

UNIVERSIDADE ESTADUAL DE CAMPINAS
Faculdade de Engenharia Elétrica e de Computação
Departamento de Engenharia de Computação e Automação Industrial



**Heurísticas para sistemas APS utilizando janelas de
processamento: Interesse, conceitos e abordagens**

Autor: **Marcosiris Amorim de Oliveira Pessoa**

Orientador: **Prof. Dr. Luis Gimeno Latre**

Co-orientadora: **Prof. Dra. Maria Teresa Moreira Rodrigues**

Banca Examinadora:

Dra. Evelin Maria Abreu Teixeira (Consultora/Campinas/SP)

Prof. Dr. Luis Gimeno Latre (DCA/FEEC/UNICAMP)

Prof. Dr. Márcio Luiz de Andrade Netto (DCA/FEEC/UNICAMP)

Prof. Dr. Paulo Morelato França (DENSIS/FEEC/UNICAMP)

Dissertação submetida à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, para o preenchimento dos pré-requisitos parciais para obtenção do Título de Mestre em Engenharia Elétrica.

Dezembro - 2003

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

Pessoa, Marcosiris Amorim de Oliveira

P439h Heurísticas para sistemas APS utilizando janelas de
processamento: interesse, conceitos e abordagens /
Marcosiris Amorim de Oliveira Pessoa. -- Campinas, SP
:[s.n.], 2003.

Orientadores: Luis Gimeno Latre e Maria Teresa
Moreira Rodrigues.

Dissertação (mestrado) – Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Planejamento de Produção. 2. Escalonamento de
produção. 3. Programação heurística. I. Latre, Luis
Gimeno. II. Rodrigues, Maria Teresa Moreira. III.
Universidade Estadual de Campinas. Faculdade de
Engenharia Elétrica e de Computação. IV. Título.

Resumo

Este trabalho constitui a primeira parte do projeto de desenvolvimento de heurísticas de alocação de operações utilizando janelas de processamento. O objetivo deste trabalho é o de propor a utilização de heurísticas de alocação em ambientes “Advanced Planning and Scheduling” (APS) utilizando as janelas de tempo de processamento para problemas de programação da produção em que as datas de entrega dos produtos finais são importantes. Inicialmente são apresentados os tipos de heurísticas utilizadas nos sistemas APS comerciais. A seguir, discutem-se as técnicas utilizadas para as propostas de novas heurísticas, especificamente o sistema de planejamento PCPIP (Planejamento e Controle de Produção em Indústrias de Processos) para geração das janelas (desenvolvido em 2000 pela equipe de pesquisa em que foi desenvolvido este trabalho), as técnicas de propagação de restrições e os sistemas de programação da produção baseados em busca orientada por restrições (CBS). Finalmente foram propostos três tipos de heurísticas: heurísticas alocando no início ou fins das janelas de processamento, heurísticas utilizando a informação do carregamento dos equipamentos para detectar equipamentos gargalo e heurísticas de busca em árvore limitada com decisão de programação na forma de modificação das janelas por parte do usuário.

Abstract

This work constitutes the first part of the project developing schedules heuristics using tasks processing time windows. The objective of this work is to consider scheduling heuristics in APS environments using the tasks processing time windows for scheduling problems with important due dates. Initially some types of heuristics used in commercial Advanced Planning and Scheduling (APS) systems are presented. Some techniques utilized in the development of the new heuristics are discussed, specifically the planning system PCPIP for generation of the tasks time windows (developed in 2000 for the research team where this work was developed), the techniques of propagation of constraints and the scheduling systems based in constrained based search (CBS). Finally three types of heuristics have been proposed: heuristics allocating at the beginning or end of the task time window, heuristics using the aggregated demand of the equipments to detect equipment units bottlenecks, and limited tree search heuristics with programming decisions in the form of modification of the tasks time windows by the user.

Aos meus pais Sonia e Bernardo - i.m.

À minha esposa Simone

Agradecimentos

Ao professor Luis Gimeno Latre pela orientação, por todos os ensinamentos concedidos, e pelo fato de além de ter sido um grande orientador sempre foi um grande amigo.

À professora Maria Teresa Moreira Rodrigues por toda a orientação e apoio durante o desenvolvimento deste trabalho.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela bolsa concedida, o que tornou possível a realização deste trabalho.

À diretoria da Empresa TECMARAM por toda a atenção e suporte na utilização do sistema PREACTOR durante o desenvolvimento deste trabalho.

Aos meus pais Sonia e Bernardo - i.m., que sempre me apoiaram em todos os meus passos e me deram todo o suporte necessário para nunca ter me faltado nada.

À minha esposa Simone por toda a compreensão, apoio e incentivos.

Aos meus grandes amigos José Antonio Sánchez, Diana e Elizabeth por terem me acolhido como parte da família, e por todo apoio fornecido.

Aos amigos Richard Montesco, Wesley Alvarenga, Alessandro Borjas, Luis A. Ramírez, Lizet Liñero, José Federico, Eduardo, Tatiane, Vicente Delgado, Claudio Aparecido, Giovani, Renato Cazangi, Plinio, Sarajane, Ivette Luna, Mabilia, Ivana, Naur, Paulo e a todos os outros amigos da FEE/UNICAMP e da FEQ/UNICAMP que sempre me apoiaram.

A todos os professores e funcionários da FEE/UNICAMP pelo companheirismo demonstrado durante todos os anos em que convivemos.

Agradeço a Deus por tudo o que ele nos oferece.

Sumário

Resumo	i
Abstract	ii
Agradecimentos	iv
1 Introdução	1
1.1 Visão geral da tese	2
1.2 Sistemas APS e heurísticas de alocação	2
1.3 Sistema PREACTOR	8
2 Análise das técnicas de alocação em sistemas APS (Advanced Planning and Scheduling)	9
2.1 Introdução	9
2.2 Alocação por ordens de produção	10
2.2.1 Alocação no instante mais cedo	12
2.2.2 Alocação no instante mais tarde	14
2.2.3 Alocação bidirecional	16
2.2.4 Minimização do tempo de espera (Min Work in Process - Min WIP) . .	18
2.2.4.1 Minimização do tempo de espera para frente	18
2.2.4.2 Minimização do tempo de espera para trás	21
2.3 Alocação por operações, heurística APS Relógio	23

2.3.1	Alocação de acordo com a lista de ordens de produção	23
2.3.2	Alocação utilizando regras heurísticas de seleção	28
2.4	Representação rede estado-tarefa	31
2.5	Emulador	32
2.5.1	Heurísticas de alocação no instante mais cedo e no instante mais tarde .	33
2.5.2	Heurística APS Relógio	36
2.6	Exemplos	38
3	Janelas de processamento, propagação das restrições e técnicas CBS	44
3.1	Introdução	44
3.2	Janelas de processamento e a sua geração na fase de planejamento	44
3.3	Propagação de restrições	51
3.3.1	Restrições de capacidade	51
3.3.2	Restrições de precedência por receita	54
3.3.3	Restrições por monotonicidade	55
3.3.4	Restrições de armazenagem	55
3.3.5	Exemplos	56
3.4	Utilização das janelas de processamento nas técnicas de scheduling da busca orientada pelas restrições (CBS)	60
3.4.1	Representação de gargalos	62
3.4.2	Exemplo	64
3.5	Heurísticas APS em um cenário com datas de entrega importantes	65
4	Propostas de utilização das janelas de processamento nas heurísticas de alocação	71
4.1	Introdução	71
4.2	Heurísticas de alocação no instante mais cedo ou mais tarde	72

4.3	Heurísticas orientadas pelo gargalo	78
4.4	Heurísticas de modificação interativa das janelas de processamento com busca limitada	82
5	Conclusões	84
	Referências Bibliográficas	86

Capítulo 1

Introdução

Este trabalho constitui a primeira parte do projeto de desenvolvimento de heurísticas de alocação de operações utilizando janelas de processamento. A segunda parte é a Tese de Mestrado “*Heurísticas para sistemas APS utilizando janelas de processamento: Propostas, implementação e exemplos*” (Montesco, 2003).

A geração e utilização de janelas de processamento em problemas de planejamento e programação é uma linha de pesquisa em andamento desde algum tempo na Faculdade de Engenharia Elétrica e Computação, e na Faculdade de Engenharia Química da UNICAMP. Como principais referências cabe citar: Rodrigues M. (Rodrigues et al., 2004), (Rodrigues et al., 1996), (Rodrigues, Latre e Rodrigues, 2000a), (Rodrigues, Latre e Rodrigues, 2000b), Gimeno (Gimeno et al., 2000), Rodrigues L. (Rodrigues, Graells, Canton, Gimeno, Rodrigues, Espuña e Puigjaner, 2000), (Rodrigues, 2000), Alvarenga (Alvarenga, 2001).

O objetivo deste trabalho é propor a utilização de heurísticas de *scheduling* em sistemas APS (*Advanced Planning and Scheduling*) utilizando janelas de processamento para resolver problemas em que as datas de entrega dos produtos finais sejam importantes. Inicialmente são apresentados os tipos de heurísticas utilizadas nos sistemas APS comerciais analisando as suas vantagens e inconvenientes. A seguir discutem-se as técnicas utilizadas para as propostas de novas heurísticas, especificamente o sistema de planejamento PCPIP (Rodrigues, Latre e Rodrigues, 2000a) para geração das janelas, as técnicas de propagação de restrições e os sistemas de programação da produção baseados em busca orientada por restrições. Finalmente à luz dos problemas das heurísticas APS e das ferramentas analisadas propõe as

linhas de desenvolvimento de novas heurísticas.

Na segunda parte do projeto o objetivo é propor, implementar e testar estas novas heurísticas de alocação para situações em que as datas de entrega dos produtos são importantes. Para isto foram utilizadas diversas técnicas que se apóiam na existência de janelas de processamento para os lotes.

1.1 Visão geral da tese

No capítulo 2 são analisadas algumas das técnicas de alocação utilizadas em sistemas *APS* comerciais, com o objetivo de discutir os mecanismos que permitem uma execução rápida e a incorporação de restrições na planta. Utilizam-se exemplos simples para ilustrar o seu comportamento. Descreve-se um emulador desenvolvido para teste daquelas técnicas em exemplos mais complexos.

O capítulo 3 contém uma descrição resumida de várias técnicas e ferramentas utilizadas para o desenvolvimento das heurísticas objeto desta tese, especificamente a obtenção das janelas de processamento, os mecanismos de propagação das restrições e as técnicas de propagação de restrições utilizando busca orientada por restrições.

A proposta de novas heurísticas é o objeto do capítulo 4. Nele discutem-se três propostas à luz das limitações das heurísticas *APS* apresentadas no capítulo 2 e das ferramentas descritas no capítulo 3.

A tese de Mestrado “*Heurísticas para sistemas APS utilizando janelas de processamento: Propostas, implementação e exemplos*” (Montesco, 2003) apresenta a descrição detalhada e implementação destas heurísticas assim como o seu teste e análise sobre exemplos.

1.2 Sistemas APS e heurísticas de alocação

O texto desta seção foi extraído da referência (Gimeno e Rodrigues, 2002) com a devida autorização dos autores.

Dada a complexidade do problema global de planejamento e programação

diversos procedimentos heurísticos tem sido propostos na literatura e sucessivas famílias de softwares comerciais tem aparecido no mercado, os sistemas *APS* sendo os mais recentes. De alguma forma as abordagens heurísticas simplificam o problema através de um tratamento seqüencial dos diferentes aspectos envolvidos: determinação das necessidades de intermediários, atribuição de tarefas a equipamentos, determinação do tamanhos dos lotes e programação. Os sistemas *MRP* (*Material Requirements Planning*) também são um exemplo desta abordagem na medida em que tratam os três primeiros aspectos e deixam para a fase de análise de capacidade a resolução dos problemas eventuais de programação de produção. Deve-se enfatizar que o tratamento seqüencial pode ter uma justificativa prática quando as restrições do problema são tais que, por exemplo, as diversas opções de atribuição e/ou tamanhos do lote podem ser reduzidos drasticamente devido ao plano de demanda específico (*mix* de produtos e datas de entrega).

O tratamento seqüencial geralmente inicia com a seleção de uma demanda de um produto final procedendo-se ao cálculo das necessidades de intermediários e de matérias primas. Isto implica a seleção da(s) receita(s) de produção que serão utilizadas se existirem receitas alternativas e em qualquer caso a seleção dos tamanhos do lote. A seguir é selecionado o equipamento que será utilizado para cada tarefa ou lote e finalmente o instante de início de cada lote. O procedimento segue através da escolha de uma segunda demanda e assim sucessivamente.

A característica principal de um procedimento deste tipo é a construção seqüencial da solução de programação, ou seja a carta de Gantt. A satisfação de uma demanda $n + 1$ de um produto final implica um número determinado de lotes de determinadas tarefas; a sua alocação no tempo é feita partindo do estado de ocupação dos equipamentos, ocupação esta determinada pelas alocações anteriores necessárias para satisfazer as demandas $1, 2, \dots, n$ processadas anteriormente. Este procedimento seqüencial de alocação pode ser extremamente rápido e, mais importante, pela sua forma construtiva pode levar em consideração todas as restrições operacionais (ocupação de equipamentos, recursos cumulativos, *setups* dependentes de seqüência, turnos de trabalho, etc). Obviamente a contraparte é que a solução ótima da programação ou mesmo uma solução boa pode não ser encontrada.

Independentemente do procedimento de alocação as decisões de seleção comentadas anteriormente estão baseadas em heurísticas, por exemplo a escolha do equipamento que está disponível mais cedo, o que também não pode garantir a obtenção de uma boa solução. Tem-se assim a necessidade de algum tipo de retorno (*backtracking*) explorando outras seleções ou a geração de diversos cenários como proposto em (Mockus e Reklaitis, 1999) em que algumas seleções são randomizadas escolhendo o melhor cenário dentre os cenários gerados. Algumas referências anteriores a trabalhos nesta linha são (Egli e Rippin, 1986) e (Lazaro et al., 1989).

Os sistemas *APS* podem ser vistos como tendo dois objetivos principais interligados. Por um lado a obtenção rápida de soluções da programação da produção a partir dos dados de planejamento, por outro permitir ao usuário a modificação dos dados do planejamento em função dos resultados da programação.

Do ponto de vista do primeiro objetivo estes sistemas seguem a linha dos sistemas de programação com capacidade finita (*Finite Capacity Scheduling - FCS*) que foram os seus antecessores. Estes sistemas, por exemplo o Rhythm da empresa i2 (i2, 1996), dispõem de ferramentas para resolver os problemas de capacidade eventuais resultantes da explosão de necessidades no *MRP* distribuindo a carga nos intervalos de tempo discretos. Para tanto utilizam procedimentos eminentemente heurísticos mas rápidos e freqüentemente proprietários.

Em termos deste primeiro objetivo os sistemas *APS* respondem à crítica dos sistemas *ERP* (*Enterprise Resources Planning*) por parte dos responsáveis pela operação no chão de fábrica por não considerar as inúmeras restrições operacionais na execução do resultado do planejamento.

O segundo objetivo é enfatizado no próprio nome *Planning and Scheduling* em relação aos sistemas *FCS* cujo objetivo principal é resolver os problemas de capacidade finita na programação de produção. As tendências nos últimos anos de utilizar eficientemente a capacidade instalada, analisar a viabilidade de atender demandas suplementares e, no caso atende-las com precisão levaram a uma necessidade de interligar o planejamento e a programação. Esta interligação deve permitir medir corretamente as possibilidades reais de execução de planejamentos alternativos por parte do chão de fábrica, evitando tanto avaliações

pessimistas que levam a sub-utilização dos equipamentos quanto avaliações otimistas que levam a não satisfação de prazos de entrega.

A interligação é feita através da avaliação e visualização rápida do resultado da programação da produção para cada planejamento alternativo. São comuns avaliações de atrasos, tempos totais de processamento de lotes que podem ser facilmente medidas sobre a carta de Gantt. A questão é então a geração rápida de soluções de programação e isto é conseguido geralmente através da utilização de procedimentos heurísticos rápidos. Estes procedimentos heurísticos evidentemente não podem garantir que a solução ótima ou mesmo uma solução boa seja obtida. Geralmente os sistemas *APS* disponibilizam para o usuário diversas heurísticas e este se defronta com diversos cenários de programação e as suas avaliações respectivas para escolher a solução que julgar mais conveniente. A seguir são discutidos alguns dos procedimentos heurísticos utilizados e documentados na literatura, e as possibilidades de interação entre planejamento e programação.

As técnicas heurísticas para a programação das tarefas utilizadas pelos sistemas *APS* supõem sempre que as tarefas e a quantidade de lotes de cada uma delas tem sido previamente definidas. Este resultado procede de uma fase de planejamento anterior, à qual pode ser necessário voltar caso a programação obtida não seja satisfatória, mas que para cada cenário fornece os lotes envolvidos na satisfação de cada demanda de produtos finais. A partir desta informação as técnicas de programação utilizam basicamente três abordagens (Keng et al., 1998): a programação por demanda de produtos (*job based*) para frente (*forward*) ou para trás (*backward*), a programação por tarefas (*event based*) e a programação por gargalos (*resource based*).

Na programação por demanda de produtos é definida pelo usuário uma ordem de prioridades para as diferentes demandas. Seguindo esta ordem cada demanda é programada alocando no tempo em um único passo todos os lotes envolvidos. Estas alocações podem ser realizadas para frente (*forward*) a partir do instante mais cedo possível dada a ocupação dos equipamentos e a disponibilidade de matérias primas ou para trás (*backward*) a partir da data de entrega. Não se trata de um procedimento em que o tempo avance ou retroceda, porque cada vez que uma nova demanda é tratada o instante de início da programação deve voltar ao instante mais cedo em que é possível iniciar as alocações ou ao instante mais tarde anterior à data de

entrega.

A programação por tarefas utiliza um mecanismo de relógio para avançar na construção da carta de Gantt. No instante inicial da programação determinam-se os lotes, envolvidos em qualquer demanda, que podem ser iniciados dada a disponibilidade de matérias primas e eles são alocados sequencialmente utilizando algum tipo de classificação. Estas alocações levam a ocupações de equipamentos e o relógio é avançado até o menor instante de tempo em que algum dos equipamentos fica livre, repetindo-se o procedimento.

A programação por gargalos parte de uma definição do equipamento gargalo procedendo a alocação dos lotes atribuídos a este equipamento. A partir destas alocações utiliza dois procedimentos de alocação dos lotes restantes: para frente e para trás. Aloca para frente os lotes seguintes na receita e para trás os lotes precedentes na receita.

A descrição sucinta anterior permite discutir o interesse e os problemas com estas heurísticas. Por um lado são mecanismos extremamente rápidos e que podem levar em consideração todas as restrições na alocação das tarefas pela sua característica construtiva. Quando uma tarefa é alocada seja na opção para frente ou para trás é conhecida a alocação das tarefas que devem precede-la por balanço de massa ou as tarefas que precisam dela, portanto o seu instante de início mais cedo ou de término mais tarde está definido. Por outro lado qualquer outra restrição, como por exemplo tempos de preparação dependentes da sequência, recursos cumulativos compartilhados limitados, turnos de trabalho, etc. apenas implicam em modificações fáceis de levar em consideração nos instantes de início ou de término. Esta alocação sequencial pode ser corrigida para evitar estoques desnecessários de intermediários sincronizando, quando possível, a produção com o consumo na programação para frente e o consumo com a produção na programação para trás.

Os problemas com estas heurísticas construtivas são originados pela necessidade de mecanismos de estabelecimento de prioridades na programação por demandas de produtos e na programação por tarefas. Estes mecanismos definem a ordem em que serão feitas as alocações e portanto tem implicações diretas sobre a qualidade da solução; qualquer mudança nestas prioridades poderá dar lugar a uma carta de Gantt completamente diferente. Já na programação por gargalos o problema inicial é a programação do equipamento gargalo o que é em si mesmo um problema difícil e obviamente também com implicações fortes sobre a solução total.

As descrições das técnicas de programação pelos fabricantes (e.g. *i2*, *Oracle*, *Preactor*, *SAP*,...), empresas de consultoria (*AMR*, *Andersen - Ascet*,...) ou periódicos especializados (*APICS*,...) enfatizam a tendência à incorporação de técnicas de otimização como de programação mista ou programação orientada pelas restrições e busca em árvore; por exemplo o sistema *APO* da *SAP* (SAP, 1998) propõe um algoritmo *Branch & Bound* para minimizar o tempo total consumido em preparação (*setups*) para a programação do equipamento gargalo. Esta tendência é óbvia mas sempre deverá ser balanceada pelo acréscimo em tempo que traz; o compromisso será entre uma multiplicidade de cenários gerados rapidamente e um número menor de possibilidades resultado de otimizações locais mais demoradas.

A interação entre planejamento e programação da produção nos sistemas *APS* é viabilizada pela rapidez com que são obtidas as soluções da programação. Desta forma é possível analisar diferentes cenários de planejamento onde, por exemplo, sejam modificados dados de demanda e/ou de disponibilidade de recursos e materiais. Duas das funções mais discutidas na literatura são as denominadas *ATP - Available To Promise* e *CTP - Capable To Promise*. A primeira não é específica de sistemas *APS* porque não implica na análise de novos cenários de planejamento, trata-se de contabilizar as disponibilidades futuras de produtos finais e não comprometidas dada a programação de produção em execução, e as existências de estoques, para responder a novas solicitações de vendas. Já o *CTP* é um passo a mais porque o objetivo é analisar se novas demandas podem ser incluídas no plano de produção vigente, o que requer modificar os dados de planejamento e passar a fase de programação. Obviamente a rapidez com que podem ser analisados estes novos cenários permitem uma resposta rápida a estas novas indagações do setor de vendas, contudo a utilização de heurísticas pode levar a respostas negativas mesmo existindo soluções no problema de modificação da demanda. A literatura da área apresenta diversas propostas sobre outras funções na interação entre planejamento e programação, todas elas viabilizadas pela rapidez com que podem ser obtidas as soluções de programação.

1.3 Sistema PREACTOR

A análise das heurísticas de alocação típicas dos sistemas *APS* foi realizada sobre o sistema PREACTOR disponibilizado pela empresa TECMARAN representante responsável pela distribuição na América do Sul do Sistema PREACTOR. Agradecemos aqui toda a colaboração dada por esta empresa através da utilização do sistema e das diversas reuniões para esclarecimentos sobre as funcionalidades dos vários módulos do sistema.

O sistema PREACTOR na versão utilizada importa os resultados do planejamento de um módulo externo. Este pode ser um módulo *MRP*, por exemplo fazendo parte de um sistema *ERP* ou um módulo desenvolvido especificamente para interagir com os sistemas *APS*. Os dados de planejamento importados relevantes para esta tese são as ordens de produção, lotes ou bateladas e as relações de precedência entre eles dada a receita dos produtos a serem fabricados.

As heurísticas de alocação serão discutidas no capítulo 2. Para maiores informações sobre o sistema PREACTOR sugere-se consulta dos endereços www.preactor.com e www.tecmaran.com.br.

As discussões posteriores de heurísticas *APS*, mesmo falando genericamente, obviamente fazem referência ao sistema PREACTOR. Isto porque foi o único ao qual se teve acesso, mas os conceitos sobre a utilização das janelas, propagação de restrições, curvas de demanda agregada e as heurísticas propostas nesta tese podem ser utilizadas em qualquer sistema *APS*.

Capítulo 2

Análise das técnicas de alocação em sistemas APS (Advanced Planning and Scheduling)

2.1 Introdução

Neste capítulo são discutidas algumas das heurísticas de alocação mais freqüentes em sistemas *APS* comerciais utilizando como estudo de caso o sistema PREACTOR. Inicialmente são utilizados exemplos simples para ilustrar o comportamento das heurísticas sobre o próprio sistema PREACTOR (seções 2.2 e 2.3).

Os exemplos objeto deste trabalho são receitas de produção complexas em plantas descontínuas (operando por bateladas ou lotes). Elas envolvem tamanhos de lotes diferentes em operações sucessivas, condições de armazenagem limitada e produtos intermediários compartilhados por várias operações consumidoras. Uma representação adequada para este tipo de receitas é a rede estado-tarefa (*“state-task-network”*) descrito na seção 2.4.

A utilização do sistema PREACTOR com receitas complexas mostrou dificuldades. Algumas delas provavelmente poderiam ter sido resolvidas através de uma maior interação com os fornecedores do sistema, mas, com o objetivo de dispor de uma plataforma totalmente conhecida, optou-se pela construção de um emulador simples. Este emulador é descrito na seção 2.5 e utilizado com um exemplo na seção 2.6.

2.2 Alocação por ordens de produção

Nesta estratégia alocam-se ordens de produção no tempo, sendo que todas as operações que compõem uma ordem de produção são alocadas num mesmo passo. Para tanto, utiliza-se uma lista de ordens de produção previamente ordenada por prioridades ou simplesmente pela ordem de declaração, e uma lista ordenada de recursos habilitados.

Alocam-se todas as operações da primeira ordem de produção, em seguida, alocam-se as da segunda e assim por diante, até que não exista mais ordens para alocar. Cada operação é alocada no primeiro instante de tempo disponível do primeiro recurso habilitado para o processamento da operação. Desta forma quando uma operação está habilitada em mais de um recurso, ela é alocada no recurso da lista de recursos que estiver disponível mais cedo (seguindo a ordem declarada da lista de recursos em caso de empate).

É importante observar, portanto, que as alocações utilizam uma lista ordenada (por prioridades ou declaração) de ordens de produção, e uma lista ordenada de recursos habilitados para cada alocação. As prioridades podem ser prioridades de produtos, onde escolhe-se para alocar o produto mais priorizado, ou pode ser por data de entrega, onde alocam-se primeiro as operações que serão entregues o mais cedo. As estratégias *APS* de alocação por ordens de produção analisadas foram:

- Alocação no instante mais cedo, partindo do início do horizonte de planejamento;
- Alocação no instante mais tarde, partindo das datas de entrega das ordens de produção;
- Bidirecional partindo da programação de todas as ordens de produção em um equipamento gargalo intermediário na rota de produção da ordem, este caso compreende as duas estratégias anteriores: i) uma alocação no instante mais tarde das operações precedendo a alocação na máquina gargalo e ii) uma alocação no instante mais cedo das operações que seguem a partir da alocação na máquina gargalo;
- Minimização do tempo de espera para frente, utiliza dois passos: i) uma alocação inicial no instante mais cedo e ii) fixando a última operação e realocando as operações restantes a fim de diminuir os intervalos de tempo entre operações (sempre com operações da mesma ordem de produção);

- Minimização do tempo de espera para trás, utiliza dois passos: i) uma alocação inicial no instante mais tarde e ii) fixando a primeira operação e realocando as operações restantes a fim de diminuir os intervalos de tempo entre operações (sempre com operações da mesma ordem de produção).

As estratégias de alocação no instante mais cedo, e no instante mais tarde serão discutidas nas seções 2.2.1 e 2.2.2 utilizando o exemplo da tabela 2.1, as demais serão discutidas nas seções seguintes. A ordem para tratamento das ordens de produção é a ordem de declaração ou seja *A*, *B*, *C*, *D*, *E* e *F*. Quando uma operação tem mais de um recurso habilitado a ordem para seleção do recurso é a ordem declarada, ou seja, por exemplo, operação *B2*, lista de recursos habilitados (*R1*, *R2* e *R3*).

Ordens de produção	Seqüência de operações na ordem de produção	Recursos habilitados	Tempo de processamento	Prazo de entrega
A	A1	R1	3	18:00
	A2	R2	4	
B	B1	R2	5	18:00
	B2	R1 ou R2 ou R3	3	
C	C1	R1	3	18:00
	C2	R1 ou R2 ou R3	3	
D	D1	R1 ou R2 ou R3	3	18:00
E	E1	R1 ou R2 ou R3	3	18:00
F	F1	R1 ou R2 ou R3	3	18:00

Tabela 2.1: Ordens de produção, operações constituindo as ordens, recursos habilitados para as operações, tempos de processamento, e prazos de entrega.

2.2.1 Alocação no instante mais cedo

É uma alocação no sentido crescente de tempo, partindo do início do horizonte de planejamento $T = 0$, e começando pela primeira operação da primeira ordem até chegar na última operação da última ordem. A figura 2.1 representa a carta de Gantt resultante.

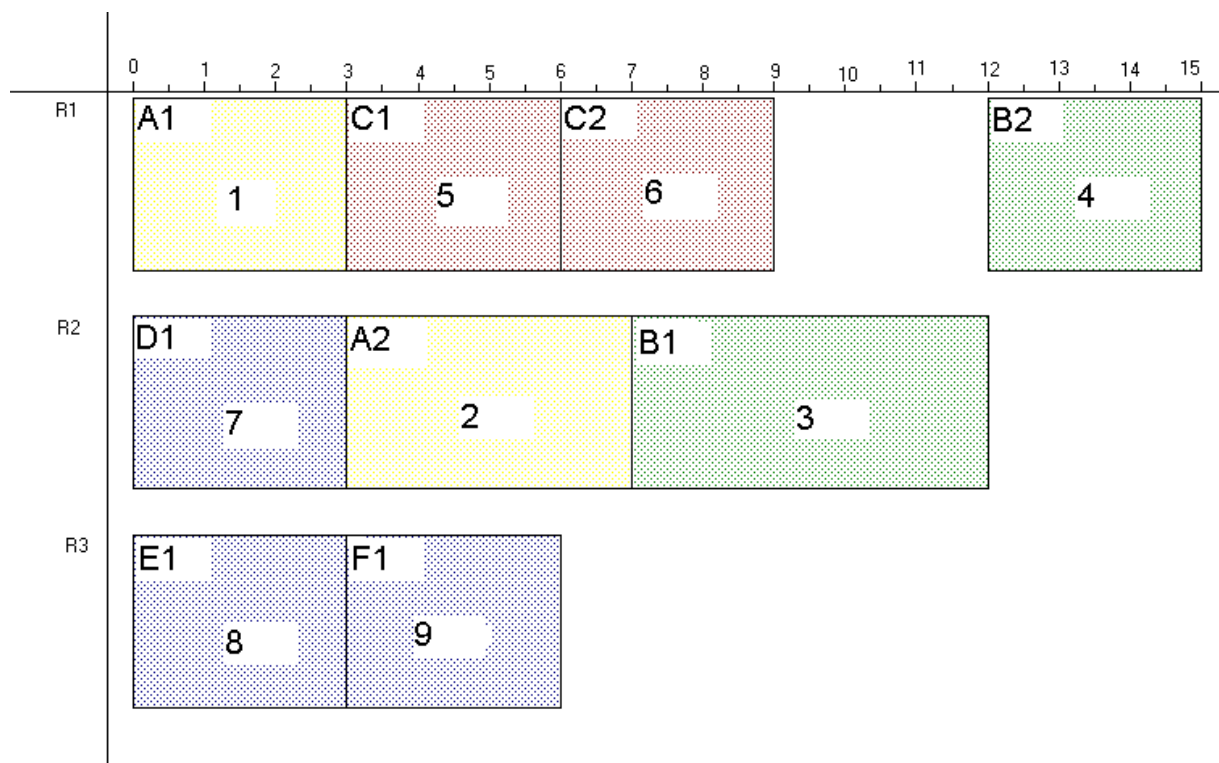


Figura 2.1: Alocação no instante mais cedo - Carta de Gantt enumerando as operações de acordo com a ordem de alocação conforme declarado na tabela 2.1.

As figuras 2.2 a 2.5 mostram sucessivamente a alocação das quatro primeiras ordens de produção (A-D) da tabela 2.1. A figura 2.2 mostra o primeiro passo, o de alocar as operações da ordem A. Aloca-se a operação A1 no primeiro tempo disponível do recurso R1, neste caso $T = 0$. Como a operação A2 é posterior a A1, ela é a próxima a ser alocada. Observa-se que fica uma janela de tempo não utilizada no recurso R2. A operação B1 somente pode ser alocada no recurso R2, e como a primeira janela da figura 2.3 é insuficiente, a operação é alocada no próximo tempo disponível possível $T = 7$. A operação B2 pode ser realizada em todos os recursos, mas ela é posterior a operação B1, desta maneira somente pode ser alocada após $T = 12$. Como todos os recursos estão disponíveis em $T = 12$, o critério de escolha seleciona o primeiro recurso da lista de recursos habilitados, no caso o recurso R1.

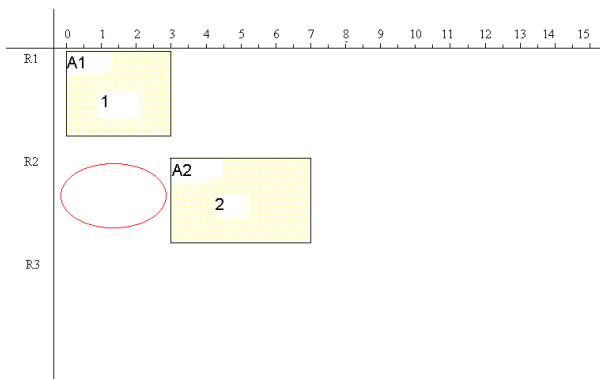


Figura 2.2: 1º passo, alocação das operações da ordem A.

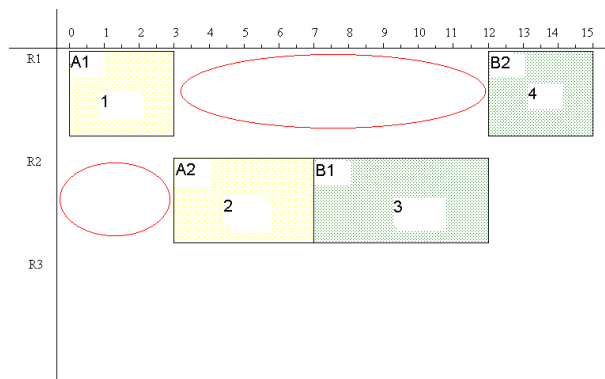


Figura 2.3: 2º passo, Alocação das operações da ordem B.

Na figura 2.4 observa-se que ao alocar a ordem C, a operação C1 é alocada no primeiro instante de tempo disponível do recurso R1. Desta maneira sempre que for possível o sistema ocupa os intervalos de tempo disponíveis. A operação C2 segue o mesmo critério de alocação da operação B2.

Ao alocar a ordem D, a operação D1 pode ser alocada em todos os recursos. Como o menor tempo de disponibilidade é $T = 0$ nos recursos R2 e R3 e ambos estão disponíveis no mesmo instante de tempo, a alocação é feita no primeiro recurso disponível da lista de recursos, o recurso R2, alocação esta representada pela figura 2.5.

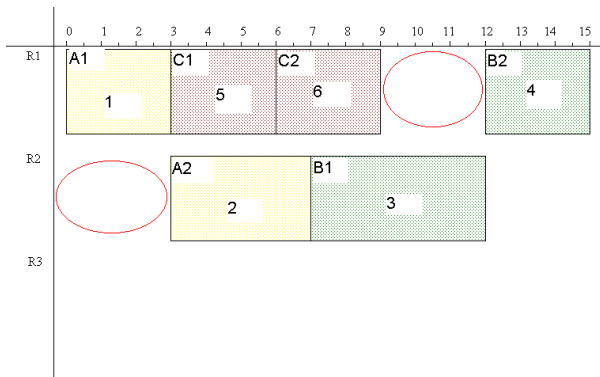


Figura 2.4: 3º passo, alocação das operações da ordem C.

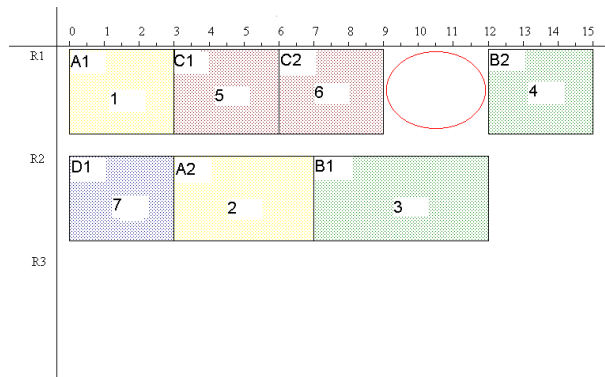


Figura 2.5: 4º passo, alocação das operações da ordem D.

2.2.2 Alocação no instante mais tarde

É uma alocação em sentido decrescente de tempo, iniciando a alocação pela data de entrega das ordens. Aloca-se inicialmente a última operação da ordem de menor prioridade ou da última ordem declarada, até chegar na primeira operação da ordem de maior prioridade ou a primeira declarada. A figura 2.6 mostra o resultado final.

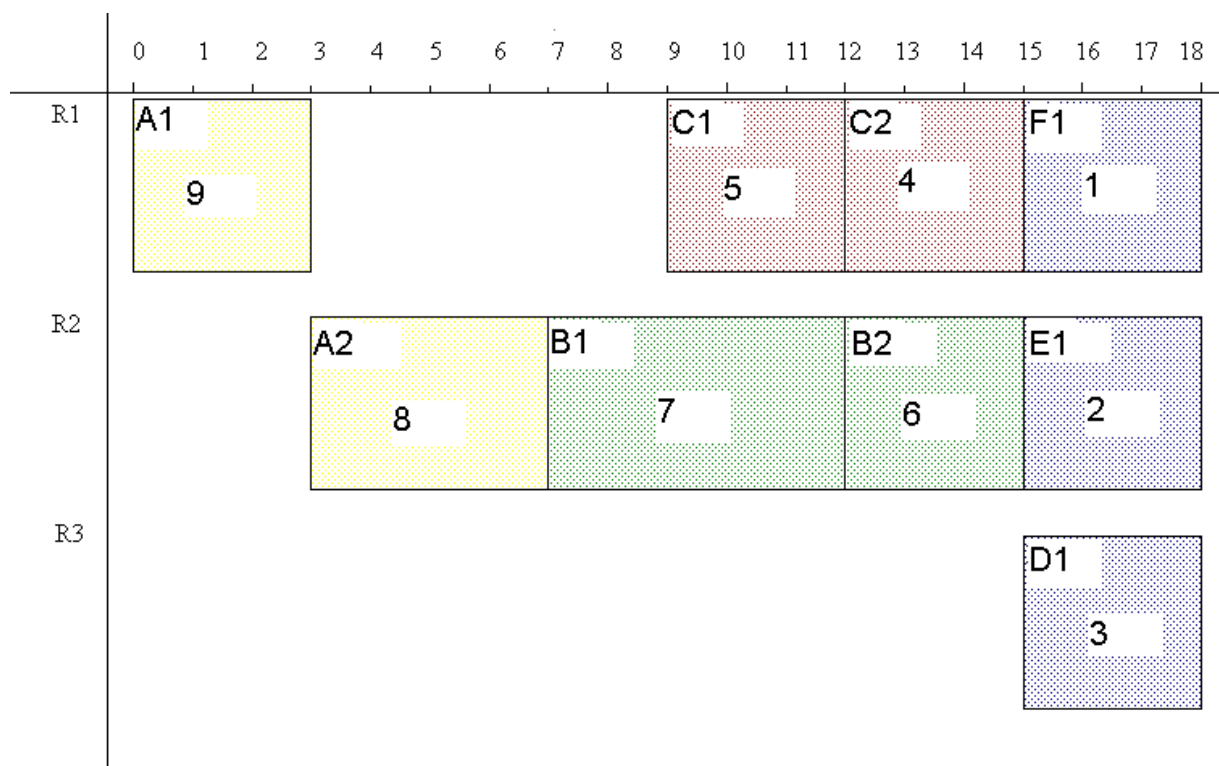


Figura 2.6: Alocação no instante mais tarde - Carta de Gantt enumerando as operações de acordo com a ordem de alocação conforme declarado na tabela 2.1.

As figuras 2.7 e 2.8 exemplificam a alocação de algumas ordens de produção da tabela 2.1. Na alocação da figura 2.7 a operação *F1* da ordem *F* pode ser realizada por todos os recursos. É escolhido o recurso que estiver disponível no intervalo de tempo mais próximo da data de entrega da operação *F1* ($T = 18$). Desta forma a alocação tem que ser feita no intervalo de tempo ΔT : (15 ; 18). Como neste caso os três recursos estão disponíveis neste intervalo, o sistema aloca a operação no primeiro recurso da lista de recursos habilitados, o recurso *R1*. A situação final é mostrada na figura 2.8, que ficou com uma janela de tempo não utilizada entre as operações *A1* e *C1*.

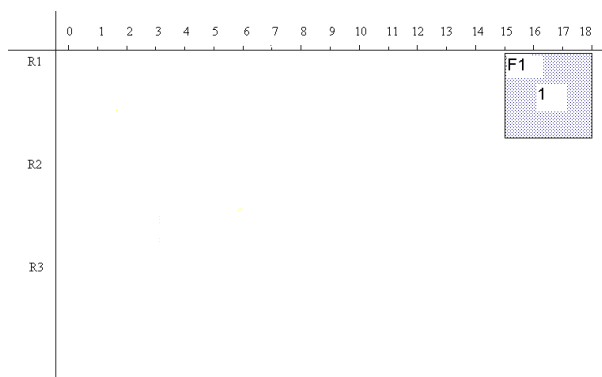


Figura 2.7: 1º passo, alocação da ordem F.

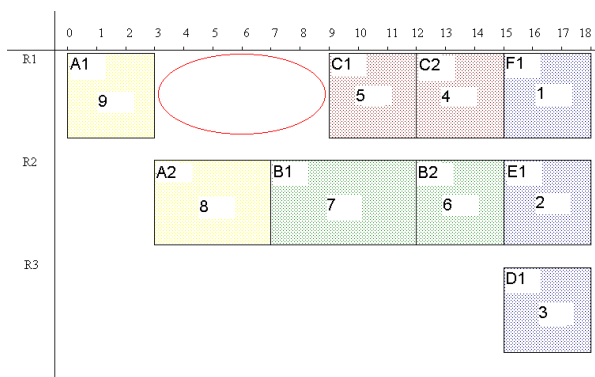


Figura 2.8: 6º passo, alocação da ordem A.

Adicionando-se uma nova ordem de produção G com a data de entrega em $T = 18$ antes da ordem de produção A na lista de ordens de produção, com a operação $G1$ habilitada em todos os recursos, e a operação $G2$ no recurso $R1$, obtém-se o resultado da figura 2.9. Pela figura 2.9 percebe-se que sempre que for possível, o sistema ocupa os intervalos de tempo disponíveis.

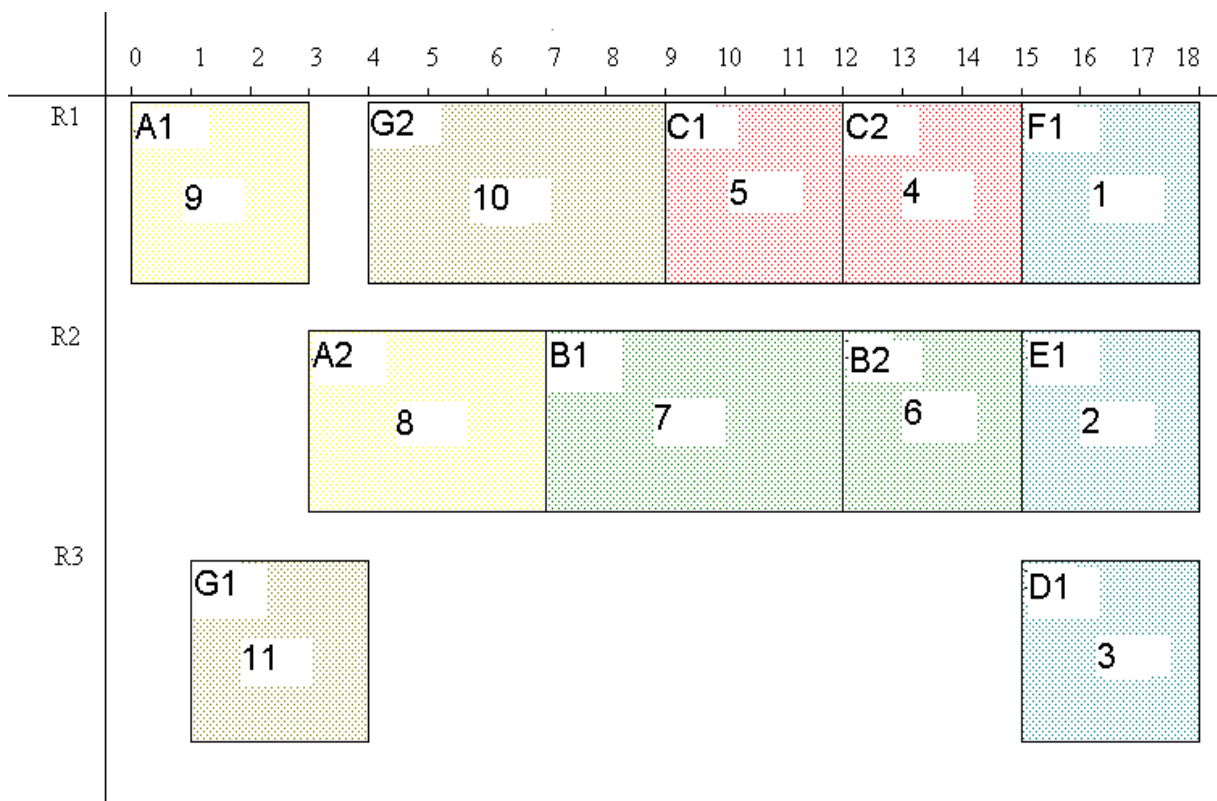


Figura 2.9: 7º passo, alocação da nova ordem a ordem G.

2.2.3 Alocação bidirecional

Esta estratégia parte da alocação de todas as ordens de produção na máquina gargalo. Consiste em: i) alocar inicialmente todas as ordens de produção no instante mais cedo ou no instante mais tarde, ii) fixar as operações que estão alocadas no recurso gargalo, iii) desalocar todas as operações que não estão fixas, iv) uma realocação no instante mais tarde das operações precedentes às operações fixas e uma realocação no instante mais cedo das operações posteriores às operações fixas, v) uma realocação no instante mais cedo ou mais tarde das operações restantes (operações que não tenham relação direta com as operações fixadas).

Esta heurística é útil quando está bem definido qual é o recurso gargalo, nela priorizam-se as ordens que utilizam os recursos gargalos alocando-as antes das outras ordens de produção, independentemente de qual seja a ordem de declaração ou do ordenamento da lista de prioridades das ordens de produção. Considerando o exemplo descrito pela tabela 2.2, e supondo que o usuário escolheu como recurso gargalo o recurso *R2*. A escolha do recurso gargalo é um dos problemas desta heurística dado que esta escolha compete ao usuário.

Ordens de Produção	Seqüência de operações na Ordem de Produção	Recursos habilitados	Tempo de Processamento
A	A1	R1	4
	A2	R2	5
	A3	R3	4
	A4	R1	5
B	B1	R1	2
	B2	R2	3
	B3	R3	3
	B4	R1	3
D	D1	R3	4
E	E1	R3	3
C	C1	R2	3
	C2	R3	3
	C3	R1	2
	C4	R1 ou R2	4

Tabela 2.2: Ordens de produção, operações constituindo as ordens, recursos habilitados para as operações, tempos de processamento.

A figura 2.10 mostra o primeiro passo, que é de alocar todas as ordens da lista de ordens de produção, neste caso foram alocadas no instante mais cedo. No segundo passo fixam-se as operações do recurso gargalo, em seguida desalocam-se as operações não fixadas,

conforme figura 2.11.

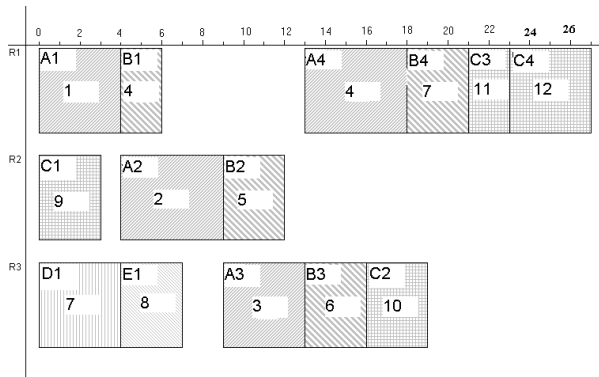


Figura 2.10: 1º passo, alocação das ordens de produção constantes na tabela 2.2.

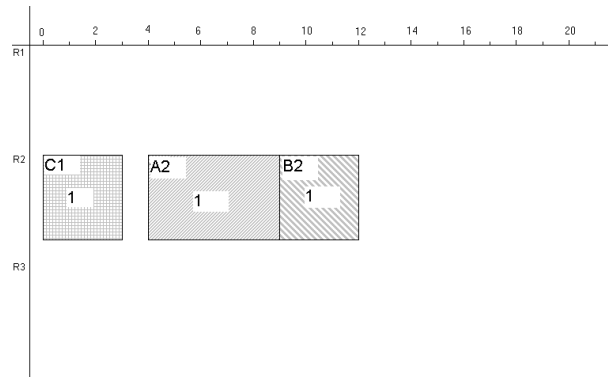


Figura 2.11: Operações do recurso gargalo fixadas.

O quarto passo é o de realocar bidirecionalmente as operações que tenham relação com as operações fixadas. Pode-se observar pela figura 2.12, que a estratégia bidirecional utiliza tanto a alocação no instante mais cedo, como no caso das operações *C2*, *C3* e *C4*, que utilizaram a alocação mais cedo a partir do instante de tempo final da operação *C1* ($T = 3$) quanto utiliza a alocação no instante mais tarde como no caso da operação *B1*, que foi realocada para trás a partir da operação *B2*. Percebe-se também que as operações que não pertencem as ordens de produção das operações fixadas, não foram realocadas (operações *D1* e *E1*). A figura 2.13 mostra o último passo que é o de realocar as operações restantes (*D1* e *E1*).

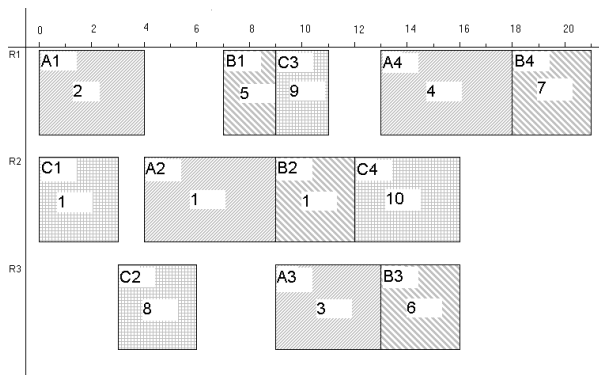


Figura 2.12: 4º passo, realocação das operações em sentido bidirecional.

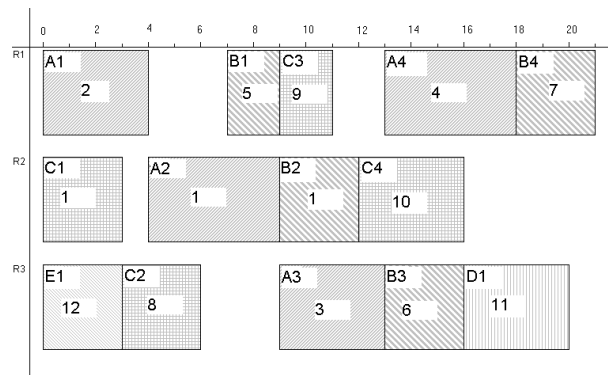


Figura 2.13: Último passo, realocação das operações restantes.

Percebe-se pela figura 2.10 que a ordem de produção *C* seria entregue em $T = 27$ (operação *C4*, a décima segunda operação alocada). Mas como a ordem de produção *C* pertence ao recurso gargalo, a sua alocação é priorizada. Pela figura 2.13 percebe-se que: a heurística aloca a operação *C4* mais cedo (nova data de conclusão prevista para a operação *C4* é $T = 16$).

2.2.4 Minimização do tempo de espera (Min Work in Process - Min WIP)

As estratégias de alocação por ordens de produção discutidas nas seções anteriores podem dar lugar a soluções (cartas de Gantt) que apresentam intervalos de tempo não utilizados entre operações sucessivas da mesma ordem de produção. Isto implica em um tempo de espera entre operações da mesma ordem de produção desnecessário, e também na existência de estoques intermediários desnecessários.

Na literatura de sistemas APS, as estratégias adicionais que visam eliminar estes tempos de espera desnecessários são conhecidas como estratégias de minimização da produção em processo (*Min Work in Process - Min WIP*) para frente (*forward*) ou para trás (*backward*). As estratégias de *Min WIP* para frente e *Min WIP* para trás são similares, e elas têm em comum o fato de construir a carta de Gantt em dois passos.

2.2.4.1 Minimização do tempo de espera para frente

Esta estratégia é dividida em dois passos, o primeiro é idêntico à alocação por ordens de produção no instante mais cedo. O segundo consiste em fixar a última operação da ordem de produção (de acordo com a declaração ou com a prioridade da ordem de produção) e deslocar as operações anteriores visando minimizar o tempo de espera entre as operações. A figura 2.14 representa a carta de Gantt resultante das alocações das ordens de produção da tabela 2.3.

Ordens de Produção	Seqüência de operações na Ordem de Produção	Recursos habilitados	Tempo de Processamento
A	A1	R1	5
	A2	R3	6
	A3	R4	1
	A4	R2	2
B	B1	R2	4
	B2	R1	4
	B3	R3	3
C	C1	R4	4
	C2	R2	3
	C3	R1	1
	C4	R3	2

Tabela 2.3: Ordens de produção, operações constituindo as ordens, recursos habilitados para as operações, tempos de processamento.

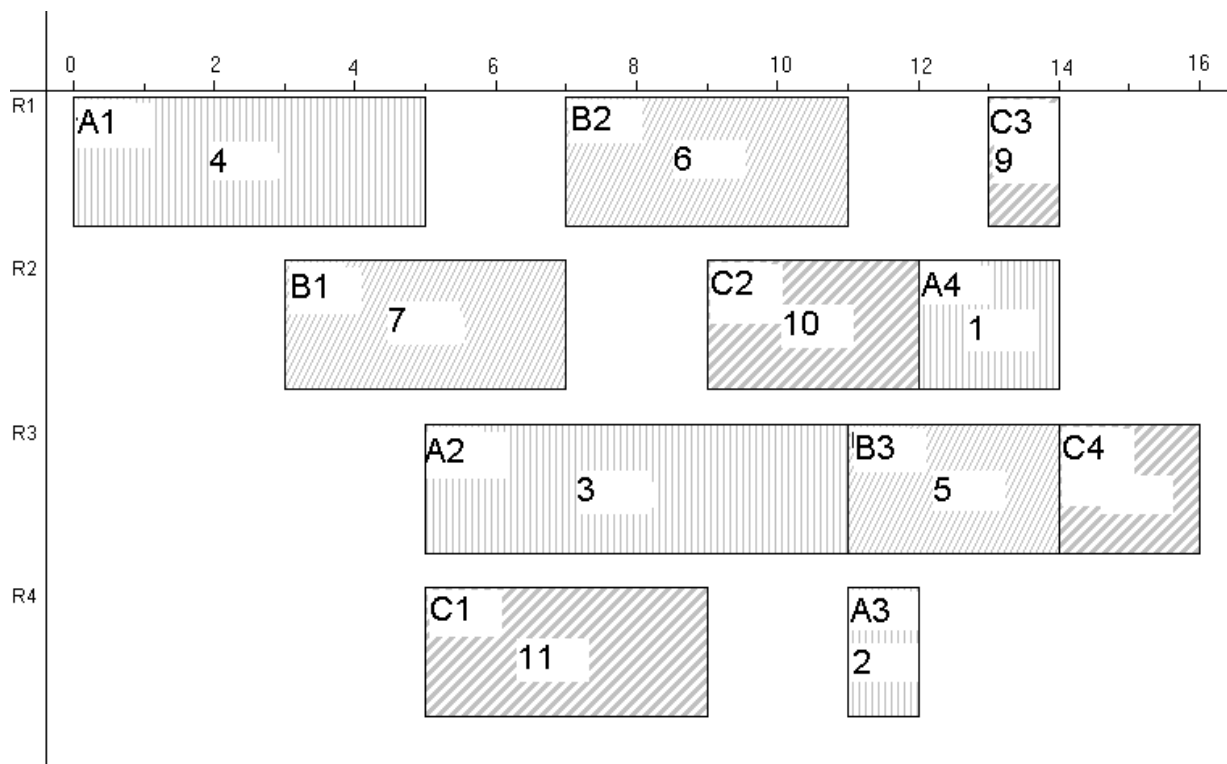


Figura 2.14: Minimização do tempo de espera para frente - Gráfico de Gantt enumerando as operações de acordo com a ordem de alocação conforme declarado na tabela 2.3.

A alocação inicial (ordem A) não tem tempo de espera, pois como ela é a primeira a ser alocada a carta de Gantt estava vazia, conforme mostrado na figura 2.15. Desta forma, não é necessário o segundo passo. Na alocação da ordem B pode-se perceber a utilização dos dois passos (figuras 2.16 a 2.18). A figura 2.16 representa: i) a alocação para frente; ii) a operação B3 fixada e iii) a diferença de tempo que tem que ser minimizado entre B2 e B3. As etapas do segundo passo estão mostradas nas figuras 2.17 e 2.18.

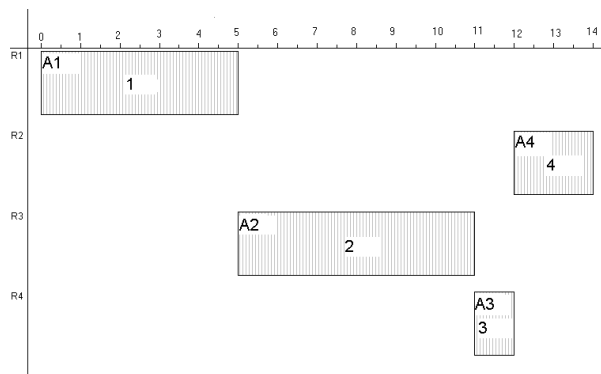


Figura 2.15: Alocação da ordem A.

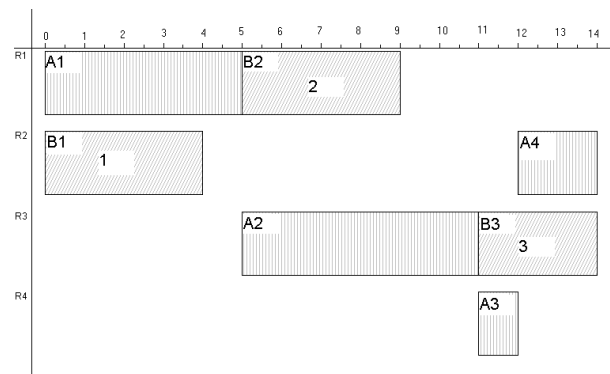


Figura 2.16: 1º passo, operação B3 fixada e distância entre B2 e B3 (alocação da ordem B).

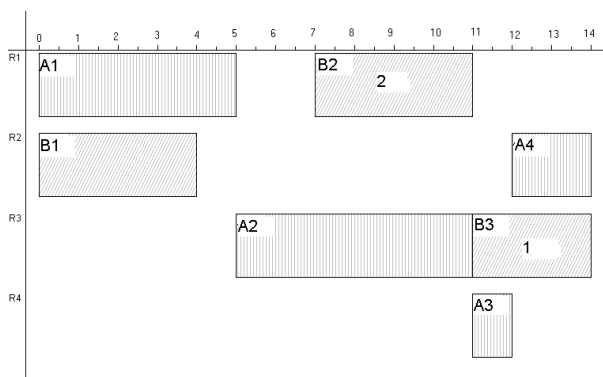


Figura 2.17: Deslocamento de $B2$ (etapa do 2º passo da alocação da ordem B).

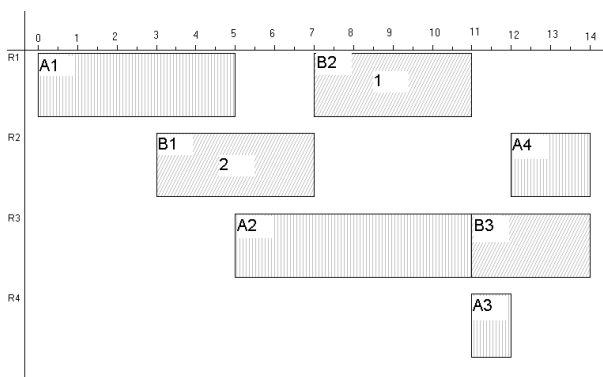


Figura 2.18: Deslocamento de $B1$ (última etapa do 2º passo da alocação da ordem B).

A alocação da ordem C é semelhante a alocação da ordem B . A figura 2.19 mostra o primeiro passo da alocação, e a figura 2.20 mostra a última etapa do segundo passo, e o tempo de espera que ainda ficou entre $C2$ e $C3$.

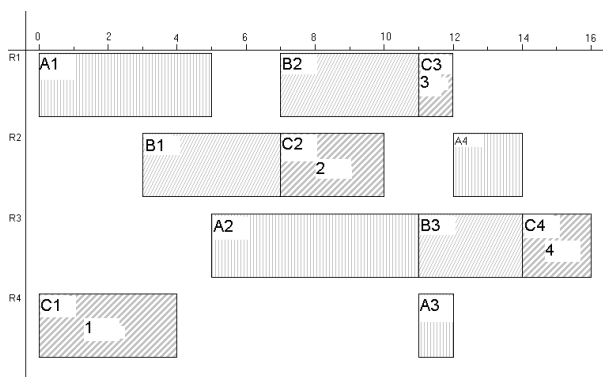


Figura 2.19: 1º passo da alocação da ordem C .

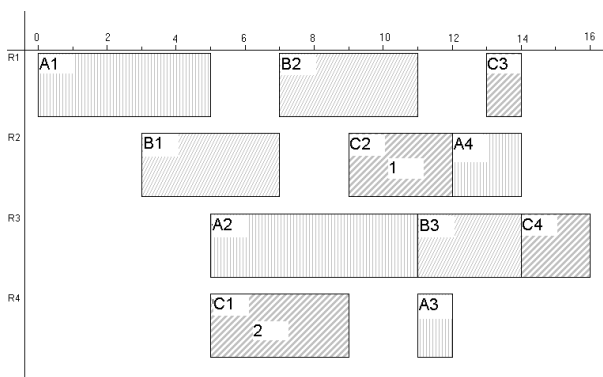


Figura 2.20: Última etapa do 2º passo da alocação da ordem C .

2.2.4.2 Minimização do tempo de espera para trás

Esta estratégia possui dois passos como no caso anterior, o primeiro é semelhante à alocação por ordem de produção no instante mais tarde. Neste inicia-se a alocação pela última operação da ordem de maior prioridade ou da primeira declarada. No segundo passo fixa-se a primeira operação da ordem e desloca as operações seguintes, tentando assim, minimizar o tempo de espera entre operações. A figura 2.21 representa a carta de Gantt resultante das alocações das ordens de produção da tabela 2.4.

Ordens de produção	Seqüência de operações	Recursos habilitados	Tempo de processamento	Prazo de entrega
A	A1	R1	5	14:00
	A2	R3	6	
	A3	R4	1	
	A4	R2	2	
B	B1	R3	3	16:00
	B2	R2	3	
	B3	R1	2	
C	C1	R2	2	16:00
	C2	R1	2	
	C3	R4	3	
	C4	R3	3	

Tabela 2.4: Ordens de produção, operações constituindo as ordens, recursos habilitados para as operações, tempos de processamento, e prazos de entrega.

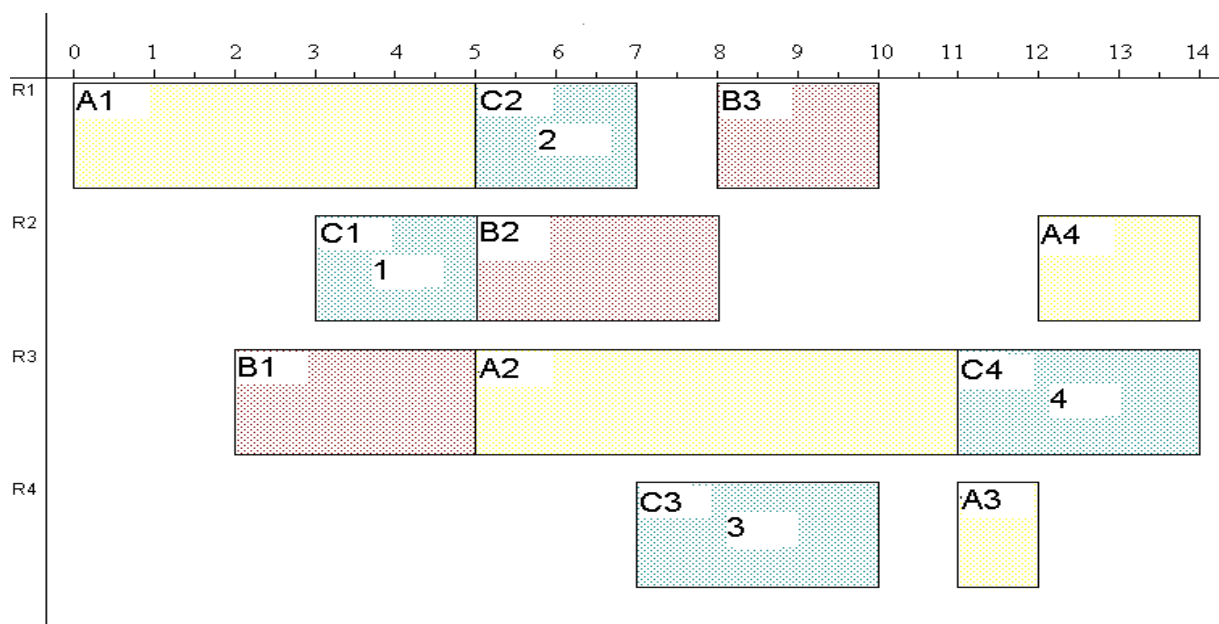


Figura 2.21: Minimização do tempo de espera para trás - Gráfico de Gantt enumerando as operações de acordo com a ordem de alocação conforme declarado na tabela 2.4.

A alocação da ordem *A* não tem tempo de espera, pois é a primeira a ser alocada, conforme mostrado na figura 2.22. As figuras de 2.23 a 2.27 exemplificam as alocações das ordens *B* e *C*.

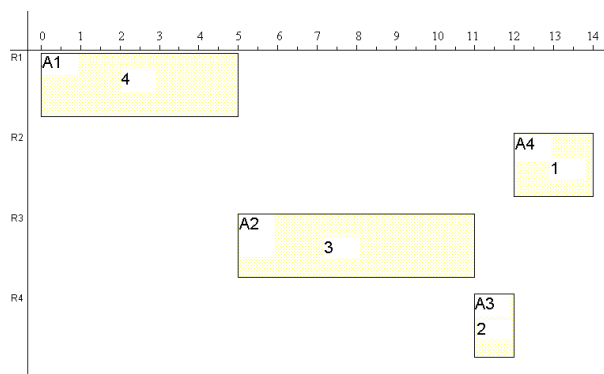


Figura 2.22: Alocação da ordem *A*.

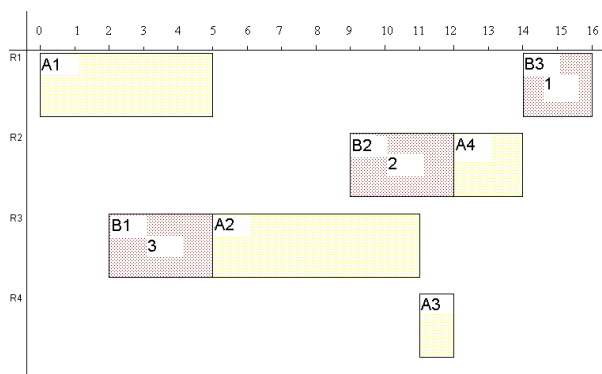


Figura 2.23: 1º passo, operação *B1* fixada e distância entre *B1* e *B2* (alocação da ordem *B*).

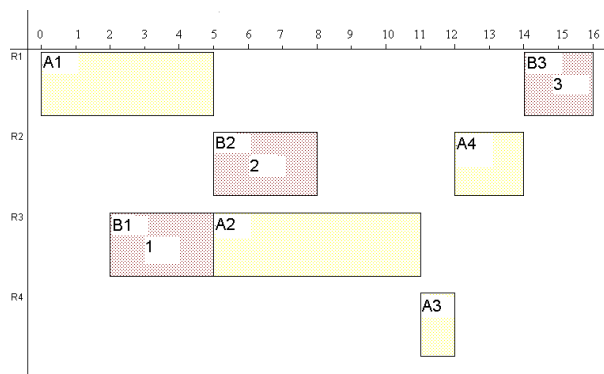


Figura 2.24: Deslocamento de *B2* (etapa do 2º passo da alocação da ordem *B*).

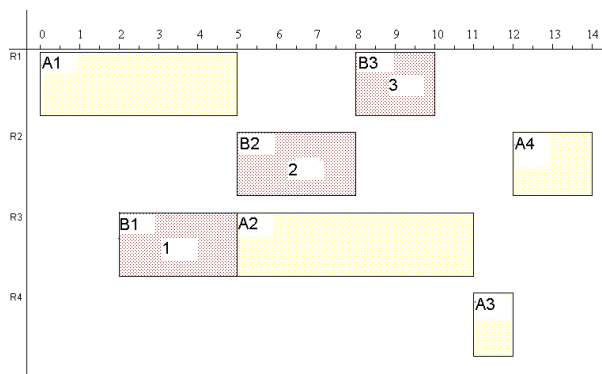


Figura 2.25: Deslocamento de *B3* (última etapa do 2º passo da alocação da ordem *B*).

Observa-se pela figura 2.27, que não foi possível reduzir o tempo de processamento para $T = 0$ entre todas as operações da ordem *C* (*C3-C4*).

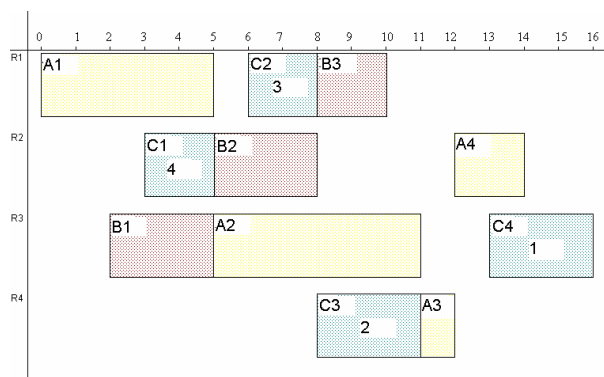


Figura 2.26: 1º passo da alocação da ordem *C*.

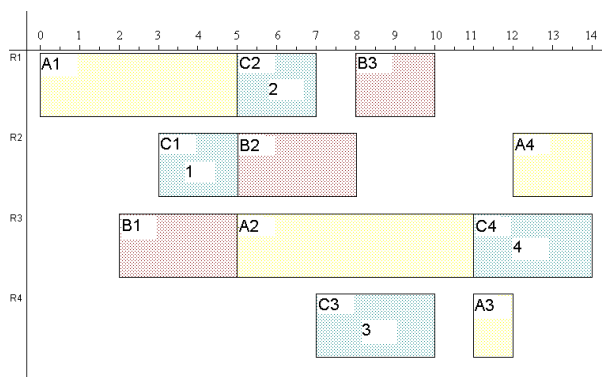


Figura 2.27: Última etapa do 2º passo da alocação da ordem *C*.

2.3 Alocação por operações, heurística APS Relógio

Nesta estratégia a carta de Gantt é construída utilizando um mecanismo de relógio, que avança alocando as operações em sentido crescente do tempo, na construção da carta de Gantt. Esta estratégia aloca as operações em função da lista de liberação de recursos (sempre alocando a partir do primeiro recurso da lista), independentemente da ordem de produção à qual pertençam, diferente da alocação por ordens de produção que aloca primeiro todas as operações de cada ordem de produção em um mesmo passo.

A lista de liberação de recursos é associada a lista de recursos ordenados pelo tempo de disponibilidade em ordem crescente. Na alocação inicial como todos os recursos estão disponíveis no mesmo instante de liberação ($T = 0$), inicia-se a lista de tempos de liberação dos recursos de acordo com a ordem em que os recursos foram declarados, conforme exemplificado no passo 1 da tabela 2.5.

O mecanismo de relógio é responsável pela atualização da lista de liberação de recursos, e pela escolha do recurso a alocar. Após cada operação alocada atualiza-se o tempo de liberação (TL) do recurso em que foi alocada, e reordena a lista de liberação de recursos com o novo valor. Caso o novo tempo de liberação seja igual a algum outro da lista, o recurso será reordenado para a posição posterior ao que estava já ordenado.

Uma vez escolhido um recurso é necessário definir a operação que será alocada, dado que podem existir várias operações habilitadas no recurso e que já podem ser alocadas. Os sistemas *APS* utilizam regras heurísticas específicas ou simplesmente a lista ordenada das ordens de produção para escolher qual é a próxima operação candidata a alocação.

2.3.1 Alocação de acordo com a lista de ordens de produção

Nesta estratégia aloca-se de acordo com a prioridade das ordens de produção fornecido pelo usuário. Para cada ordem de produção é utilizada a seqüência ordenada de operações que compõem a ordem (precedência entre operações). Aloca-se a primeira operação possível de ser alocada (no recurso escolhido pelo tempo de liberação). A tabela 2.5, mostra a seqüência de alocação utilizando os dados da tabela 2.1 repetida abaixo.

Ordens de produção	Seqüência de operações na ordem de produção	Recursos habilitados	Tempo de processamento	Prazo de entrega
A	A1	R1	3	18:00
	A2	R2	4	
B	B1	R2	5	18:00
	B2	R1 ou R2 ou R3	3	
C	C1	R1	3	18:00
	C2	R1 ou R2 ou R3	3	
D	D1	R1 ou R2 ou R3	3	18:00
E	E1	R1 ou R2 ou R3	3	18:00
F	F1	R1 ou R2 ou R3	3	18:00

Tabela 2.1 Ordens de produção, operações constituindo as ordens, recursos habilitados para as operações, tempos de processamento, e prazo de entrega.

Este é um critério estático, uma vez que o ordenamento das ordens de produção é dado pelo usuário no início do procedimento. No início (passo 1 na tabela 2.5) todos os recursos estão com o mesmo tempo de liberação, neste caso, é escolhido o primeiro recurso da lista de liberação. Dentre as operações habilitadas e ainda não alocadas, escolhe-se a primeira operação habilitada no recurso. Se não for possível alocar a primeira operação da lista por alguma restrição de produção, avança-se para a próxima até encontrar uma operação possível. A figura 2.28 apresenta a carta de Gantt resultante, de acordo com os dados da tabela 2.1, mostrando a ordem em que foram alocadas as operações.

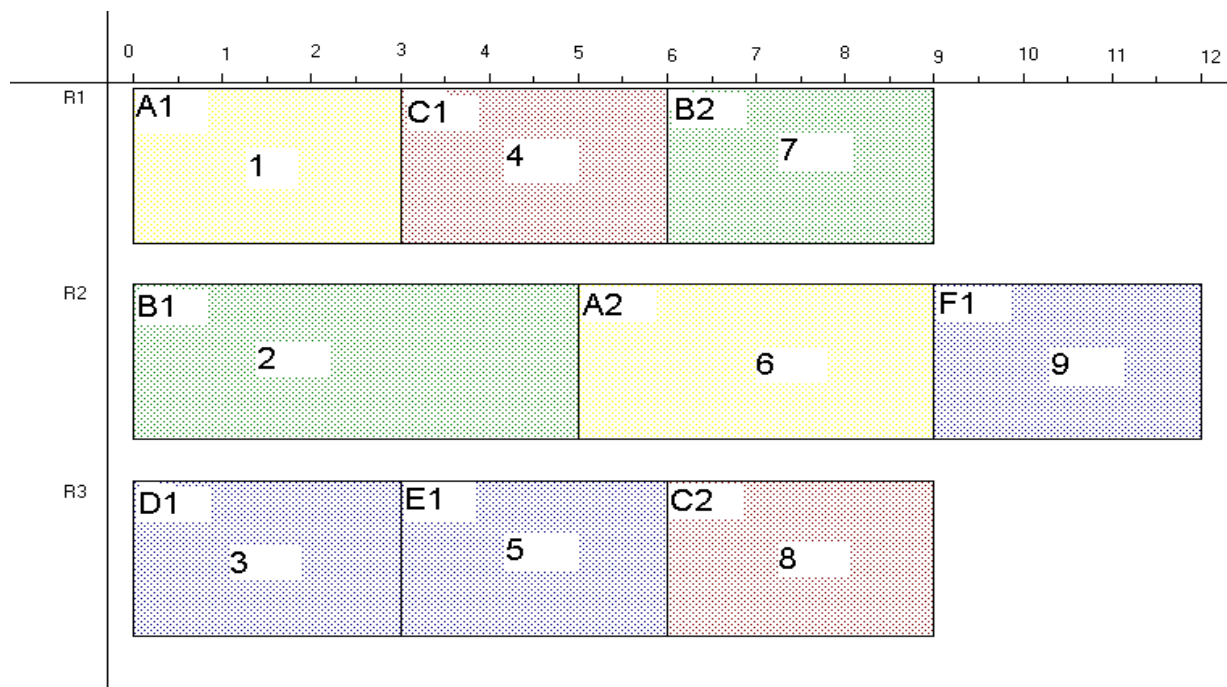


Figura 2.28: Carta de Gantt utilizando o ordenamento das ordens de produção para o exemplo da tabela 2.1.

As figuras de 2.29 a 2.31 mostram sucessivamente a alocação de algumas operações da tabela 2.1 e da figura 2.28. A figura 2.29 apresenta o primeiro passo da tabela 2.5, que é o de alocar a primeira operação (A1) do primeiro recurso listado (R1).

É importante notar a grande diferença no sistema de alocações se comparado com a alocação por ordem de produção. Na alocação por ordem de produção seria necessário alocar a operação A2 também, conforme figura 2.2 já referida na seção 2.2.1. O próximo recurso a ser considerado é o R2, a primeira operação habilitada é a A2, mas como ela é posterior a operação A1, não é possível alocá-la. Desta forma seleciona-se a próxima operação da lista de operações habilitadas, a operação B1. A figura 2.30 mostra a alocação da operação B1 no recurso R2. Percebe-se que pode ocorrer um melhor aproveitamento do recurso, pois a alocação não vai induzindo intervalos de tempo não utilizados (folgas) nos recursos, como ocorre no recurso R2 na alocação por ordem de produção da ordem A conforme figura 2.3 já referida na seção 2.2.1.

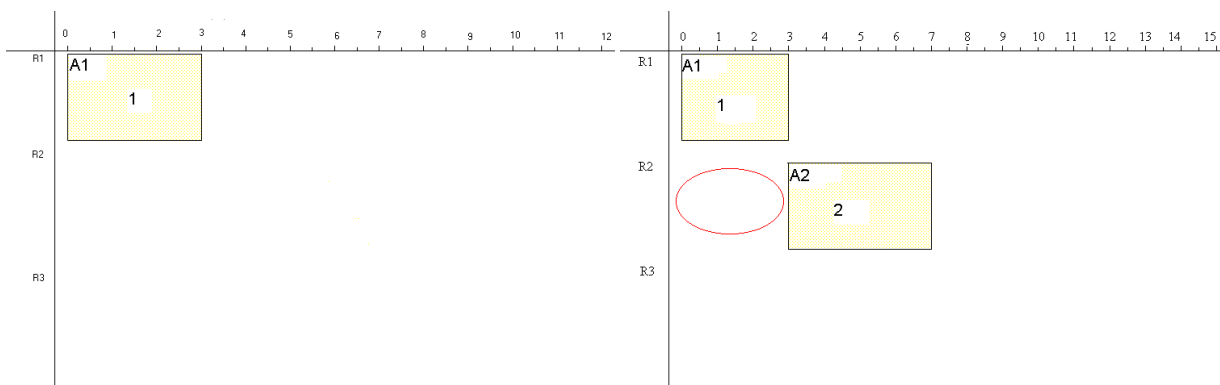


Figura 2.29: 1º passo, alocação da operação A1.

Figura 2.2: 1º passo, alocação das operações da ordem A.

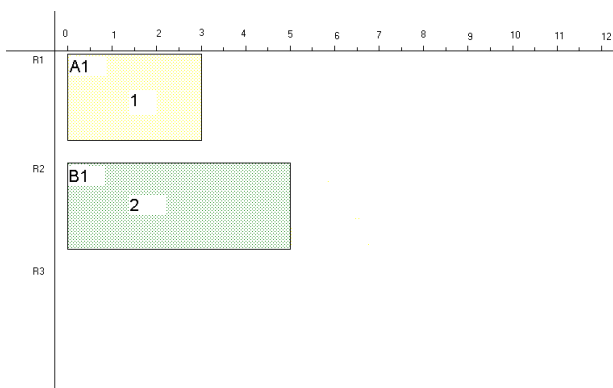


Figura 2.30: 2º passo, alocação da operação B1.

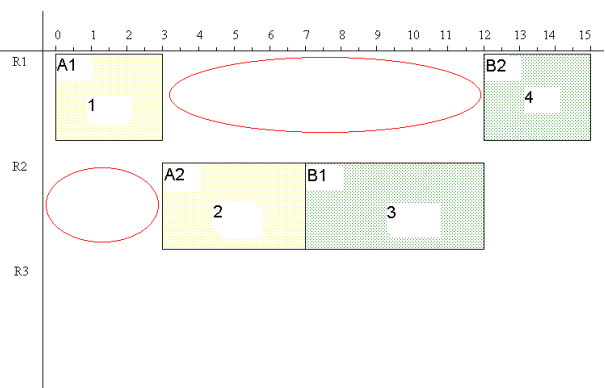


Figura 2.3: 2º passo, alocação das operações da ordem B.

Com um melhor aproveitamento do recurso, pode-se ter um tempo de término total de produção menor que o da alocação por ordem de produção, conforme apresentado nas figuras 2.31 e 2.1 já referida, onde o tempo total foi reduzido de 15 horas para 12 horas.

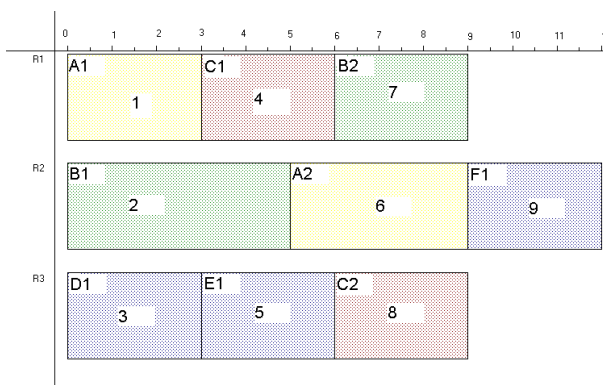


Figura 2.31: 9º passo, alocação da operação F1.

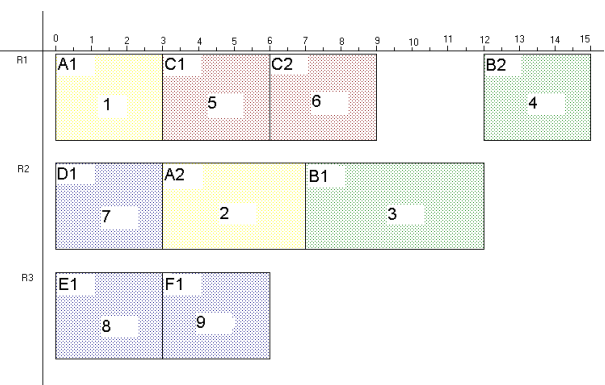


Figura 2.1: Alocação no instante mais cedo - Carta de Gantt enumerando as operações de acordo com a ordem de alocação conforme declarado na tabela 2.1.

Passo	Lista / Tempo de liberação de recursos	Tempo mínimo (relógio)	Recurso escolhido	Lista ordenada de operações remanescentes	Lista ordenada de operações disponíveis para o recurso	Operação escolhida	Temp de lib. / recurso escolhido
1	$R1 = 0$ $R2 = 0$ $R3 = 0$	0	R1	A1, A2, B1, B2, C1, C2, D1, E1, F1	A1 , B2 , C1 , C2 , D1 , E1 , F1	A1	3 / R1
2	$R2 = 0$ $R3 = 0$ $R1 = 3$	0	R2	A2, B1, B2, C1, C2, D1, E1, F1	A2 , B1 , B2 , C2 , D1 , E1 , F1	B1	5 / R2
3	$R3 = 0$ $R1 = 3$ $R2 = 5$	0	R3	A2, B2, C1, C2, D1, E1, F1	B2 , C2 , D1, E1 , F1	D1	3 / R3
4	$R1 = 3$ $R3 = 3$ $R2 = 5$	3	R1	A2, B2, C1, C2, E1, F1	B2 , C1 , C2 ,E1 , F1	C1	6 / R1
5	$R3 = 3$ $R2 = 5$ $R1 = 6$	3	R3	A2, B2, C2, E1, F1	B2 , C2 , E1 , F1	E1	6 / R3
6	$R2 = 5$ $R1 = 6$ $R3 = 6$	5	R2	A2, B2, C2, F1	A2 , B2 , C2 , F1	A2	9 / R2
7	$R1 = 6$ $R3 = 6$ $R2 = 9$	6	R1	B2, C2, F1	B2 , C2 , F1	B2	9 / R1
8	$R3 = 6$ $R2 = 9$ $R1 = 9$	6	R3	C2, F1	C2 , F1	C2	9 / R3
9	$R2 = 9$ $R1 = 9$ $R3 = 9$	9	R2	F1	F1	F1	12 / R2
10	$R1 = 9$ $R3 = 9$ $R2 = 12$		R1	—	—	—	—

Tabela 2.5: Detalhamento do procedimento de alocação das operações conforme os dados da tabela 2.1

2.3.2 Alocação utilizando regras heurísticas de seleção

Neste caso são utilizados diversos critérios para a escolha da operação a ser alocada, dentre eles: i) razão crítica; ii) tempo de processamento; iii) tempo de preparação; iv) data de entrega; e v) atributos. Os valores podem ser utilizados em ordem crescente ou decrescente e ainda pode ser utilizado mais de um critério para a escolha da operação.

Estes critérios são aplicados sobre as operações habilitadas no recurso que tem o menor tempo de liberação. As principais regras heurísticas utilizadas são descritas a seguir:

- A razão crítica (*RC*) é o quociente entre: i) o tempo restante até a data de entrega da ordem de produção (tempo da data de entrega subtraído do tempo de liberação do recurso em que será alocada a operação) e ii) o tempo necessário para completar a ordem de produção (tempo de processamento total da ordem subtraído da soma dos tempos das operações já alocadas). Quanto menor for o valor mais crítica é a ordem de produção. Escolhendo a operação da ordem de produção mais crítica, busca-se diminuir a possibilidade de ordens de produção atrasadas.
- A heurística de tempo de processamento escolhe a operação considerando os tempos de processamento das operações candidatas. Pode, por exemplo, escolher a operação que demora menos, o que pode ser útil para processar o maior numero de operações primeiro e deixar as mais demoradas por último.
- A heurística de tempo de preparação prioriza as operações de acordo com o tempo de preparação do recurso para processar cada operação.
- A heurística de data de entrega escolhe a operação de acordo com a data de entrega da ordem de produção. Pode-se, por exemplo, escolher a operação que pertença à ordem que tenha que ser finalizada mais cedo, buscando-se assim evitar atrasos na entrega dos produtos.
- A heurística de atributos escolhe a operação de acordo com atributos introduzidos pelo usuário, por exemplo, é possível priorizar a produção por um atributo de cor.

Para exemplificar, considere-se o caso descrito na tabela 2.1, supondo que o

tempo de processamento da operação *C2* é de 5 horas. As figuras 2.28 e 2.32 exemplificam a diferença entre uma alocação de acordo com o ordenamento da lista declarada de ordens de produção (figura 2.28) e uma alocação ordenada de acordo com a heurística de razão crítica em ordem crescente (figura 2.32).

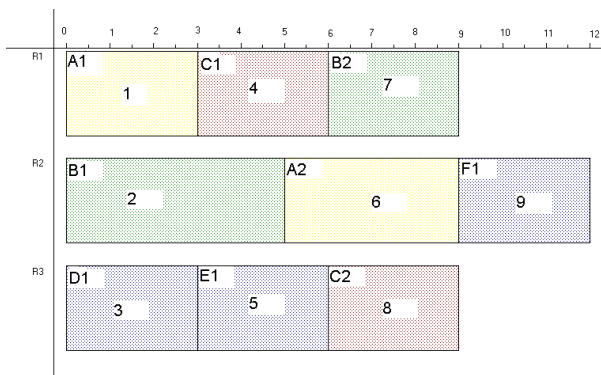


Figura 2.28: Carta de Gantt utilizando o ordenamento das ordens de produção para o exemplo da tabela 2.1.

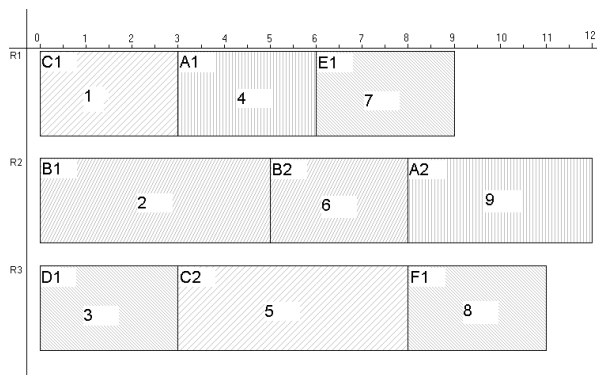


Figura 2.32: Ordenamento pela regra de seqüência de razão crítica.

A tabela 2.6 detalha os passos utilizados nas alocações e o cálculo da razão crítica. A razão crítica da primeira operação do passo 1 da tabela 2.6 (operação *A1*) é definida pelo quociente do tempo restante até a data de entrega da ordem de produção pelo tempo necessário para completar a ordem de produção ou seja $RC_{A1} = \frac{18-0}{7-0} = 2,57$. Como no primeiro passo as operações com a menor razão crítica ($RC = 2,25$) foram *B2*, *C1*, e *C2*, neste caso a operação a alocar seria a operação *B2*, pois ela é a primeira operação da lista declarada de ordens de produção com o valor de $RC = 2,25$. Por restrição de produção não é possível alocá-la pois a operação *B2* é posterior a operação *B1* (ainda não alocada), neste caso aloca-se a próxima operação possível que contenha o menor RC (operação *C1* com $RC = 2,25$). No passo 3 da tabela 2.6, também não é possível alocar a operação com o menor RC ($RC_{C2} = 3,6$), e a operação possível de alocar é a *D1*.

Passo	Lista de liberação de recursos	Tempo mínimo (relógio)	Recurso escolhido	Lista ordenada de operações remanescentes	Lista ordenada de operações disponíveis para o recurso	Menor razão Crítica	Operação escolhida menor <i>CR</i> possível	Tempo de liberação/ recurso escolhido
1	$R1 = 0$ $R2 = 0$ $R3 = 0$	0	R1	A1, A2, B1, B2, C1, C2, D1, E1, F1	A1, B2, C1, C2, D1, E1, F1	B2, C1, C2	C1	3 / R1
2	$R2 = 0$ $R3 = 0$ $R1 = 3$	0	R2	A1, A2, B1, B2, C2, D1, E1, F1	A2, B1, B2, C2, D1, E1, F1	B1, B2	B1	5 / R2
3	$R3 = 0$ $R1 = 3$ $R2 = 5$	0	R3	A1, A2, B2, C2, D1, E1, F1	B2, C2, D1, E1, F1	C2, B2, D1, E1, F1	D1	3 / R3
4	$R1 = 3$ $R3 = 3$ $R2 = 5$	3	R1	A1, A2, B2, C2, E1, F1	A1, B2, C2, E1, F1	A1	A1	6 / R1
5	$R3 = 3$ $R2 = 5$ $R1 = 6$	3	R3	A2, B2, C2, E1, F1	B2, C2, E1, F1	C2	C2	8 / R3
6	$R2 = 5$ $R1 = 6$ $R3 = 8$	5	R2	A2, B2, E1, F1	A2, B2, E1, F1	A2, B2, E1, F1	B2	8 / R2
7	$R1 = 6$ $R3 = 8$ $R2 = 8$	6	R1	A2, E1, F1	E1, F1	E1, F1	E1	9 / R1
8	$R3 = 8$ $R2 = 8$ $R1 = 9$	8	R3	A2, F1	F1	F1	F1	11 / R3
9	$R2 = 8$ $R1 = 9$ $R3 = 11$	8	R2	A2	A2	A2	A2	12 / R2
10	$R1 = 9$ $R3 = 11$ $R2 = 12$		R1	—	—	—	—	—

Tabela 2.6: Detalhamento do procedimento de alocação das operações relacionadas na figura 2.32.

2.4 Representação rede estado-tarefa

A representação rede estado tarefa (“*state-task network*” - *STN*) foi apresentada em (Kondili et al., 1993) e tem sido utilizada freqüentemente em problemas de planejamento e programação da produção. Uma descrição detalhada encontra-se em (Rodrigues, 2000). A *STN* será descrita aqui utilizando os dois exemplos de aplicação que serão utilizados neste trabalho. Estes exemplos foram apresentados por Papageorgiou e Pantelides (Papageorgiou e Pantelides, 1993), e (Papageorgiou e Pantelides, 1996).

As matérias primas, produtos intermediários e produtos finais são denominados estados e representados por círculos. As operações produtoras e consumidoras são representadas por retângulos. Os estados e operações são caracterizados por:

- Condições de armazenagens dos estados: UIS, FIS e NIS (“*unlimited intermediate storage*”, “*finite intermediate storage*”, e “*no intermediate storage*”);
- Política de transferência: sem limitações ou ZW (“*Zero Wait*”);
- Tamanho de lote para cada operação;
- Tempo de processamento da cada operação;
- Fração massica dos estados produzidos e consumidos por cada operação como fração do tamanho de lote da operação.

As regras para construção da representação *STN* de um processo são:

- Uma operação tem a quantidade de estados de entrada (saída) igual ao número de materiais diferentes consumidos (gerados) pela operação.
- Duas ou mais entradas (saídas) em um estado são necessariamente de materiais iguais.

Exemplo 1.

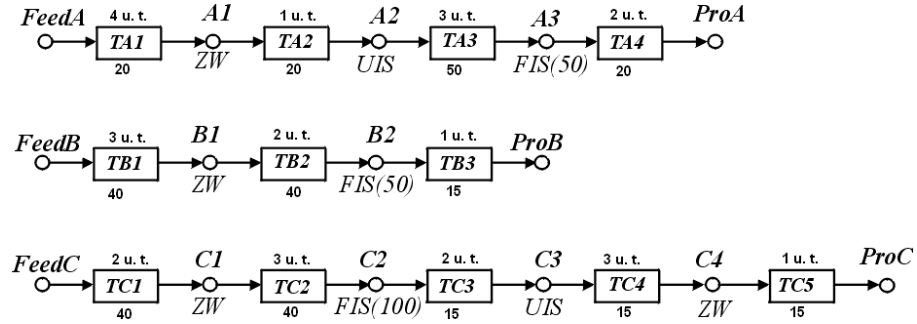


Figura 2.33: Representação STN do exemplo apresentado por Papageorgiu (Papageorgiu & Pantelides 1993).

Exemplo 2.

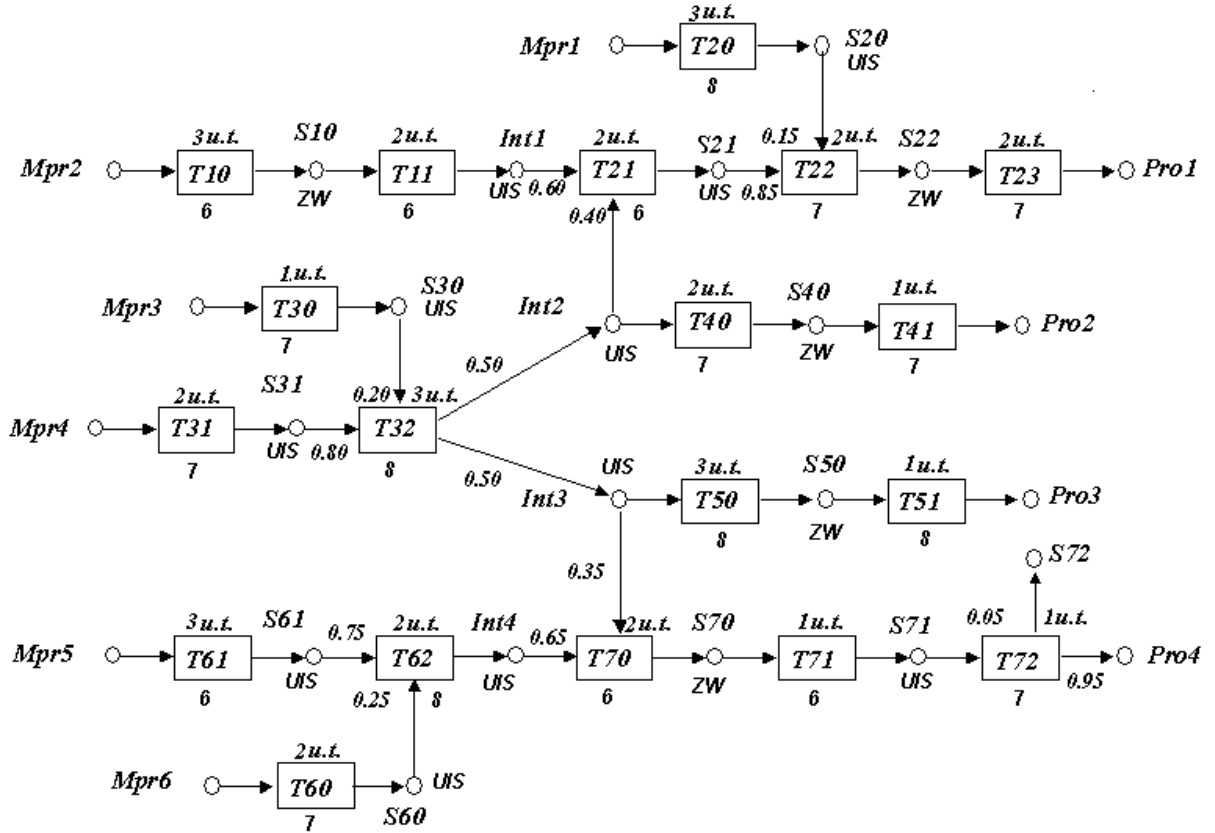


Figura 2.34: Representação STN do exemplo apresentado por Papageorgiu (Papageorgiu & Pantelides 1996).

2.5 Emulador

O emulador desenvolvido utiliza como dados de entrada um conjunto de dados fornecido pelo sistema de planejamento PCPIP (Rodrigues, Latre e Rodrigues, 2000a) e procede

a alocação seguindo as heurísticas *APS* implementadas. O sistema de planejamento fornece ao emulador os seguintes dados:

- Número de lotes de cada operação para satisfazer uma demanda específica de produtos finais;
- Precedências entre lotes originadas pelo balanço de massa;
- Processadores habilitados para cada lote. Quando mais de um processador está habilitado para um lote os tamanhos dos lotes devem ser iguais (caso contrario não poderia ser definido o número de lotes antes da alocação nos processadores).
- Prioridades dos lotes para a alocação.

No emulador foram implementadas as heurísticas de alocação no instante mais cedo (seção 2.2.1), a de alocação no instante mais tarde (seção 2.2.2), e a heurística *APS* Relógio (seção 2.3).

As heurísticas *APS* que utilizam prioridades normalmente utilizam prioridades das ordens de produção, onde uma ordem de produção pode estar constituída por vários lotes de cada uma das operações presentes na receita. As heurísticas implementadas no emulador que utilizam prioridades utilizam prioridades de lotes de operações, o que não exclui a possibilidade de implementar prioridades por ordens de produção (neste caso todos os lotes pertencendo à ordem de produção teriam a mesma prioridade).

2.5.1 Heurísticas de alocação no instante mais cedo e no instante mais tarde

Nestas heurísticas primeiro escolhe-se o lote a alocar e o processador, a seguir determina-se o instante de início da alocação. Estas heurísticas alocam no instante mais cedo ou mais tarde. No primeiro caso este instante depende do instante de tempo em que o processador está livre e do instante de tempo em que os materiais intermediários de entrada gerados pelos lotes produtores (já alocados) estão disponibilizados. No segundo caso dependem do processador e do instante de tempo em que os materiais intermediários de saída devem estar

disponíveis para os lotes consumidores (já alocados). Por outro lado obviamente não pode existir sobreposição no tempo de lotes no mesmo processador. Estas condições devem ser verificadas pelo emulador, e quando não satisfeitas o emulador deve determinar o instante de início de alocação mais cedo ou mais tarde satisfazendo as condições anteriores.

A heurística está descrita no fluxograma abaixo, e logo a seguir serão detalhados os principais cuidados que devem-se tomar para seleccionar o tempo de início das alocações.

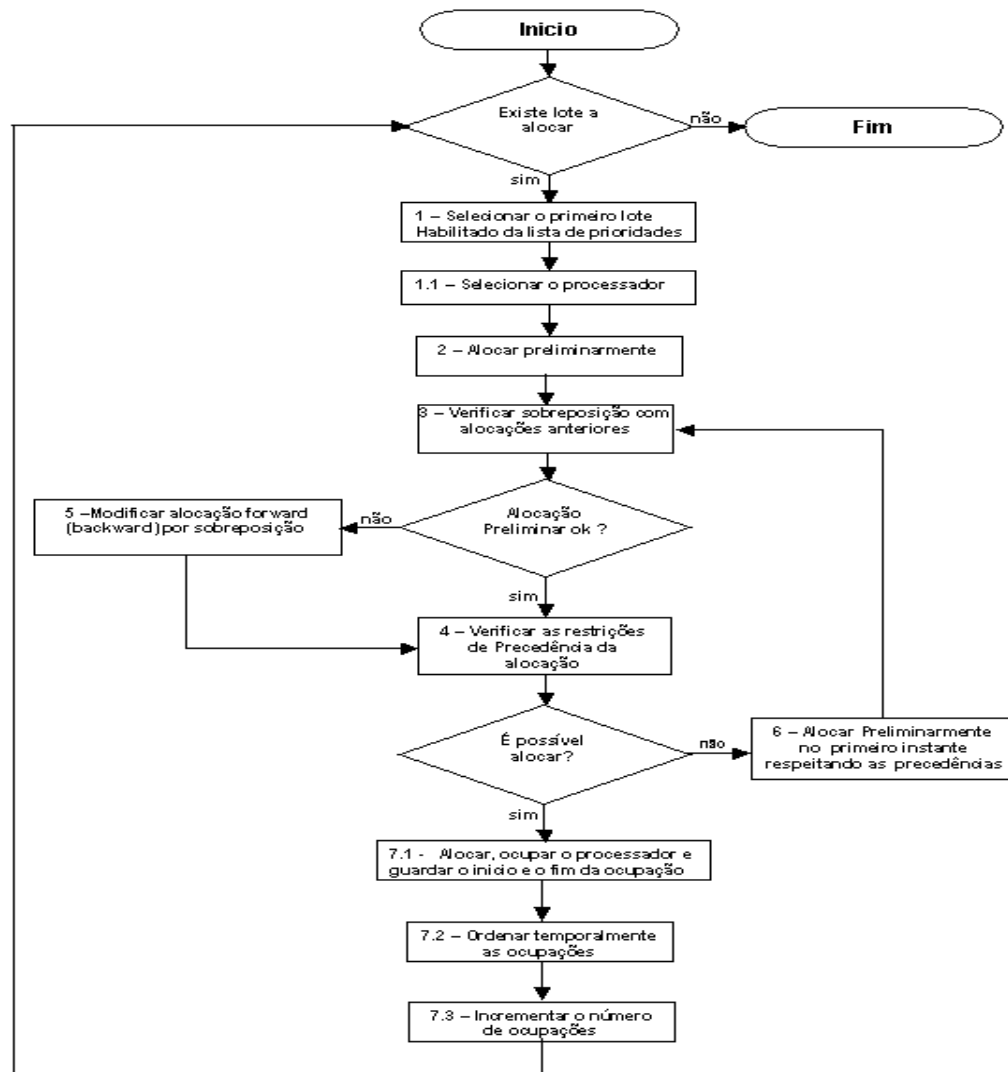


Figura 2.35: Fluxograma da heurística de alocação no instante mais cedo ou mais tarde

Os lotes são tratados seguindo a lista de prioridades dada pelo usuário. A cada passo, uma vez escolhido o lote, determina-se o processador onde será alocado e o instante de alocação. Se o lote estiver habilitado em mais de um processador escolhe-se o primeiro na

lista de prioridades de processadores (figura 2.35 passo: 1.1). O instante de início da alocação preliminar deve ser o instante de liberação inicial do processador, ou seja o primeiro instante de liberação do processador antes das ocupações, mas o processador pode ter diversos intervalos de tempo já ocupados por alocações de lotes anteriores. O emulador determina o primeiro (último) intervalo de tempo livre (e de tamanho suficiente) em que o lote pode ser alocado e o correspondente instante de tempo de início de alocação (figura 2.35 passos: 2; 3; 5). Por outro lado a alocação do lote deve satisfazer as relações de precedência entre lotes produtores e consumidores. Caso não sejam satisfeitas o instante de início é aumentado (diminuído) até que estas restrições sejam satisfeitas. Esta modificação no instante de início implica uma volta à verificação de sobreposição do lote com lotes anteriormente alocados (figura 2.35 passos: 4; 6; volta 3).

Considerando o caso de alocação mais cedo o módulo responsável pela verificação de sobreposição ativa duas rotinas. A primeira rotina é responsável por verificar se a alocação no valor preliminar do tempo de início sobrepõe-se com o de algum lote já alocado. Se não existir sobreposição o valor preliminar não é alterado. Entretanto se existir sobreposição é ativada a segunda rotina que é responsável por encontrar um novo intervalo de tempo onde pode-se preliminarmente alocar o lote. Esta segunda rotina tenta encontrar primeiro um intervalo de tempo o mais cedo possível entre as ocupações onde o intervalo seja maior que o tempo de processamento do lote, se não existir nenhum intervalo nesta condição o valor preliminar do tempo de início do lote passa a ser igual ao valor de tempo final da última ocupação do processador, neste caso este é o instante mais cedo em que o lote preliminarmente pode ser alocado.

O módulo responsável pela verificação da restrição de precedência verifica se o lote selecionado será alocado após o seu lote produtor. Se o lote atender a esta restrição ele é alocado, mas se não atender a restrição é encontrado o lote produtor e é associado preliminarmente o tempo final da alocação do lote produtor com o tempo de início do lote selecionado. Após a associação é necessário verificar a sobreposição novamente.

2.5.2 Heurística APS Relógio

Nesta heurística escolhe-se primeiro o instante de início da alocação e o processador, a seguir seleciona-se o lote a alocar. Os principais cuidados que se devem tomar são: a atualização correta da lista de liberação (lista de disponibilidade), e a verificação das precedências entre lotes. A heurística está descrita no fluxograma abaixo, e a seguir serão detalhados os principais pontos dela.

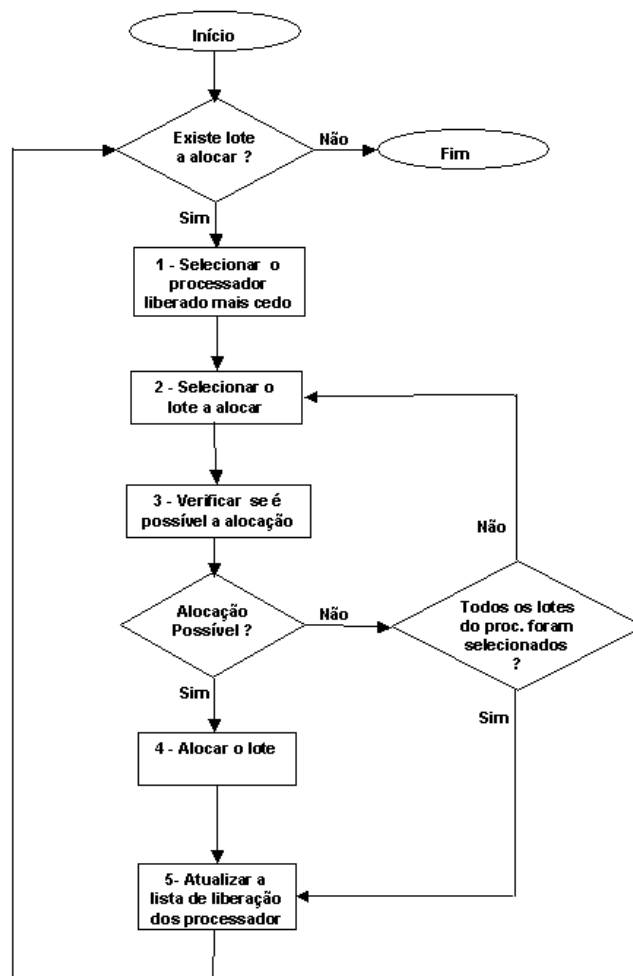


Figura 2.36: Fluxograma da heurística APS relógio

Esta heurística utiliza duas listas importantes, uma para a escolha do processador e outra para a escolha do lote a alocar, são elas: i) a lista de liberação dos processadores, e ii) uma lista de lotes habilitados e ainda não alocados por processadores. A lista de liberação é a responsável por manter o relógio atualizado, enquanto que a lista dos lotes habilitados por

processadores facilita a busca pelo lote a alocar.

A lista dos lotes habilitados por processadores reduz a busca pelo lote a alocar, pois como primeiro escolhe-se o processador onde será feita a alocação, depois é necessário apenas selecionar um lote habilitado nesta lista que atenda as restrições de precedência da alocação em questão. Esta lista é ordenada pelas ordens de prioridades fornecidas pelo usuário, e sempre que possível será selecionado o primeiro lote da lista. Após a alocação do lote selecionado ele muda de estado para alocado e não pertence mais a lista.

O processador selecionado sempre será o primeiro processador da lista de liberação. Para manter a lista de liberação corretamente atualizada é necessário seguir os seguintes passos: i) manter um ordenamento temporal a partir do tempo de liberação dos processadores, e um “*flag*” atualizado dos estados do processador, os estados possíveis são disponível e não disponível, ii) após a seleção e alocação de um lote atualizar o tempo de liberação do processador a partir do instante de tempo final do lote alocado.

O ordenamento temporal do primeiro passo é um ordenamento dos processadores por tempo de liberação em ordem crescente. Na primeira alocação, e somente nela caso exista empate entre os tempos de liberações, o processador escolhido será o primeiro da lista ordenada de processadores. Após a escolha do processador, seleciona-se para alocação o primeiro lote da lista de lotes habilitados. Após a escolha do lote é necessário verificar se o mesmo satisfaz as relações de precedência entre lotes produtores e consumidores (figura 2.36 passos: 1; 2; 3). Se o lote satisfizer as relações de precedência ele é alocado (figura 2.36 passo: 4) e o tempo de liberação do processador é atualizado para o instante final da alocação do lote, e a lista de liberação é reordenada (figura 2.36 passo: 5). Se o lote selecionado não satisfizer as relações de precedência é escolhido o próximo lote da lista de lotes habilitados por processadores, e assim por diante até encontrar um lote que satisfaça (figura 2.36 passo: 2). Caso não exista nenhum lote que satisfaça as relações de precedências é necessário atualizar o estado do processador para não disponível, e reordenar a lista de liberação (figura 2.36 passo: 5).

Quando um processador selecionado for marcado com o estado de “não disponível” é necessário atualizar o seu tempo de liberação. Este será atualizado com o valor do primeiro tempo de liberação da lista de liberação dos processadores de um processador com o estado de “disponível. Após esta atualização o processador selecionado muda de estado para

“disponível” e é necessário reordenar a lista de liberação (figura 2.36 passo: 5). Este passo que é o responsável por manter as alocações avançando no tempo quando não existir lote a alocar no tempo de liberação do processador selecionado. Sem este passo a heurística não iria mais avançar o tempo, isto porque como não existe lote a alocar no tempo de liberação do processador selecionado, este tempo de liberação não tem como ser atualizado, sendo assim este sempre será o menor tempo de liberação da lista de liberação dos processadores.

Sempre que houver alguma atualização do tempo de liberação é necessário um reordenamento na lista de liberação, caso já exista um processador com o mesmo tempo de liberação, o processador recém selecionado é reordenado abaixo do processador que já tinha este tempo de liberação (figura 2.36 passo: 5).

2.6 Exemplos

Considera-se o exemplo 1 descrito na seção 2.4, modificado de forma a não ter limitações de armazenagem (porque as heurísticas *APS* analisadas não garantem que restrições *FIS*, *NIS* ou *ZW* sejam respeitadas na evolução). Neste exemplo são utilizados os dados das tabelas 2.7 e 2.8 que apresentam respectivamente: as habilitações dos lotes, e as demandas dos produtos finais e datas de entrega. A disponibilidade de matéria prima é em $T = 1$. As figuras de 2.37 a 2.41 mostram os resultados obtidos com as heurísticas implementadas. Neste exemplo o nome das ordens de produção é o mesmo que dos produtos.

Processadores	Operações
P1	TA1,TC1
P2	TA2,TA4
P3	TB1
P4	TB1,TC4
P5	TB2,TC2
P6	TB3,TC3
P7	TA3,TC5

Tabela 2.7: habilitação das operações.

Produtos	Demanda	Data de entrega	Demanda	Data de entrega
ProA	100	32	90	56
ProB	270	32		
ProC	75	56		

Tabela 2.8: Demanda de produtos finais, e datas de entrega.

A figura 2.37 apresenta as alocações do exemplo 1 utilizando a heurística de alocação no instante mais cedo para a ordem de prioridades de produção a seguir: *ProdA*, *ProdB* e *ProdC*, e para a seguinte ordem de prioridades dos processadores: *P1*, *P2*, e *P3*.

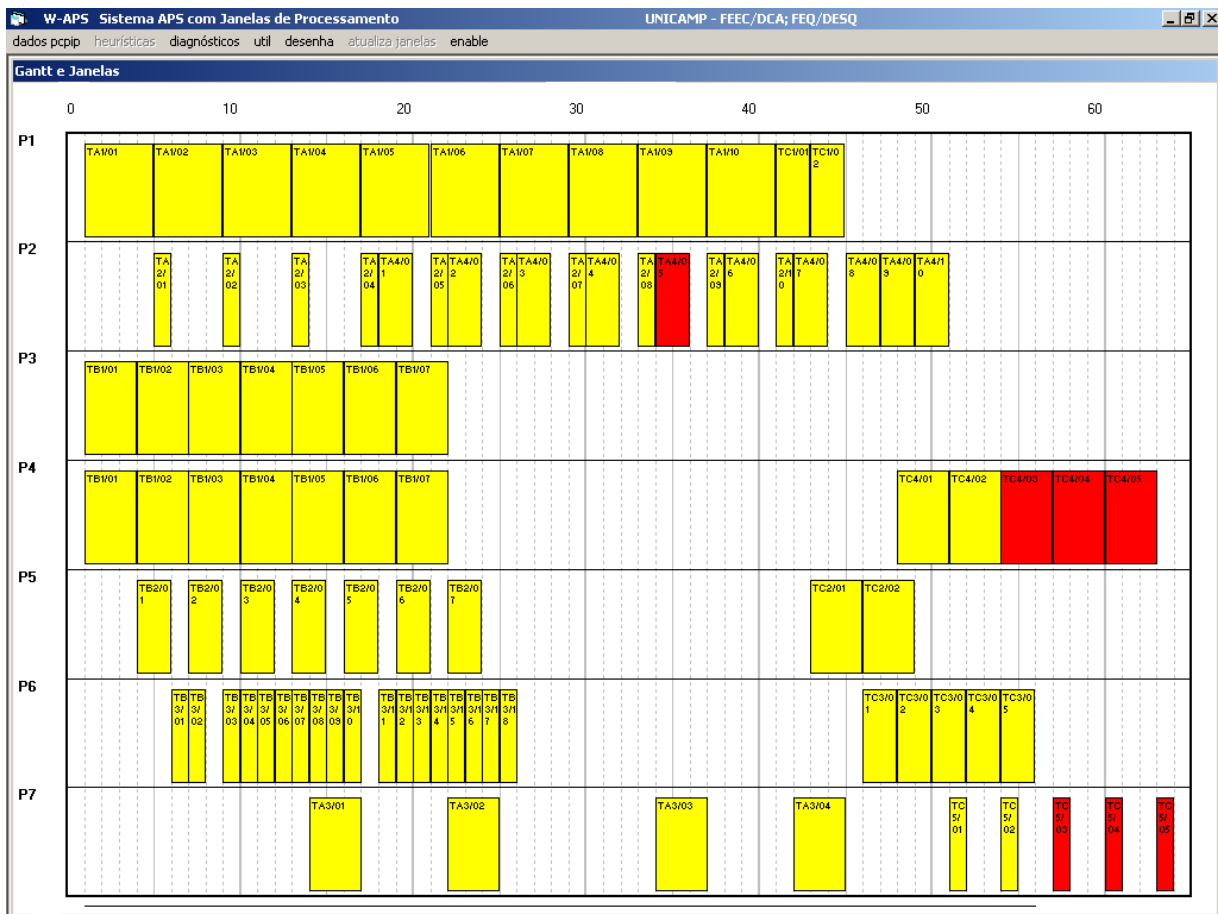


Figura 2.37: Alocações do exemplo 1 utilizando a heurística de alocação no instante mais cedo para a ordem de prioridades a seguir: *ProdA*, *ProdB* e *ProdC*.

Pela figura 2.37 é possível verificar que os lotes *TA4/5*, *TC4/3, 4, 5*, *TC5/3, 4, 5* (achurados) não serão entregues dentro do prazo, o prazo de entrega para a primeira demanda do produto *ProdA* era $T = 32$ e o lote já ultrapassou este prazo, o mesmo aconteceu com os lotes achurados do produto *ProdC*. Nestes casos o sistema deixa identificado os lotes que não cumpriram com as datas de entrega. Fica a critério do usuário aceitar o resultado, ou fazer alguma modificação, por exemplo mudar a ordem de alocação das operações, ou modificar a data de entrega dos produtos *ProdA* e *ProdC* e realocar novamente.

A figura 2.38 apresenta as alocações do exemplo 1 utilizando a heurística de alocação no instante mais tarde para a ordem de prioridades a seguir: *ProdA*, *ProdB* e *ProdC*,

e para a seguinte ordem de prioridades dos processadores: $P1$, $P2$, e $P3$.

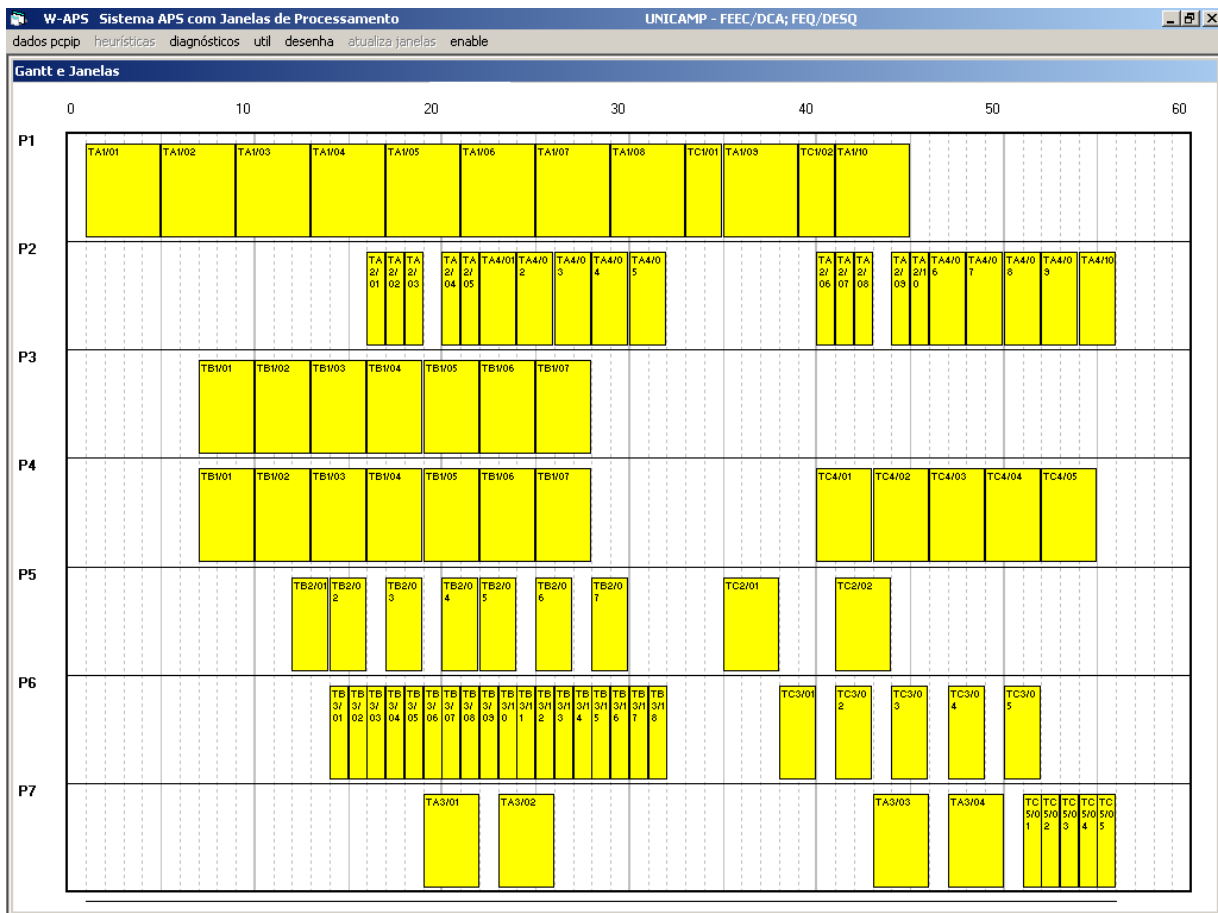


Figura 2.38: Alocações do exemplo 1 utilizando a heurística de alocação no instante mais tarde para a ordem de prioridades a seguir: $ProdA$, $ProdB$ e $ProdC$.

Pela figura 2.38 é possível verificar que todos os produtos serão entregues de acordo com a data de entrega.

A figura 2.39 apresenta as alocações do exemplo 1 utilizando a heurística de alocação APS Relógio para a ordem de prioridades a seguir: $ProdA$, $ProdB$ e $ProdC$. Todos os exemplos desta seção que utilizam esta heurística terão a seguinte ordem de prioridades dos processadores: $P1$, $P2$, e $P3$. A figura 2.40 apresenta as alocações do exemplo 1 utilizando a heurística de alocação APS Relógio para a ordem de prioridades a seguir: $ProdB$, $ProdA$ e $ProdC$.

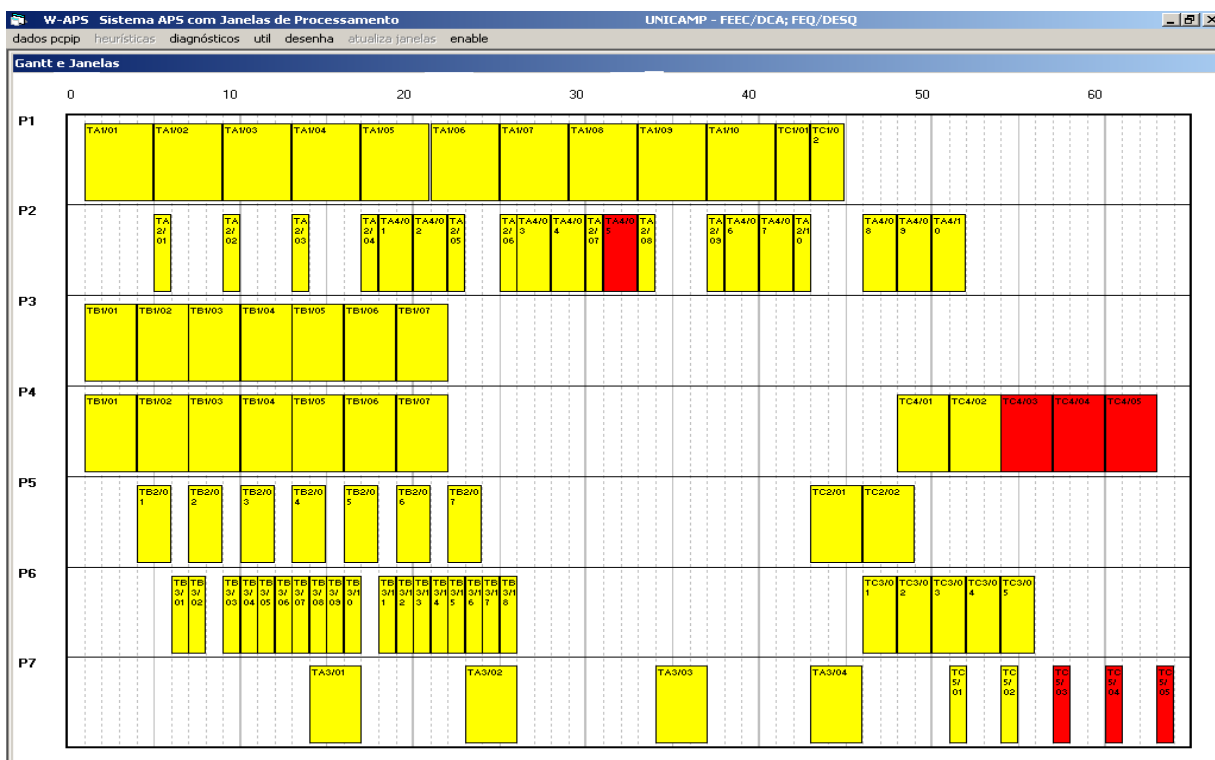


Figura 2.39: Alocações do exemplo 1 utilizando a heurística de alocação APS Relógio para a ordem de prioridades a seguir: *ProdB*, *ProdB* e *ProdB*.

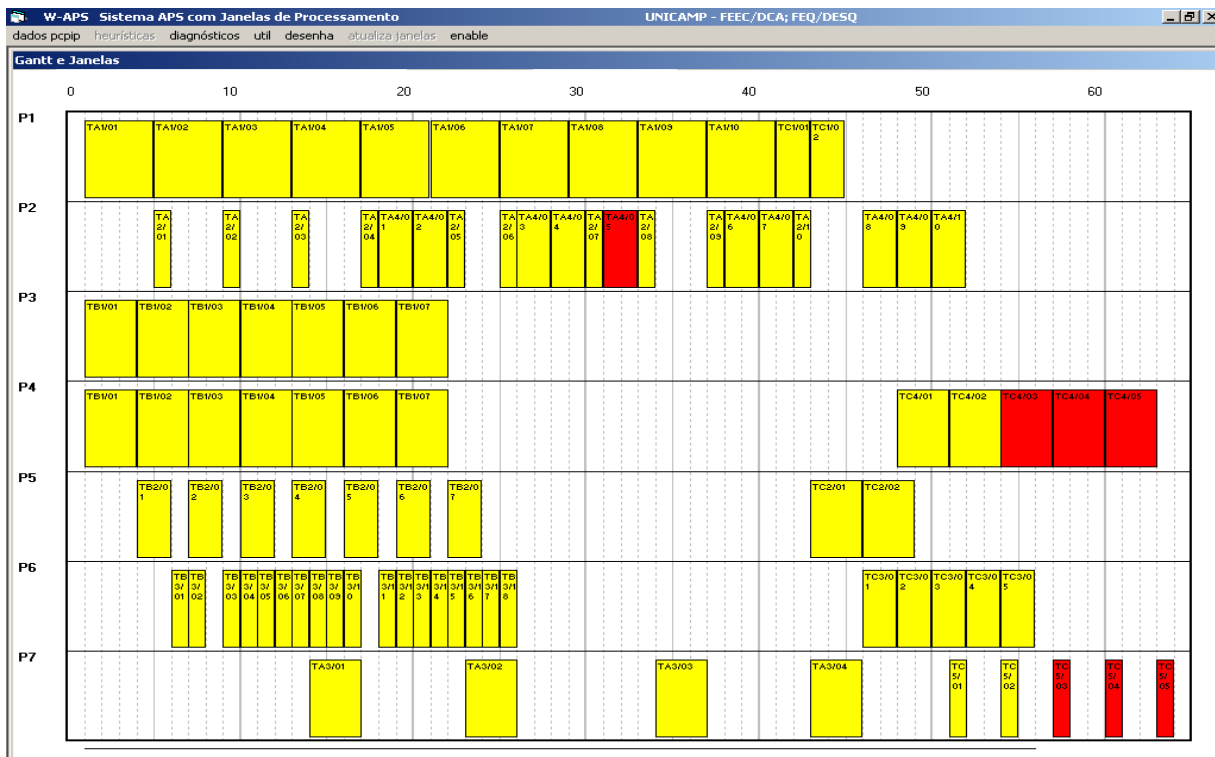


Figura 2.40: Alocações do exemplo 1 utilizando a heurística de alocação APS Relógio para a ordem de prioridades a seguir: *ProdB*, *ProdB* e *ProdB*.

A figura 2.41 apresenta as alocações do exemplo 1 utilizando a heurística de alocação *APS Relógio* para a ordem de prioridades a seguir: *ProdC*, *ProdB* e *ProdB*.

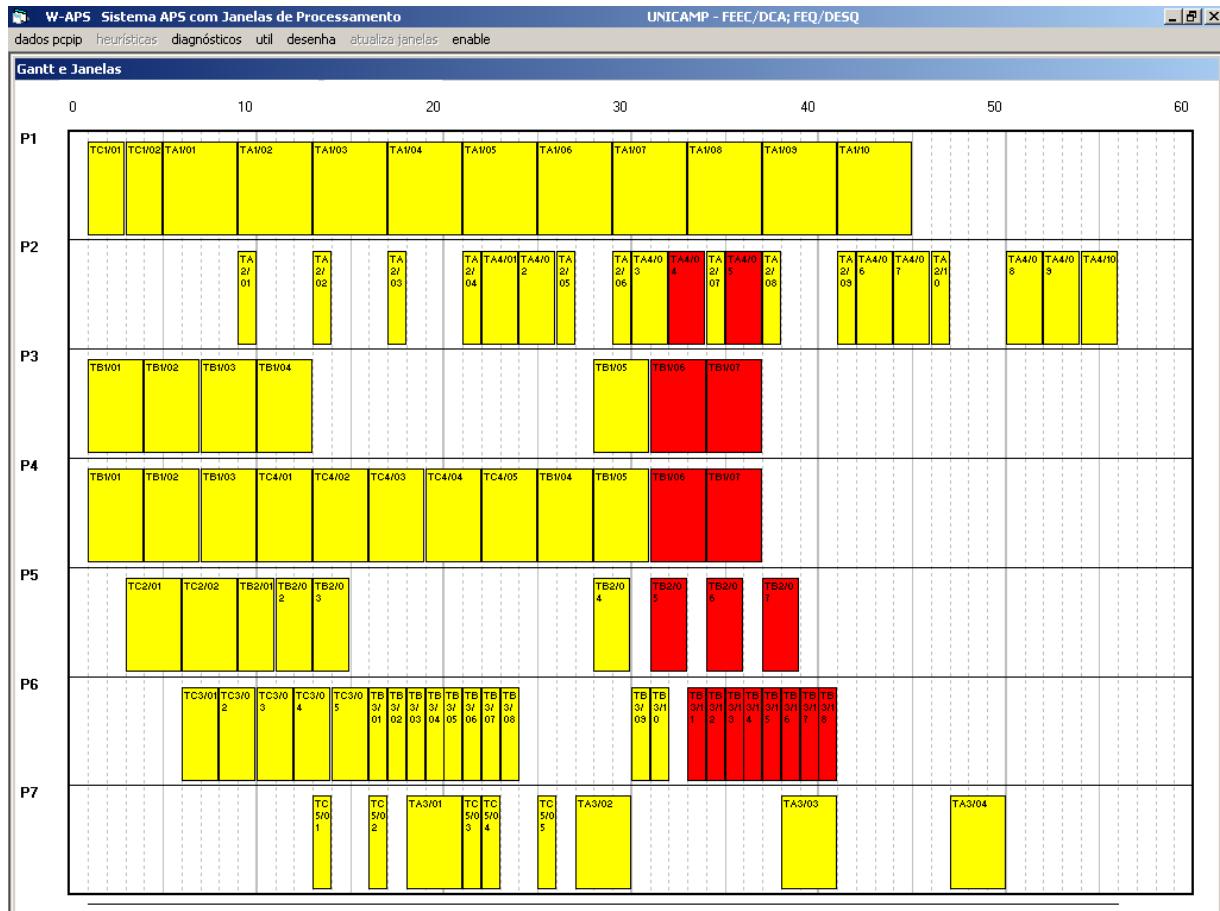


Figura 2.41: Alocações do exemplo 1 utilizando a heurística de alocação *APS Relógio* para a ordem de prioridades a seguir: *ProdC*, *ProdB* e *ProdB*.

Pelas figuras 2.37 e 2.39 percebe-se que não é possível cumprir com a data de entrega dos produtos *ProdB* e *ProdC* em ambas as heurísticas (alocação no instante mais cedo, e *APS relógio*) utilizando as prioridades fornecidas. Entretanto é possível perceber pela carta de Gantt da heurística *APS relógio* um melhor aproveitamento do processador *P2*, pois nas alocações utilizando a heurística de alocação no instante de tempo mais cedo (figura 2.37) houveram 10 intervalos de tempo não utilizados entre os lote, enquanto que na heurística *APS relógio* houveram apenas 6 intervalos. Este fato ocorre pois com a utilização do mecanismo de relógio a alocação é feita de forma gradativa no tempo, sendo que somente existirá intervalos de tempos não utilizados se não existir um lote que possa ocupar este intervalo.

Pelas figuras 2.39 a 2.41 é possível perceber que mesmo mudando as ordens

de prioridades de alocação dos produtos não houve nenhuma solução final em que todos os produtos cumprissem com o prazo de entrega (os lotes que estão hachurados são lotes que não cumpriram o prazo de entrega). É importante frisar que para este problema com apenas três produtos finais existem seis combinações possíveis de ordens de prioridade dos produtos, e entre as seis nenhuma ordem de prioridade faz com que as alocações cumpram as datas de entrega. Obviamente é uma responsabilidade muito grande para o usuário o fato de ser ele quem prioriza as ordens de produção para alocação.

Capítulo 3

Janelas de processamento, propagação das restrições e técnicas CBS

3.1 Introdução

O objetivo deste trabalho é o desenvolvimento de heurísticas tipo *APS* para problemas com datas de entrega importantes. Para estes tipos de problemas o conceito de janelas de processamento dos lotes é muito útil. A sua obtenção na fase de planejamento é descrita resumidamente na seção 3.2. A seção 3.3 contém um resumo das técnicas de propagação de restrições. Estas técnicas tem como finalidade garantir que as janelas de processamento dos lotes satisfaçam as restrições de balanço de massa, armazenagem limitada e capacidade finita dos equipamentos. A seção 3.4 apresenta rapidamente um exemplo de aplicação das técnicas de propagação baseadas em busca orientada pelas restrições. Estas técnicas são orientadas pelos gargalos e as ferramentas utilizadas para detecção de gargalos serão utilizadas nas propostas das heurísticas no capítulo 4. Finalmente a seção 3.5 retorna ao exemplo da seção 2.6 que utilizou diversas heurísticas *APS* para analisar o seu desempenho em um cenário em que as datas de entrega são importantes.

3.2 Janelas de processamento e a sua geração na fase de planejamento

Em problemas de produção com datas de entrega é possível determinar, para cada um dos lotes necessários para atender a demanda, o instante de tempo mais tarde em que este

lote deve estar pronto. Este cálculo é um dos pontos essenciais da técnica *MRP*. Por outro lado os instantes de tempo a partir dos quais as matérias primas ou estados de entrada na receita estão disponíveis permite determinar o instante de início mais cedo em que cada lote pode ser fabricado. Este instante de início mais cedo (“*earliest starting time*” *Est*) e o instante de término mais tarde (“*latest finishing time*” *Lft*) definem a janela de tempo de processamento do lote.

Para calcular as janelas de processamento é necessário que sejam fornecidos os seguintes dados: i) a receita do produto; ii) as demandas de produtos finais e suas datas de entrega; iii) a disponibilidade temporal das matérias primas; iv) os estoques existentes de produtos finais e intermediários.

As janelas de processamento são geradas em duas etapas: a primeira é um procedimento de explosão de requisitos para trás; a segunda é um procedimento de explosão para frente que utiliza o plano de disponibilidade de matérias primas. A primeira etapa utiliza a demanda de produtos finais para calcular o número de lotes de cada operação e seus tempos de término mais tarde (*Lft*). Esta etapa fornece um plano mínimo de necessidades de matérias primas. A segunda etapa utiliza o plano de disponibilidades de matérias primas fornecido pelo usuário para calcular o instante de início mais cedo da cada lote (*Est*). O plano de disponibilidade de matéria prima fornecido pelo usuário deve satisfazer o plano mínimo calculado pela explosão para trás.

A partir do instante de início mais cedo e a data de entrega do lote, calcula-se a sua janela de processamento. O cálculo do número de lotes de cada operação só poderá ser feito após a atribuição das operações aos processadores, pois somente é possível saber o tamanho do lote de cada operação quando for definido o processador que irá produzi-la.

As figuras 3.2 a 3.5 exemplificam a explosão para trás do *STN* representado pela figura 3.1.

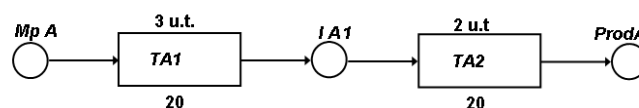


Figura 3.1: STN do exemplo

Utilizando as demandas do produto final (*Proda*) representadas pela figura 3.2,

calcula-se a curva da demanda acumulada total de *ProdA* que é representada pela figura 3.3. Este é um cálculo simples, pois cada degrau da curva representa a somatória da massa que é necessária para suprir a demanda no tempo.

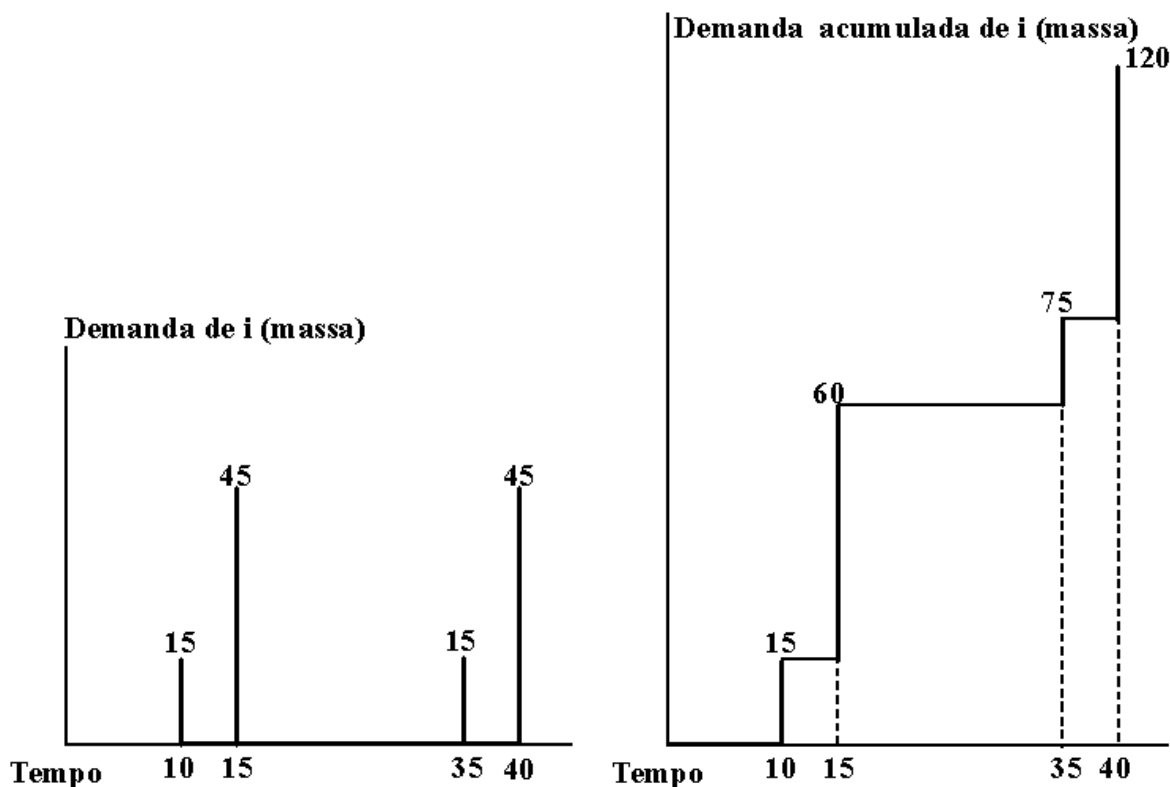


Figura 3.2: Demanda inicial.

Figura 3.3: Demanda acumulada.

É necessário produzir um número de lotes suficientes para suprir a demanda acumulada. O número de lotes a produzir é o quociente entre a demanda acumulada total e o tamanho do lote da operação, neste exemplo o número de lotes a produzir é seis lotes ($120/20 = 6$). O número de lotes tem que ser inteiro, assim, sempre que for necessário, arredonda-se para cima esse valor. A partir do número de lotes a produzir, é necessário calcular o instante de término mais tarde de cada lote (*Lft*).

A figura 3.4 apresenta as datas de entrega dos lotes do *ProdA* necessários para satisfazer a demanda acumulada total. Para construir a curva da figura 3.4 é necessário iniciar o cálculo pela última data de entrega do produto, neste caso o ponto A está representando este valor. Partindo do ponto A, calcula-se para trás (no tempo) e para baixo (demanda) os valores da figura 3.3, criando uma seqüência de degraus com altura igual ao tamanho do lote da operação e largura maior ou igual ao tempo de processamento da operação. A largura do degrau tem que

ser no mínimo igual ao tempo de processamento, porque, os instantes de término de dois lotes sucessivos tem que ter um intervalo de tempo mínimo igual ao valor do tempo de processamento do lote.

Com os valores das datas de entrega dos lotes da figura 3.4, calcula-se a demanda do produto intermediário *IA1* induzida pelo produto *ProA* (explosão para trás). As datas da demanda do produto intermediário *IA1* são as datas de entrega dos lotes subtraídas do tempo de processamento da operação *TA2*. A figura 3.5 representa a demanda do produto intermediário *IA1*.

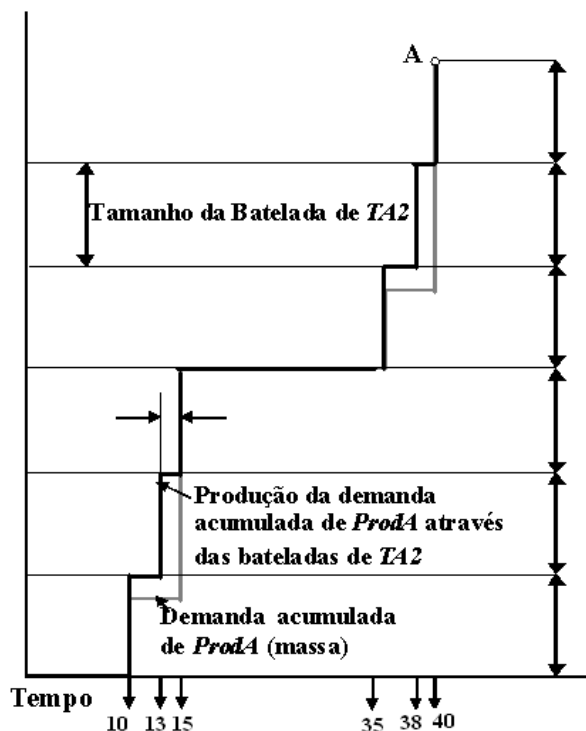


Figura 3.4: Datas de entrega dos lotes do produto *ProA*.

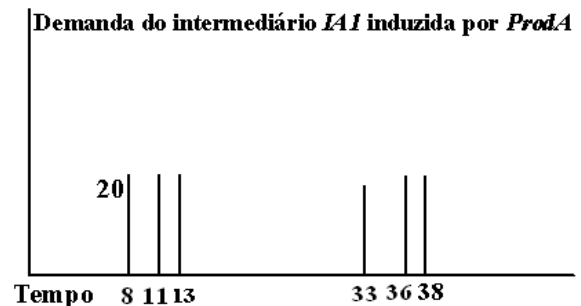


Figura 3.5: Demanda do intermediário *IA1*.

É necessário utilizar o procedimento de explosão para trás até chegar no plano de necessidades de matérias primas. Feito isto se encerra a primeira etapa. A segunda etapa é uma explosão para frente, partindo de um plano de disponibilidade de matéria prima introduzido pelo usuário. O plano de disponibilidade de matéria prima é limitado pelo plano de necessidade de matérias primas. A explosão para frente é um balanço de massa, utilizado para determinar os instantes de início mais cedo dos lotes das operações.

As explosões para trás e para frente geram os instantes de início mais cedo (*Est*) e

os instantes de término mais tarde dos lotes (Lft), criando assim as janelas de processamento dos lotes. A figura 3.6 mostra uma janela de processamento com os seus quatro instantes principais. Os quatro instantes principais de uma janela de processamento são:

- instante de início mais cedo (Est), é o instante mais cedo que um lote pode ser iniciado na janela.
- instante de término mais tarde (Lft), é o instante de término mais tarde que um lote deve ser finalizado na janela.
- instante de término mais cedo (Eft), é o instante de término mais cedo que um lote pode ser finalizado na janela, ele é definido pela soma do Est com o tempo de processamento do lote.
- instante de início mais tarde (Lst), é o instante de início mais tarde que um lote deve ser iniciado na janela, ele é definido pelo Lft subtraído do tempo de processamento do lote.

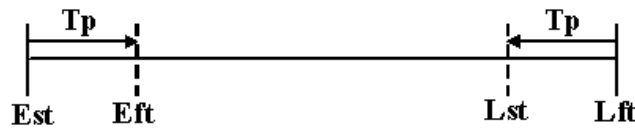


Figura 3.6: Instantes principais de uma janela de processamento.

A seguir será exemplificada a determinação das janelas de processamento. Considerando, por exemplo, o STN da figura 3.7, onde constam as operações, tempos de processamento, tamanhos dos lotes, condições de armazenagem, as tabelas 3.1, e 3.2, apresentam, respectivamente: a habilitação dos lotes nos processadores e condições de armazenagem, as demandas e datas de entrega dos produtos finais fornecidas pelo planejamento. As matérias primas estarão disponibilizadas a partir do instante $T = 1$.

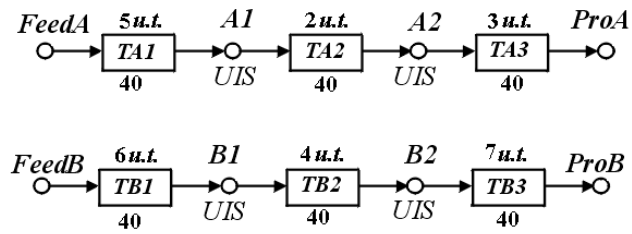


Figura 3.7: Representação *STN* do exemplo considerado.

Equipamentos	Lotes	Armazenagem
P1	TA1,TB1	UIS
P2	TA2,TB2	UIS
P3	TA3,TB3	UIS

Tabela 3.1: habilitação dos lotes e condições de armazenagem.

Produtos	Demanda	Data de entrega
ProA	40	21
ProB	40	21

Tabela 3.2: Demanda de produtos finais e datas de entrega.

Utilizando os dados da figura 3.7 e das tabelas 3.1 e 3.2, as figuras 3.8 e 3.9 ilustram a determinação das janelas iniciais de processamento do exemplo em questão. A figura 3.8 mostra o procedimento de explosão de requisitos para trás. Nesta etapa é utilizada a demanda de produtos finais para calcular o número de lotes de cada operação e suas datas de entrega. A figura 3.9 ilustra o procedimento de explosão para frente que utiliza o plano de disponibilidade de matérias primas. Neste exemplo as matérias primas estão disponibilizadas em $T = 1$.

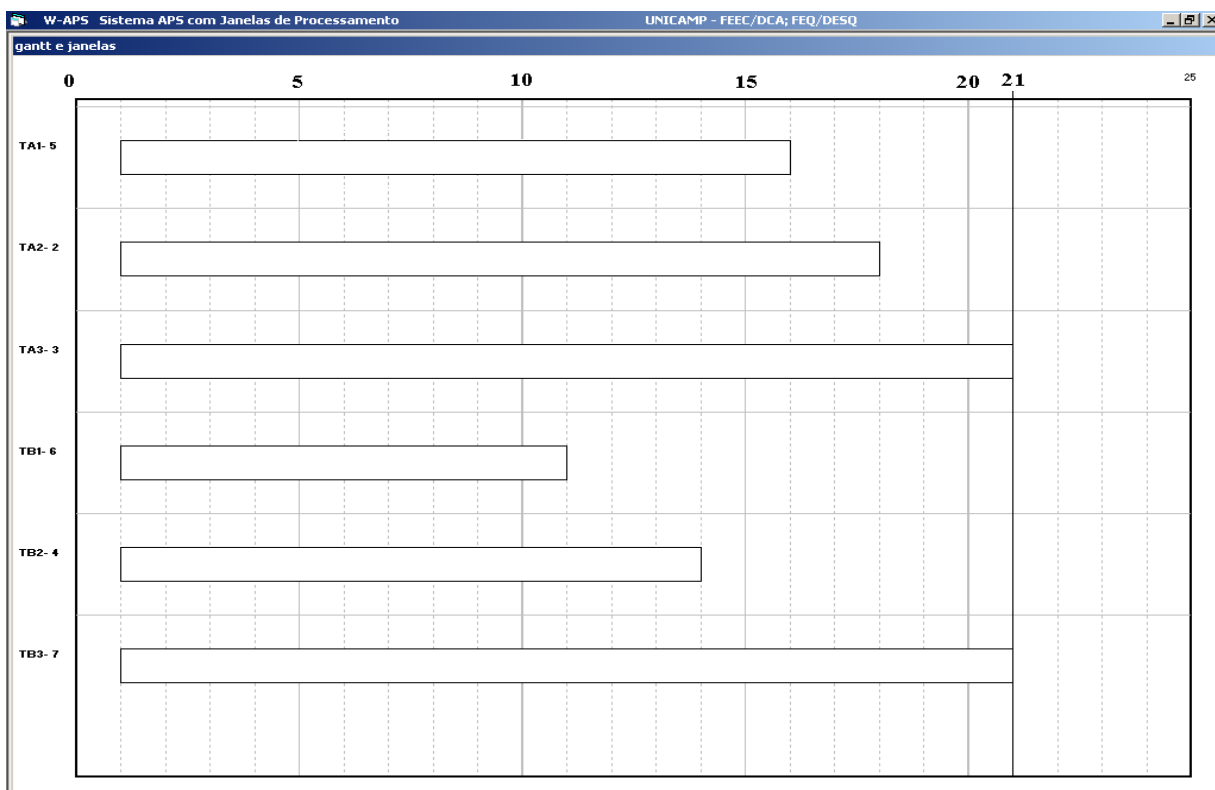


Figura 3.8: Procedimento de explosão para trás.

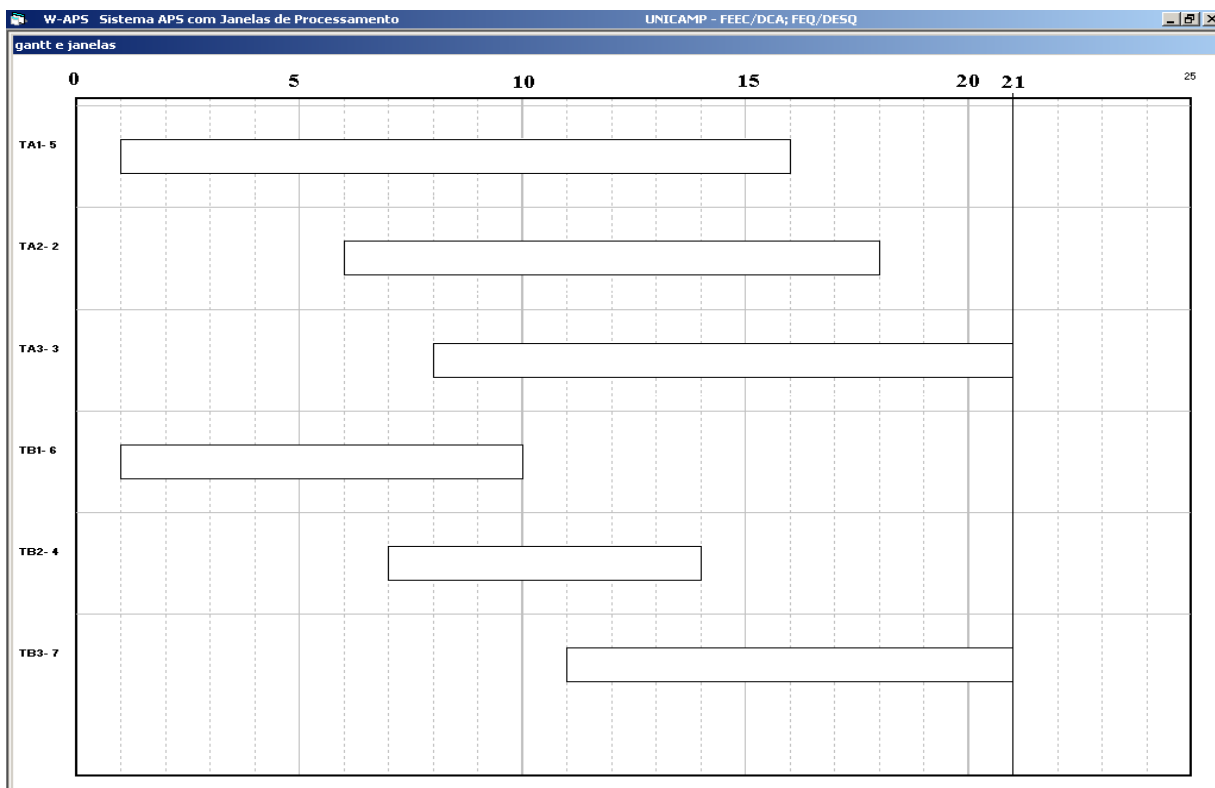


Figura 3.9: Procedimento de explosão para frente (janelas iniciais).

3.3 Propagação de restrições

Este é o mecanismo responsável por manter compatíveis com as restrições os limites das janelas de processamento (Est e Lft) de todos os lotes. Neste trabalho são propagadas as seguintes restrições:

- restrições de capacidade (recurso unitário), os lotes atribuídos ao mesmo processador não podem se sobrepor;
- restrições de precedência por receita, o instante de início mais cedo do lote consumidor deve ser igual ou maior que o instante de término mais cedo do lote produtor, e o instante de término mais tarde do lote produtor deve ser menor ou igual ao instante de início mais tarde do lote consumidor;
- restrições por monotonicidade: serão produzidos consecutivamente os lotes da mesma operação que estejam atribuídos ao mesmo processador;
- restrições de armazenagem.

3.3.1 Restrições de capacidade

A restrição de capacidade garante que os lotes não se sobrepõem no tempo, ou seja, não é possível processar dois lotes ao mesmo tempo no mesmo processador. Para garantir esta restrição, existem duas situações em que podem levar a modificações nas janelas de processamento, as quais são:

- Janela de processamento com um intervalo de tempo que obrigatoriamente será utilizado pelo lote (“*slots of total reliance*”) (Sadeh, 1991).
- Janelas de processamento com um ordenamento obrigatório entre si.

A primeira situação acontece quando uma janela de processamento tem um intervalo de tempo que será obrigatoriamente utilizado pelo lote. Isto acontece quando na janela de processamento tem-se $Eft > Lst$, gerando assim um intervalo obrigatoriamente utilizado

(Sadeh, 1991). Neste caso é necessário excluir este intervalo das janelas dos outros lotes que o contenham, evitando assim sobreposição de lotes. A exclusão deste intervalo das outras janelas gera aumentos de Est , ou diminuições de Lft ou janelas disjuntas.

A segunda situação é quando existem ordenamentos obrigatórios entre lotes. Os ordenamentos obrigatórios entre lotes são necessários quando apesar de ser possível alocar os lotes com janelas competindo entre si pelo mesmo processador, é obrigatório um ordenamento entre os lotes para que a alocação seja factível. As equações de 3.1 a 3.5 em conjunto com as figuras 3.10 e 3.11 exemplificam a questão para dois lotes TA e TB , com os tempos de processamento de TP_{TA} e TP_{TB} , e existindo sobreposição entre suas janelas de processamento.

Como os dois lotes não podem ocupar o processador ao mesmo tempo, mas as suas janelas estão sobrepostas, é necessário verificar se é possível alocar os dois lotes no intervalo de tempo definido pela união de suas janelas, pois se não for possível o problema é infactível. Somente é possível alocar dois lotes com janelas sobrepostas se o intervalo da união das janelas for maior ou igual a soma dos tempos de processamento dos lotes, ou seja a factibilidade é garantida se:

$$\max(Lft_{TA}, Lft_{TB}) - \min(Est_{TA}, Est_{TB}) \geq TP_{TA} + TP_{TB} \quad (3.1)$$

Se o problema for factível, verifica-se a necessidade de um ordenamento obrigatório entre lotes. Somente poderá ser obrigatório o ordenamento se a intersecção das janelas for menor que a soma dos tempos de processamento. O lote que precede é o lote que tiver o instante de término mais tarde de sua janela (Lst) menor que o instante de início mais cedo da outra janela (Eft). As equações 3.2 e 3.3 representam as condições que implicam em ordenamentos, caso necessário.

$$Eft_{TB} > Lst_{TA} \quad \text{então} \quad TA < TB \quad (TA \text{ precede } TB) \quad (3.2)$$

$$Eft_{TA} > Lst_{TB} \quad \text{então} \quad TB < TA \quad (TB \text{ precede } TA) \quad (3.3)$$

Como os lotes não podem se sobrepor, determinados ordenamentos obrigatórios podem implicar em reduções nas janelas de processamento. Estas reduções podem gerar aumentos de Est e diminuições de Lft . Como o lote precedente tem que estar pronto para começar o outro lote, então o instante final do lote precedente tem que ser menor ou igual ao instante de início do outro lote. As equações 3.4 e 3.5 representam as reduções. Existem reduções quando TA precede TB ou quando TB precede TA .

No primeiro caso.

$$\text{Se } Est_{TB} < Est_{TA} + TP_{TA} \text{ obriga-se à } Est_{TB} = Est_{TA} + TP_{TA}$$

$$\text{Se } Lft_{TA} > Lft_{TB} - TP_{TB} \text{ obriga-se à } Lft_{TA} = Lft_{TB} - TP_{TB} \quad (3.4)$$

Analogamente, caso TB precede TA .

$$\text{Se } Est_{TA} < Est_{TB} + TP_{TB} \text{ obriga-se à } Est_{TA} = Est_{TB} + TP_{TB}$$

$$\text{Se } Lft_{TB} > Lft_{TA} - TP_{TA} \text{ obriga-se à } Lft_{TB} = Lft_{TA} - TP_{TA} \quad (3.5)$$

As figuras 3.10 e 3.11 mostram, respectivamente, para a restrição TA precede TB , o ordenamento obrigatório e a redução das janelas pela propagação da restrição.

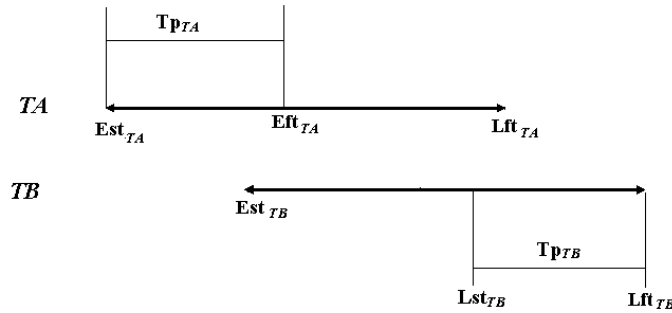


Figura 3.10: Ordenamento obrigatório TA precede TB

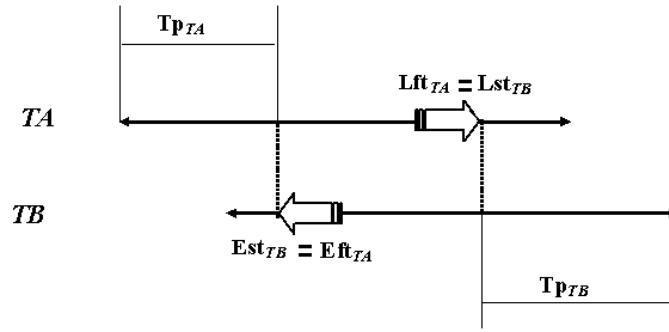


Figura 3.11: Propagação a partir das janelas iniciais da restrição TA precede TB

É possível generalizar o ordenamento entre lotes para um ordenamento entre um lote A e um conjunto de lotes que competem pelo mesmo processador. As condições que deduzem esse ordenamento foram apresentadas em (Bel et al., 1989) e em (Caseu e Laburthe, 1995). Neste tipo de ordenamento existem duas possibilidades de análise: i) quando o lote A pertence ao conjunto de lotes (*edge finding*); e ii) quando o lote A não pertence ao conjunto de lotes (exclusão). A descrição detalhada desta análise pode ser encontrada em (Rodrigues, 2000).

3.3.2 Restrições de precedência por receita

Como o lote produtor tem que estar pronto para poder iniciar o lote consumidor, ou seja o lote produtor *precede* o lote consumidor, portanto é necessário utilizar restrições entre os inícios e fins das janelas de processamento dos lotes envolvidos por precedências. As restrições utilizadas são:

- restrição entre o instante de início mais cedo da janela do lote consumidor e o instante de término mais cedo do lote produtor, ou seja:

$$Est_{consumidora} \geq Eft_{produtora}$$

- restrição entre o instante de término mais tarde do lote produtor e o instante de início mais tarde do lote consumidor, ou seja:

$$Lft_{produtora} \leq Lst_{consumidora}$$

3.3.3 Restrições por monotonicidade

As restrições de monotonicidade supõem que se forem necessários vários lotes da mesma operação e se eles forem produzidos no mesmo processador, estes lotes podem ser considerados como lotes consecutivos. Para representar esta condição é necessário que:

$$Est_{lote+1} \geq Est_{lote} + Tp$$

$$Lft_{lote} \leq Lft_{lote+1} - Tp$$

3.3.4 Restrições de armazenagem

Existem quatro tipos de armazenagem elas são:

- Ilimitada (*UIS* ou *unlimited intermediate storage*)
- Limitada (*FIS* ou *finite intermediate storage*)
- Inexistente (*NIS* ou *no intermediate storage*)
- Sem espera (*ZW* ou *zero wait*)

A armazenagem do tipo *UIS* considera que exista capacidade de armazenagem ilimitada, e sendo assim não há nenhuma restrição a ser imposta ao problema. Os outros tipos de armazenagem serão descritos a seguir.

Armazenagem Limitada (FIS)

A armazenagem limitada é utilizada quando existe limitação de capacidade de estocagem do produto processado. Neste caso as janelas dos lotes produtores estão interligadas com as janelas dos lotes consumidores de forma que não seja ultrapassada a capacidade de estocagem. Uma descrição mais detalhada pode ser encontrada em (Rodrigues, 2000).

Armazenagem inexistente (NIS)

Neste tipo de armazenagem, não existem local para armazenar o produto processado. Entretanto o produto pode ficar armazenado no processador que o produziu até que exista demanda para ele, ou até que o processador seja requisitado por outra operação. A mesma referência (Rodrigues, 2000) contém uma descrição detalhada.

Armazenagem sem espera (ZW)

No caso em que os produtos intermediários são produtos instáveis, é obrigatório que exista uma sincronização do lote produtor com o lote consumidor. Assim sempre que houver uma redução na janela de processamento no lote produtor, será propagada uma redução para o lote consumidor e vice-versa. Para que a restrição por política *ZW* garanta a factibilidade entre janelas é necessário que:

- $Eft_{produtor} = Est_{consumidor}$;
- $Lst_{consumidor} = Lft_{produtor}$.

3.3.5 Exemplos

Considera-se o exemplo utilizado na seção 3.2 com a receita indicada na figura 3.7(*STN*). As janelas de processamento dos lotes obtidas inicialmente são indicadas na figura 3.9 repetida abaixo. A propagação de restrições leva às janelas finais da figura 3.12.

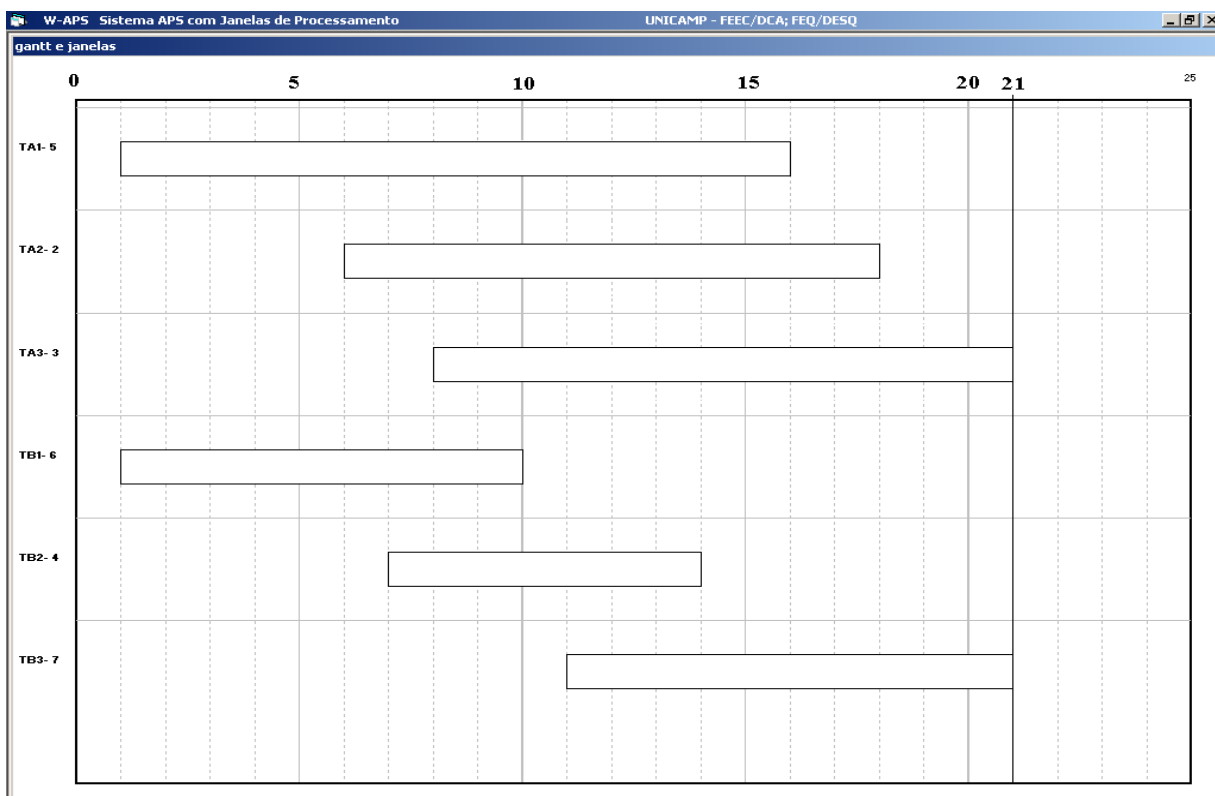


Figura 3.9: Procedimento de explosão para frente (janelas iniciais).

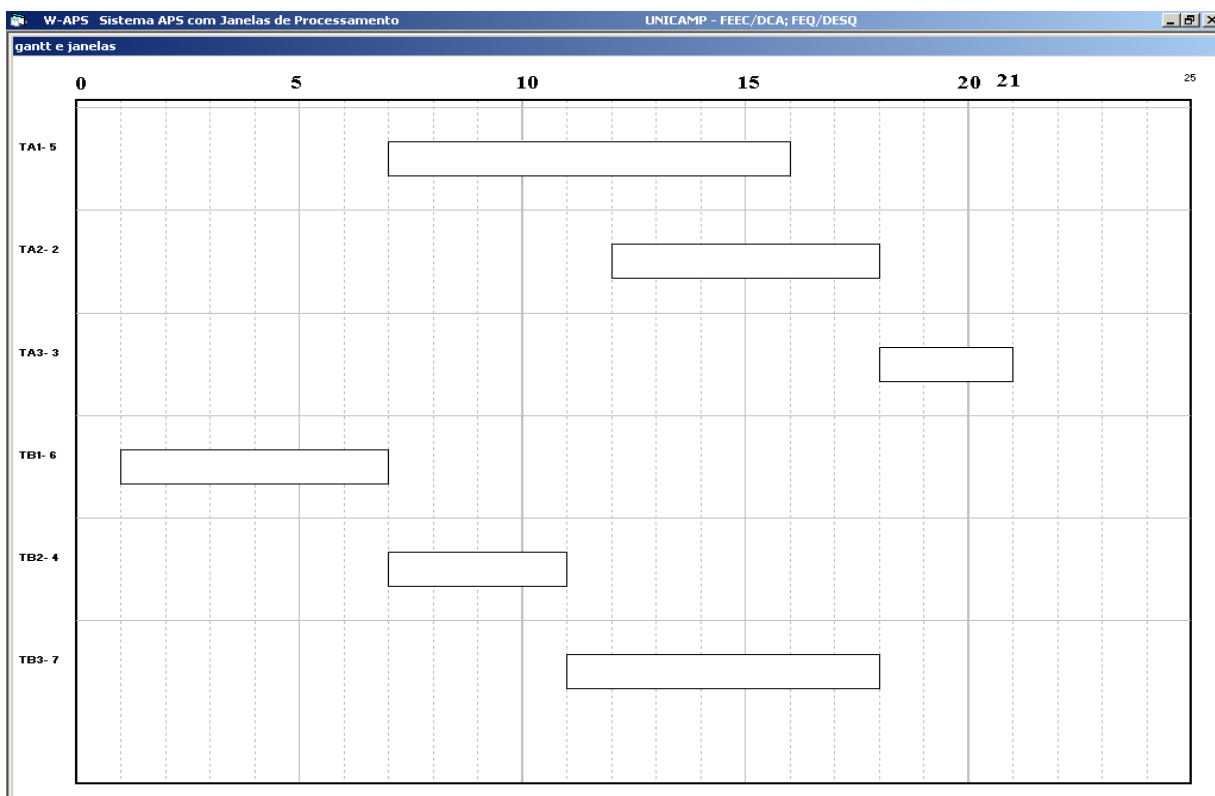


Figura 3.12: Janelas de processamento (finais) do *STN* da figura 3.7, utilizando os dados da tabela 3.2.

As modificações nas janelas foram originadas pelas condições seguintes:

O lote $TB1$ tem a janela (1; 10), dado que o tempo de processamento é $Tp_{TB1} = 6$ o intervalo (4; 7) é um intervalo que com certeza será ocupado pelo lote (“*slot of total reliance*” seção 3.3.1). Este intervalo ocupado no processador $P1$ não estará disponível para o lote $TA1$ atribuído ao mesmo processador. Dado que a janela deste lote é (1; 16) e que o intervalo (1; 4) não é suficiente ($Tp_{TA1} = 5$) o início da janela de $TA1$ deve ser aumentado para $Est_{TA1} = 7$ ou seja a janela passa a ser (7; 16).

As restrições de balanço de massa fazem com que as janelas de $TA2$ e $TA3$ sejam modificadas para (12; 18) e (14; 21). O lote $TB3$ tem a janela (11; 21) e com o tempo de processamento $Tp_{TB3} = 7$ gera o intervalo de ocupação obrigatória (14; 18) que reduz por sua vez a janela de $TA3$ para (18; 21). Como $Tp_{TA3} = 3$ o intervalo (18; 21) é de ocupação obrigatória e portanto a janela de $TB3$ passa a ser (11; 18). A propagação por balanço de massa leva a modificação dos Lft de $TB2$ e $TB1$ respectivamente dando lugar as janelas (7; 11) e (1; 7).

Considera-se a seguir o Exemplo 1 da seção 2.4 sem limitações de armazenagem, e utilizando os dados das tabelas 2.7 e 2.8. A figura 3.13 apresenta as janelas iniciais do planejamento, e a figura 3.14 apresenta as janelas finais, ou seja após a propagação de restrições.

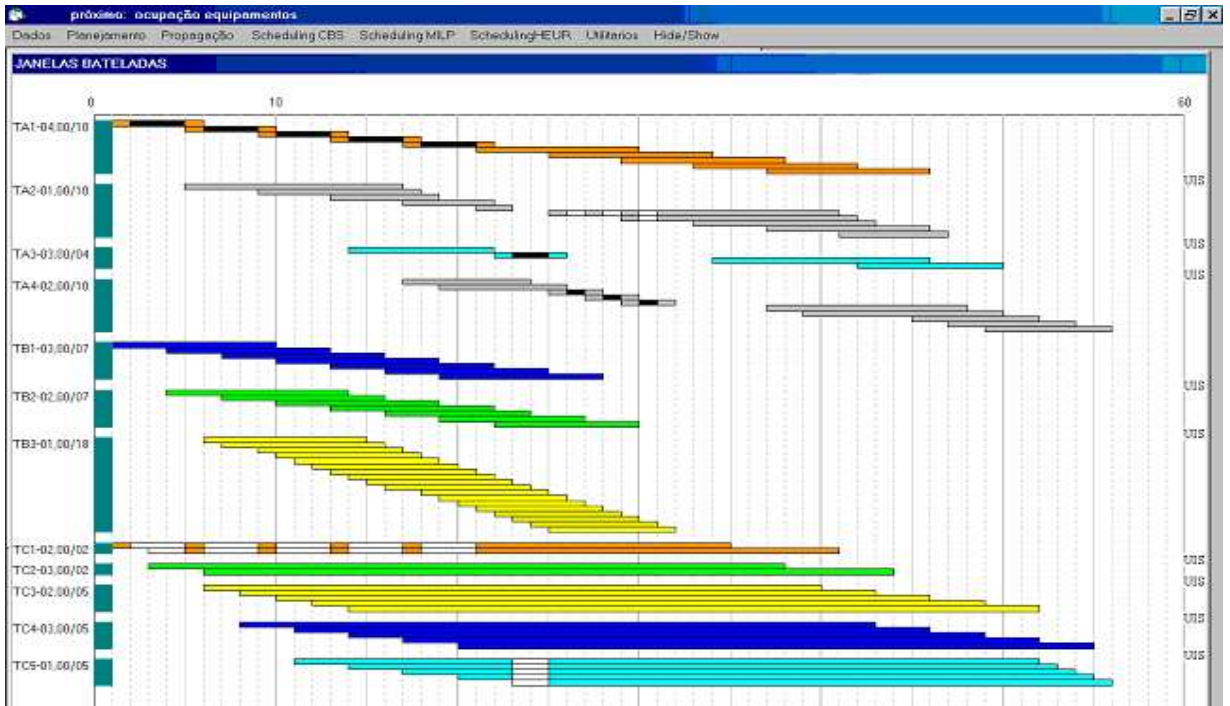


Figura 3.13: Janelas de processamento (iniciais) do STN da figura 2.33, utilizando os dados das tabelas 2.7 e 2.8.

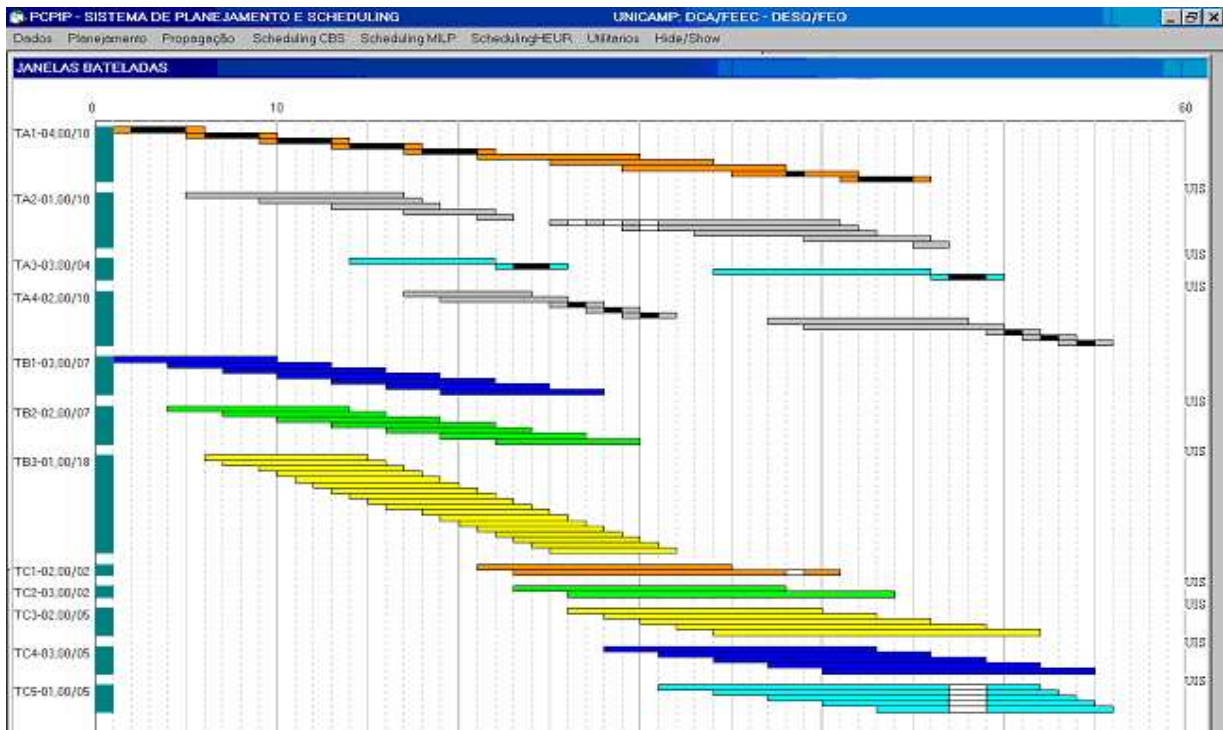


Figura 3.14: Janelas de processamento (finais) do *STN* da figura 2.33, utilizando os dados das tabelas 2.7 e 2.8.

Uma das funções do mecanismo de propagação é manter os limites das janelas atualizados, ou seja, limites reais, e não é possível prever os recortes que o mecanismo irá fazer. Caso algum recorte seja infactível, ou seja o tempo de processamento do lote for maior que o tamanho da janela, o sistema envia uma mensagem de infactibilidade e não gera as janelas.

Pelas figuras 3.13 e 3.14 é possível perceber o grande recorte que o mecanismo de propagação de restrições pode fazer nas janelas de processamento (janelas dos lotes do produto *ProdC*). A janela de processamento do lote *TC1/1* que inicialmente era de (2; 35) passou a ser de (21; 35). Este fato ocorreu porque as janelas dos lotes *TA1/1*, *TA1/2*, *TA1/3*, *TA1/4*, e *TA1/5* são janelas que contêm “*slot of total reliance*” e estes *slots* geraram recortes nas janelas do lotes *TC1/1* e *TC1/2*. Após os recortes na janela de *TC1/1*, sobraram pequenos intervalos disponíveis na janela de *TC1/1* antes de $T = 21$, mas como estes intervalos eram menores que o tempo de processamento da operação ($T_{p_{TC1}} = 2$) eles foram eliminados. A propagação por balanço de massa leva a modificação dos *Est* de *TC1/2*, *TC2/1,2*, *TC3/1,2,3,4,5*, *TC4/1,2,3,4,5* e *TC5/1,2,3,4,5* respectivamente dando lugar as novas janelas.

3.4 Utilização das janelas de processamento nas técnicas de scheduling da busca orientada pelas restrições (CBS)

A busca orientada por restrições (*Constraint Based Search, CBS*) é uma estratégia de busca em árvore, onde os nós são soluções parciais do problema de programação e os nós finais representam soluções completas do problema. Uma decisão de programação leva de um nó pai a um nó filho. As decisões de programação são decisões de alocação ou de ordenamento, onde a seleção do(s) lote(s) a alocar ou a ordenar são orientadas pelo gargalo.

A alocação decide o momento em que um lote é iniciado, enquanto que o ordenamento define uma relação de precedências entre dois lotes. No primeiro caso, o problema de programação está resolvido quando são alocados todos os lotes. Neste caso, o instante específico em que o lote inicia e termina é conhecido, e a partir destes é obtida a Carta de Gantt diretamente. No segundo caso (ordenamento), o problema de programação (*Scheduling*) está finalizado quando todas as precedências estão fixadas.

A busca orientada por restrições tem como características principais a utilização de procedimentos para determinar a decisão de programação em cada nó e a utilização de técnicas de propagação de restrições após cada decisão de programação para reduzir a árvore de busca. O processo heurístico da busca orientada pelas restrições é composto por vários componentes: o nó da árvore que será escolhido para a abertura, o(s) lote(s) que serão escolhidos pelo cenário representado e a decisão de programação sobre este(s) lotes(s).

Sua principal característica é que o processo de busca é orientado pelos gargalos do processo, onde o gargalo é originado pela competição entre operações por um mesmo processador. Após a detecção do gargalo, procede-se à escolha do(s) lote(s) sobre os quais serão tomadas as decisões de alocação ou de ordenamento. As regras utilizadas para esta escolha são conhecidas na literatura *CBS* (Sadeh, 1991) e (Smith e Cheng, 1993) por heurísticas de escolha de variável (*variable ordering heuristics*) e as decisões tomadas sobre o(s) lotes(s) escolhido(s), são conhecidas como heurísticas de escolha de valor (*value ordering heuristics*).

A outra característica importante das estratégias *CBS* é a utilização dos mecanismos de propagação de restrições descritos na seção 3.3, que são ativados após cada decisão de programação, ou quando alguma janela sofrer um recorte devido a propagação de

restrições. Estes mecanismos propagam as restrições entre as janelas de processamento, gerando novos cenários para o problema e mantendo sempre as janelas atualizadas. Esse mecanismo é o responsável pela redução das janelas, gerando também uma redução no espaço de busca na árvore. Tratando-se de uma busca em árvore é necessário manter informações adequadas para permitir uma volta (*backtracking*) inteligente quando o processo leva a uma situação infactível (*dead end*).

Alguns softwares comerciais utilizam estratégias *CBS*. O sistema ILOG (ILO, 1997) considera apenas as restrições por recursos compartilhados unitários, ou seja, recursos que não podem ser utilizados por mais de uma operação ao mesmo tempo (por exemplo processadores). Entretanto o sistema Claire (Cla, 1999) vem sendo desenvolvido com extensões que consideram os recursos compartilhados cumulativos, ou seja, recursos que podem ser utilizados por várias operações ao mesmo tempo (por exemplo mão de obra, água).

Como mencionado na seção 3.2, supõe-se que exista uma fase anterior de planejamento onde são definidos previamente os lotes a serem alocados e a sua atribuição aos equipamentos. Para estes lotes supõe-se também que são conhecidas as relações de precedência entre eles, assim como uma janela de processamento para cada lote (podendo ser do tamanho do horizonte de programação).

Quando a decisão de programação for de alocação, a decisão estará concluída quando todos os lotes tiverem sido alocados, satisfazendo as restrições se o que se trata é obter uma solução factível. Caso as decisões de programação de produção na árvore representarem decisões de ordenamento de lotes no mesmo equipamento, surge o aspecto de que todos os ordenamentos de lotes no mesmo equipamento podem estar definidos sem que a alocação de todos os lotes esteja obrigatoriamente definida, simplesmente porque algumas janelas restantes após a propagação das restrições são maiores que os tempos de processamento necessários. Esta solução pode ser apropriada porque permite um grau de flexibilidade no chão de fábrica que, pode permitir absorver perturbações durante a execução do programa de produção.

3.4.1 Representação de gargalos

Na programação da produção com técnicas *CBS* define-se gargalo como sendo o intervalo de tempo no qual existe uma competição grande das operações por recursos compartilhados unitários. Neste trabalho serão utilizados os procedimentos que utilizam representação do carregamento para detectar o gargalo. Estes procedimentos foram utilizados na literatura das técnicas *CBS* por (Sadeh, 1991) e (Keng et al., 1988).

O carregamento é uma medida de competição entre lotes pelo mesmo processador, ou seja uma medida de gargalo. Neste caso esta competição é determinada pela estimativa no tempo da demanda de cada lote dos recursos necessários ao seu processamento. A estimativa da demanda é feita a partir da janela de processamento de cada lote.

São expostas na literatura duas propostas sobre estimativas da demanda de um lote: a criticalidade apresentada por Keng (Keng et al., 1988) e a demanda individual por Sadeh (Sadeh, 1991). Estas propostas só se aplicam a recursos unitários, por exemplo, processadores que somente podem ser utilizados exclusivamente por uma operação por vez.

Define-se a criticalidade de um lote como o quociente entre o tempo de processamento do lote (Tp) e a duração da sua janela de tempo (DW). Um valor unitário para a criticalidade significa que o lote tem uma única opção de alocação dentro da sua janela, um valor menor implica que existem diversas opções de alocações possíveis (têm uma certa flexibilidade de alocação), e um valor maior representa uma situação infactível (janela menor que tempo requerido pelo processamento).

A demanda individual de um lote é a medida de que um intervalo de tempo discreto seja utilizado na alocação do lote. Esta medida é avaliada a partir das alocações possíveis que utilizam o intervalo de tempo e o total das alocações possíveis. Considerando o tempo discretizado a quantidade de possíveis alocações é definida da seguinte maneira: $DW - Tp + 1$.

Na determinação da “intensidade” de um gargalo é preciso considerar todos os lotes que competem entre si pelo processador. Keng e Sadeh utilizam a criticalidade e a demanda individual de cada lote para determinar a crucialidade e a demanda agregada de um processador respectivamente.

O procedimento é o mesmo para os dois casos: somam-se, para cada intervalo de tempo discreto, as contribuições dos lotes cujas janelas englobam o intervalo em questão. Desta forma, a demanda agregada de cada intervalo discreto é construída somando as demandas individuais e a crucialidade é obtida somando-se as crucialidades das janelas de processamento. Considerando por exemplo dois lotes TA e TB , com seus tempos de processamento de $Tp_{TA} = 2$ e $Tp_{TB} = 3$, e suas janelas de processamento $DW_{TA} = (3; 10)$ e $DW_{TB} = (5; 9)$, as figuras 3.15 e 3.16 ilustram respectivamente os métodos da crucialidade e demanda agregada.

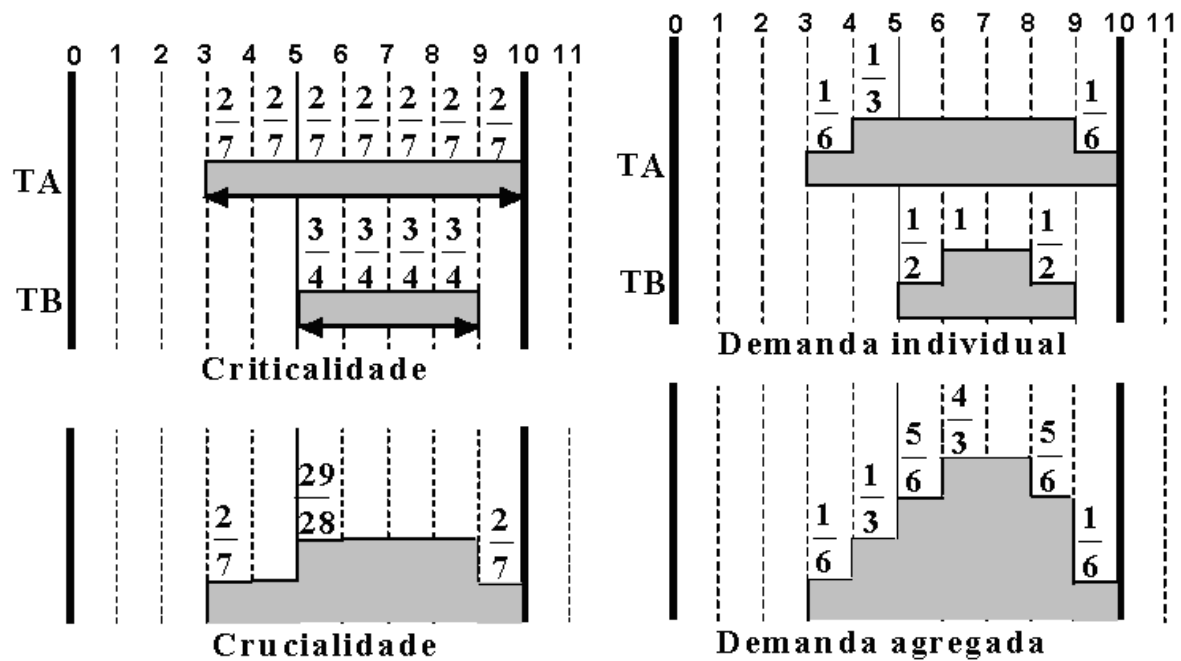


Figura 3.15: Curvas de crucialidade e crucialidade. Figura 3.16: Curvas de demanda individual e demanda agregada.

O conceito de crucialidade e de demanda agregada são utilizados nas heurísticas propostas por Keng (Keng et al., 1988) e Sadeh (Sadeh, 1991).

3.4.2 Exemplo

A figura 3.17 representa as janelas de processamento e as curvas de carregamento (demanda agregada) induzidas por estas janelas para o exemplo 1 da seção 2.4 sem limitações de armazenagem, e utilizando os dados das tabelas 2.7 e 2.8 (já referidas) que apresentam respectivamente: as habilitações dos lotes, e as demandas dos produtos finais e datas de entrega.

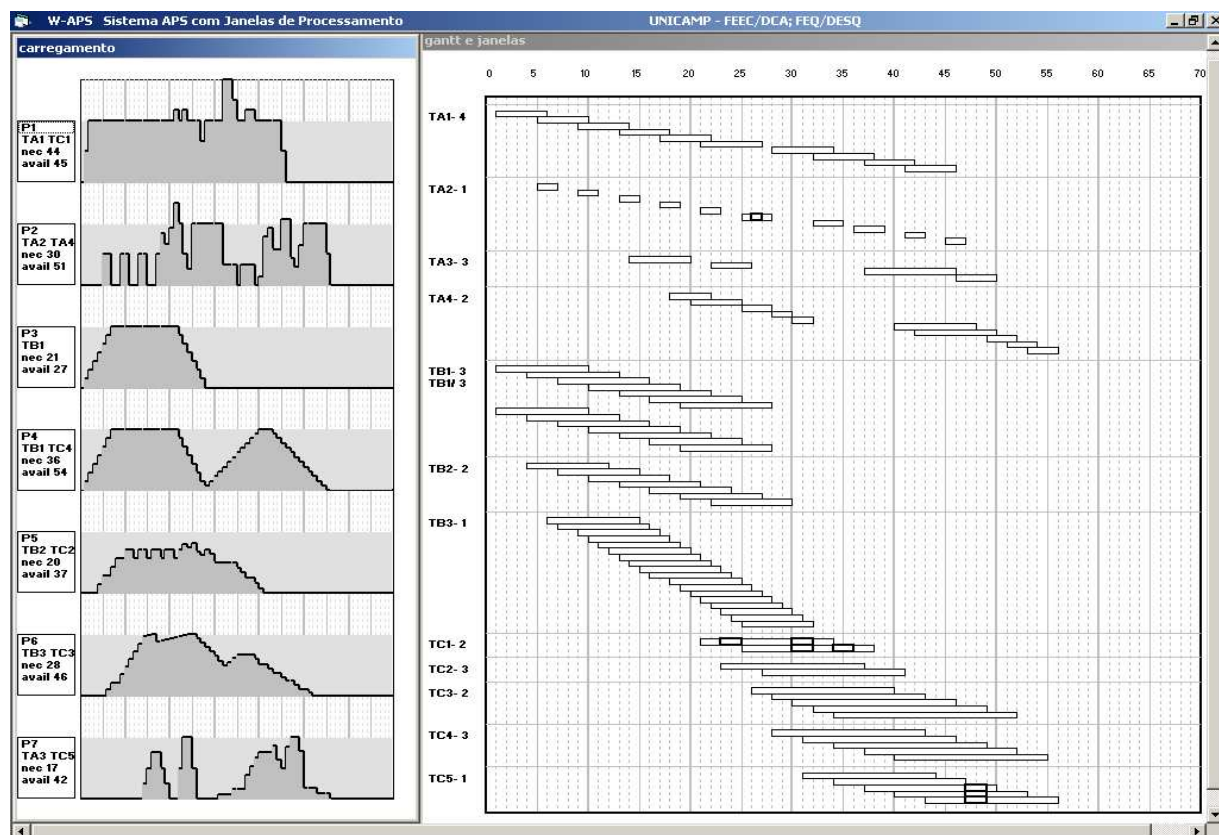


Figura 3.17: Janelas de processamento do *STN* da figura 2.33 e as curvas de carregamento, utilizando os dados das tabelas 2.7 e 2.8.

Junto às curvas de demanda agregada estão indicados o tempo necessário e o tempo disponível para cada processador. O primeiro é a soma dos tempos de processamento de todos os lotes atribuídos ao processador, o segundo a extensão da união das janelas correspondentes. Valores de folgas absolutas e relativas podem ser obtidos a partir destes valores.

É importante notar que quando as janelas são disjuntas, como por exemplo em *P2* o tempo disponível não é uma informação válida porque não considera os intervalos de tempo não utilizados.

3.5 Heurísticas APS em um cenário com datas de entrega importantes

O exemplo tratado na seção 2.6 é retornado para mostrar as dificuldades das heurísticas *APS* em problemas onde as datas de entregas são importantes. As figuras a seguir são as mesmas figuras apresentadas na seção 2.6 incluindo as janelas de processamento.

A figura 3.18 apresenta as alocações do exemplo 1 utilizando a heurística de alocação no instante mais cedo para a ordem de prioridades de produção *ProdA*, *ProdB* e *ProdC* e para a ordem de prioridades dos processadores *P1*, *P2* e *P3*.

Pela figura 3.18 é possível verificar que os lotes *TA1/9* e *TA1/10* foram alocados fora de suas janelas de processamento (35; 42) e (41; 46) respectivamente. Este fato não permitiu que os lotes de *TC1/1* e *TC1/2* fossem alocados dentro de suas janelas (21; 35) e (23; 41), atrasando assim a alocação de todos os outros lotes subseqüentes do produto *ProdC*, gerando um atraso na entrega do produto *ProdC* que era para ser entregue em $T = 56$ e somente será entregue em $T = 64$. O lote *TA4/4* também foi alocado fora de sua janela de processamento (27; 30). Este fato ocorreu porque o lote *TA4/3* estava alocado em (26; 28) e o lote *TA2/7* estava alocado em (29; 30), como o tempo de processamento do lote *TA4/4* é de $T_p = 2$ somente era possível alocá-lo depois de *TA2/7*. Esta alocação fez com que o lote *TA4/5* também fosse alocado fora de sua janela de processamento, gerando atraso na entrega da primeira demanda do produto *ProdA* que tinha a data de entrega de $T = 32$ e somente será entregue em $T = 36$.

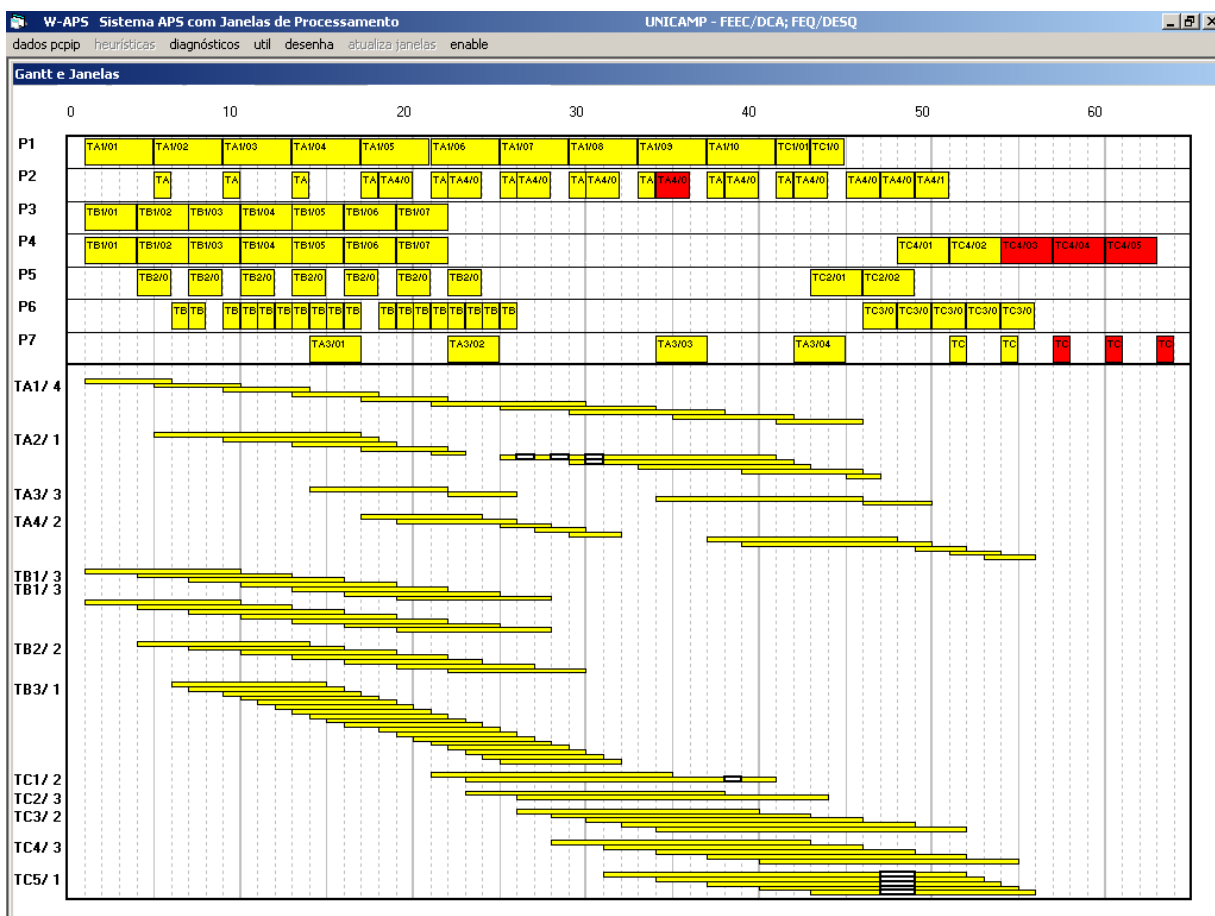


Figura 3.18: Alocações e janelas de processamento do exemplo 1 utilizando a heurística de alocação no instante mais cedo para a ordem de prioridades *ProdA*, *ProdB* e *ProdC*.

A figura 3.19 apresenta as alocações do exemplo 1 utilizando a heurística de alocação no instante mais tarde para a ordem de prioridades *ProdA*, *ProdB* e *ProdC* e para a ordem de prioridades dos processadores *P1*, *P2* e *P3*. Pela figura 3.19 é possível verificar que, como todos os lotes foram alocados dentro de suas janelas de processamento, todos os produtos serão entregues de acordo com a data de entrega.

A figura 3.20 apresenta as alocações do exemplo 1 utilizando a heurística de alocação *APS Relógio* para a ordem de prioridades *ProdA*, *ProdB* e *ProdC*. Todos os exemplos desta seção que utilizam esta heurística terão a ordem de prioridades dos processadores *P1*, *P2* e *P3*.

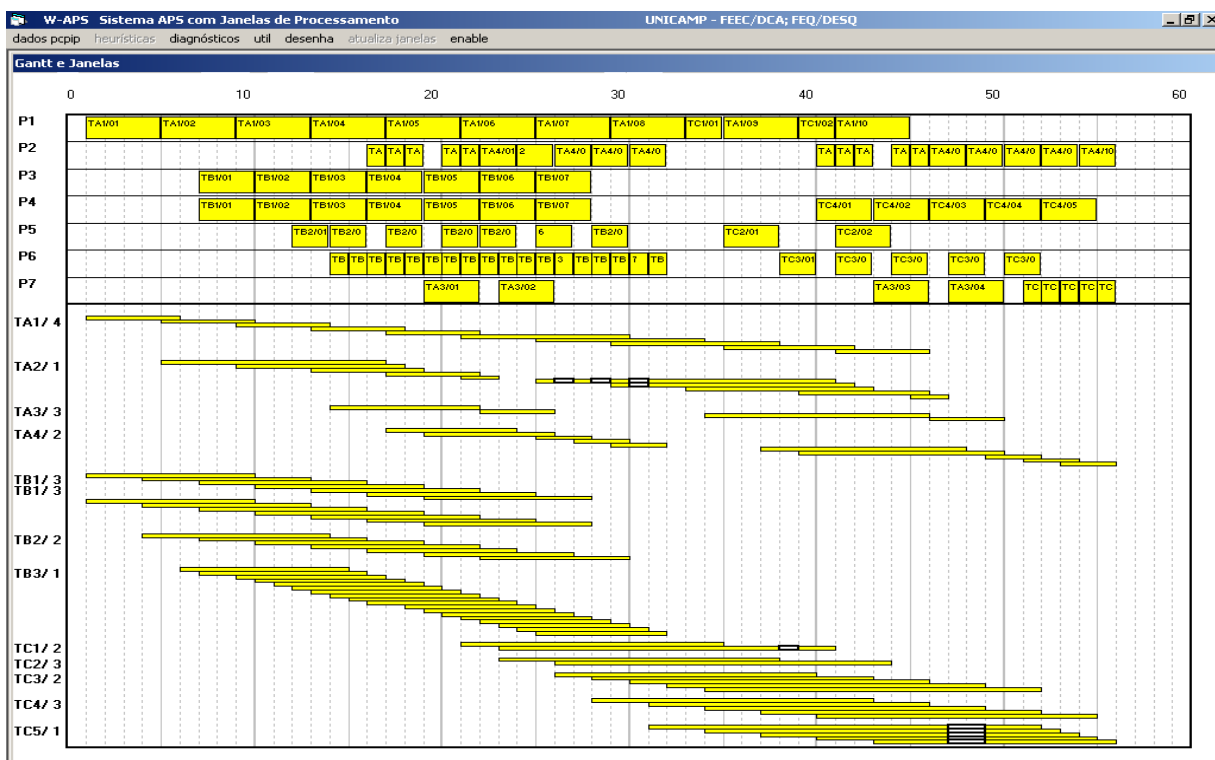


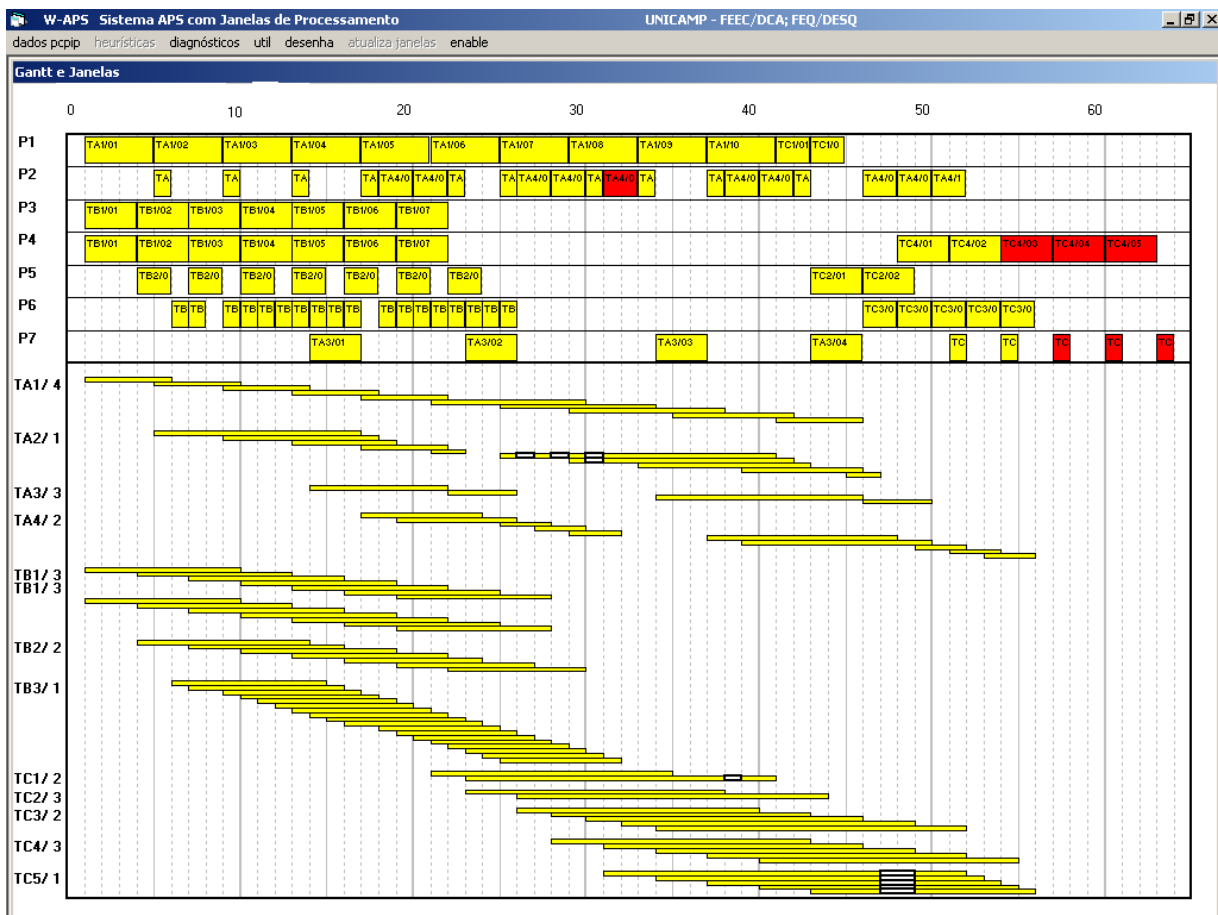
Figura 3.19: Alocações e janelas de processamento do exemplo 1 utilizando a heurística de alocação no instante mais tarde para a ordem de prioridades *ProDA*, *ProDB* e *ProDC*.



Figura 3.20: Alocações e janelas de processamento do exemplo 1 utilizando a heurística de alocação *APS* Relógio para a ordem de prioridades *ProDA*, *ProDB* e *ProDC*.

Pela figura 3.20 é possível perceber que o produto *ProdC* não cumpriu com a data de entrega pelo mesmo motivo da figura 3.18. Entretanto a primeira demanda do produto *ProdA* não cumpriu com a data de entrega porque o lote *TA4/5* foi alocado fora de sua janela de processamento (29; 32). Isto porque os lotes *TA4/4* e *TA2/7* estavam alocados respectivamente em (28; 30) e (30; 31). Pelas figuras 3.20 e 3.18 é possível perceber que utilizando a heurística *APS Relógio* o produto *ProdA* teve um pequeno atraso (de apenas 1 *u.t.*) em relação a heurística de alocação no instante de início mais cedo (atraso de 4 *u.t.*). Este fato é devido ao melhor aproveitamento do processador pela heurística *APS*.

A figura 3.21 apresenta as alocações do exemplo 1 utilizando a heurística de alocação *APS Relógio* para a ordem de prioridades *ProdB*, *ProdA* e *ProdC*.



mesmo processador com as operações do produto *ProdB*. Desta forma não houve nenhuma interferência nas ordens de alocações dos lotes, ou seja alocando com a prioridade (*ProdA*, *ProdB*, e *ProdC*) a solução final é a mesma que alocando com a prioridade (*ProdB*, *ProdA* e *ProdC*).

A figura 3.22 apresenta as alocações do exemplo 1 utilizando a heurística de alocação *APS Relógio* para a ordem de prioridades *ProdC*, *ProdA* e *ProdB*.



Figura 3.22: Alocações e janelas de processamento do exemplo 1 utilizando a heurística de alocação *APS Relógio* para a ordem de prioridades *ProdC*, *ProdA* e *ProdB*.

Pela figura 3.22 percebe-se que as alocações dos lotes *TC1/1,2* fora de suas janelas fizeram que os lotes *TA1/1, 2, 3, 4, 5* também fossem alocados fora de suas janelas. As alocações dos lotes *TC5/3, 4, 5* fora de suas janelas fizeram que o lote *TA3/2* fosse alocado fora também. Desta forma, e por consequência, os lotes *TA4/3, 4, 5* foram alocados fora atrasando assim a entrega da primeira demanda do produto *ProdA*. As alocações dos lotes *TC4/1, 2, 3, 4, 5* fora de suas janelas de processamento fizeram que os lotes *TB1/5, 6, 7* fossem alocados fora de

suas janelas. Por consequência os demais lotes do produto *ProdB* foram alocados fora de suas janelas de processamento (alocados mais tarde) gerando assim também um atraso na entrega do produto *ProdB*.

Como já comentado anteriormente, se para o problema o importante for cumprir com as datas de entrega, as janelas de processamento auxiliam muito este objetivo. As janelas não podem garantir que o problema tenha solução onde todos os produtos sejam entregues dentro do prazo, mas se existir esta solução ela será dentro das janelas. Portanto se algum lote for alocado fora de sua janela de processamento existirá algum produto entregue fora do prazo, conforme já descrito na seção 3.2 e demonstrado pelos exemplos desta seção.

Capítulo 4

Propostas de utilização das janelas de processamento nas heurísticas de alocação

4.1 Introdução

Neste capítulo será proposta a utilização das janelas de processamento dos lotes em heurísticas de alocação aplicáveis a sistemas *APS*. A principal característica das heurísticas *APS*, que é a rapidez, tem que ser mantida, pois é ela a responsável por permitir aos sistemas *APS* avaliar novos cenários de demanda em tempo hábil. Outra característica importante que deve ser mantida é a possibilidade de considerar as diversas restrições existentes nos casos reais. Esta característica já é suportada naturalmente pelas janelas de processamento em conjunto com a propagação de restrições. É esta característica que possibilita um resultado de programação confiável, mesmo em situações de restrições fortes sobre os recursos compartilhados pelas operações.

A utilização de janelas de processamento dos lotes implica que as datas de entrega dos produtos finais são importantes no problema de programação em questão. São estas datas de entrega que determinam os valores de Lft para as janelas através do procedimento de explosão para trás apresentado no capítulo 3.2. Estas datas de entrega podem não ser firmes no sentido que o usuário as defina como forma de construir um cenário de produção, mas de qualquer forma as janelas de processamento dos lotes somente existem após a definição de datas de entrega dos produtos finais. Um outro fator importante é que com a utilização das janelas de processamento é possível criar heurísticas onde a escolha de prioridades de alocação não dependam de decisões

de prioridades tomadas pelos usuário, conforme será discutido nesta seção.

O contexto para o estudo das heurísticas é rapidez na execução e aplicação em problemas com datas de entrega importantes. A exigência de rapidez será conseguida mantendo-se a forma de alocação sequencial utilizada nos sistemas *APS*, o que também permite considerar convenientemente as diversas restrições na planta.

Dos três tipos de heurísticas propostos neste capítulo um foi inspirado nas heurísticas normalmente encontradas em sistemas *APS* comerciais, heurísticas de alocação no instante de início mais cedo (H1-Est) ou heurísticas de alocação no instante de início mais tarde (H1-Lst). A heurística orientada por gargalos e a heurística de busca em árvore limitada foram inspiradas nos sistemas *CBS*. Como as janelas de processamento prevêm problemas de armazenagens, os exemplos desta seção serão mais complexos, ou seja com mais restrições.

4.2 Heurísticas de alocação no instante mais cedo ou mais tarde

As heurísticas deste tipo presentes nos sistemas *APS* e discutidas no capítulo 2 têm duas características básicas: a utilização de ordens de prioridades a serem definidas pelo usuário e em algumas delas a utilização de um mecanismo de relógio. As heurísticas *APS* utilizam esquemas de alocações por ordens de produção (seção 2.2) ou esquemas de alocações por operações (seção 2.3), as duas heurísticas propostas nesta seção não consideram o primeiro caso, porque em problemas com datas de entrega somente em situações de baixo carregamento o esquema de alocações por ordens de produção pode conseguir uma solução onde todos os produtos finais cumpram a data de entrega.

Nas heurísticas *APS* a utilização de ordens de prioridades é necessária para a escolha de um lote a ser alocado a cada passo do procedimento sequencial. O esquema mais utilizado para esse ordenamento nas heurísticas *APS* é o esquema de prioridades por ordens de produção, ou seja, é priorizado com um mesmo valor todos os lotes que compõem a ordem de produção. A ordem de prioridades tem uma influência muito grande sobre a solução de *scheduling*, e como esta ordem de prioridades é introduzida pelo usuário, este é um dos grandes problemas dos sistemas *APS*. Isto porque em situações complexas, por exemplo, com alto

carregamento dos equipamentos, não é óbvio o melhor ordenamento de prioridades, e conforme apresentado no exemplo da seção 3.5 pode ser que não exista solução seguindo este ordenamento de prioridades (de acordo com as alocações utilizando a heurística *APS* relógio).

A figura 4.1, correspondente ao exemplo apresentado na seção 3.5 acrescido das restrições de armazenagens da tabela 4.1, ilustra o comentário anterior. Considerando que este exemplo tem uma ordem de produção do produto *ProdA* com duas demandas, e cada uma delas com 5 lotes de produto final, uma ordem de produção do produto *ProdB* com 18 lotes de produto final, e uma ordem do produto *ProdC* com 5 lotes de produto final. Considerando a segunda demanda de produção de *ProDA* e a ordem de produção de *ProdC*, as janelas dos lotes *TA3/3, 4* e *TC5/1, 2, 3, 4, 5* com ambas operações (*TA3* e *TC5*) competindo pelo processador *P7* mostram que não será possível processar os dois conjuntos de lotes um após o outro (será necessário uma alternância entre os lotes de diferentes ordens de produção). Esta alternância dificilmente pode ser conseguida através de um esquema de prioridades por ordens de produção.

Estados	Armazenagem
A1, B1, C1, C4	ZW
A3,B2	FIS(50)
C2	FIS(100)
A2,C3	UIS

Tabela 4.1: Condições de armazenagem.

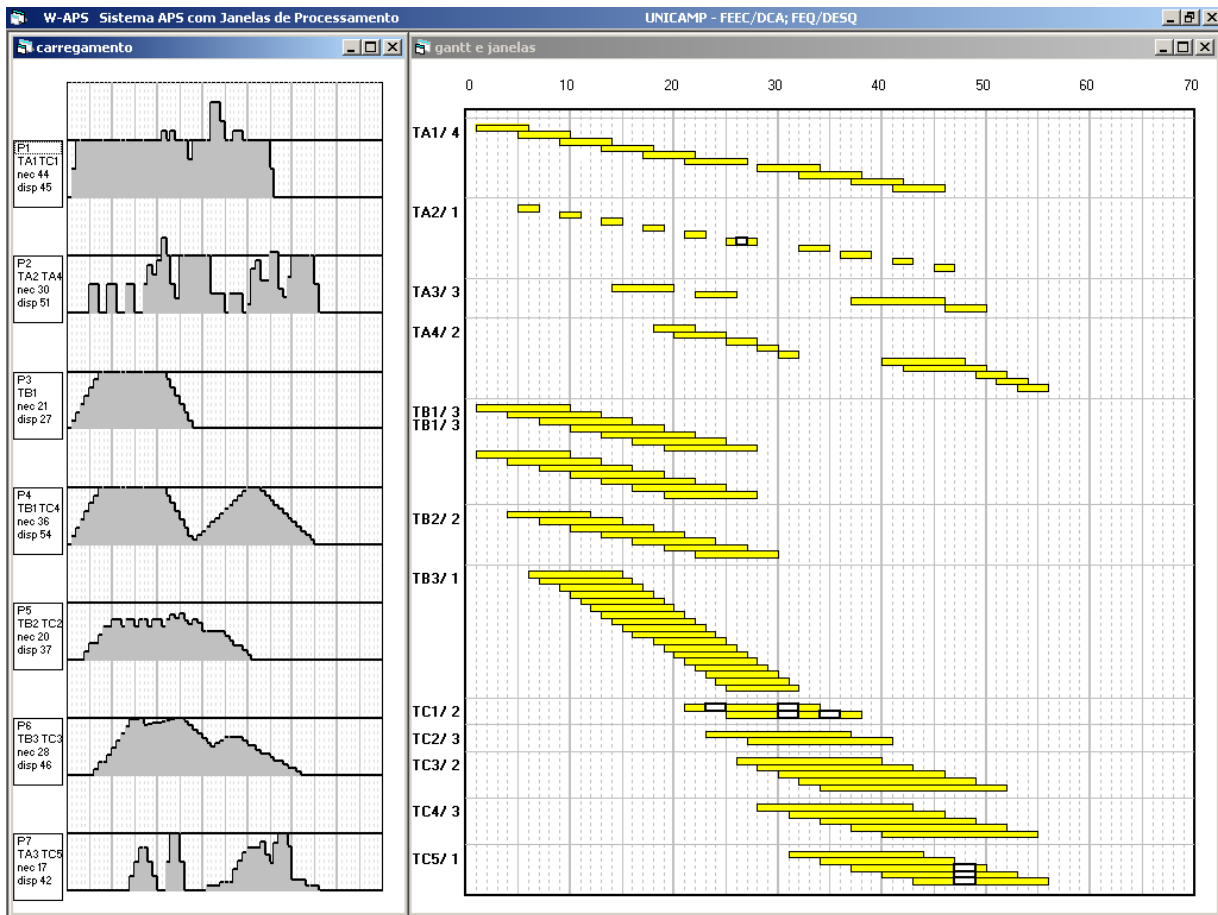


Figura 4.1: Janelas de processamento, e curvas de carregamento.

Foi devido a esta alternância obrigatória entre os lotes que no exemplo da seção 2.6 apenas uma heurística conseguiu uma solução final onde todos os produtos finais cumprissem com a data de entrega. A heurística que obteve esta solução final foi a heurística de alocação no instante mais tarde, isto porque *coincidentemente* houve alternância entre as alocações dos lotes $TA1/8$, $TC1/1$, $TA1/9$, $TC1/2$, $TA1/10$ no processador $P1$.

Como o objetivo final deste processo de ordenamento de prioridades é a seleção de um lote a ser alocado no procedimento seqüencial, a melhor forma de priorizar esta seleção é utilizando um ordenamento de prioridades por lotes. A adoção de um procedimento de ordenamento por lotes nas heurísticas *APS* convencionais complicaria ainda mais a vida do usuário, isto porque aumentaria bastante o número de decisões de prioridades que o usuário tem que tomar, ficando inviável tal atitude. Como uma das intenções deste trabalho é de propor heurísticas em que a seleção do lote a ser alocado seja feito sem a necessidade de prioridades

fornecidas pelo usuário, a utilização de ordenamentos de prioridades por lotes é viável.

As janelas de processamento trazem duas vantagens importantes para o processo de escolha do lote a alocar:

- As janelas eliminam a necessidade de um mecanismo de relógio, pois o instante de tempo a partir do qual o lote pode ser alocado já está definido pelo *Est* da sua respectiva janela (analogamente o instante de tempo mais tarde, *Lst*). Convém notar que este instante de tempo independe da alocação anterior de lotes que precedam o lote em questão, isto porque as janelas já incorporam as condições de precedência entre lotes;
- A eliminação da responsabilidade do usuário ter que informar uma ordem de prioridade de operações, pois a escolha de um lote pode ser feita baseada na dificuldade esperada na alocação deste lote, que pode ser estimada pela criticalidade de cada lote (Keng et al., 1988), ou seja o quociente entre o tempo de processamento e a duração da sua janela. Assim por exemplo na figura anterior, num procedimento de alocação no instante mais cedo, inicialmente os lotes candidatos seriam *TA1/1* e *TB1/1*, com suas respectivas criticalidades $4/5$ e $3/9$ e seria lógico iniciar o procedimento sequencial alocando o lote *TA1/1*.

O lote mais crítico pode porém não ser um dos lotes com início da janela mais próximo do instante inicial, ou seja, seria necessário considerar um horizonte de tempo para a seleção do lote. Isto é possível dado que, como mencionado anteriormente, a heurística não exige que a alocação de um lote seja precedida pela alocação dos lotes que o precedem.

Com a utilização das janelas os instantes de início mais tarde também estão bem definidos e, como elas já incorporam as condições de precedência entre lotes, estas condições possibilitaram a criação da heurística de alocação no instante de início mais tarde. Esta é semelhante a heurística de alocação no instante de início mais cedo, entretanto a escolha do lote a alocar é a partir dos *due dates* dos lotes, e a alocação é no instante de início mais tarde.

Após cada alocação as janelas dos lotes ainda não alocados podem sofrer modificações através dos mecanismos de propagação das restrições na forma de aumentos de *Est* e/ou reduções de *Lft*. A figura 4.2 mostra o resultado após a alocação de *TA1/10* em um

procedimento de alocação no instante de início mais tarde.

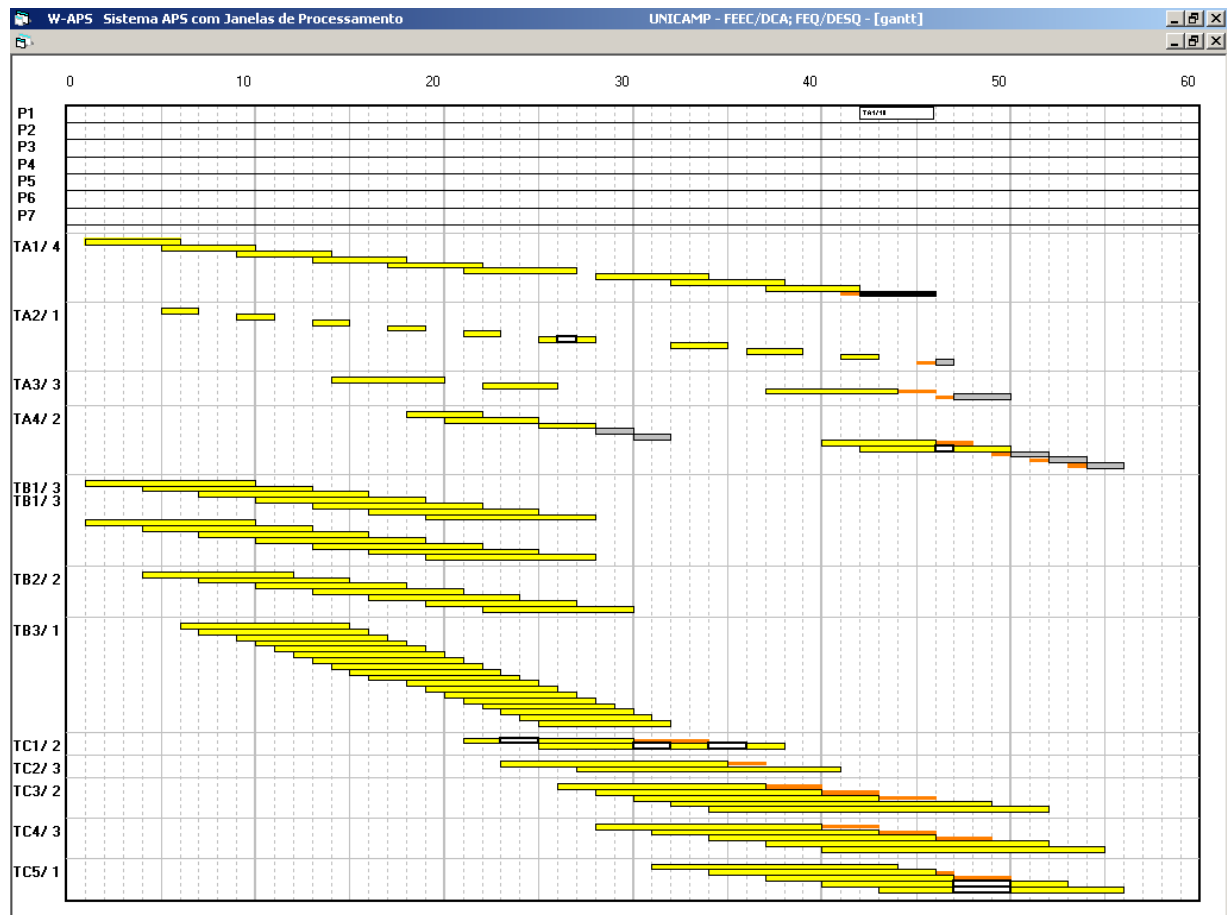


Figura 4.2: Carta de Gantt, e janelas de processamento resultante da alocação de $TA1/10$ no instante de início mais tarde, e janelas iniciais.

A propagação de restrições resultante da alocação de $TA1/10$ levou a reduções das janelas de $TA2/10$ (restrição de armazenagem ZW), $TA3/4$ (restrição de balanço), $TC5/1, 2, 3$; $TA3/3$ (restrição de capacidade em $P5$), $TC4/1, 2, 3$; $TC3/1, 2, 3$; $TC2/2$; $TC1/1$ (restrição de balanço). Isto leva, por exemplo, a alocações obrigatórias para os lotes: $TA2/10$, $TA3/4$, $TA4/8, 9, 10$. Torna-se assim necessário proceder à propagação das restrições após cada passo da heurística.

A figura 4.3 mostra o resultado final das alocação do exemplo da figura 4.1 utilizando o procedimento de alocação no instante de início mais cedo.

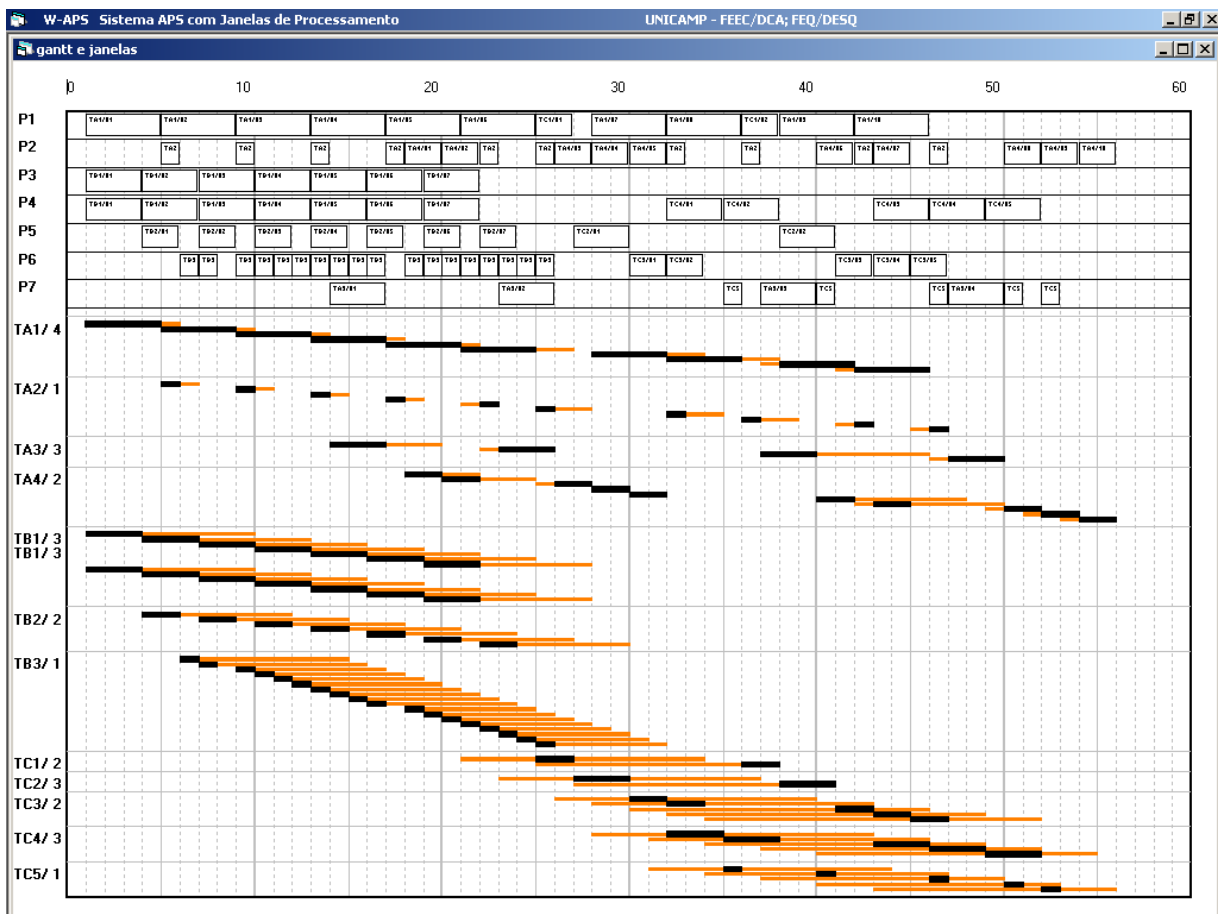


Figura 4.3: Carta de Gantt, e janelas de processamento resultante das alocações do exemplo da figura 4.1 utilizando o procedimento de alocação no instante de início mais cedo

É possível perceber pela figura 4.3 que todos os lotes foram alocados dentro de suas janelas de processamento. Isto já garante que todos os produtos finais cumpriram as datas de entrega.

A heurística proposta está descrita no fluxograma abaixo.

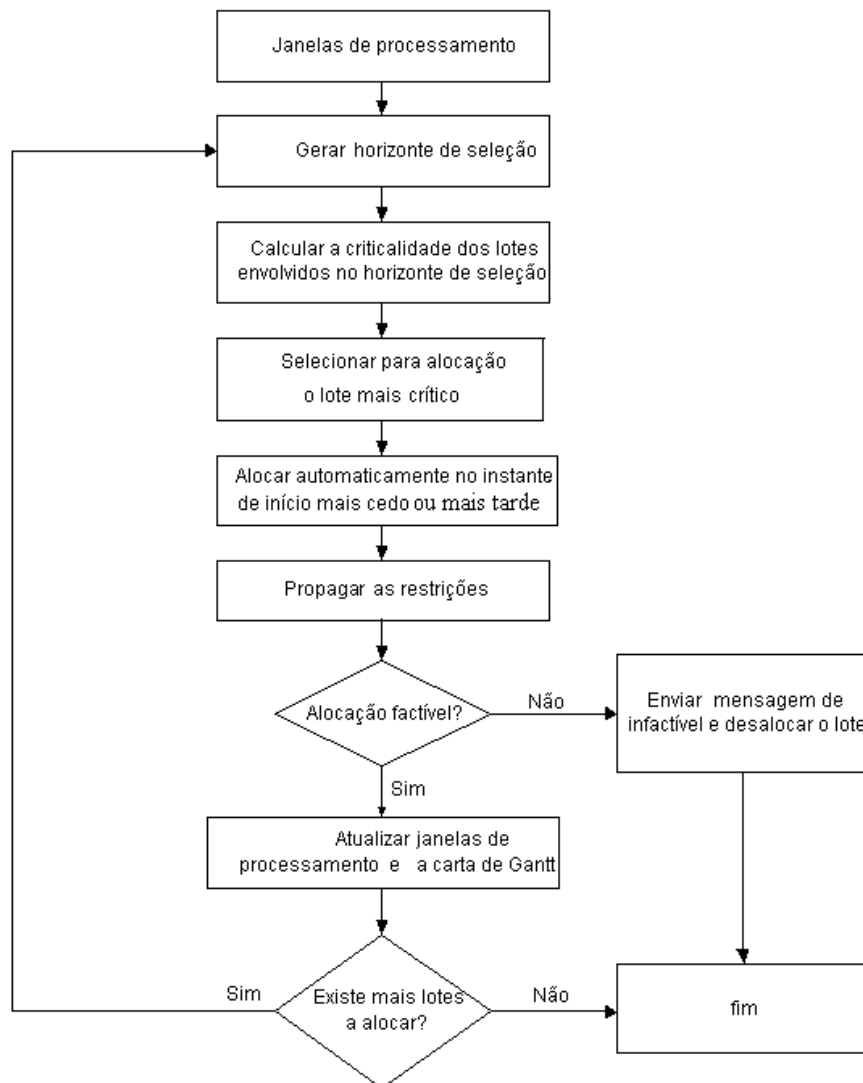


Figura 4.4: Fluxograma da heurística proposta.

4.3 Heurísticas orientadas pelo gargalo

As heurísticas orientadas pelos gargalos nos sistemas *APS* têm como principais características uma determinação estática de gargalos e a ausência de ferramentas para a alocação dos lotes no gargalo, frequentemente deixada por conta do usuário. Assim por exemplo a heurística descrita na seção 2.2.3 supõe uma alocação no equipamento gargalo seguida da alocação “*upstream*” o mais tarde e da alocação “*downstream*” o mais cedo. Um dos grandes problemas desta heurística é o fato de ser o usuário que determina o processador gargalo, e a

heurística aloca todos os lotes baseando-se neste processador gargalo.

É bem conhecido que a alocação de lotes em um intervalo gargalo tem grandes conseqüências sobre a solução final. Os sistemas de busca orientada pelas restrições determinam o maior gargalo e procedem a alocação de um lote ou o ordenamento de dois lotes, mas dentro de um procedimento de busca em árvore que permite o retrocesso (“*backtracking*”) quando a solução não é julgada adequada. Porém um mecanismo do tipo busca em árvore não é imaginável dentro de um sistema *APS* dada a exigência de rapidez. A ausência desta característica limita muito a possibilidade de boas soluções utilizando as heurísticas normalmente encontradas nos sistemas *APS*.

A proposta descrita a seguir utiliza as janelas de processamento, as curvas de carregamento e a sua atualização através dos mecanismos de propagação de restrições junto com uma abordagem diferente de escolha do lote a alocar. Por um lado trata-se de uma heurística sequencial em que cada passo é alocado um lote e a situação é novamente avaliada em função das novas curvas de carregamento. Esta nova avaliação possibilita a utilização de um gargalo dinâmico, ou seja o gargalo pode mudar de um processador para outro em função da última alocação. Entretanto as grandes conseqüências de uma decisão de alocação no gargalo são eliminadas porque as decisões são tomadas sobre os lotes atribuídos ao equipamento que contém o gargalo, mas com pequena influência sobre o gargalo.

A idéia básica é a de que a solução do gargalo sempre será difícil pelos grandes impactos que origina. O que parece ser mais razoável é delimitar o gargalo alocando os lotes com pequena contribuição sobre ele. Desta forma o problema no gargalo torna-se mais simples no mínimo pela sua dimensão dado que o número de lotes envolvidos será menor. Por outro lado a importância do gargalo será igual ou menor como resultado destas alocações. Eventualmente a solução do gargalo final vai requerer estratégias sofisticadas mas para um problema de dimensão menor.

A determinação dinâmica dos gargalos é feita através da análise das curvas de carregamento (demanda agregada) apresentada por Sadeh (Sadeh, 1991), e dos tempos de folga apresentado por Smith (Smith e Cheng, 1993). Utiliza-se um valor de demanda agregada mínimo (*dagmin*) para determinar os intervalos de tempo em processadores em que a demanda agregada excede este valor. Para cada intervalo determinam-se os lotes envolvidos com o

gargalo e o tempo de folga do conjunto de lotes. Seleciona-se a seguir o processador e o intervalo com menor tempo de folga. Os lotes candidatos a serem alocados são lotes atribuídos a este equipamento e não estão envolvidos com nenhum intervalo gargalo. Dentre estes lotes escolhe-se para alocação aquele que apresente maior criticalidade. A alocação na sua janela de processamento é feita no instante de tempo mais longe possível do intervalo gargalo.

A figura 4.5 mostra o resultado desta heurística num exemplo emblemático, utilizando um valor de demanda agregada mínimo de $dagmin = 0,8$. Na esquerda tem-se o carregamento inicial dos equipamentos, na direita o carregamento final resultado da heurística, onde os retângulos de carregamento com altura unitária correspondem aos intervalos de tempo onde foram feitas as alocações. De um total de 54 lotes correspondendo a 27 operações, a heurística alocou 39 lotes fora dos intervalos gargalo. Neste exemplo a situação remanescente é de fácil solução, o que obviamente não pode ser garantido que aconteça. É porém um heurística rápida que permite determinar se existe uma solução fácil para os gargalos.

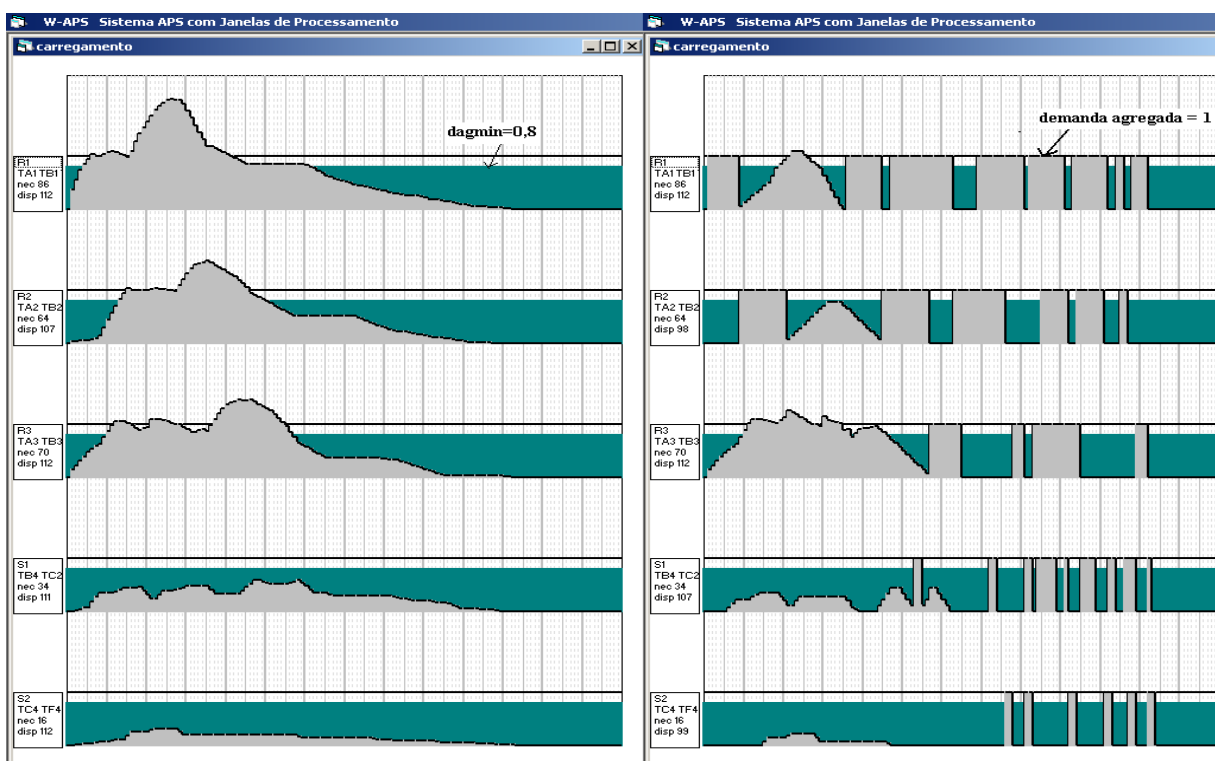


Figura 4.5: Curvas de carregamento iniciais (esquerda) e finais.

Esta heurística se adaptará bem para situações de carregamento médio ou alto. Na situação oposta, de baixo carregamento, a heurística terá a tendência de alocar os lotes em

instantes muito distantes dos gargalos o que pode levar a uma sub-utilização dos processadores.

A heurística proposta está descrita no fluxograma abaixo.

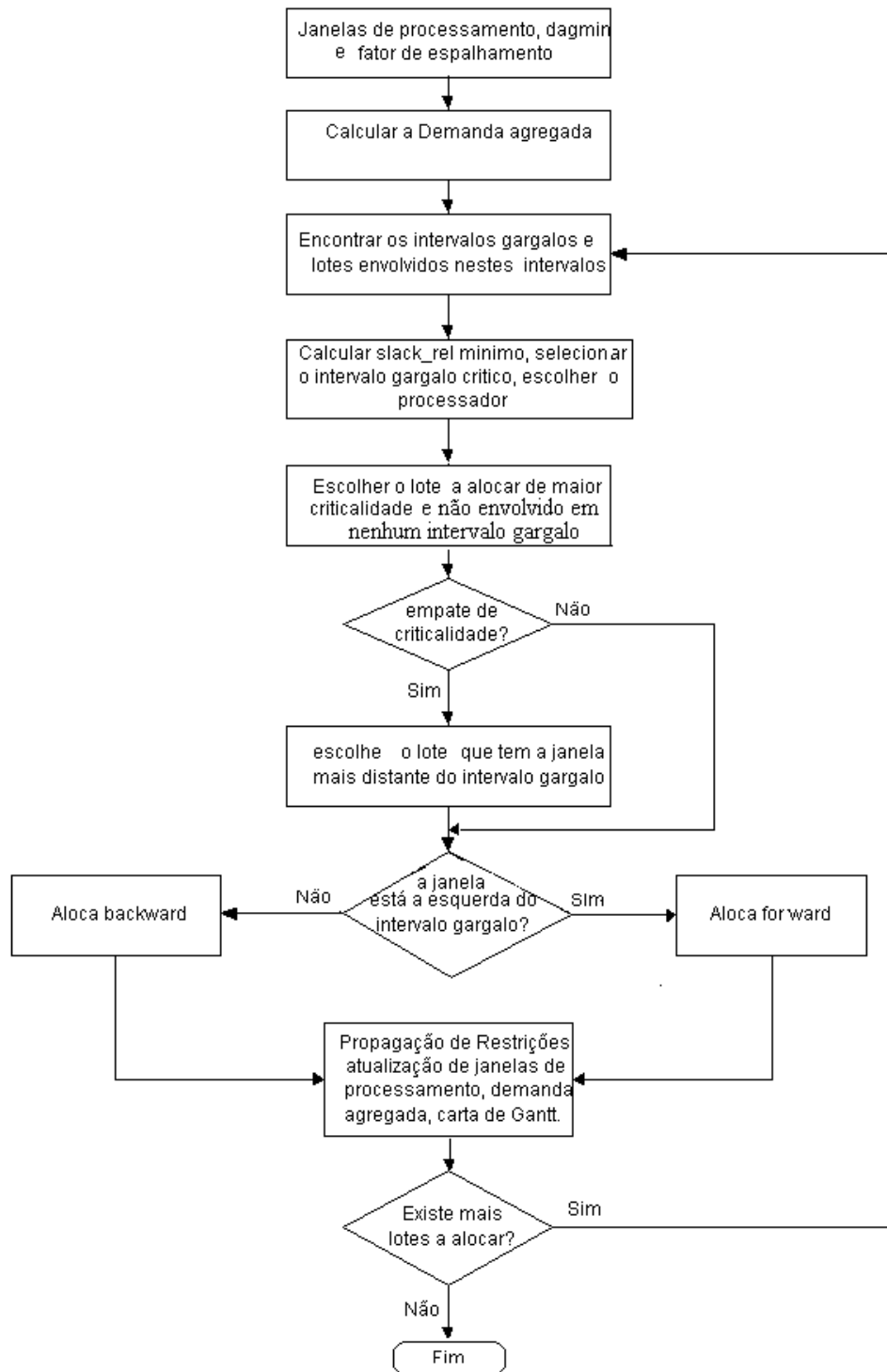


Figura 4.6: Fluxograma da heurística proposta.

4.4 Heurísticas de modificação interativa das janelas de processamento com busca limitada

Esta heurística foi desenvolvida para situações de baixo carregamento. Considere por exemplo a situação representada na figura 4.7. As curvas de carregamento e os tempos de folga totais configuram uma situação de baixo carregamento em que existirão inúmeras soluções ao problema de programação dos lotes. Em uma situação deste tipo é bastante possível que o usuário acrescente condições adicionais ao problema de programação, por exemplo a separação na planta da fabricação dos diferentes produtos, a escolha de determinados períodos de tempo para a fabricação de certos produtos, etc. Em suma o usuário desejará restringir as soluções possíveis no tempo. Este efeito pode ser conseguido através de uma manipulação interativa das janelas de processamento dos lotes, aumentando os *Est* ou reduzindo os *Lft*. A figura 4.8 mostra um exemplo desta manipulação e suas janelas iniciais, onde o usuário tomou a decisão de produzir primeiramente o produto *ProdC* e depois o produto *ProdA* junto com o produto *ProdB* dado que ambos não compartilham os mesmos processadores.

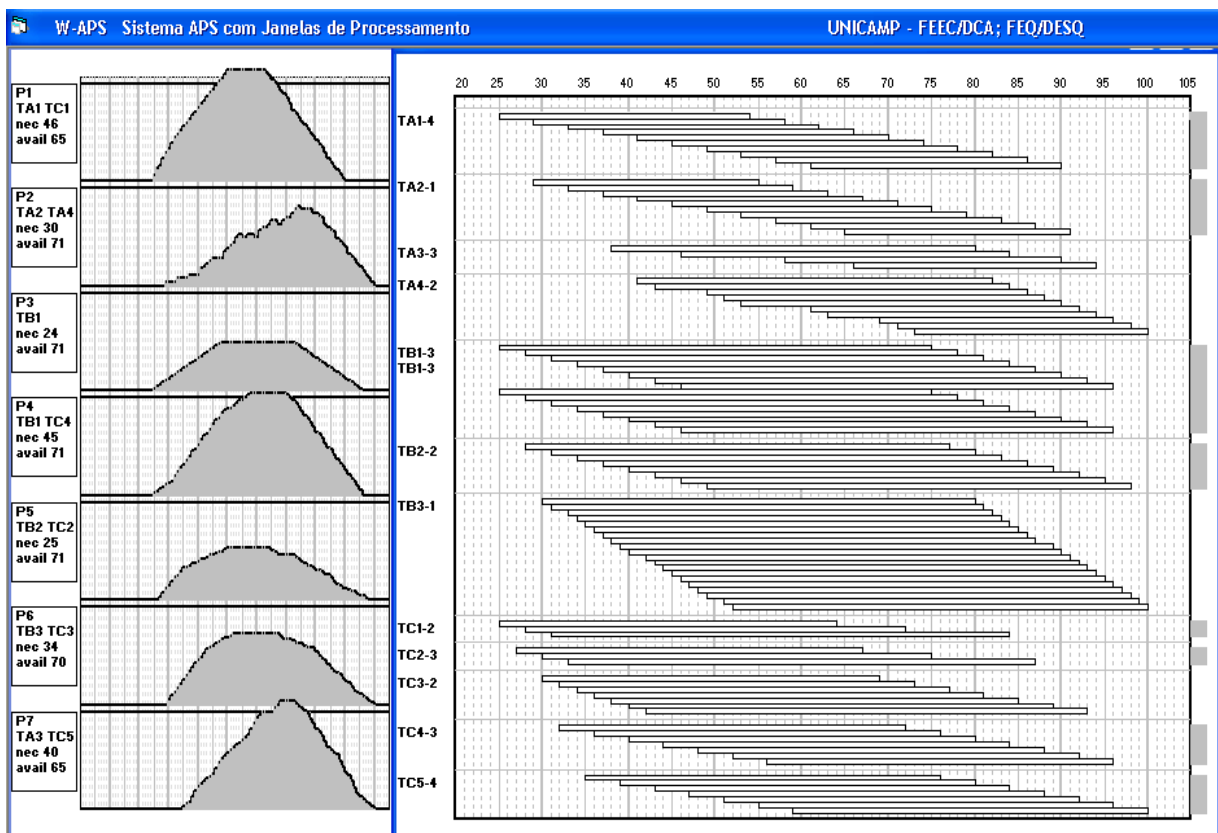


Figura 4.7: Janelas de processamento e curvas de carregamento.

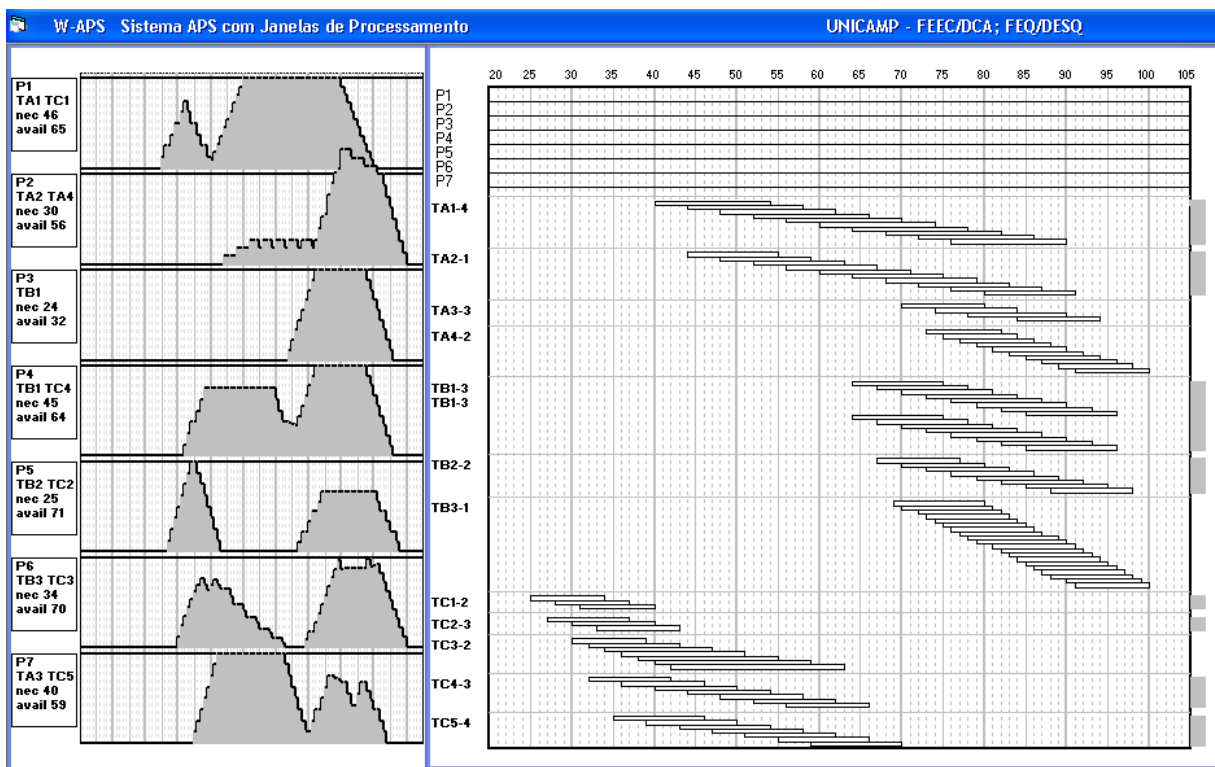


Figura 4.8: Resultado dos recortes das janelas de processamento do problema proposto, após aplicação da heurística de recorte de janelas.

Qualquer heurística que seja utilizada com estas janelas levará a cartas de Gantt em que a separação pretendida será conseguida. A heurística proposta consiste em modificações das janelas dos lotes por parte do usuário, sendo que após cada modificação é ativado o mecanismo de propagação de restrições para atualizar as janelas. Dado que a propagação de restrições que segue cada modificação pode evidenciar situações infactíveis, ou seja, situações onde o tempo de processamento do lote seja maior que o tamanho de sua janela, estas modificações são implementadas em uma estrutura de busca em árvore com “*backtracking*” interativo por parte do usuário.

Capítulo 5

Conclusões

O objetivo deste trabalho foi o de desenvolver heurísticas de alocação em ambientes *APS* utilizando as janelas de tempo de processamento dos lotes para problemas de programação da produção com datas de entrega para os produtos finais. As razões foram basicamente duas: a grande utilização na indústria dos sistemas *APS* dado o seu preço e rapidez na geração de cenários para a programação da produção e a dificuldade geral destes sistemas para lidar com problemas com datas de entrega em situações de carregamento alto dos equipamentos.

Disponha-se do sistema de planejamento *PCPIP* desenvolvido pela equipe de pesquisa em que foi desenvolvido este trabalho assim como da grande experiência desta equipe em técnicas de otimização para a programação em problemas com datas de entrega.

Os resultados principais e contribuições deste trabalho foram:

- a análise detalhada do desempenho de um sistema *APS* comercial representativo e a implementação de um emulador. Isto exigiu o desenvolvimento de ferramentas de alocação, como a manutenção do estado de ocupação dos processadores, úteis para qualquer heurística de alocação que venha a ser analisada no futuro;
- a proposta, implementação e teste de heurísticas típicas de alocação mais cedo ou mais tarde com características importantes, basicamente a ausência de esquema de prioridades e um mecanismo de relógio com menor impacto na solução final. A primeira característica é importante para evitar um problema básico nas heurísticas equivalentes nos sistemas *APS* qual seja a grande variação na qualidade da carta de Gantt obtida em função do esquema de prioridades induzido pelo usuário . A eliminação de esquema de prioridades

foi possível pela utilização das janelas de processamento de lotes e pela estimação da carga dos processadores, conceito este utilizado nas técnicas de programação da produção orientadas pelas restrições;

- Estas heurísticas estão baseadas na utilização das janelas de tempo geradas pelo sistema PCPIP e atualizadas após cada decisão de programação através dos mecanismos de propagação de restrições. Quando para o problema for importante cumprir com as datas de entrega, as janelas de processamento auxiliam muito este objetivo;
- a proposta, implementação e testes de uma heurística orientada pelos gargalos com uma abordagem nova. Esta heurística aloca preferencialmente os lotes menos envolvidos nos gargalos diminuindo a sua importância. Isto tem levado a soluções boas para problemas com carregamento alto resolvendo totalmente o problema ou reduzindo a dimensão do problema, que seria atacado a seguir por técnicas de otimização. A heurística proposta utiliza exclusivamente a informação de carregamento dos processadores obtida a partir das janelas de processamento e de uma forma dinâmica; isto possibilita focalizar a evolução dos gargalos induzidos pelas sucessivas alocações. Em contraste as heurísticas orientadas pelos gargalos em sistemas *APS* comerciais supõem uma solução do gargalo utilizando critérios estáticos avaliados sobre a situação inicial e a solução do problema remanescente utilizando esquemas de prioridades.

Como trabalhos futuros cabe citar:

- aprimoramento das heurísticas desenvolvidas tanto nos seus aspectos metodológicos quanto no aspecto de interação com o usuário para a sugestão de ajustes nos parâmetros das heurísticas;
- complementação das heurísticas desenvolvidas quando levam a uma situação infactível com heurísticas relaxando as restrições impostas pelas janelas;
- inclusão de medidas de desempenho para caracterizar e comparar as soluções;
- análise de outros sistemas *APS* comerciais.

Referências Bibliográficas

- Alvarenga, W. L. (2001). *Estudo da estratégia de busca orientada por restrições em plantas multipropósito operando em batelada*, Tese de mestrado, Faculdade de Engenharia Química, Universidade Estadual de Campinas, Campinas, São Paulo, Brasil.
- Bel, G., Bensana, E., Dubois, D., Erschler, J. e Esquirol, P. (1989). *A knowledge-based approach to industrial job-shop scheduling*, in *Knowledge-Based Systems in Manufacturing*, Ed. A. Kusiak. Taylor and Francis.
- Caseu, Y. e Laburthe, F. (1995). Improving branch and bound for job shop scheduling with constraint propagation, *Proceedings of 8th Franco-Japanese 4th Franco-Chinese Conference*.
- Cla (1999). *Reference Manual*, Claire Schedule 1.0, Mountain View, EUA. <http://www.ens.fr/laburthe>.
- Egli, U. M. e Rippin, D. T. (1986). Short-term scheduling for multiproduct batch chemical plants, *Computers and Chemical Engineering* **10**(4): 303–325.
- Gimeno, L. e Rodrigues, M. T. M. (2002). Planejamento e programação da produção: Abordagens de otimização para indústria de processos, *Technical report*, FEEC, FEQ, UNICAMP, Campinas, São Paulo, Brasil. notas de aula.
- Gimeno, L., Rodrigues, M. T. M. e Rodrigues, L. (2000). Constraint propagation tools in multipurpose batch plants short term planning, *Proceedings of 2nd Conference on Management and Control and Production and Logistics*, Grenoble, France. em CD-ROM.
- i2 (1996). *Rhythm Tutorial Manual*, i2 Technologies Inc, Irving, Texas, EUA.

ILO (1997). *Scheduler 4.0 User's Manual*, ILOG, Mountain View, EUA.

Keng, N. P., Yun, D. Y. Y. e Rossi, M. (1988). Interaction sensitive planning system for job shop scheduling, *Expert systems and intelligent manufacturing*, Elsevier pp. 57–59.

Keng, N. P., Yun, D. Y. Y. e Rossi, M. (1998). Finite capacity scheduling methods, *APICS - The Performance Advantage*.

Kondili, E., Pantelides, C. C. e Sargent, R. W. H. (1993). A general algorithm for short term scheduling of batch operations - i. milp formulation, *Computers and Chemical Engineering* **17**(2): 211–227.

Lazaro, M., España, A. e Puigjaner, L. (1989). A comprehensive approach to production planning in multipurpose batch plants, *Computers and Chemical Engineering* **13**.

Mockus, L. e Reklaitis, G. V. (1999). Continuous time representation approach to batch and continuous process scheduling.1. minlp formulation, *Industrial Engineering Chemistry Research* **38**: 197–203.

Montesco, R. A. E. (2003). *Heurísticas para sistemas aps utilizando janelas de processamento: Propostas, implementação e exemplos*, Tese de mestrado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, São Paulo, Brasil.

Papageorgiou, L. G. e Pantelides, C. C. (1993). A hierarchical approach for campaign planning of multipurpose batch plants, *Computers and Chemical Engineering* **17** (supplement): 27–32.

Papageorgiou, L. G. e Pantelides, C. C. (1996). Optimal campaign planning/ scheduling of multipurpose batch/semicontinuous plants. 2. a mathematical decomposition approach, *Industrial Engineering and Chemical Research* **35**: 510–529.

Rodrigues, L. C. A. (2000). *Planejamento e programação da produção em plantas multipropósito operando em batelada: Proposta de uma estratégia de decomposição usando janelas de tempo*, Tese de doutorado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, São Paulo, Brasil.

- Rodrigues, L. C. A., Graells, M., Canton, J., Gimeno, L., Rodrigues, M. T. M., Espuña, A. e Puigjaner, L. (2000). Utilization of processing time windows to enhance planning and scheduling in short term multipurpose batch plants, *Computers and Chemical Engineering* **24**: 353–359.
- Rodrigues, M. T. M., Gimeno, L., Passos, C. A. S. e Campos, M. D. (1996). Reactive scheduling approach for multipurpose batch plants, *Computers and Chemical Engineering* **20 (supplement)**: S1215–S1220.
- Rodrigues, M. T. M., Latre, L. G., Pessoa, M. A. O. e Montesco, R. A. E. (2004). Scheduling heuristics based on tasks time windows for aps systems, *European Symposium on Computer Aided Process/Engineering - ESCAPE 14*, Lisboa , Portugal.
- Rodrigues, M. T. M., Latre, L. G. e Rodrigues, L. C. A. (2000a). Production planning using time windows for short term multipurpose batch plants scheduling problems, *Industrial Engineering Chemistry Research* **39**: 3823–3834.
- Rodrigues, M. T. M., Latre, L. G. e Rodrigues, L. C. A. (2000b). Short-term planning and scheduling in multipurpose batch chemical plants: A multi-level approach, *Computers and Chemical Engineering* **24**: 2247–2258.
- Sadeh, N. (1991). *Look-ahead techniques for micro-opportunistic job shop scheduling*, Tese de doutorado, School of Computer Science, Carnegie Mellon University, Pittsburg, Pennsylvania, USA.
- SAP (1998). *SAP Advanced Planner & Optimizer. Production Planning and Detailed Scheduling*. www.sap.com.
- Smith, S. F. e Cheng, C. C. (1993). Slack-based heuristics for constraint satisfaction scheduling. proc, *Proceedings of 11th National Conf. on Artificial Intelligence*, Washington DC, EUA, pp. 139–144.