

1 Introdução

1.1 Motivação

Na resolução de problemas reais é necessário ter em conta os aspectos temporais sobre as ações a executar, que podem ter, cada uma, uma duração distinta. Da mesma forma, as ações podem requerer para sua execução o compartilhamento de recursos. Por exemplo, no contexto da pesquisa e da investigação científica, freqüentemente é necessário o envolvimento de aplicativos e processos que se encontram dispersos em várias fontes e que utilizam diferentes tipos de recursos computacionais. Normalmente, neste tipo de trabalho, um determinado dado produzido como resultado num processo i é utilizado como entrada do processo j .

Por exemplo, considere uma instituição responsável pela coleta e tratamento de informações na área de Geoinformática. O processo de analisar imagens de sensoriamento remoto e enviar as imagens para as agências de controle e proteção ambiental envolve a seguinte seqüência de ações:

1. Busque no repositório de imagens de sensoriamento remoto um conjunto R de imagens tal que R cobre toda a área de interesse e cada imagem em R foi adquirida em um intervalo de tempo t e cobre uma parte área da região que esta sendo procurada.
2. Para cada imagem em R , pré-processe a imagem, criando um novo conjunto S .
3. Para cada imagem pré-processada em S , analise a imagem buscando identificar atividades ilegais, criando uma lista de áreas com ameaças potenciais para o meio ambiente.
4. Envie um relatório para as autoridades responsáveis indicando as áreas com maior probabilidade de ocorrência de atos ilícitos.

Como um segundo exemplo, considere uma versão modificada do problema do *Ferry Boat* (figura 1). O objetivo deste problema é transportar os veículos V_1 e V_2 de uma margem do rio para outra, dentro de um determinado intervalo de tempo, tendo como recursos disponíveis o caminhão, a ponte e o *ferry boat*. Neste problema, podemos observar a existência de algumas restrições temporais, relacionadas ao posicionamento da ponte e ao carregamento dos veículos no *ferry boat* e no caminhão.

A ponte não é um recurso compartilhado, sendo portanto um fator limitador, já que deve estar suspensa para a utilização do *ferry boat* e abaixada para a utilização por parte do caminhão. O carregamento tanto do caminhão quanto do *ferry boat* dependem do posicionamento da ponte. Neste caso, temos restrições temporais em três níveis: no nível das ações, que são as restrições baseadas na duração das ações, estando relacionadas com a ordem de execução destas ações; no nível de recursos, onde as restrições estão relacionadas com a duração e com a não utilização compartilhada dos recursos; e no nível de atributos, onde são representadas as restrições entre os estados de um ou mais atributos dinâmicos como, por exemplo, um determinado atributo não pode alterar seu valor durante a janela de tempo em que esteja aguardando.

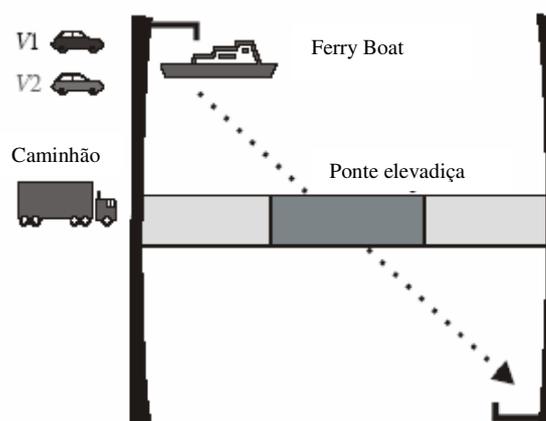


Figura 1 – Problema do Ferry Boat

Como um terceiro exemplo, considere uma lavanderia, onde cada etapa do processo deve ser realizada num intervalo de tempo T , com restrições de uso dos equipamentos, para realizar as seguintes tarefas:

1. Colocar a roupa na máquina de lavar.
2. Depois de lavada, colocar a roupa na máquina de secar.
3. Depois de seca, passar a ferro.
4. Depois de passada, arrumar a roupa no armário.

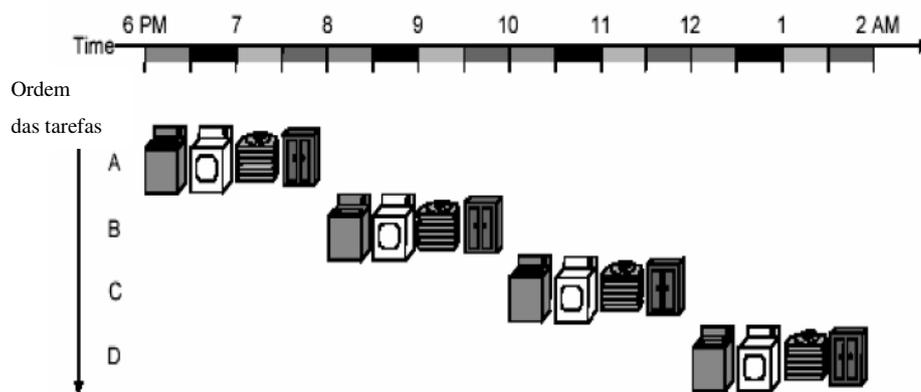


Figura 2 – Problema da Lavanderia

Supondo-se que cada uma destas etapas leve 30 minutos para ser realizada, a lavagem de um cesto de roupas continuará levando 2 horas para ser realizada. Entretanto, podemos iniciar a lavagem de um cesto de roupas a cada 30 minutos, até que tenhamos 4 cestos sendo lavados simultaneamente, um em cada etapa. Depois das primeiras 2 horas, teremos um cesto de roupa lavada a cada 30 minutos. Neste caso, com as restrições temporais impostas, em caso de atraso de uma tarefa, o processo como um todo pode sofrer atrasos e vir a não ser realizado, já que uma tarefa depende da outra.

Os exemplos apresentados descrevem situações em que devemos tomar decisões que estão sujeitas a restrições, sejam elas temporais, resultantes do uso de recursos ou ambos. Quando pensamos em soluções para estes problemas, pensamos em técnicas que nos permitam lidar com uma quantidade grande de variáveis e restrições, onde pode ser conveniente a utilização da paralelização ou distribuição do problema.

Estes problemas se agravam quando não temos um ambiente totalmente controlado, pois temos vários fatores que podem influenciar a obtenção de uma solução. Por exemplo, caso uma nova restrição seja adicionada, o espaço de busca de soluções aumenta. Em particular, este caso pode ser tratado como um *problema de satisfação de restrições* (PSR) (ver Figura 3), discutido em mais detalhe na seção seguinte.

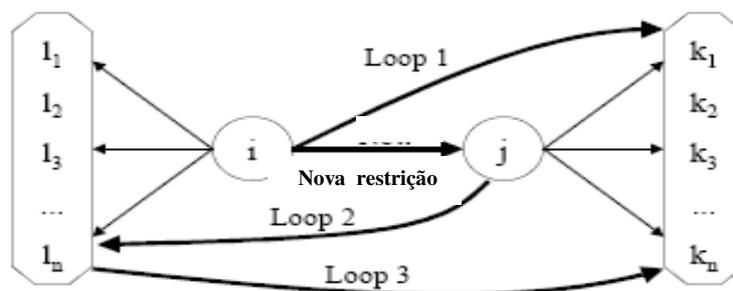


Figura 3 – Esquema de fluxo de tarefas com restrições

1.2 Abordagem Proposta

Os exemplos apresentados na seção 1.1 podem ser modelados como *plano*, no sentido em que um *plano* diz respeito à automatização de procedimentos, onde documentos, informações ou tarefas são passadas entre os participantes de acordo com um conjunto pré-definido de regras, para alcançar ou contribuir para um objetivo global de um negócio. Apesar de um *plano* poder ser manualmente organizado, na prática, a maioria dos *plano* são organizados dentro de um contexto de um sistema de informação para prover um apoio automatizado aos procedimentos [WfMC, 1995; WfMC, 1999; WfMC, 2004].

A modelagem dos *planos* deve considerar ainda restrições relacionadas com intervalos de duração, término de tarefas e programação de recursos. Assim, podemos pensar em diferentes tipos de restrições a serem consideradas na modelagem de um *plano*:

1. Restrições sobre a utilização de recursos;
2. Restrições sobre a ordem das ações; e
3. Restrições sobre início e término de atividades.

O conjunto de todas as restrições do problema, incluindo as derivadas dos objetivos que se perseguem, definem o *espaço solução* admissível. Note que podem existir ainda restrições preferenciais, ou seja, que tem prioridade em relação a outras, caracterizando a qualidade das decisões do escalonamento como, por exemplo, os tempos de término, custos, produtividade, níveis de disponibilidades de recursos [Fox, 1990]. Note ainda que as restrições podem conflitar umas com as outras e, portanto, a resolução de um problema de escalonamento também pode considerar quais restrições devem ser satisfeitas e quais deveriam ser relaxadas.

Esta classe de problemas é chamada de *problemas de satisfação de restrições* (PSR). A variante onde o tempo é o principal fator a ser considerado é chamada de *problemas de satisfação de restrições temporais* (PSRT), conforme trabalho de Decther, onde são definidas formalmente as relações e propriedades. A solução é obtida mediante a declaração das restrições do sistema e o conjunto solução é o conjunto que satisfaça a todas as restrições de forma ótima. O conceito de custo pode ser considerado, por exemplo, no cálculo do tempo total de uso de determinado recurso, processamento das tarefas, etc.

Este tipo de plano com restrições requer a utilização integrada de um planejador da execução das ações e de um escalonador para as ações. O escalonador deve receber informações sobre as ações planejadas, a lista de recursos a serem utilizados em cada ação e a ordem de execução das ações. O papel do escalonador será buscar uma solução dos possíveis conflitos que venham a ocorrer, reordenando de forma consistente as ações.

Este trabalho propõe inicialmente uma arquitetura que permita integrar o *planejamento* da execução de um conjunto de ações com o seu *escalonamento*, identificando os módulos necessários, suas funções e componentes. Em seguida, o trabalho descreve um algoritmo realizando planejamento e o escalonamento de forma integrada, respeitando as restrições impostas.

Como complemento, o trabalho apresenta duas ontologias com o objetivo de auxiliar na análise do domínio e na construção efetiva de um plano, permitindo

otimizar o processo de reparo e replanejamento quando necessário. Uma ontologia específica descreve formalmente as características da natureza do escalonamento, independentemente do domínio da aplicação e de como o problema será resolvido. Uma segunda ontologia provê uma estrutura que formaliza as distinções sobre estados e ações, funcionando mais como uma biblioteca de planos, independente da definição do domínio. Assim torna-se um veículo para a reutilização do conhecimento, facilitando a integração entre o processo de geração do plano e do escalonamento das tarefas de acordo com as restrições.

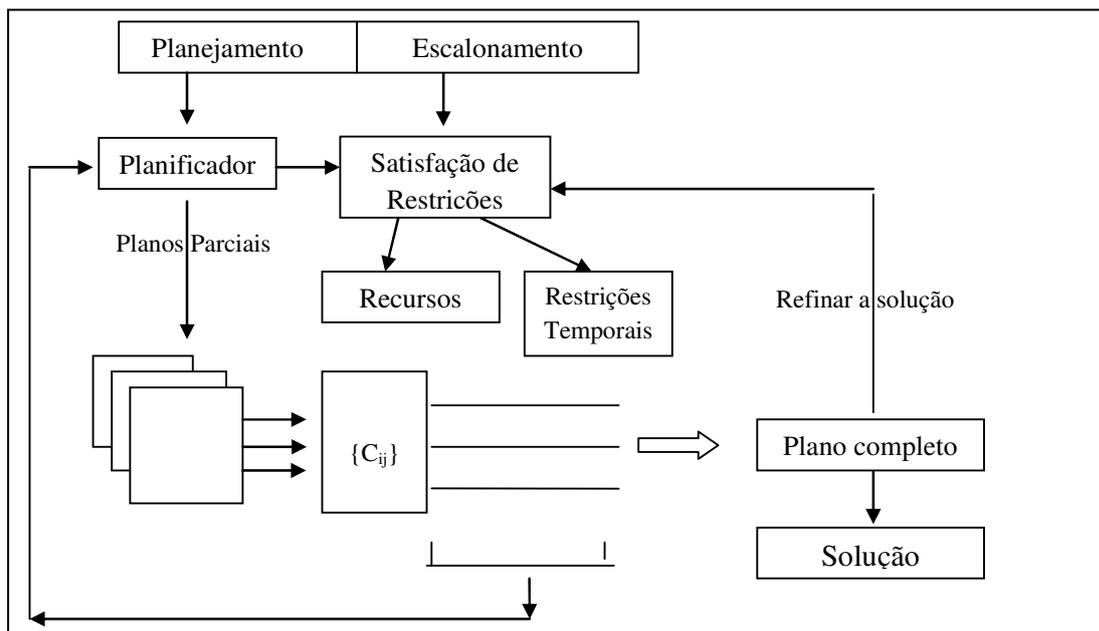


Figura 4 – Arquitetura de Integração

1.3 Contribuição da Dissertação

A principal contribuição da dissertação está relacionada a planejamento em presença de restrições temporais, com capacidade de reparar os conflitos, seja através do re-escalonamento dos recursos, seja realizando o replanejamento das ações ou ainda reparando o plano.

1.4 Organização da Dissertação

Esta dissertação está dividida nos seguintes capítulos. O Capítulo 2 apresenta uma breve discussão sobre os conceitos de planejamento,

escalonamento e tratamento de restrições temporais, conceituando formalmente o assunto e apresentando alguns trabalhos realizados na área. O Capítulo 3 define formalmente o problema de satisfação de restrições. O Capítulo 4 apresenta a solução proposta, que adapta o algoritmo QPRODIGY para acomodar as características e restrições do problema em estudo. Finalmente, o Capítulo 5 resume o trabalho realizado, apresenta as contribuições e sugere trabalhos futuros.