouco para habilitá-lo como uma facilidade comunitária [6]: i) O sistema deve ficar disponível para todos os membros da comunidade. Isso implica a necessidade um sistema de contas de acesso para múltiplos usuários. Também exige a padronização do método de acesso, o que é obtido pelo uso de tecnologias web. ii) Deve prover um conjunto de ferramentas adequadas às necessidades dessa comunidade.

4.3 Mediador de organização virtual

Para atender a esta finalidade, o sistema deve [6]: i) Ser acessível de qualquer parte do mundo. ii) Contar com um robusto suporte para trabalhos em colaboração. Este suporte deve permitir acesso compartilhado aos arquivos e tarefas que compõem os cálculos de varredura de parâmetros. Deve também incluir ferramentas de visualização e análise de dados que proporcionem visão sincronizada dos dados. Adicionalmente, deve preferencialmente trabalhar em conjunto com aplicativos de teleconferência.

Tanto o papel de mediador de organização virtual quanto o de facilidade comunitária, requerem adicionalmente que tal sistema seja facilmente extensível e adaptável a novos problemas.

5 Próximos passos

O sistema VLab [19,20] foi projetado para executar workflows de algumas centenas de tarefas. O estudo de ligas e soluções sólidas gera um número de tarefas em escala muito maior, o que exige a agregação de performance de um número de recursos computacionais também muito maior. O manejo de workflows dessa natureza exige facilidades de escalabilidade dinâmica e níveis de confiabilidade ainda ausentes no VLab. Para atingir essas metas e permitir uma estrutura extensível e adaptável é necessário empregar novos conceitos. Uma possibilidade promissora é importar conceitos da computação em nuvens (cloud computing). Uma estrutura proposta para atender a essas necessidades, e radicalmente diferente daquela das referências [19] e [20] pode ser vista na figura 4.

Não se espera que os fatores econômicos favoreçam a criação de nuvens de recursos a curto prazo. Isso pode ser viabilizado a longo prazo mediante implementação de uma organização virtual, em que os participantes contribuem com recursos computacionais para a nuvem. Embora a criação de nuvens autênticas seja inviável no curto prazo, ainda podemos usar muitos conceitos da computação em nuvem como princípio de projeto para garantir que a disponibilização destes recursos seja efetuada facilmente. Somente serviços de controle do sistema, como metadados e registro tem localizações pré-determinadas. Os demais serviços são localizados sob demanda mediante um serviço de broker do tipo publica/assina (publish/subscribe). Outro aperfeiçoamento crucial é a incorporação de meta-schedulers hierárquicos para viabilizar escalabilidade para milhares de tarefas. Adicionalmente é necessário introduzir o uso de repositórios para facilitar o arquivamento e re-uso de resultados gerados anteriormente, pseudo-potenciais, etc ...

6 Sumário

Neste artigo discutiu-se que estudos estado da arte, em muitas áreas, envolvem extensiva amostragem de parâmetros. Quando estas amostragens geram subproblemas desacoplados, os paradigmas de desenvolvimento tradicionais, baseados no modelo de computador de programa armazenado, se mostram inadequados. Esses estudos requerem a execução independente de muitas tarefas. Conclui-se que, para abordar adequadamente esse tipo de problemas, devemos

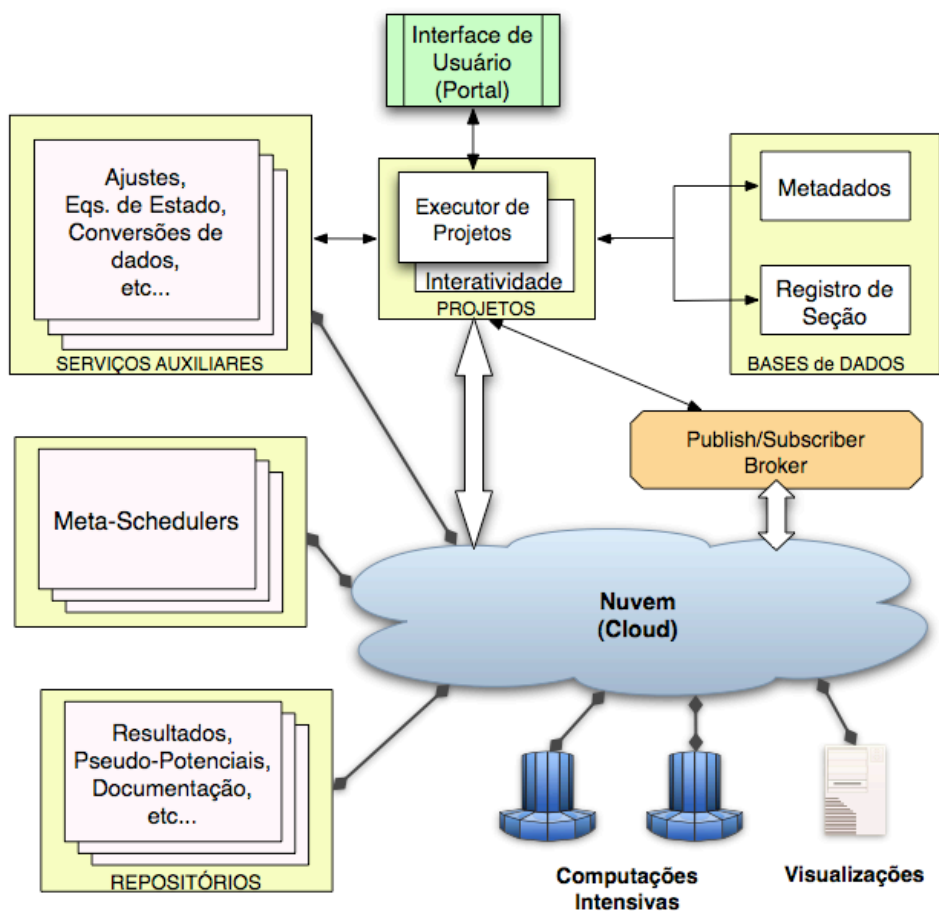


Fig. 4. Diagrama de visão de uso da nova SOA proposta para proporcionar execução de workflows compostos por milhares de tarefas. Neste caso, ênfase especial é dada aos aspectos de escalabilidade e tolerância a falhas.

estender os conceitos de física computacional, englobando codificação de workflows como parte integral da solução do problema. Sistemas em infraestrutura cibernética capazes de resolver esses problemas devem atender aos requerimentos mínimos de agregação de capacidade computacional, controle automático de execução de workflows e confiabilidade. Se o sistema for destinado ao uso como facilidade comunitária, devemos adicionar diversidade de ferramentas, disponibilidade e usabilidade. A existência de uma biblioteca de classes para facilitar o desenvolvimento de novos workflows é muito conveniente embora não seja estritamente necessária. O atendimento desses requerimentos resultou em um sistema pioneiro, VLab, cuja descrição foi revisada.

Novos requerimentos foram estabelecidos para sistemas futuros, capazes de manejar um número de tarefas duas ou mais ordens de grandeza maior: Escalabilidade dinâmica mediante agregação de capacidade computacional sob demanda e scheduling otimizado. Propõe-se usar conceitos de computação em nuvens como forma de atender esses requerimentos. Propõe-se também a

formação de organizações virtuais como forma de disponibilizar recursos computacionais suficientes para a criação de nuvens autênticas.

References

- [1] L F Magana and G J Vazquez J. Phys.: Condens. Matter 7 L393 (1995);
- [2] Wentzcovitch, RM, Justo, JF, WU, Z., da SILVA, Cesar R. S., Yuen, DA, Kohlstedt, D, PNAS. Proceedings of the National Academy of Sciences of the United States of America. , v.106, p.8447 - 8452, 2009.
- [3] Foster, I.; Kesselman, C.. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers (1999). ISBN 1-55860-475-8.
- [4] Gannon D, Plale B, Christie M, Fang L, Huang Y, Jensen S, Kandaswamy G, Marru S, Pallickara SFL, Shirasuna S, Simmhan Y, Slominski A, Sun YM, Service oriented architectures for science gateways on Grid systems, Lecture Notes in Computer Science 3826: 21-32 2005.
- [5] R. Buyya, C. S. Yeo, S. Venugopal, In HPCC '08: Proceedings of the 2008 10th IEEE International Conference on High Performance Computing and Communications (2008), pp. 5-13.
- [6]] Cyberinfrastructure Vision for 21st Century Discovery. National Science Foundation publication NSF-2007-28, Arlington, VA (2007).
- [7] R. E. Allen and F. W. de Wette, Phys. Rev. 179, 873 (1969).
- [8]] D. C. Wallace, Thermodynamics of Crystals, Wiley, New York, (1972).
- [9] Hohenberg, P., Kohn, W., 1964. Inhomogeneous electron gas. Phys. Rev. 136, B864B871.
- [10] Kohn, W., Sham, L., 1965. Self-consistent equations including exchange and correlation effects. Phys. Rev. 140, A1133A1138.
- [11] M. Parrinello and A. Rahman, J. App. Phys. 52, 12, 7182-7190 (1981).
- [12] Charles L. Cleveland, J. Chem. Phys. 89, 8, 4987-4993 (1988).
- [13] P. Giannozzi, S. de Gironcoli, P. Pavone, and S. Baroni, Phys. Rev. Lett. 43, 7231 (1991).
- [14] S. Baroni, A. Dal Corso, P. Giannozzi, and S. de Gironcoli, Rev. Mod. Phys. 73, 515 (2001).
- [15] P. Carrier, et. al., comunicao privada, a ser publicada.
- [16] D. De Fontaine, in Solid State Physics, ed. H. Ehrenreich, F, Seitz, and D. Turnbull, Academic, New York 73, 357 (1979).

- [17] A. Van de Walle, and G. Ceder, *J. of Phase Equilibria* 23 4, 348 (2002).
- [18] C. R. S. da Silva, D. A. Yuen, and A. P. Vincent, Performance and Scalability of a Simple User Serviceable Parallel Fast Fourier Transform For Geodynamical Applications. To be published.
- [19] P. R. C. da Silveira, C. R. S. da Silva, and R. M. Wentzcovitch, Metadata Management for Distributed First Principles Calculations in VLab - A Collaborative Grid/Portal System for Geo-materials Computation, *Comp. Phys. Comm.* 178, p. 186 (2008).
- [20] C. R. S. da Silva et al., Virtual laboratory for planetary materials: System service architecture overview, *Phys. Earth Planet. Int.* 163, 321 (2007). See also E. F. Bollig et al., *Phys. Earth Planet. Int.* 163, 333 (2007).
- [21] Y. Zhang, C. Koelbel, and K. Kennedy, Relative performance of scheduling algorithms in grid environments, In proc. of 7th IEEE International Symposium on Cluster Computing and the Grid. Rio de Janeiro, May 2007.