

UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA ELÉTRICA

DEZEMBRO DE 1989

PROGRAMAÇÃO DE PRODUÇÃO EM LOTES EM CÉLULA DE
MANUFATURA BASEADA EM TECNOLOGIA DE GRUPO

Por : Eng. Antonio Freitas Rentes

*Este trabalho corresponde à pesquisa final de seu doutorado
por Antonio Freitas Rentes e aprovado pelo Conselho
Julgador em 07/12/1989*

Orientador : Prof. Dr. Fernando A. C. Gomide

Fernando A. C. Gomide

Tese apresentada à Faculdade de Engenharia Elétrica, da Universidade Estadual de
Campinas, como requisito parcial para a obtenção do título de Mestre em Engenharia
Elétrica.

Ao meu filho Victor

À minha esposa Patrícia

Aos meus pais, Antonio (em memória) e Isaura

AGRADECIMENTOS

Ao professor Fernando Antonio Campos Gomide pela segura orientação, pela dedicação e pela confiança oferecidas à mim e a este trabalho.

Aos amigos Mitur Sakoda Jr. e Francisco S. Sawada pelo constante apoio e pelo desenvolvimento das interfaces do sistema heurístico.

Aos amigos Evelin M. A. Teixeira e Carlos A. Fernandes pelo suporte matemático, computacional e pelo estímulo oferecido a este trabalho.

Aos amigos Dirceu T. Carvalho e Newton Giachetta Filho, da Equipamentos Clark Ltda, pela participação na formulação do problema e pelo suporte técnico oferecido.

Aos professores Plínio Stange e Evandro O. Araújo pelas proveitosas discussões sobre o problema aqui tratado.

Ao Centro Tecnológico para a Informática (CTI) pelo apoio institucional.

À Equipamentos Clark Ltda pela receptividade, interesse pelo problema e pela oportunidade de aplicação do meu conhecimento.

À Regina Berretta pelo eficiente trabalho de diagramação.

E, especialmente, à minha esposa Patrícia pela compreensão e estímulo.

RESUMO

Este trabalho objetiva o estudo do problema de programação da produção voltado para célula de manufatura baseada em Tecnologia de Grupo, com ênfase no problema de sequenciamento da produção.

É proposta uma abordagem heurística visando amenizar o efeito da explosão combinatorial, típica desta classe de problema. Com isso, foi obtido um procedimento de sequenciamento simples o suficiente para uma rápida execução em microcomputadores, e com desempenho adequado para aplicações práticas.

Um modelo matemático é também desenvolvido, com o objetivo de validar o desempenho computacional e a qualidade dos resultados, através de estudos comparativos com o método heurístico.

Finalmente, são apresentados os resultados computacionais obtidos, as conclusões e as possíveis extensões para a continuidade deste trabalho.

ABSTRACT

This work considers production scheduling problems for manufacturing cells based on Group Technology, with emphasis in job sequencing task.

A heuristic approach is proposed aiming at a procedure that is simple enough for a microcomputer implementation, but sufficiently reliable in providing good solutions for practical applications.

A mathematical model is also developed in order to verify both, computational performance and solutions quality. The verification is made by comparing the solutions provided by the mathematical with the results of the heuristic approach.

Finally the conclusions are drawn and future work addressed.

Índice

Lista de Símbolos	0 - 1
TERMINOLOGIA USADA NO MODELO HEURÍSTICO0 - 2
TERMINOLOGIA USADA NO MODELO MATEMÁTICO0 - 4
Capítulo 1 - Apresentação	1 - 1
INTRODUÇÃO1 - 2
MOTIVAÇÃO1 - 3
ESCOPO DO TRABALHO1 - 5
ORGANIZAÇÃO DA TESE1 - 6
Capítulo 2 - Células de Manufatura	2 - 1
INTRODUÇÃO2 - 2
CÉLULAS DE FABRICAÇÃO2 - 3
Ciclo de fabricação2 - 3
Tecnologia de Grupo2 - 5
Sistema de fabricação por Tecnologia de Grupo2 - 6
A Célula baseada em Tecnologia de Grupo2 - 7
O SEQUENCIAMENTO DA PRODUÇÃO2 - 9

Geração de Requisições2 - 9
Complexidade de Processamento2 - 10
Critérios de Avaliação2 - 10
EXEMPLO DE UMA CÉLULA DE FABRICAÇÃO : A CÉLULA PILOTO2 - 14
FORMULAÇÃO DO PROBLEMA DE SEQUENCIAMENTO2 - 17
RESUMO2 - 20

Capítulo 3 - Modelagem do Problema de Sequenciamento . . . 3 - 1

INTRODUÇÃO3 - 2
ENTRADAS DO SISTEMA3 - 3
PARÂMETROS DO SISTEMA CÉLULA3 - 3
Configuração da célula3 - 3
Volumes mínimos e máximos de estoque em cada nível.3 - 4
Velocidade de produção3 - 5
Matrizes de tempos de preparação3 - 5
INFORMAÇÕES DE ESTADO DA CÉLULA3 - 5
Montagem inicial das linhas3 - 6
Volume atual dos estoques3 - 6
INFORMAÇÕES DE DEMANDA3 - 6
MODELAGEM HEURÍSTICA3 - 7
PROCESSAMENTO DE INFORMAÇÕES3 - 7
Conversão de demanda3 - 7

Cálculo do Tempo de Processamento de Cada Lote	3 - 8
Datas Mais Ceddo e Datas Mais Tarde de produção	3 - 8
Data mais tarde de fabricação	3 - 8
Data mais cedo de produção	3 - 10
Data mais cedo primária - DMCp	3 - 10
Data mais cedo secundária - DMCs	3 - 10
SEQUENCIAMENTO DA PRODUÇÃO USANDO O ALGORÍTMO	3 - 11
O algoritmo de sequenciamento	3 - 11
Comentários sôbre o algoritmo.	3 - 13
MODELAGEM MATEMÁTICA	3 - 16
TERMINOLOGIA DO MODELO MATEMÁTICO	3 - 16
FUNÇÃO OBJETIVO	3 - 17
RESTRIÇÕES DO MODELO MATEMÁTICO	3 - 18
Restrição classe I - Garantia do Prazo de Entrega	3 - 18
Restrição classe II - Relações de precedência	3 - 19
Restrição classe III - Controle de estoques	3 - 19
Restrição classe IV - Integridade de $Betal[i,j]$	3 - 21
Restrição classe V - Integridade das variáveis $Alfab[i,j]$ e $Alfac[i,j]$	3 - 22
Restrição classe VI - Integridade da variável $Gamal[i,j]$	3 - 22
Restrição classe VII - Integridade da variável $Deltal[i,j]$	3 - 23
Restrição classe VIII - Integridade da variável $Deltal[i,j]$ (II)	3 - 24
Restrição classe IX - Integridade de $Deltal[i,j]$ (III)	3 - 24
Restrição classe X - Integridade de $Deltal[i,j]$ (IV)	3 - 25

RESUMO	3 - 26
Capítulo 4 - Implementação dos Modelos e Resultados	4 - 1
INTRODUÇÃO	4 - 2
IMPLEMENTAÇÃO DO MODELO HEURÍSTICO	4 - 3
IMPLEMENTAÇÃO DO MODELO MATEMÁTICO	4 - 5
Geração do Modelo Matemático	4 - 5
Resolução do Problema Matemático	4 - 5
EXEMPLOS DE UTILIZAÇÃO DAS A BORD. HEURÍSTICA E MATEMÁTICA.	4 - 7
Descrição dos Parâmetros da Célula	4 - 7
Descrição do Estado Inicial	4 - 10
Demandas de Produção Consideradas	4 - 10
Resultados obtidos.	4 - 11
Exemplo 1	4 - 12
Exemplo 2.	4 - 14
Exemplo 3.	4 - 16
Exemplo 4.	4 - 18
Exemplo 5.	4 - 20
Exemplo 6	4 - 24
Exemplo 7.	4 - 28
ANÁLISE COMPARATIVA ENTRE OS MÉTODOS	4 - 30
Desempenho computacional	4 - 30

Análise dos resultados	4 - 30
RESUMO	4 - 32
Capítulo 5 - Conclusão	5 - 1
INTRODUÇÃO	5 - 2
AVALIAÇÃO DOS RESULTADOS GLOBAIS OBTIDOS	5 - 3
TRABALHOS A SEREM DESENVOLVIDOS NO FUTURO	5 - 4
Referências Bibliográficas	6 - 1
Bibliografia	7 - 1

0. Lista de Símbolos

0.1. TERMINOLOGIA USADA NO MODELO HEURÍSTICO

p_1, \dots, p_n : Tipos diferentes de peças processadas pela célula.

i, j, y : Lotes genéricos de peças a serem produzidos pela célula.

E_1, E_2, E_3 : Estágios ou linhas de fabricação pertencentes à célula.

I_1, I_2, I_3 : Estágios ou linhas de fabricação pertencentes à célula.

n_1, n_2 : Níveis de fabricação.

$ATR[I]$: Variável que indica o nível de fabricação ao qual pertence a linha l .

$ATRIB[n,p]$: Variável que indica a linha de fabricação pertencente ao nível n , por onde são processadas as peças de tipo p .

$EMI[n,p]$: Quantidade mínima de peças do tipo p a serem estocadas ao final do nível n .

$EMA[n,p]$: Quantidade máxima de peças do tipo p a serem estocadas ao final do nível n .

$PR[l,p]$: Quantidade de peças do tipo p passíveis de serem produzidas pela linha l por unidade de tempo.

$SETUP[n,p,p']$: Tempo de preparação necessário para uma troca de produção de peças do tipo p para p' no nível n .

$SETUP_INIC[l]$: Indica o tipo de peça para o qual está montada a linha l no instante inicial ($t=0$) do sequenciamento.

$ESTOQUE[n,p]$: Quantidade de peças do tipo p estocadas ao final do nível n no instante inicial do sequenciamento ($t=0$).

$XN[i]$: Quantidade de peças existentes no lote i a serem demandadas pela operação posterior à célula.

$PE[i]$: Prazo de entrega final do lote i no último processamento da célula.

$TF[l,i]$: Tempo de duração do processamento do lote i na linha l .

$DMCp[n,i]$: Data mais cedo primária - data mais cedo permitida para o início da produção do lote i no nível n , baseada no critério de máximo adiantamento da produção.

$DMCs[n,i]$: Data mais cedo secundária - data mais cedo permitida para o início da produção do lote i no nível n a partir de informações de sequenciamento do nível anterior.

DMC[n,i]: Data mais cedo de produção - valor máximo obtido entre $DMCp[n,i]$ e $DMCs[n,i]$.

SEQ: Vetor de armazenamento das posições de sequência obtidas pelo algoritmo de sequenciamento.

TSEQ[l,i]: Instante de término previsto para o processamento do lote i na linha l .

0.2. TERMINOLOGIA USADA NO MODELO MATEMÁTICO

a,b,c: Índices que representam as linhas E1,E2 e E3 respectivamente.

i,j,n: Lotes genéricos.

l: Linha genérica.

I: Quantidade de lotes existentes.

TIPO[i]: Tipo de peça que constitui um determinada lote i.

SETUP_l [TIPO[i],TIPO[j]]: Tempo de preparação necessário para a linha l na troca de produção de peças do TIPO[i] para a produção de peças do TIPO[j].

AT[TIPO[i],l] = 1: Indicação de que as peças TIPO[i] são processadas na linha l no segundo nível.

AT[TIPO[i],l] = 0: Caso isto não ocorra.

ISEQ_l [i]: Instante de início de processamento do lote i na linha l.

TF_l [i]: Tempo de processamento do lote i na linha l.

PE[i] : Prazo final de entrega do lote i no final da célula.

XN[i] : Quantidade de peças a serem produzidas no lote i.

Delta_l [i,j] = 1: Caso o lote i é processado imediatamente antes que o lote j na linha l.

Delta_l [i,j] = 0: Caso isto não ocorra.

Alfa_l [i,j] = 1: Caso o lote i é consumido antes que o lote j no estoque da linha l.

Alfa_l [i,j] = 0: Caso isto não ocorra.

Beta_l [i,j] = 1: Caso o lote i é produzido na linha l antes de se iniciar o consumo do lote j no estoque da linha l.

Beta_l [i,j] = 0: Caso isto não ocorra.

Gama_l [i,j] = 1: Caso o lote i está em processamento na linha l no instante de início de consumo do lote j no estoque da linha l.

Gama_l [i,j] = 0: Caso isto não ocorra.

G: Número tão grande quanto se queira.

1. Capítulo 1 - Apresentação

1.1. INTRODUÇÃO

Neste capítulo são descritos os principais elementos que motivaram o desenvolvimento deste trabalho. O tipo do problema a ser abordado bem como os objetivos finais a serem atingidos são explicitados.

Finalmente, é apresentada uma visão geral sobre a organização dos demais capítulos desta dissertação.

1.2. MOTIVAÇÃO

A realidade econômica do Brasil vem impondo às indústrias de manufatura, dentre outros, um custo pesado de manutenção de estoques. A administração de materiais (matérias primas, estoques intermediários, e estoques de produtos acabados) é, portanto, de fundamental importância pelo seu impacto nos resultados econômicos de curto e médio prazo nas empresas.

As indústrias vêm ao longo dos anos, adquirindo esta consciência e, conseqüentemente, vêm dispendendo esforços para reduzir estes volumes de estoque. Os estoques que se formam ao longo do sistema produtivo, por ter a maior parte de sua formação independente de elementos externos à empresa, é o estoque com maior potencial de redução, sendo o problema de fluxo de produção uma das chaves para o seu equacionamento.

À medida em que o ciclo de materiais (ou *lead-times* de compras e fabricação) aumenta, é comum a utilização do MRP (Material Requirement Planning) como instrumento de programação de produção. As várias versões de programas de computador que usam o conceito de MRP têm em comum a necessidade de definição, a priori, de valores de *lead-times* que comandarão o deslocamento das necessidades no tempo. Este valor leva em consideração não apenas os tempos de preparação (*setup*) e de operação, mas também, e principalmente, os tempos de fila. Na prática, o tempo de fabricação de um determinado componente é função da prioridade sobre outros na fila, até o limite do tempo real de máquina. Quanto maior a diferença entre estes dois valores para todas as peças fabricadas, maior o valor imobilizado de estoques intermediários potencialmente passíveis de serem desativados.

Apesar de ter-se revelado uma ferramenta poderosa no planejamento da produção, o MRP não é, sozinho, capaz de resolver o problema do fluxo de materiais. A organização do fluxo passa pela aplicação sistemática dos conceitos de tecnologia de grupo no dimensionamento de células de usinagem e comunização dos ferramentais por família de peças. Duas fortes restrições, entretanto, colocam limites à generalização deste procedimento:

1. O mercado exige produtos cada vez mais diversificados e em volumes cada vez menores.
2. O alto custo de aquisição de equipamentos modernos (cujos tempos de preparação sejam suficientemente baixos para permitir alta flexibilidade) em número suficiente para absorver variações de carga decorrentes das mudanças da composição (*mixing*) dos produtos.

O tratamento da primeira restrição passa pela necessidade de aumento de flexibilidade de produção, a qual se dá através da redução dos tempos improdutivos e de preparação. A solução de redução dos tempos de preparação via aquisição de equipamento moderno, por sua vez, esbarra na segunda restrição, que é o alto custo de equipamento.

Uma tentativa que vem sendo desenvolvida nas empresas para a resolução deste problema é a formação de células compostas baseadas em conceitos de tecnologia de grupo. Estas células são composições de linhas de produção que compartilham uma ou mais máquinas entre si, visando racionalizar a utilização de equipamento.

Desta forma, é possível manter o processo produtivo em estrutura celular com flexibilidade relativamente alta e ao mesmo tempo utilizar ao máximo os equipamentos já existentes, ou seja, obtem-se os benefícios de uma estrutura flexível sem incorrer no alto custo associado a esta opção. O problema que surge então, considerando que é desejável um nível baixo de inventário em processo, é o de coordenação de fluxo de produção entre estas linhas, devido às variações de velocidade de produção existentes entre os diversos tipos de peças e as flutuações de demanda verificadas ao longo do horizonte de programação.

Esta preocupação tem motivado o desenvolvimento de metodologias e ferramentas para o suporte às tomadas de decisão de supervisores ou programadores de produção, objetivando a geração de sequências de fabricação condizentes com a realidade física imposta pelos equipamentos e a realidade econômica, imposta pelo sistema produtivo.

1.3. ESCOPO DO TRABALHO

Este trabalho objetiva a consideração de uma classe de problemas de sequenciamento da produção em indústrias de manufatura de partes discretas, como por exemplo, transmissões mecânicas, eixos e engrenagens. O desenvolvimento deste trabalho se dá em três etapas principais.

Na primeira etapa, é desenvolvido um modelo heurístico que procura atender às necessidades de sequenciamento da produção dentro de um determinado ambiente de programação da produção. Isto significa que deve ser processado em equipamentos computacionais de pequeno porte, tipo computadores pessoais (PC's) e que tenha um desempenho computacional bom o suficiente para acompanhar o ritmo de execução das tarefas de programação.

A segunda etapa aborda uma modelagem matemática que reflete o mesmo problema de sequenciamento, viabilizando assim, uma comparação entre os dois modelos a nível de resultados gerados. A implementação deste modelo matemático foi efetuada em ambiente computacional de médio porte (VAX 11/785).

A terceira etapa considera a solução de vários problemas de sequenciamento, a partir da abordagem heurística e do modelo matemático. Uma análise dos resultados é feita de forma a validar e refinar tanto o método heurístico como o baseado no modelo matemático.

Para o desenvolvimento desta etapa, considerou-se como um estudo de caso, uma célula de fabricação de engrenagens para transmissões mecânicas da Equipamentos Clark Ltda.

1.4. ORGANIZAÇÃO DA TESE

Como colocado anteriormente, este trabalho aborda uma classe de problema de sequenciamento de produção, apresentando duas formas de resolução: heurística e matemática .

A primeira forma, a ser utilizada operacionalmente em uma empresa real, é baseada em algoritmo heurístico, desenvolvido a partir de diversas consultas aos elementos responsáveis pela programação de produção da empresa. Não se trata, no entanto, de um sistema baseado apenas no conhecimento destes técnicos. É na verdade uma heurística desenvolvida em consenso com estes elementos e em consonância com as realidades operacionais.

A segunda forma de resolução se dá através de métodos clássicos de programação matemática. Neste caso, o modelo é de programação inteira mista, e o método utilizado é o *Branch and Bound*.

O próximo capítulo apresenta uma pequena introdução à tecnologia de formação de células de manufatura, no que diz respeito ao raciocínio empregado para justificar a mudança do conceito clássico de produção para o conceito de fabricação em células. É também, apresentado um exemplo de célula de fabricação, que servirá como piloto para os estudos realizados.

A seguir, a formulação do problema é abordada, ressaltando as limitações associadas à sua resolução, bem como as premissas adotadas para viabilizar, na prática, a sua solução.

O terceiro capítulo apresenta os modelos heurístico e matemático desenvolvidos.

O capítulo quatro discute as implementações computacionais feitas para o problema. Nesta oportunidade descreve-se também o sistema **SEQMAN**, desenvolvido para ser utilizado no ambiente de produção, e a implementação feita para o modelo matemático. Uma análise comparativa entre os métodos é também incluída.

Finalmente, o capítulo cinco, apresenta as conclusões obtidas. A partir destas conclusões, proposições para futuras extensões e abordagens a este problema são estabelecidas.

2. Capítulo 2 - Células de Manufatura

2.1. INTRODUÇÃO

Este capítulo apresenta o conceito básico de formação de células de manufatura, sob o ponto de vista dos motivos que levaram ao desenvolvimento desta técnica de produção.

É efetuado um levantamento bibliográfico específico à questão do problema de sequenciamento de produção em ambientes convencionais e de células de Tecnologia de Grupo.

Finalmente, é descrita uma célula de manufatura piloto, a qual motivou o trabalho de sequenciamento propriamente dito, e a formulação do problema de sequenciamento, ressaltadas as premissas relevantes à prática.

2.2. CÉLULAS DE FABRICAÇÃO

2.2.1. Ciclo de fabricação

O ciclo de fabricação de uma peça é o tempo compreendido entre o início de industrialização até o momento em que ela se encontra totalmente pronta (Sério,[1988]).

Para que ela possa ser fabricada, a empresa retira do seu capital de giro uma parcela e investe nos insumos de produção : matéria prima, energia elétrica, ferramental, mão de obra, corpo técnico, processamento de dados, transporte, etc.

Para que a empresa possa obter sucesso é necessário trabalhar bem todos esses insumos. A redução do ciclo de fabricação é muito importante, pois garante a empresa a reincorporação do capital investido ao capital de giro.

A relação "menor ciclo, maior lucro" é direta e deve ser alvo constante de todos os elementos da empresa, desde de a abertura de um pedido de fabricação até o embarque do produto.

Para efeito de compreensão do processo, tomemos uma estação de trabalho em um centro produtivo qualquer. Nesta estação chegam peças que devem sofrer uma operação de fabricação, e saem peças ou direcionadas para uma segunda estação de trabalho ou prontas.

A uma estação de trabalho estão associados os tempos que irão compor o ciclo de fabricação da peça.

Para deixar a estação de trabalho apta a fabricar, gastamos um certo tempo na montagem do dispositivo, ajuste e calibração do ferramental, posicionamento da primeira peça, teste do processo de fabricação ou do programa C.N., etc. A soma desses tempos é conhecida como tempo de preparação, de montagem ou de *setup* (**ts**).

À manufatura propriamente dita, de cada peça, também está associado um tempo padrão (**tp**). Temos também um tempo de carga e descarga de peças (**td**).

Terminado o trabalho, na primeira estação, as peças são transportadas para a segunda, associando-se um tempo de transporte (**tt**).

No caso geral, nem o transporte é imediato e nem a segunda estação de trabalho está livre para receber as peças, como a primeira não estava. A programação vincula a cada estação de trabalho uma ordem de prioridade, o que obriga algumas ordens de fabricação a esperarem sua vez. Portanto a cada estação de trabalho também está associado um tempo de estocagem (**te**).

O tempo total (**TT**) que compõe a fabricação de uma peça em uma estação de trabalho elementar, é então :

$$TT = tp + td + ts + tt + te$$

No caso de fabricação em lotes, com lotes de L peças, temos :

$$TT = (ts + tt + tc) + L * (tp + td)$$

Se ao abrir uma ordem de fabricação, N estações de trabalho forem utilizadas para produzir as peças, o ciclo de fabricação destas será :

$$CF = \sum_{i=1..N} L * (tp_i + td_i) + ts_i + tt_i + te_i$$

ou seja, a soma de todos os tempos gastos nas estações de trabalho elementares.

Agora, a empresa possui em fabricação uma certa quantidade M de peças e tem um parque de N estações de trabalho aptas a produzi-las. Para esta conceituação não há necessidade de saber se as máquinas são de usinagem, corte, solda ou qualquer outra.

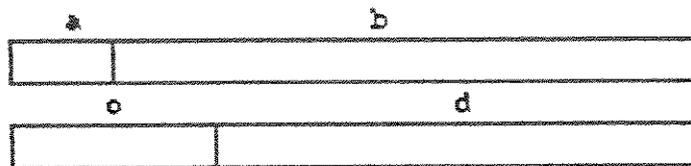
Se, em determinado instante, tirarmos uma fotografia da empresa, teremos todas as peças que estão em produção. Isto dá um ciclo de fabricação total de :

$$CFT = \sum_{i=1..N} \sum_{j=1..M} L_j * (tp_{ij} + td_{ij}) + ts_{ij} + tt_{ij} + te_{ij}$$

Esta fotografia é o inventário em processo. O capital de giro foi retirado da empresa e precisa ser rapidamente reincorporado. Portanto temos que reduzir os ciclos de fabricação com a consequente redução do inventário em processo.

O trabalho publicado por Merchant em 1977 (citação de Sérgio [1988]) mudou completamente a visão de trabalho nas empresas, sendo considerado como uma importante contribuição para a automatização da manufatura.

Industria de producao com usinagem



- a. Tempo de maquina 5%
- b. Movendo e esperando 95%
- c. 30% do tempo produtivo removendo material
- d. 70% do tempo produtivo para posicionamento, carregamento, medição, ociosidade, etc.

Figura 2.1. Figura de Merchant

Pesquisando as fábricas americanas (confirmado também na Europa) Merchant chegou à seguinte conclusão, representada pela figura 2.1.

A interpretação da figura 2.1 mostra que um ciclo de fabricação é composto por :

5 % de horas produtivas (tp + td)

95 % de horas improdutivas (ts + tt + te)

Completando a análise, 30 % do tempo produtivo era realmente de estação de trabalho operando, nos restantes 70 % o equipamento não estava operando e sobre a peça havia trabalhos tais como : controle de qualidade, medição, troca de ferramentas, etc.

A necessidade de atuar melhor nos 95 %, sem esquecer os outros 5 %, obrigou a consideração de melhores técnicas e equipamentos, tais como : Tecnologia de Grupo, *Layout* celular, máquinas CNC, CAD/CAM, etc.

Considerando o escopo deste trabalho, discorreremos somente sobre as duas primeiras técnicas indicadas, de forma a focalizar o contexto da utilização do método heurístico proposto.

2.2.2. Tecnologia de Grupo

Tecnologia de Grupo (TG) ou Produção por Família de Peças é um conceito que se aplica na produção de pequenos e médios lotes (Serio,[1988]).

Como 75 % das peças produzidas nas indústrias metalúrgicas são produzidas em lotes pequenos (aproximadamente 50 peças), nota-se a incidência de empresas que tem a difícil tarefa de planejar sua produção em pequenos e médios lotes, considerando a crescente diversificação de peças e produtos.

O conceito de Tecnologia de Grupo (TG) é baseado no agrupamento de peças em famílias, de acordo com certos atributos e então desenvolver o planejamento de todo um sistema produtivo para cada família. Cada família de peças pode então ser considerada como sendo uma peça composta pelos atributos de todas as peças do grupo.

Uma vez feito isto, utilizando-se técnicas de Engenharia de fabricação semelhantes àquelas usadas na produção em série, conseguiu-se benefícios significativos, tais como : redução dos ciclos de fabricação, redução de material em processo, confiabilidade nos prazos de entrega, simplificação do planejamento e controle da produção, racionalização do projeto e do processo.

Além disso, tem-se reconhecido que TG é um elemento essencial na fundação de uma estruturação de Fabricação Assistida por Computador.

2.2.3. Sistema de fabricação por Tecnologia de Grupo

A idéia básica original de fabricação de família em peças consiste em formar as famílias com características similares de fabricação, agrupadas para formar lotes e programadas num arranjo funcional..

Em desenvolvimentos posteriores, o lote de peças com características similares de fabricação são programadas e executadas em grupos de máquinas destinadas à fabricação específica de famílias de peças.

Os sistemas de fabricação TG podem estar relacionados às seguintes formas (Sério [1988]) :

TG Centro : Sistema de uma só máquina

TG Linha : Sistema composto por máquinas especiais

TG Célula : Sistema composto por grupos de máquinas

As três formas de arranjo localizam-se num campo ou espectro entre o arranjo funcional, onde as seções são organizadas por equipamentos com características semelhantes de fabricação, típico para pequenos lotes de fabricação, e o sistema de linha de fabricação para grandes lotes das indústrias com produção seriada.

Os sistemas acima diferenciam-se entre si principalmente pelo grau de similaridade das peças sobre o ponto de vista de fabricação.

O **TG-Centro** se desenvolve a partir do arranjo funcional, consistindo de um posto de trabalho, o qual, tanto do ponto de vista técnico como econômico, é organizado de modo que um espectro de peças semelhantes possam ser executados por um mesmo tipo de operação, por exemplo torneamento, dobramento, etc.

Este sistema se aplica a um espectro de peças, cujo grau de similaridade reside apenas em um tipo de operação, o qual pode ser executado num único posto de trabalho ou numa única máquina. As máquinas com controle numérico (CN), particularmente os centros de fabricação (por ex. centros de usinagem, centros de estampagem, etc.), ocupam um papel especial dentro deste sistema. A racionalização é aqui obtida pela formação de famílias de programas CN, através do desenvolvimento de instruções de programação limitadas às famílias obtidas.

Na **TG-Linha** os posto de trabalho para fabricação e/ou montagem de um espectro de peças são organizados de acordo com uma sequência fixa de operações. Este sistema representa o mais alto grau de racionalização. É a racionalização da produção em massa, adequada às famílias de peças com grande volume anual de vendas e pouca sazonalidade. É uma forma clássica de arranjo, sendo as suas vantagens e desvantagens bem conhecidas e já bem divulgadas (Sério,[1988]).

Em termos de escopo deste trabalho, vamos nos manter mais no desenvolvimento de conceitos do **TG-Célula**.

2.2.4. A Célula baseada em Tecnologia de Grupo

Na produção em massa, a demanda para um aumento na diversificação dos produtos e necessidade de diminuir horas inativas tem tornando as linhas de montagem anti-econômicas em alguns casos, de modo que pequenos grupos tem sido formados para executarem uma variedade de tarefas sob controle próprio.

O **TG-Célula**, consiste em organizar o sistema de fabricação em grupo de máquinas, de modo que todas as operações necessárias para a fabricação de uma família de peças são executadas normalmente dentro da área delimitada pelo grupo. Este sistema possibilita a execução da sequência de operação de modo mais flexível, constituindo um grau intermediário de racionalização entre o TG-Centro e o TG-Linha.

Implicitamente, este sistema se baseia também no princípio de trabalho em grupo, onde um pequeno número de pessoas trabalham juntas de forma coesa, conscientes de que formam um grupo e iteragem de modo a atingir um objetivo em comum.

Benefícios deste sistema :

Redução do ciclo de fabricação e material em processo.

Fluxo de material mais organizado.

Diminuição de ocorrência de atrasos de produção.

Controle de produção mais simples e eficaz.

Diminuição da utilização de equipamentos de transporte.

Controle de qualidade assegurado, feito de forma mais simples e eficaz.

Diminuição na quantidade de retrabalhos.

Melhoria do nível de motivação, devido aos aspectos sociológicos do trabalho em grupo com objetivos comuns.

Supervisão mais objetiva e racional, dando ao supervisor visão clara dos objetivos de produção.

Criação de facilidades infra-estruturais para implantação de tecnologias mais avançadas.

A título de exemplo a **figura 2.2** mostra um fluxo de peças em um arranjo funcional e a **figura 2.3** mostra o mesmo fluxo em estrutura celular.

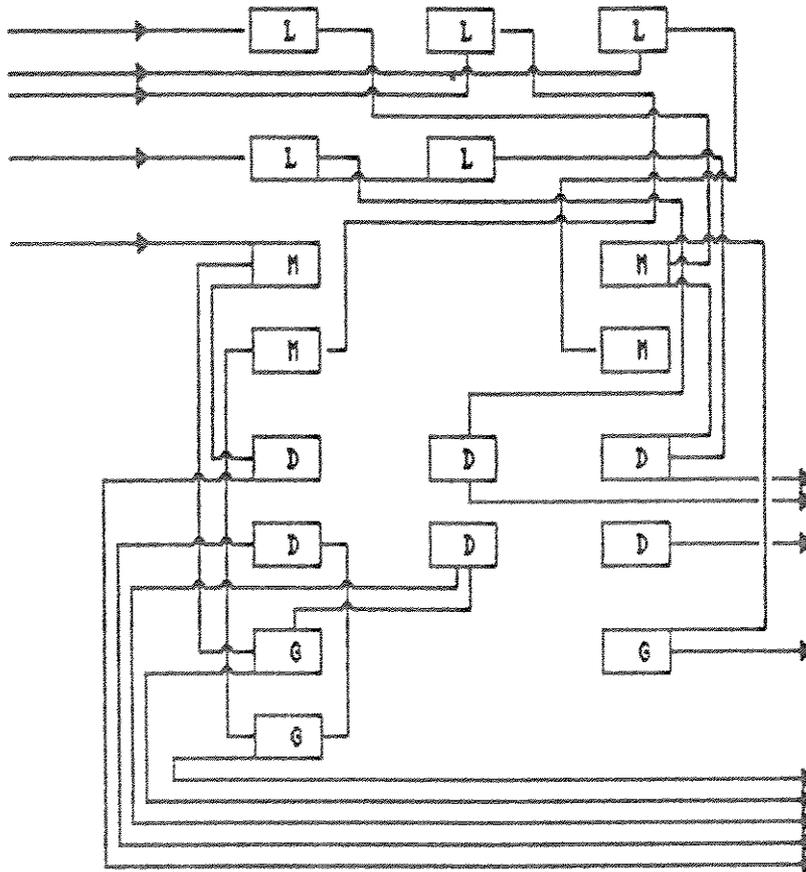


Figura 2.2. Fluxo de Peças em Arranjo Funcional

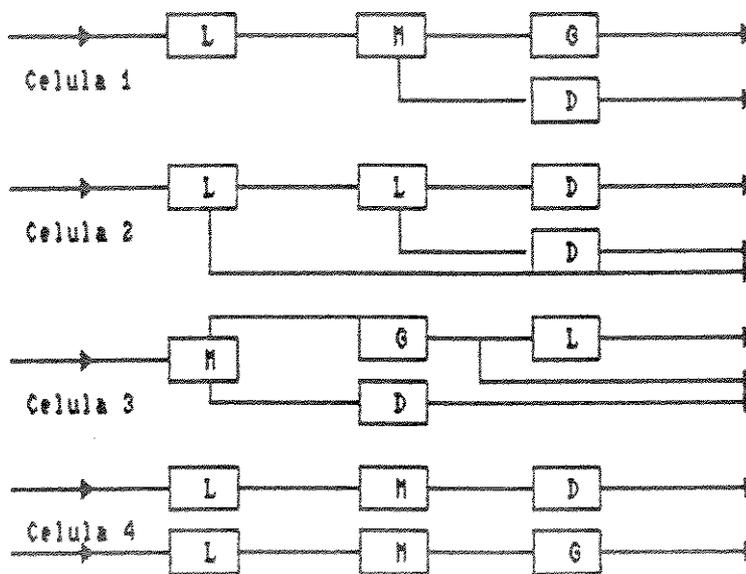


Figura 2.3. Fluxo de Peças em Arranjo IG

2.3. O SEQUENCIAMENTO DA PRODUÇÃO

O problema de sequenciamento tem sido amplamente estudado nos últimos 30 anos, constando na literatura uma farta documentação das diversas abordagens a nível de análise do problema, desenvolvimento de modelos matemáticos, heurísticos, simulações, dentre outros.

Alan S. Manne [1960], até onde pode ser verificado na literatura consultada, foi um dos primeiros a tratar o problema de sequenciamento da produção industrial de maneira formal, matematicamente, introduzindo a questão da programação linear booleana no tratamento de restrições dicotômicas, envolvendo precedência entre lotes ou peças. Neste artigo, o ambiente de produção é um job shop.

Hitomi [1979] apresenta o algoritmo de Johnson, para minimização de makespan em jobshop com duas máquinas, originariamente apresentado em 1954.

Um apanhado geral de alguns dos principais esforços dispendidos para tratar problemas de sequenciamento da produção, pode ser encontrado em artigo publicado por Graves [1981]. Neste artigo, a partir de uma visão clássica do problema, é proposta uma classificação que nos permite enquadrar as características mais gerais tanto no aspecto prático quanto no teórico dos problemas clássicos de sequenciamento da produção. As dimensões de classificação propostas são três :

1. Geração das Requisições
2. Complexidade de Processamento
3. Critérios de Avaliação

Estas classificações são desejáveis uma vez que permitem, além de uma formalização da nomenclatura empregada, o conhecimento de um determinado tipo de problema, através de suas particularidades, dentro do universo clássico dos problemas de sequenciamento.

2.3.1. Geração de Requisições

Dentro desta primeira dimensão, a requisição de demanda pode ser gerada diretamente através do usuário, por meio de ordens de serviço, ou indiretamente, visando reposição de consumo de estoques. Na literatura foi detectada esta distinção expressa em termos de *open shop* ou *closed shop*.

Em um *open shop*, todas as ordens de produção são geradas diretamente através de requisições do usuário, não gerando inventário em processo. No caso do *closed shop*, a programação da produção é feita não apenas em função das decisões de sequenciamento, mas também em função do dimensionamento dos lotes de produção associados ao processo de reposição dos inventários.

2.3.2. Complexidade de Processamento

Esta segunda dimensão é baseada no número de passos e no tipo de fluxo de processamento associados a cada tarefa de produção ou item. As classes mais comuns dentro desta dimensão são as seguintes :

Um estágio, um processador

Um estágio, processadores em paralelo

Multiestágios, flow shop

Multiestágios, job shop

O problema de um estágio com um processador é também conhecido como o problema *one-machine* e é o mais simples dentro desta dimensão. Aqui, todas as tarefas requerem apenas um passo de processamento que deve ser feito em uma única máquina ou processador.

O problema de um estágio com processadores paralelos é similar ao problema anterior, apenas diferindo no fato de existir um conjunto de máquinas que podem processar paralelamente diversas tarefas do problema.

O problema multiestagio é aquele no qual uma tarefa, ou item, requer uma sequência de operações em mais de uma máquina, existindo normalmente uma relação de precedência entre estas operações.

No problema *flow shop*, todas as tarefas são processadas em vários estágios em diversas máquinas, mantendo a mesma ordem de precedência entre os estágios.

No caso *job shop* não existem restrições quanto a variações desta ordem para as tarefas, ou seja, as tarefas a serem sequenciadas podem ter rotas tecnológicas diferentes umas das outras, podendo a tarefa ter, inclusive, rotas alternativas. É o caso mais geral e mais complexo de problemas dentro da classificação.

2.3.3. Critérios de Avaliação

Nesta dimensão é levada em conta a medida através da qual o sequenciamento é avaliado. Duas classes são sugeridas por Graves [1981] : *Schedule Costs* e *Schedule Performance*.

Em *Schedule Costs* os parâmetros de avaliação são os diversos custos envolvidos na tomada de decisões, tais como custos de preparação, custos de atrasos, custos de manutenção do inventário em processo, etc.

Já em *Schedule Performance* a avaliação pode ser efetuada de diversas maneiras. Algumas das mais comuns são : níveis de utilização dos recursos, percentual de tarefas atrasadas, média de

término das tarefas (*average tardiness*), instante de término da última tarefa (*maximum tardiness*), etc.

Convém salientar que a questão de complexidade de processamento é constantemente levantada ao longo de diversos artigos.

Norbis e Smith [1986] fazem uma análise de uma série de procedimentos de resolução ótima de problemas de sequenciamento utilizando programação inteira, inteira mista ou fluxo em redes. Eles colocam a questão do insucesso destas abordagens no tratamento de problemas reais devido à natureza não polinomial completa (*NP-Completeness*) destes problemas. Os autores citam para esta afirmativa um artigo de Karp de 1975 onde isto é demonstrado.

Yamamoto e Nof [1985] reforçam este fato, citando também como causa a enorme variação de fatores a serem considerados para cada caso específico no processo real de manufatura convencional.

A partir de meados da década passada, novos avanços significativos no sistema produtivo foram divulgados a nível de cenário internacional. A necessidade de diversificação da produção, associada a um menor ciclo de vida do produto, obrigou o planejamento da produção a ser efetuado de forma mais integrada. O conceito de utilização de Tecnologia de Grupo e de MRP tornou-se uma realidade (Burbidge, 1978 ; Hitomi e Ham, 1977).

Na concepção de Hitomi e Ham [1977a], o problema de programação da produção em ambiente de Tecnologia de Grupo (TG) difere do problema de programação convencional devido à contemplação dos seguintes fatores pela TG :

1. **Redução do tempo e custo de preparação**, através do agrupamento de peças semelhantes em células de manufatura.

2. **Decisão a nível de sequência de grupos e de tarefas (peças)**. A questão de sequência de grupo diz respeito a quando diversas famílias compartilham a mesma célula. Nestes casos, existe a necessidade de ordenação dos grupos existentes e da ordenação das tarefas individuais dentro de cada grupo, demandando um sequenciamento de dois estágios.

3. **Possibilidade de padrão flow shop**. O padrão *flow shop* é inerente ao problema de sequenciamento TG, permitindo a utilização de algoritmos mais simples, em relação ao *job shop*. Esta idéia é reforçada por Choobineh [1984].

4. **Necessidade de consideração de ordem econômica**, direta ou indiretamente, no que diz respeito à questão de formação de estoques intermediários.

Desta forma, os estudos de sistemas de sequenciamento da produção passaram a enfatizar estes conceitos, procurando abordar os problemas de programação da produção dentro deste novo contexto.

Hitomi e Ham [1977b] tratam matematicamente o problema de carregamento de máquinas em uma linha TG, apresentando uma formulação de programação inteira mista. Em outro artigo do

mesmo ano [1977a], os autores analisam a questão de programação da produção de produtos multiestágios em células baseadas em TG. Neste artigo a questão principal é a minimização de *makespan*, considerando o conceito de sequenciamento de grupos e peças dentro dos grupos, utilizando o método *Branch and Bound*.

Choobineh [1984] extrai as vantagens da coexistência entre TG e MRP a partir de um procedimento de planejamento hierárquico, propondo um modelo de programação linear multiperíodo para determinar programação de menor custo em uma célula de manufatura.

Mozier, Elvers e Kelly [1984] efetuam uma análise de regras para sequenciamento em *job shop* e *flow shop* (ambientes não TG) aplicadas a células de TG. O interessante no enfoque é que a célula analisada também não é uma "célula TG pura", mas linhas interligadas tais como a do modelo desenvolvido nesta dissertação. Os autores, no entanto, levaram em consideração o tempo de preparação associado ao tempo de processamento, observando prazos de entrega, mas não levaram em consideração a questão de volume dos estoques intermediários.

Ozden, Egbelu e Iyer [1985] apresentam uma resolução de problema de sequenciamento em uma célula TG de uma única máquina usando programação dinâmica, focalizando o problema na formação de um grupo ótimo e em uma sequência de produção que minimize os tempos de preparação total. O modelo aqui empregado considera inicialmente a divisão das diversas "peças" em grupos com processos similares e trata da minimização dos tempos de preparação na mudança de um grupo para outro e de uma peça a outra dentro de cada grupo. O enfoque de tempo de preparação dinâmico baseado em estrutura matricial atende perfeitamente à resolução do problema proposto nesta dissertação, mas o modelo de Ozden, Egbelu e Iyer [1985] não levam em consideração a questão de prazos de entrega e também a questão de estoques intermediários.

Pourbabai [1986] também reforça a idéia de programação de célula TG como um problema *flow shop*, estabelecendo uma série de regras ou premissas de programação que são transformadas em um modelo de programação linear inteira mista, levando em conta a questão de prazo de entrega em uma linha baseada em TG. A ocorrência de tempos de preparação não foi tratada, no modelo de Pourbabai, de forma matricial, sendo associados ao tempo de processamento do lote. O modelo, no entanto, levou em consideração a possibilidade de redução do tempo de preparação através da junção de dois ou mais lotes com prazos diferentes no sequenciamento, sendo a questão da determinação do tamanho do lote (*lotsizing*) abordada de forma a atender uma demanda previamente estabelecida.

A idéia de que a condição de otimalidade do sequenciamento é dependente da minimização dos tempos de preparação também está presente no artigo de Lee e Mirchandani [1986]. Isto ocorre quando os tempos de preparação são significativos, tal como ocorre no caso em estudo nesta dissertação. Lee e Mirchandani [1986] fazem uma distinção entre os tempos de preparação dos grupos e os das peças dentro dos grupos, elaborando duas heurísticas de sequenciamento considerando uma célula TG compostas de duas máquinas.

A nível de utilização para resolução do problema tratado nesta dissertação, algumas idéias levantadas na literatura se tornaram úteis, apesar da natureza do problema de sequenciamento, como já reforçado por Yamamoto e Nof [1985], impossibilitar generalizações devido a particularidades específicas associadas ao ambiente produtivo.

Teixeira e outros [1987] fazem uma extensão do modelo de programação linear inteira apresentado por Baker [1974], focalizando o problema de sequenciamento da produção em células flexíveis de manufatura, com as alocações das peças nas máquinas feitas por um robô. A questão das precedências entre os itens é tratada através de restrições dicotômicas, objetivando minimizar o tempo de espera das peças entre os processamentos. O modelo, no entanto, não leva em consideração prazos de entrega e tempos de preparação.

No que diz respeito ao desenvolvimento heurístico, Norbis e Smith [1986] apresentam uma heurística em dois níveis para sistemas tipo *job shop*, onde em um primeiro nível os trabalhos são priorizados e ordenados de acordo com a criticalidade dos caminhos de produção, podendo indiretamente ser relacionados à dificuldade no cumprimento do prazo de entrega de cada trabalho. A partir desta primeira ordenação, os trabalhos são ajustados, em um segundo nível da heurística, maximizando-se a utilização dos recursos ou máquinas. Em termos lógicos, isto se assemelha à heurística desenvolvida ao longo desta dissertação, diferindo na aplicação (*job shop* X célula TG) e nas variáveis que são controladas (utilização de máquina X inventários em processo + mínimo tempo de preparação).

Em termos de modelagem matemática, a relação de precedência entre os lotes é tratada a nível de variáveis booleanas, estando as restrições amarradas em termos de dicotomias, análogos ao conceito apresentado por Baker [1974] e por Teixeira e outros [1987].

A diferença básica apresentada nesta dissertação é o tratamento de precedência imediata, ou seja, a variável booleana indicará apenas a precedência entre um dado lote *i* e um dado lote *j* se o lote *i* preceder imediatamente o lote *j*, não existindo na linha de produção nenhum outro lote sendo processado entre o término de processamento de *i* e o início do processamento de *j*. Isto se deve ao fato do tempo de preparação ser considerado no modelo.

2.4. EXEMPLO DE UMA CÉLULA DE FABRICAÇÃO : A CÉLULA PILOTO

A célula que motivou este trabalho poderia também ser denominada uma semi-linha de usinagem. Ela produz engrenagens para um dos produtos de mais alto volume de produção e venda da empresa, transmissões mecânicas. Portanto, o número de itens que compõem as duas famílias de peças feitas pela célula é pequeno. Cada tipo de peça processada passa por dois conjuntos de operações. O torneamento dos *blanks* e o corte de dentes das engrenagens no primeiro e o corte do sincronizado, chanfro e acabamento dos perfis dos dentes no segundo, como mostra a figura 2.4.

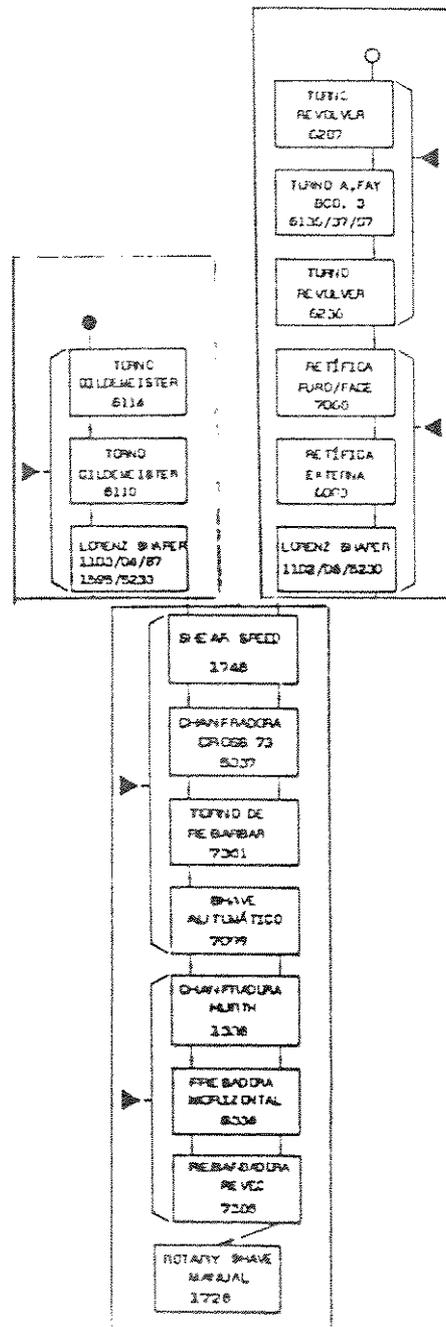


Figura 2.4. Configuração da célula piloto

Neste exemplo, a programação de montagem determina uma dada demanda para cada tipo de peça produzida na célula, no caso os tipos p1, p2, p3 e p4. A célula é composta por três linhas de máquinas, E1, E2 e E3, que são sequencias independentes de máquinas que trabalham em série, compartilhadas não simultaneamente por um ou mais tipos de peças em uma dada etapa do processo produtivo. A figura 2.5 representa esquematicamente a célula

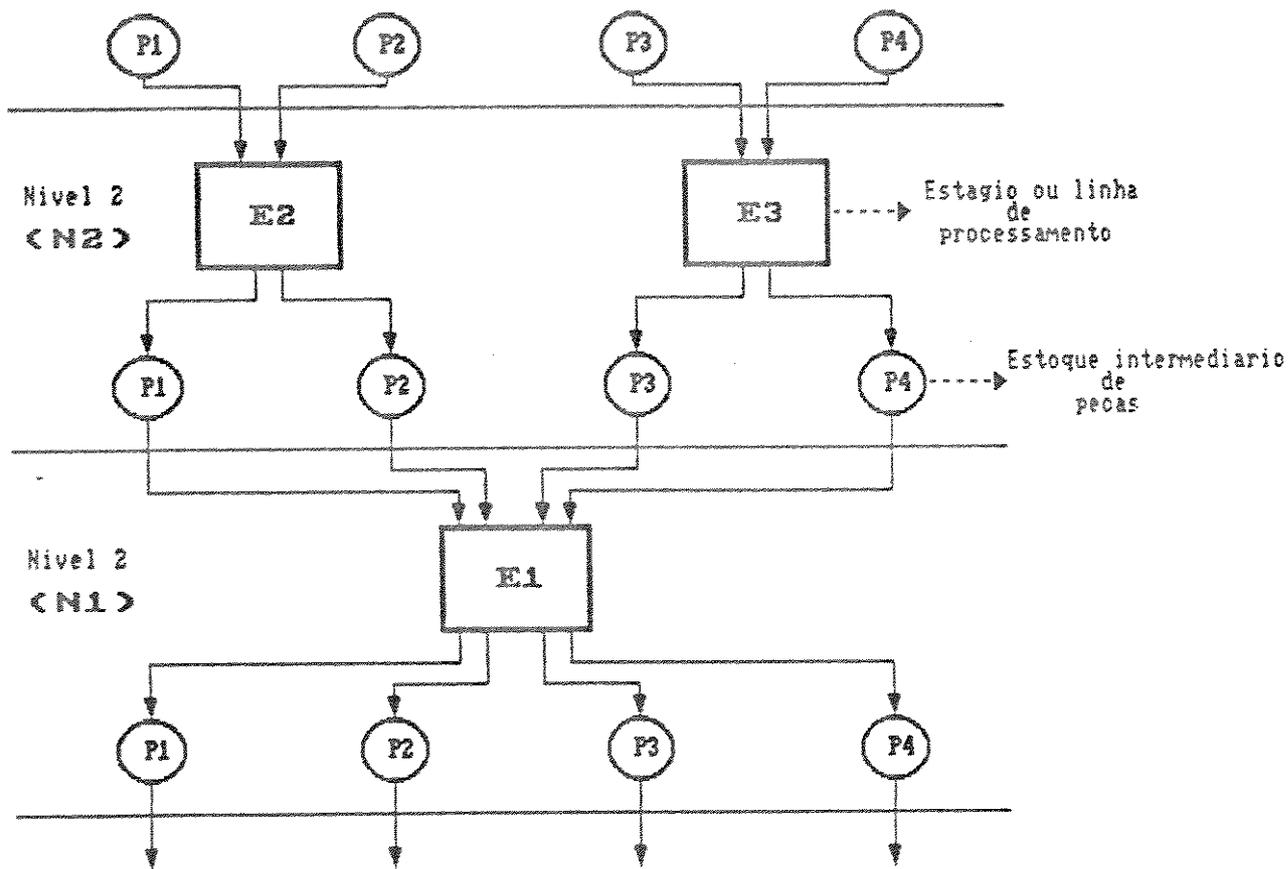


Figura 2.5. Representação Esquemática da Célula

Estas linhas estão dispostas em um *layout* funcional de dois níveis, n1 e n2, que convencionamos enumerar a partir da demanda final da célula. Os níveis são identificáveis devido à necessidade de tomadas de decisões sobre o que produzir em um dado instante.

Portanto, o fluxo de produção apresentado para a peça p1, por exemplo, inicia no nível n2, na linha E2 e termina no nível 1 na linha E1, existindo a possibilidade de estocagem ao final de cada nível.

Depois de terminada a usinagem na célula, a peça ainda sofre operações de tratamento térmico e retífica. É necessário programar cada estágio da célula para produzir um determinado tipo de peça. Para esta tarefa dispõe-se de várias informações, dentre elas :

- Prioridade de Produção. Projetando uma demanda diária de montagem e abatendo os saldos existentes nas operações posteriores, é possível estabelecer uma ordem de prioridades para as peças no nível n1.
- Estoques intermediários e quantidade de forjados disponíveis.
- Ferramentas disponíveis para usinagem e controle de qualidade.
- Manutenção de máquinas, programadas ou não.
- Operadores e montadores de máquinas.
- Para quais peças estão montadas as linhas no instante em que a decisão deve ser tomada.

Elaborar uma boa programação nesta célula piloto não é uma tarefa difícil para, por exemplo, um supervisor experiente. Mesmo assim, ele pode enfrentar dificuldades em um determinado estágio, causando um impacto sobre os outros dois que leve à necessidade de uma reprogramação. Nem sempre isto fica claro ao supervisor no momento oportuno, aquele em que a perda pudesse ser a menor possível. Outra variável importante sobre a qual há pouco controle é o valor do inventário em processo que ele manipula.

Os princípios desenvolvidos para o equacionamento desta célula simples podem ser aplicadas a configurações bem mais complexas. Na empresa hoje, já estão em andamento projetos de formação de células com um número maior de níveis, cada um com vários estágios, estoques e pontos de convergência para os níveis seguintes. A programação, neste caso, não é óbvia, mesmo a um supervisor experiente. Para tomar decisões sobre a programação em tal ambiente é necessário uma ferramenta que seja ágil e proporcione uma visão geral do desempenho de toda a célula face às alternativas de produção disponíveis no momento.

2.5. FORMULAÇÃO DO PROBLEMA DE SEQUENCIAMENTO

A questão de desenvolvimento de sistemas de sequenciamento é, hoje, bastante delicada e polêmica. O desejável em qualquer situação de sequenciamento é a obtenção de uma sequência ótima global, onde todos os parâmetros do sistema estejam alocados de tal forma que não exista alterações possíveis de estado que leve a uma melhor solução.

O problema aqui tratado é o de programação de produção de células compostas de manufatura, ou seja, células formadas a partir de mais de uma linha serial de máquinas de usinagem. A produção é processada em lotes, atendendo a prazos de entrega previamente definidos por uma programação de montagem final dos componentes, sendo que dois lotes não podem ser simultaneamente processados em uma mesma linha. A composição estrutural de cada estágio da célula é feita embasada em conceitos e tecnologias de processos, tempos e métodos, ferramentaria e tecnologia de grupo.

Estas células processam normalmente mais de um tipo de peça. Apesar de haver alguma semelhança entre os diversos tipos processados, uma vez que a configuração da célula é dimensionada para peças com processos semelhantes, existe a ocorrência de tempos de preparação (ou setup) entre a execução de dois lotes de tipos diferentes em uma mesma linha.

Outra consideração importante à modelagem do problema é a existência de estoques intermediários entre as linhas, que funcionam como niveladores de diferentes velocidades de produção.

Estes estoques, ou inventários em processo, também evitam rompimento do fluxo de produção, em casos de paradas não programadas de uma ou mais linhas, funcionando, assim, como estoques de segurança.

Em termos de modelagem o problema em questão tem mais de um objetivo a ser considerado. Trata-se de uma adequação do atendimento aos prazos de entrega versus minimização dos estoques em processo versus minimização dos tempos de preparação.

A priori, observando os objetivos em questão, um tratamento matemático mais adequado ao problema requereria a manipulação de um problema do tipo multiobjetivo, mas, algumas premissas adotadas em consenso com os usuários do sistema permitem uma abordagem mais simples e igualmente eficiente sob o ponto de vista operacional.

A primeira premissa adotada foi a consideração da demanda (prazos de entrega) como um fato determinístico e conhecido a priori, dentro de um determinado horizonte de programação. Isto torna o atendimento ao prazo uma restrição do problema. As situações em que atrasos a estes prazos se tornam inevitáveis fazem com que o programa de produção seja infactível. Nestes casos, os prazos necessariamente devem ser negociados com o cliente, ou seja, a linha de montagem, não podendo o sistema simplesmente atribuir uma nova data de entrega segundo critérios de otimização, como seria no caso de uma minimização das ocorrências de atrasos.

Outra premissa adotada foi o estabelecimento de patamares máximos de estoque para cada tipo de peça ao final de cada linha. Se o volume de peças, ao longo do horizonte, se manter abaixo destes níveis, o sequenciamento é considerado satisfatório. Este fato transforma o controle de estoques em mais uma classe de restrições do problema.

Assim, com os prazos de entrega e o volume de estoques sendo garantidos a nível de restrição, os objetivos se reduzem à minimização dos tempos de preparação. Portanto, a programação da célula tem, neste contexto, a função de atingir a seguinte situação:

- * Atender aos prazos de entrega definidos pelo Sistema de Programação de Produção (restrição).
- * Manter os estoques dentro de patamares pré estabelecidos, proporcionando um controle sobre o inventário em processo (restrição).
- * Minimizar os tempos de preparação necessários, mantendo sob controle as restrições anteriores (objetivo).

Apesar desta simplificação quanto ao enfoque, ainda temos o problema do complexo universo de variáveis que interferem no processo produtivo e da natureza dinâmica de algumas destas variáveis. Na situação real, temos paradas não programadas de manutenção, trocas de ferramentas, absenteísmo, produção em caráter de urgência, impossibilidade de execução de *setup* em troca de turno e outras variáveis que tornam praticamente impossível a resolução do problema de sequenciamento *off line*.

A idéia, portanto, foi desenvolver um modelo que envolvesse apenas as principais variáveis do problema e desse um caráter determinístico às entradas de dados.

Foram, portanto, considerados como entradas de demanda o prazo de entrega de cada lote ao final da célula, a quantidade e o tipo das peças que compõem cada lote. Como parâmetros de configuração do sistema produtivo, foram consideradas as velocidades de produção de cada tipo de peça, os estoques mínimos e máximos para cada tipo de peça ao final de cada linha e as matrizes de *setup* entre os diversos tipos em cada linha.

Como velocidade de produção, decidiu-se considerar a velocidade da linha como sendo igual à velocidade de sua máquina gargalo. Esta premissa é aceitável, considerando que a produção se dá em lotes de fabricação de tamanhos consideráveis.

Outra concessão feita na modelagem é quanto à não existência de estoques dentro da linha. São considerados apenas os estoques formados ao final de cada linha. Estes estoques funcionam como *buffers*, que evitam rompimentos de fluxo de produção.

Em termos de modelo não é considerada a possibilidade de refluxo de produção, ou seja, uma peça não retorna a uma etapa de processo pela qual ela já passou, e também não é permitido a uma mesma linha fazer dois tipos diferentes de etapas de processo para uma mesma peça. Também não é considerada a possibilidade de fluxos alternativos de produção para cada tipo de peça.

Finalmente, como parâmetros de estado da célula, são considerados o tipo de montagem existente em cada linha e o volume de estoque de cada tipo de peça ao final de cada linha, no instante inicial do sequenciamento (instante 0).

2.6. RESUMO

Vimos neste capítulo as justificativas técnicas para as situações de migração do sistema funcional convencional de produção para a formação de estruturas celulares.

Foi apresentada também, a título de exemplo, a configuração básica da célula piloto a ser considerada como estudo de caso (desenvolvida pela Equipamentos Clark Ltda), bem como a formulação do problema de sequenciamento da produção baseado em fluxo tecnológico, ressaltando-se as premissas operacionais adotadas para uma abordagem prática.

No próximo capítulo serão apresentados os modelos heurístico e matemático, empregados na resolução do problema.

3. Capítulo 3 - Modelagem do Problema de Sequenciamento

3.1. INTRODUÇÃO

Neste capítulo são apresentados os modelos heurístico e matemático associados ao problema formulado.

Os dois modelos podem ser considerados modelos de "dois estágios" uma vez que, além dos entradas, aqui apresentados na forma descritiva, é necessária a geração de alguns parâmetros pré-processados, para utilização tanto no algoritmo heurístico e quanto no matemático.

Deve-se ressaltar que o método heurístico apresentado é apropriado apenas aos sistemas produtivos distribuídos em estrutura celular, pois estes garantem um fluxo bem estabelecido e em apenas uma direção. Além disso, a estrutura celular permite um controle eficaz sobre os pontos de estocagem, sem o que, a aplicação do método se torna, na prática, contraproducente.

No modelo heurístico, o "primeiro estágio" abrange a determinação da "demanda líquida" a ser processada em cada linha de produção, os cálculos dos tempos de processamento de cada lote e as "datas mais cedo" e "datas mais tarde" de produção de cada lote. No modelo matemático é efetuado apenas o cálculo dos tempos de processamento dos lotes. Os significados lógicos destes cálculos são devidamente explicados ao longo deste capítulo.

3.2. ENTRADAS DO SISTEMA

O modelo heurístico empregado utiliza as seguintes categorias de entradas:

Parâmetros do sistema célula
 Informações de estado da célula
 Informações de demanda

3.2.1. PARÂMETROS DO SISTEMA CÉLULA

São as informações inerentes e estáveis ao sistema. Através destas informações, são caracterizados tanto o sistema produtivo quanto os tipos de peças que este produz.

São parâmetros do sistema:

Configuração da célula,
 Volumes mínimos e máximos de estoque para cada nível da célula,
 Velocidade de produção de cada linha,
 Matrizes dos tempos de preparação ou setup.

3.2.1.1. Configuração da célula

Trata-se das informações necessárias para definição do cenário que representará para o modelo o sistema produtivo. No nosso exemplo tem-se a necessidade de atribuir ao modelo quais linhas pertencem a cada nível. Cada nível é definido como uma etapa ou conjunto de etapas em série de

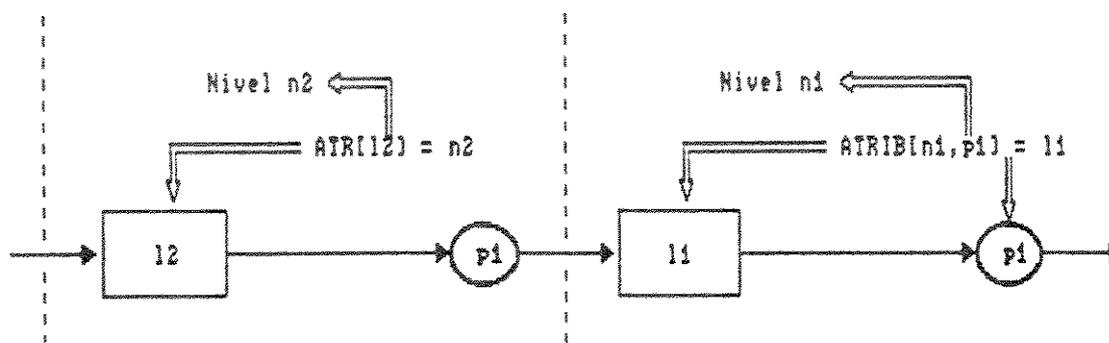


Figura 3.1. Indicações de Fluxos Tecnológicos

um processo antes de uma bifurcação de fluxo ou de um ponto de estocagem. Sendo assim, tem-se para as atribuições das linhas :

$ATR[I1] = n1$; $ATR[I2] = n2$ e $ATR[I3] = n2$ onde ATR é uma variável indexada que atribui uma linha a um nível.

Da mesma forma devemos indicar a rota tecnológica de cada peça, ao longo das linhas.

$ATRIB[n1,p1] = I1$; $ATRIB[n2,p1] = I2$ e assim por diante, onde $ATRIB$ é uma variável que atribui a passagem de um determinado tipo de peça por uma linha em um determinado nível. A figura 3.1 representa a atribuição de linhas a níveis e de peças a linhas.

3.2.1.2. Volumes mínimos e máximos de estoque em cada nível.

Uma das premissas admitidas para o controle sobre o inventário em processo foi o estabelecimento de patamares mínimos e máximos de estoque para cada tipo de peça em cada nível. Para o modelo, os estoques de um dado nível n estão alocados ao final das linhas pertencentes ao nível n .

O nível mínimo de estoque tem a finalidade de funcionar como estoque de segurança do sistema. O nível máximo de estoques tem a finalidade de limitar a ocorrência de inventário em processo. Logo, tem-se o seguinte :

$$EMI[n1,p1] = m$$

$EMA[n1,p1] = M$ onde m e M são as quantidade mínimas e máximas de peças $p1$ no estoque do nível $n1$, e assim sucessivamente para as demais peças nos demais níveis. A figura 3.2 representa os patamares mínimos e máximos de estoque para um determinado tipo de peça em um nível.

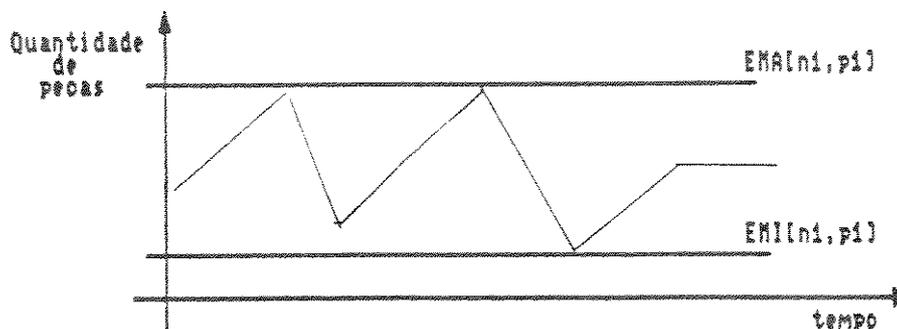


Figura 3.2. Evolucao de estoques, patamar minimo e patamar maximo.

3.2.1.3. Velocidade de produção

É a taxa de produção horária de uma determinada peça p em uma determinada linha l. Devemos notar que a capacidade em questão é nominal, não considerando tempos improdutivo e tempos de preparação. Assim, tem-se :

$PR[l,p] = c_{l1}$, que significa a quantidade de peças p1 passíveis de serem produzidas pela linha l1 por unidade de tempo.

3.2.1.4. Matrizes de tempos de preparação

São as informações dos tempos de preparação necessários para iniciar a produção dos diversos tipos de peça em cada nível. Apresenta-se na forma matricial porque existe uma relação específica de um tipo de peça para cada um dos demais em cada nível, como mostra a figura 3.3.

para de	tipo p1	tipo p2	tipo p3	tipo p4
tipo p1	0	$St_{1,1,2}$	$St_{1,1,3}$	$St_{1,1,4}$
tipo p2	$St_{1,2,1}$	0	$St_{1,2,3}$	$St_{1,2,4}$
tipo p3	$St_{1,3,1}$	$St_{1,3,2}$	0	$St_{1,3,4}$
tipo p4	$St_{1,4,1}$	$St_{1,4,2}$	$St_{1,4,3}$	0

Figura 3.3. Tabela exemplo de tempo de preparação

Portanto, tem-se que :

$SETUP[n1,p1,p2] = st_{1,1,2}$ é o tempo de preparação necessário para uma troca da produção de peças do tipo p1 para a produção de peças do tipo p2 no nível n1, sendo que $st_{1,1,2}$ pode ter um valor diferente que o de $st_{1,2,1}$.

3.2.2. INFORMAÇÕES DE ESTADO DA CÉLULA

São as informações que definem em que estado se encontra a célula no instante em que se deseja iniciar o sequenciamento (instante 0). São informações de estado:

Montagem inicial das linhas Volume atual dos estoques

3.2.2.1. Montagem inicial das linhas

Como a produção da célula é um processo dinâmico, é necessário saber que tipo de peça está sendo produzida no instante 0 (zero), de forma a referenciar o sistema sobre quais lotes poderão ser alocados inicialmente. No caso considerado, tem-se que:

$SETUP_INIC[l1] = p2$, que indica que no instante 0 a linha l1 já está montada para produzir peças do tipo p2. Em termos do modelo, isto é tratado considerando a existência de um lote 0 fictício, com tempo de processamento nulo, composto por peças do tipo p2 inicializando a programação.

3.2.2.2. Volume atual dos estoques

A necessidade desta informação se deve ao fato do modelo fazer a conversão da demanda estabelecida pela programação de montagem para uma demanda líquida em cada nível, abatendo os excedentes de estoque. Portanto, o volume atual dos estoques definem o quanto cada nível terá que efetivamente produzir.

Por exemplo :

$ESTOQUE[n1,p1] = e11$ onde e11 é o estoque inicial de peças p1 no nível n1.

3.2.3. INFORMAÇÕES DE DEMANDA

São informações provenientes da programação de montagem, determinando o quanto a célula deve fornecer de cada tipo de peça e os prazos limite para que isto ocorra.

Estas informações são discretizadas dia a dia a partir do instante 0, dentro de um horizonte de planejamento previamente determinado.

3.3. MODELAGEM HEURÍSTICA

Como mencionado anteriormente, o modelo heurístico divide-se em duas etapas. A primeira corresponde a um tratamento do problema de maneira global, chamado genericamente de Processamento de Informações. Aqui são estabelecidas algumas considerações sobre o processamento dos lotes em cada linha, de forma a permitir, em uma segunda etapa, a execução localizada do algoritmo de sequenciamento em cada linha.

3.3.1. PROCESSAMENTO DE INFORMAÇÕES

Para a execução do algoritmo empregado para o sequenciamento, é necessário que as entradas sofram uma série de transformações que vão desde óbvias conversões de unidade até a criação de parâmetros que exercem influência vital no sequenciamento da produção. Analisaremos apenas estes últimos com a finalidade de descrever a lógica empregada no modelo, dentre os quais citamos:

- Conversão de demanda em lotes operacionais
- Cálculo do tempo de processamento de cada lote
- Estabelecimento de data mais cedo e data mais tarde de produção.

3.3.1.1. Conversão de demanda

Trata-se da conversão da "demanda bruta", apresentada anteriormente, em "demanda líquida". Isto significa um abatimento do excedente de estoque da demanda bruta.

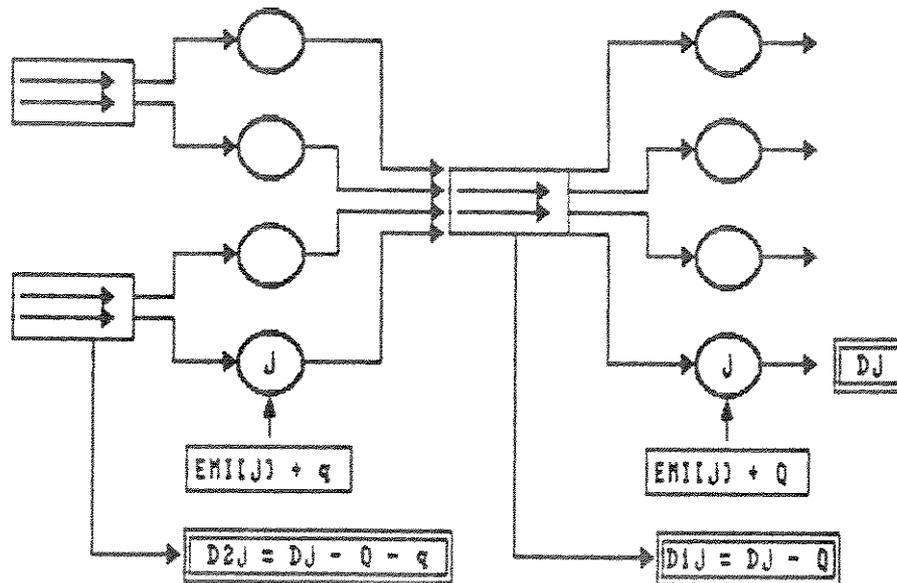


Figura 3.4. Conversão da demanda bruta em demanda líquida

Na aplicação em questão, consideramos como excedente o volume de peças acima do parâmetro de estoque mínimo do nível, como é mostrado na figura 3.4. Esta é uma premissa que poderia ser alterada para outras aplicações, bastando que se estabeleça um patamar em cada nível, a partir do qual o volume acima será considerado excedente.

A finalidade é fazer a compensação entre o que se produziu antecipadamente e a demanda bruta em vigor. Isto é válido também nos casos de atraso de produção, quando se forma um excedente negativo, ou seja, volume de estoque abaixo do patamar mínimo.

3.3.1.2. Cálculo do Tempo de Processamento de Cada Lote

Uma vez convertida a demanda e conhecendo-se a velocidade de produção de cada tipo de peça para as linhas onde estas são processadas, é possível o cálculo a priori do tempo de processamento de cada lote em cada linha. Como exemplo, se $ATR[l1] = n1$, $TIPO[y] = p1$ e $ATRIB[n1,p1] = l1$, tem-se que :

$$TF[l1,y] = XN[n1,y] / PR[n1,p1]$$

onde $TF[l1,y]$ é o tempo de processamento do lote y (composto por peças do tipo $p1$) na linha $l1$ e $XN[n1,y]$ é o tamanho operacional do lote y (em peças) no nível $n1$ (ao qual pertence a linha $l1$).

3.3.1.3. Datas Mais Cedo e Datas Mais Tarde de produção

A generalização objetivada pelo modelo pode levar a configurações razoavelmente complexas de células. Isto torna virtualmente impraticável uma heurística de sequenciamento que envolva simultaneamente todas as variáveis do problema global.

A decisão tomada considera a divisão do problema em etapas, que possam ser atacados independentemente sem, no entanto, comprometer o funcionamento da célula como um todo.

Para isto, optou-se pelo cálculo a priori de datas mais tarde e data mais cedo de produção dos lotes operacionais em cada nível, em relação ao prazo final de entrega. Desta forma, o sistema estabelece fronteiras ao longo do tempo para a produção de cada lote, semelhante à utilização de uma rede Pert, possibilitando o sequenciamento individual da produção de cada linha em cada nível.

3.3.1.3.1. Data mais tarde de fabricação

É a transferência dos prazos de entrega da demanda para os níveis anteriores. Note que na notação adotada o nível $n2$ antecede o nível $n1$.

Aqui deve ser considerada a possibilidade de execução simultânea de um mesmo lote em dois ou mais níveis, já que se trata de processamento de produção em lotes, e que o tamanho do lote de

transferência entre as linhas em uma mesma célula pode ser igual a uma única peça. Isto significa que basta existir a garantia de peça em estoque no nível n2 para que n1 possa produzir.

Tomando os parâmetros da nossa célula piloto a título de exemplo, teremos como cálculo de data mais tarde :

a) se $TF[11,y1] \leq TF[12,y1]$, como ilustrado na figura 3.5, com $TIPO[y1] = p1$, $ATR[11] = n1$ e $ATR[12] = n2$, as datas mais tarde $PE[n1,y1]$ e $PE[n2,y1]$ serão:

$$PE[n1,y1] = PE[y1]$$

$$PE[n2,y1] = PE[n1,y1]$$

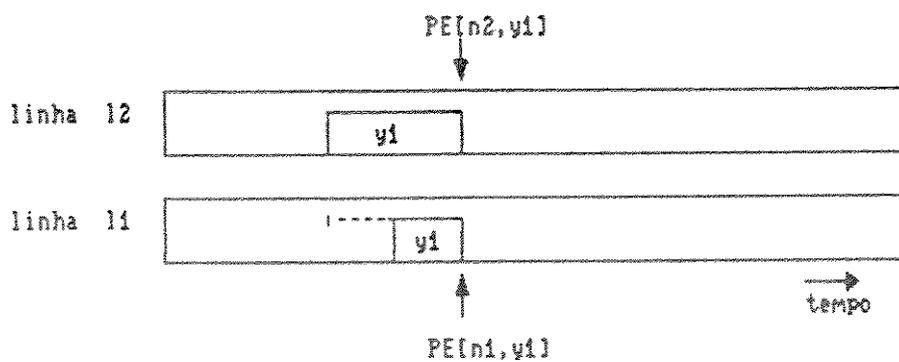


Figura 3.5. Datas mais tarde de produção para o caso a)

b) caso $TF[11,y1] > TF[12,y1]$ então teremos a seguinte situação, representada na figura 3.6: onde o cálculo será :

$$PE[n1,y1] = PE[y1]$$

$$PE[n2,y1] = PE[n1,y1] - (XN[n1,y1]/PR[n1,p1] - XN[n2,y1]/PR[n2,p1])$$

onde

p é o tipo de peça que constitui o lote y

PE[n,y] = data mais tarde de termino do lote y no nível n

XN[n,y] = tamanho do lote operacional y no nível n

PR[n,p] = velocidade de produção de peças tipo p no nível n

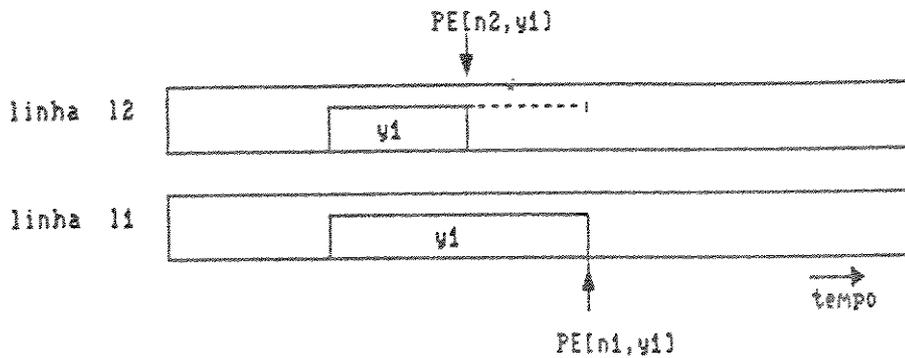


Figura 3.6. Datas mais tarde de produção para o caso b)

Para o modelo, o não atendimento ao prazo de entrega implica em redução do volume de estoques abaixo do patamar mínimo.

3.3.1.3.2. Data mais cedo de produção

Consideramos no modelo duas situações no estabelecimento de data mais cedo de produção, que são definidas a seguir:

3.3.1.3.2.1. Data mais cedo primária - DMC_p

É estabelecida com a preocupação de manter os estoques do nível em questão abaixo do limite máximo definido. O cálculo da DMC_p considera a possibilidade de máximo adiantamento da execução dos lotes de um tipo p , tal que, se estes forem consumidos nos últimos instantes possíveis, ou seja, nas suas datas mais tarde, isto não acarrete o rompimento do patamar máximo de estoque de p no nível em questão.

3.3.1.3.2.2. Data mais cedo secundária - DMC_s

Tem a finalidade de impedir que o estoque do nível anterior (n_i+1) se torne inferior ao patamar mínimo devido ao sequenciamento do nível em questão (n_i). As DMC_s's do nível n_i só são calculadas após o sequenciamento do nível n_i+1 e imediatamente antes do sequenciamento do nível n_i .

A data mais cedo de fabricação de um lote a ser usada no sequenciamento de uma linha será o máximo entre a DMC_p e a DMC_s encontradas.

$$DMC[n,y] = \max\{DMCp[n,y],DMCs[n,y]\}$$

3.3.2. SEQUENCIAMENTO DA PRODUÇÃO USANDO O ALGORÍTMO

Uma vez estabelecidos DMC e PE de cada lote em um nível, estes funcionarão como variáveis de acoplamento entre os sequenciamentos das linhas, ou seja, estará estabelecido um intervalo de tempo dentro do qual o lote deverá ser produzido. Isto torna possível fazer o sequenciamento individual de cada linha neste nível.

O sequenciamento é feito a partir do último nível em direção ao primeiro, no sentido do fluxo de produção.

O sequenciamento dos lotes de uma linha l em um nível n inicia-se colocando os lotes em uma ordem crescente das datas mais tarde de término $PE[n,y]$. É uma regra inicial, onde procura-se obter o menor atraso máximo em relação aos prazos de entrega (Baker,[1974]).

Esta alocação é armazenada em um vetor de posição (SEQ) que varia de zero ao número de lotes a serem sequenciados na linha. A posição zero é reservada a um lote fictício que indica o tipo de peça para qual a linha está montada no instante inicial do sequenciamento.

As DMC's e PE's servirão como limites de alocação de cada lote, sendo que o algoritmo empregado procura minimizar os tempos de preparação entre os lotes ordenados, respeitando estes limites.

3.3.3. O algoritmo de sequenciamento

Passo 00 - Ordene os lotes da linha por PE's crescentes em um vetor de posição de sequência (SEQ).

Passo 01 - Aloque um lote fictício vazio (y_0) na posição 0 da sequência SEQ com $tipo[y_0]$ igual à montagem ou setup inicial da linha. Faça marca (*) de alocação estável na posição 0.

Passo 02 - Retire as eventuais marcas de não candidatos (nc) da sequência SEQ. Atualize os instantes de término de processamento ($TSEQ[l,y]$) para todos os lotes nas posições da sequência, considerando os tempos de processamento, os tempos de setup e as DMC's dos lotes. Faça $Z = tipo$ do lote z alocado na última posição marcada com (*).

Passo 03 - Apanhe o lote (g) alocado na primeira posição de SEQ após z . Atribua marca de candidato (**) ao lote.

Passo 04 - O lote g é o último da sequência SEQ ?

sim - Aloque lote na primeira posição não assinalada com (*).
Coloque marca (*) na posição - FIM

não - Passo 05.

Passo 05 - Existe lote sem marca de não candidato (nc) com menor tempo de preparação do que $g (**)$ em relação ao tipo $Z (*)$ nas posições posteriores a $g (**)$ da sequência SEQ, tal que DMC do elemento seja menor que o TSEQ do lote $z (*)$ ou menor que o DMC do candidato $g (**)$?

sim - Apanhe o primeiro lote n que atenda as condições. Marque este lote como novo candidato proposto (n^*) e vá para o Passo 06.

não - Passo 08.

Passo 06 - A alocação do novo candidato proposto $n (n^*)$ na primeira posição não marcada de SEQ rompe a factibilidade das datas mais tarde (PE) dos lotes sequenciados entre a posição de $z (*)$ e a atual posição do elemento $n (n^*)$?

sim - Retire marca de novo candidato proposto (n^*) e faça marca de não candidato (nc) no lote n . Vá para o Passo 05.

não - Passo 07.

Passo 07 - Atribua marca de candidato (***) ao novo elemento g e retire marca (***) do candidato anterior. Vá ao Passo 05.

Passo 08 - A DMC do candidato $g (**)$ é maior que o TSEQ de $z (*)$?

sim - Passo 09.

não - Passo 12.

Passo 09 - Existe lote k sem marca de não candidato (nc) posterior a g na sequência SEQ, tal que seu (DMC + tempo de preparação de k em relação a Z) seja menor que (DMC + tempo de preparação do candidato $g (**)$ em relação a Z) ?

sim - Apanhe o primeiro lote k que atenda a esta condição, marque ele como novo candidato proposto (n^*) e vá ao Passo 10.

não - Passo 12

Passo 10 - A alocação do novo candidato proposto $k (n^*)$ na primeira posição de SEQ após $z (*)$ rompe factibilidade das datas mais tarde (PE) dos lotes sequenciados entre a posição de $z (*)$ e a atual posição do elemento k ?

sim - Retire marca de novo candidato proposto (n^*) e faça marca de não candidato (nc) no lote k . Vá ao Passo 09.

não - Passo 11.

Passo 11 - Atribua marca de candidato ($**$) ao lote k , tornando-o o novo elemento g , e retire marca ($**$) do candidato anterior. Vá ao Passo 09.

Passo 12 - Aloque elemento candidato g ($**$) na primeira posição após z ($*$) da sequência SEQ , fazendo marca de alocação estável ($*$) na posição e deslocando cada um dos elementos posteriores uma posição à frente. Vá ao Passo 02.

3.3.4. Comentários sobre o algoritmo.

Passo 00 - Ordenação inicial segundo critério de prazos de entrega crescente. Esta ordenação é armazenada em um vetor SEQ .

Ex.: lote 1 - $PE[1,1] = 2$
 lote 2 - $PE[1,2] = 1$
 lote 3 - $PE[1,3] = 3$

Na ordenação inicial por prazo de entrega, teremos:

$SEQ = \{2,1,3\}$, ou seja, $SEQ[1] = 2$ (lote 2 na posição 1)
 $SEQ[2] = 1$ (lote 1 na posição 2)
 $SEQ[3] = 3$ (lote 3 na posição 3)

Passo 01 - Tem a finalidade de considerar a situação de montagem inicial da linha a ser sequenciada. Isto é importante porque os tempos de preparação são considerados na obtenção da solução do problema.

Ex.: $SEQ[0] = 0$, onde o lote 0 é do tipo $SETUP_INIC[1]$
 $SEQ[1] = 2$
 $SEQ[2] = 1$
 $SEQ[3] = 3$

A marca de alocação estável ($*$) tem o significado de indicar que a posição 0 já está definitivamente ocupada pelo lote 0.

Passo 2 - O algoritmo atua o tempo todo sobre a sequência SEQ , alterando-a ao longo do seu processamento. Ele é um loop, atuando sobre uma determinada posição da sequência de cada vez.

O algoritmo inicia considerando a posição 1 da sequência, verificando qual lote deve definitivamente ser alocado nesta posição. Após determinar qual lote deve ocupá-la, é feita uma marca de estabilidade ($*$) na posição e o algoritmo toma a posição seguinte não marcada. Este

procedimento é repetido para todas as posições do vetor SEQ. Após o passo 12, o algoritmo retorna ao passo 02, abordando a posição seguinte da sequência.

A retirada de marca de não candidato (nc) deve ser considerada apenas nas interações seguintes à primeira interação inicial, quando alguns lotes ao longo do processamento foram considerados não candidatos (nc) a ocupar a posição abordada pela interação.

A atualização do TSEQ[l,y] deve ser efetuada porque na interação anterior alguma posição pode ter sido alterada, estando alterados portanto os instantes de término de processamento de cada lote.

Passo 03 - O primeiro lote após a última posição com marca de estabilidade (*) é considerado o primeiro candidato à posição por ser o lote, ainda não alocado definitivamente, com prazo de entrega mais eminente. Os demais lotes da sequência são considerados, um a um, para a posição, comparando-se as condições de antecipação com a situação do lote candidato em curso.

$$SEQ = \{ 0^*, 2^*, 1^{**}, 3 \}$$

lote a ser comparado c/ o candidato

candidato a permanecer na posição

Passo 04 - Se o lote g for o último da sequência, não existem mais lotes não estabilizados, sem a marca (*), a serem comparados com g, portanto g é automaticamente estável e o algoritmo chegou ao fim.

Passo 05 - Procura-se determinar se existe um lote subsequente ao candidato em condições de iniciar o processamento em um instante menor que o instante de início do candidato, ou seja, antes que o candidato, observando-se o tempo de preparação em relação ao ocupante da última posição marcada (*).

Caso exista lote nestas condições, ele não pode imediatamente ser candidato até ser verificado se o TF do lote não provocaria atrasos nos prazos de entrega dos lotes intermediários da sequência. Por isso, o lote é considerado candidato proposto (n^*), até ser checado no Passo 06.

$$SEQ = \{ 0^*, 2^*, 1^{**}, 3^{n^*} \}$$

Passo 06 - É verificada a possibilidade de remanejamento dos lotes candidatos propostos (n^*) para a primeira posição não marcada. Esta antecipação do lote (n^*) na sequência poderia impedir que os lotes compreendidos entre a posição (**) e a posição (n^*) atendessem aos prazos de entrega, ou aumentassem possíveis atrasos já existentes. Isto é verificado, e caso uma destas situações ocorra, o lote assinalado (n^*) é considerado não candidato, recebendo a marca (nc).

$$SEQ = \{ 0^*, 2^*, 1^{**}, 3^{nc} \}$$

Passo 07 - Se o lote candidato proposto (n^*) pode ser antecipado sem provocar atrasos ou sem dilatar atrasos já existentes, o lote (n^*) passa a ser candidato (**).

Passo 08 - É verificado se existe tempo ocioso entre o término previsto do último lote alocado (*) e o início previsto de processamento do lote candidato. Caso positivo, será verificada a possibilidade de antecipação de outro lote que possa iniciar antes, mesmo tendo tempo de preparação maior que o do lote candidato. Esta situação se deve à possibilidade do lote candidato, apesar de ter o menor tempo de preparação dentre os lotes que podem ser antecipados, ter seu início de processamento bloqueado por DMC.

Passo 09 - Nesta busca, os lotes anteriormente marcados como não candidatos (nc) não são considerados para uma possível antecipação. A busca em si é similar à efetuada no Passo 05, apenas mudando o critério de seleção para o candidato proposto.

Passo 10 - Similar ao Passo 06, observando-se a possibilidade de uma antecipação real do lote sem que ocorra atrasos ou dilatações de atrasos já existentes.

Passo 11 - Similar ao Passo 07.

Passo 12 - O candidato final (**) após passar pelas consistências acima relatadas é finalmente alocado estavelmente na primeira posição livre da sequência, obtendo a marca de estabilidade (*). O algoritmo retorna ao Passo 02 para considerar o alocamento da posição seguinte.

3.4. MODELAGEM MATEMÁTICA

O modelo que aqui será apresentado reflete exclusivamente a situação da célula piloto, considerando as limitações impostas por este sistema produtivo. Ele foi desenvolvido para permitir uma análise crítica sobre os resultados obtidos pelo modelo heurístico neste exemplo específico, nos casos em que existem poucos lotes a serem sequenciados e os estoques iniciais estão vazios.

A célula piloto tem uma característica de produção bastante peculiar que é levada em consideração por este modelo matemático, a saber: a velocidade de produção nas linhas E2 e E3 é sempre menor ou igual à velocidade de produção de E1. Isto facilita a questão de determinação de prazos de entrega para os lotes em E2 e E3 e facilita o controle sobre os estoques intermediários, que terão seus máximos locais nos instantes de início de processamento dos lotes em E1.

O modelo matemático desenvolvido parte de uma simplificação do problema, onde os estoques intermediários iniciais são iguais a zero. Desta forma, a demanda líquida inicial é igual à demanda bruta, permitindo a garantia de que o tempo de processamento de um lote na linha E2 ou E3 seja maior que o tempo de processamento deste mesmo lote em E1, sendo o tamanho de cada lote igual nos dois níveis considerados.

A terminologia adotada para o modelo matemático é um pouco diferente que a adotada no modelo heurístico, e será apresentada a seguir.

3.4.1. TERMINOLOGIA DO MODELO MATEMÁTICO

a,b,c : Índices que representam as linhas E1,E2 e E3 respectivamente.

i,j,n : Lotes genéricos.

l : Linha genérica.

I : Quantidade de lotes existentes.

TIPO[i] : Tipo de peça que constitui um determinada lote i.

SETUP_l[TIPO[i],TIPO[j]] : Tempo de preparação necessário para a linha l na troca de produção de peças do TIPO[i] para a produção de peças do TIPO[j].

AT[TIPO[i],l] = 1 : Indicação de que as peças TIPO[i] são processadas na linha l no segundo nível.

AT[TIPO[i],l] = 0 : Caso isto não ocorra.

ISEQ_l[i] : Instante de início de processamento do lote i na linha l.

TFI_l[i] : Tempo de processamento do lote i na linha l.

$PE[i]$: Prazo final de entrega do lote i no final da célula.

$XN[i]$: Quantidade de peças a serem produzidas no lote i .

$\Delta_{l[i,j]} = 1$: Caso o lote i é processado imediatamente antes que o lote j na linha l .

$\Delta_{l[i,j]} = 0$: Caso isto não ocorra.

$\alpha_{l[i,j]} = 1$: Caso o lote i é consumido antes que o lote j no estoque da linha l .

$\alpha_{l[i,j]} = 0$: Caso isto não ocorra.

$\beta_{l[i,j]} = 1$: Caso o lote i é produzido na linha l antes de se iniciar o consumo do lote j no estoque da linha l .

$\beta_{l[i,j]} = 0$: Caso isto não ocorra.

$\gamma_{l[i,j]} = 1$: Caso o lote i está em processamento na linha l no instante de início de consumo do lote j no estoque da linha l .

$\gamma_{l[i,j]} = 0$: Caso isto não ocorra.

G : Número tão grande quanto se queira.

3.4.2. FUNÇÃO OBJETIVO

O problema tem mais de um objetivo a ser considerado. Trata-se de uma adequação do atendimento aos prazos de entrega *versus* minimização dos estoques em processo *versus* minimização dos tempos de preparação. A priori, um tratamento matemático mais adequado ao problema requereria a manipulação de um problema do tipo multiobjetivo, mas, algumas premissas adotadas em consenso com os usuários do sistema permitem uma abordagem mais simples e igualmente eficiente sob o ponto de vista operacional.

A primeira premissa adotada foi a consideração da demanda (prazos de entrega) como um fato determinístico e conhecido a priori, dentro de um determinado horizonte de programação. Isto torna o atendimento ao prazo uma restrição do problema. As situações em que atrasos a estes prazos se tornam inevitáveis fazem com que o problema se torne ineficaz. Nestes casos, os prazos necessariamente devem ser negociados com o cliente, ou seja, a linha de montagem, não podendo o sistema simplesmente atribuir uma nova data de entrega segundo critérios de otimização.

A segunda premissa adotada foi o estabelecimento de patamares máximos de estoque para cada tipo de peça ao final de cada linha. Se o volume de peças, ao longo do horizonte, se manter abaixo destes níveis, o sequenciamento é considerado satisfatório. Este fato transforma o controle de estoques em mais uma classe de restrições do problema.

Desta forma, com os prazos de entrega e o volume de estoques sendo garantidos a nível de restrição, a nossa função objetivo tratará apenas de minimizar os tempos de preparação. Conseqüentemente :

$PE[i]$: Prazo final de entrega do lote i no final da célula.

$XN[i]$: Quantidade de peças a serem produzidas no lote i .

$\Delta_{l1}[i,j] = 1$: Caso o lote i é processado imediatamente antes que o lote j na linha l .

$\Delta_{l1}[i,j] = 0$: Caso isto não ocorra.

$\Delta_{l2}[i,j] = 1$: Caso o lote i é consumido antes que o lote j no estoque da linha l .

$\Delta_{l2}[i,j] = 0$: Caso isto não ocorra.

$\Delta_{l3}[i,j] = 1$: Caso o lote i é produzido na linha l antes de se iniciar o consumo do lote j no estoque da linha l .

$\Delta_{l3}[i,j] = 0$: Caso isto não ocorra.

$\Delta_{l4}[i,j] = 1$: Caso o lote i está em processamento na linha l no instante de início de consumo do lote j no estoque da linha l .

$\Delta_{l4}[i,j] = 0$: Caso isto não ocorra.

G : Número tão grande quanto se queira.

3.4.2. FUNÇÃO OBJETIVO

O problema tem mais de um objetivo a ser considerado. Trata-se de uma adequação do atendimento aos prazos de entrega *versus* minimização dos estoques em processo *versus* minimização dos tempos de preparação. A priori, um tratamento matemático mais adequado ao problema requereria a manipulação de um problema do tipo multiobjetivo, mas, algumas premissas adotadas em consenso com os usuários do sistema permitem uma abordagem mais simples e igualmente eficiente sob o ponto de vista operacional.

A primeira premissa adotada foi a consideração da demanda (prazos de entrega) como um fato determinístico e conhecido a priori, dentro de um determinado horizonte de programação. Isto torna o atendimento ao prazo uma restrição do problema. As situações em que atrasos a estes prazos se tornam inevitáveis fazem com que o problema se torne infactível. Nestes casos, os prazos necessariamente devem ser negociados com o cliente, ou seja, a linha de montagem, não podendo o sistema simplesmente atribuir uma nova data de entrega segundo critérios de otimização.

A segunda premissa adotada foi o estabelecimento de patamares máximos de estoque para cada tipo de peça ao final de cada linha. Se o volume de peças, ao longo do horizonte, se manter abaixo destes níveis, o sequenciamento é considerado satisfatório. Este fato transforma o controle de estoques em mais uma classe de restrições do problema.

Desta forma, com os prazos de entrega e o volume de estoques sendo garantidos a nível de restrição, a nossa função objetivo tratará apenas de minimizar os tempos de preparação. Consequentemente :

$$\begin{aligned}
 \text{f.o.: } \min \sum_{i=0}^I \sum_{j=1}^I \text{SETUP}_a [\text{TIPO}[i], \text{TIPO}[j]] * \text{Delta}_a[i,j] + \\
 + \sum_{i=0}^I \sum_{j=1}^I \text{SETUP}_b [\text{TIPO}[i], \text{TIPO}[j]] * \text{Delta}_b[i,j] + \\
 \text{AT}(\text{TIPO}[i], b) = 1 \\
 \text{AT}(\text{TIPO}[j], b) = 1 \\
 + \sum_{i=0}^I \sum_{j=1}^I \text{SETUP}_c [\text{TIPO}[i], \text{TIPO}[j]] * \text{Delta}_c[i,j] \\
 \text{AT}(\text{TIPO}[i], c) = 1 \\
 \text{AT}(\text{TIPO}[j], c) = 1
 \end{aligned}$$

3.4.3. RESTRICÕES DO MODELO MATEMÁTICO

3.4.3.1. Restrição classe I - Garantia do Prazo de Entrega

O prazo de entrega de um lote i deve ser obedecido. Esta é uma condição primordial do modelo matemático aqui empregado.

Considerando que a produção se dá em lotes (o tempo de produção isolado de uma única peça é considerado desprezível), e que a velocidade de produção no nível $n1$ é maior que no nível $n2$, o que deve ser observado nesta classe de restrições é a garantia de que os lotes no primeiro nível $n1$ tenham os seus processamentos terminados antes dos prazos de entrega, e que os lotes do segundo nível $n2$ tenham seus processamentos terminados no máximo simultaneamente aos seus correspondentes no nível $n1$. Logo, tem-se o seguinte :

$$\text{I.a) } \text{ISEQ}_a[i] + \text{TF}_a[i] \leq \text{PE}[i]$$

$$\text{I.b) } \text{ISEQ}_b[i] + \text{TF}_b[i] \leq \text{ISEQ}_a[i] + \text{TF}_a[i] \quad (\text{Caso } \text{AT}[\text{TIPO}[i], b] = 1)$$

$$\text{I.c) } \text{ISEQ}_c[i] + \text{TF}_c[i] \leq \text{ISEQ}_a[i] + \text{TF}_a[i] \quad (\text{Caso } \text{AT}[\text{TIPO}[i], c] = 1)$$

3.4.3.2. Restrição classe II - Relações de precedência

Esta classe de restrições estabelece a relação de precedência entre lotes em uma linha, garantindo que a produção de um lote só pode iniciar após o término da execução do lote anterior e a execução da preparação entre os tipos respectivos. Desta forma tem-se :

$$\begin{aligned} \text{II.a) } ISEQ_a[i] - ISEQ_a[i] - G * \Delta_a[i,j] &\geq \\ &\geq TF_a[i] + SETUP_a[TIPO[i], TIPO[j]] - G \\ & \quad i = 0 \dots I, \quad j = 1 \dots I \end{aligned}$$

$$\begin{aligned} \text{II.b) } ISEQ_b[j] - ISEQ_b[i] - G * \Delta_b[i,j] &\geq \\ &\geq TF_b[i] + SETUP_b[TIPO[i], TIPO[j]] - G \\ & \quad i = 0 \dots I, \quad j = 1 \dots I \\ & \quad AT[TIPO[i], b] = 1, \quad AT[TIPO[j], b] = 1 \end{aligned}$$

$$\begin{aligned} \text{II.c) } ISEQ_c[j] - ISEQ_c[i] - G * \Delta_c[i,j] &\geq \\ &\geq TF_c[i] + SETUP_c[TIPO[i], TIPO[j]] - G \\ & \quad i = 0 \dots I, \quad j = 1 \dots I \\ & \quad AT[TIPO[i], c] = 1, \quad AT[TIPO[j], c] = 1 \end{aligned}$$

Observamos que as restrições acima só são ativadas quando $\Delta_a[i,j] = 1$, ou seja, quando o lote i precede imediatamente o lote j em questão.

Devemos notar também que a garantia de sucesso desta condição Delta de precedência imediata se deve à cadeia de precedências que se forma no sequenciamento. Por exemplo, o lote i precede imediatamente um lote k que, por sua vez, precede imediatamente um lote j . Apesar de $\Delta_a[i,j] = 0$, tem-se $\Delta_a[i,k] = 1$ e $\Delta_a[k,j] = 1$, que garante o sucesso do raciocínio empregado.

3.4.3.3. Restrição classe III - Controle de estoques

Esta classe de restrições tem o objetivo de garantir a manutenção dos estoques em processo abaixo dos patamares máximos estabelecidos ao final de cada linha.

$$\text{III.a) } \sum_{i=1}^I (XN[i]*(Beta_a[i,j]-Alfa_a[i,j])) + (PE[j]-ISEQ_a[n])*PR_a[TIPO[n]] - (1-Gama_a[n,j])*G \leq EMA_a[TIPO[j]] - EMI_a[TIPO[j]]$$

$$n, j = 1..I, \\ TIPO[i]=TIPO[j]=TIPO[n]$$

$$\text{III.b) } \sum_{i=1}^I (XN[i]*(Beta_a[i,j]-Alfa_a[i,j])) - Gama_a[n,j]*G \leq EMA_a[TIPO[j]] - EMI_a[TIPO[j]]$$

$$n, j=1..I, \\ TIPO[i]=TIPO[j]=TIPO[n]$$

$$\text{III.c) } \sum_{i=1}^I (XN[i]*(Beta_b[i,j]-Alfa_b[i,j])) + (ISEQ_a[j]-ISEQ_b[n])*PR_b[TIPO[j]] - (1-Gama_b[n,j])*G \leq EMA_b[TIPO[j]] - EMI_b[TIPO[j]]$$

$$n, j=1..I, \\ TIPO[i]=TIPO[j]=TIPO[n]=K \\ AT[K,B]=1$$

$$\text{III.d) } \sum_{i=1}^I (XN[i]*(Beta_b[i,j]-Alfa_b[i,j])) - Gama_b[n,j]*G \leq EMA_b[TIPO[j]] - EMI_b[TIPO[j]]$$

$$n, j=1..I, \\ TIPO[i]=TIPO[j]=TIPO[n]=K \\ AT[K,B]=1$$

$$\text{III.e) } \sum_{i=1}^I (XN[i]*(Beta_c[i,j]-Alfa_c[i,j])) + (ISEQ_a[j]-ISEQ_c[n])*PR_c[TIPO[j]] - (1-Gama_c[n,j])*G \leq EMA_c[TIPO[j]] - EMI_c[TIPO[j]]$$

$$n, j=1..I, \\ TIPO[i]=TIPO[j]=TIPO[n]=K \\ AT[K,c]=1$$

$$\text{III.f) } \sum_{i=1}^I (XN[i]*(Beta_c[i,j]-Alfa_c[i,j])) - Gama_c[n,j]*G \leq EMA_c[TIPO[j]] - EMI_c[TIPO[j]]$$

$$n, j = 1..I, \\ TIPO[i]=TIPO[j]=TIPO[n]=K \\ AT[K,c]=1$$

As restrições desta classe representam o fato de que em um dado momento deve existir em estoque tudo o que foi produzido menos o que foi consumido até aquele instante.

Portanto, considera-se que em um instante que precede imediatamente um consumo, obtem-se um máximo local do volume de estoques. Considerando todos estes máximos locais ao longo do horizonte de programação, coloca-se o estoque sob controle, abaixo dos patamares máximos previamente estabelecidos.

Note que as restrições se apresentam na forma de dicotomias em função do valor de Gama. Explicando melhor, no exemplo, as restrições IIIa e IIIb são excludentes, ou seja, se $Gama_a[i,j] = 1$ a restrição IIIa fica ativa e a restrição IIIb fica desativada. O contrário ocorre se $Gama_a[i,j] = 0$. Isto se deve às eventuais necessidades de computarmos o quanto foi produzido de um dado lote i que está sendo processado durante um dado consumo j .

Note também que os valores de $Alfa_a[i,j]$ podem ser determinados a priori, uma vez que nós conhecemos com antecedência os valores de $PE[i]$ e $PE[j]$. Desta forma, $Alfa_a[i,j]$ não tem o caráter de variável, mas sim de valores obtidos de forma direta a partir das entradas do problema.

3.4.3.4. Restrição classe IV - Integridade de $Beta_a[i,j]$

Esta restrição tem a função de estabelecer um controle sobre a variável booleana $Beta_a[i,j]$, garantindo assim uma fidelidade dos valores obtidos de $Beta$ aos reais significados da variável.

$$\text{IV. a) } PE[j] - (ISEQ_a[i] + TF_a[i]) \leq G * Beta_a[i,j] \\ i, j = 1 \dots I, TIPO[i] = TIPO[j]$$

$$\text{IV. b) } PE[j] - (ISEQ_a[i] + TF_a[i]) > G * (Beta_a[i,j] - 1) \\ i, j = 1 \dots I, TIPO[i] = TIPO[j]$$

$$\text{IV. c) } ISEQ_a[j] - (ISEQ_b[i] + TF_b[i]) \leq G * Beta_b[i,j] \\ i, j = 1 \dots I, TIPO[i] = TIPO[j] = k, AT[K,b] = 1$$

$$\text{IV. d) } ISEQ_a[j] - (ISEQ_b[i] + TF_b[i]) > (Beta_b[i,j] - 1) * G \\ i, j = 1 \dots I, TIPO[i] = TIPO[j] = k, AT[K,b] = 1$$

$$\text{IV. e) } ISEQ_a[j] - (ISEQ_c[i] + TF_c[i]) \leq G * Beta_c[i,j] \\ i, j = 1 \dots I, TIPO[i] = TIPO[j] = k, AT[K,c] = 1$$

$$\text{IV. f) } ISEQ_a[j] - (ISEQ_c[i] + TF_c[i]) > (Beta_c[i,j] - 1) * G \\ i, j = 1 \dots I, TIPO[i] = TIPO[j] = k, AT[K,c] = 1$$

Assim, devemos garantir que, se o término da produção de um dado lote i se deu anteriormente ao consumo de um dado lote j na linha l , $Beta_l[i,j]$ deve ser igual a 1 e $Beta_l[i,j] = 0$ em caso contrário.

Verificamos, por exemplo em IVa e IVb, que se $Beta_a[i,j] = 1$, IVa ficará desativada, ao passo que em IVb, $PE[j] - (TSEQ_a[i] + TFA[i])$ necessariamente terá que ser maior que a zero. O contrário ocorrerá se $Beta_a[i,j] = 0$.

3.4.3.5. Restrição classe V - Integridade das variáveis $Alfa_b[i,j]$ e $Alfa_c[i,j]$

Esta restrição tem a função de garantir fidelidade aos valores obtidos em $Alfa_b[i,j]$ e $Alfa_c[i,j]$. Como já foi dito anteriormente, $Alfa_a[i,j]$ é um parâmetro estabelecido a priori, a sendo desnecessária

$$\text{V.a) } ISEQ_a[j] - ISEQ_a[i] \leq G * Alfa_b[i,j] \\ i, j = 1..I, TIPO[i] = TIPO[j] = K, AT[K,b] = 1$$

$$\text{V.b) } ISEQ_a[j] - ISEQ_a[i] > G * (Alfa_b[i,j] - 1) \\ i, j = 1..I, TIPO[i] = TIPO[j] = K, AT[K,b] = 1$$

$$\text{V.c) } ISEQ_a[j] - ISEQ_a[i] \leq G * Alfa_c[i,j] \\ i, j = 1..I, TIPO[i] = TIPO[j] = K, AT[K,c] = 1$$

$$\text{V.d) } ISEQ_a[j] - ISEQ_a[i] > G * (Alfa_c[i,j] - 1) \\ i, j = 1..I, TIPO[i] = TIPO[j] = K, AT[K,c] = 1$$

qualquer consistência sobre ele.

O que procura-se garantir aqui é que, em uma dada linha l , se o consumo de um lote i se der antes do consumo de um lote j , $Alfa_l[i,j] = 1$ e $Alfa_l[i,j] = 0$ em caso contrário. Devemos observar que $Alfa_l[i,j]$ não é definida para os casos em que $i = j$.

Pode-se aqui observar que o raciocínio dicotômico apresentado é igual ao utilizado na restrição IV.

3.4.3.6. Restrição classe VI - Integridade da variável $Gama_l[i,j]$

Esta restrição tem a função de garantir que exista no máximo um lote i sendo processado na linha l no instante de consumo do lote j . Sendo assim, tem-se :

$$\text{VI. a) } \sum_{i=1}^I \text{Gama}_a[i, j] \leq 1 \quad j=1..I, \text{ TIPO}[i]=\text{TIPO}[j]$$

$$\text{VI. b) } \sum_{i=1}^I \text{Gama}_b[i, j] \leq 1 \quad j=1..I, \text{ TIPO}[i]=\text{TIPO}[j]=K, \text{ AT}[K, b]=1$$

$$\text{VI. c) } \sum_{i=1}^I \text{Gama}_c[i, j] \leq 1 \quad j=1..I, \text{ TIPO}[i]=\text{TIPO}[j]=K, \text{ AT}[K, c]=1$$

3.4.3.7. Restrição classe VII - Integridade da variável Delta[i,j]

Esta restrição garante a integridade de Delta[i,j] pois não permite que um lote i possa ser precedido por um lote j que o sucede, ou seja, se um lote i precede um lote j, este não pode simultaneamente preceder o lote i.

$$\text{VII. a) } \text{Delta}_a[i, j] + \text{Delta}_a[j, i] \leq 1 \quad i=0..I, \quad j=1..I, \quad i \neq j$$

$$\begin{aligned} \text{VII. b) } \text{Delta}_b[i, j] + \text{Delta}_b[j, i] \leq 1 \quad & i=0..I, \quad j=1..I, \\ & \text{AT}[\text{TIPO}[i], b]=1 \\ & \text{AT}[\text{TIPO}[j], b]=1 \\ & i \neq j \end{aligned}$$

$$\begin{aligned} \text{VII. c) } \text{Delta}_c[i, j] + \text{Delta}_c[j, i] \leq 1 \quad & i=0..I, \quad j=1..I, \\ & \text{AT}[\text{TIPO}[i], c]=1 \\ & \text{AT}[\text{TIPO}[j], c]=1 \\ & i \neq j \end{aligned}$$

Observe que $\Delta_{a[i,j]}$ não é definida nos casos $i = j$. O lote $i = 0$ é um lote fictício com $XN[0] = 0$ que expressa o tipo de montagem em vigor na linha l no instante 0 do sequenciamento.

3.4.3.8. Restrição classe VIII - Integridade da variável $\Delta_{a[i,j]}$ (II)

Esta restrição tem a função de garantir que existe um, e somente um lote i que precede um dado lote k numa linha l de produção. Assim, tem-se :

$$\text{VIII.a)} \quad \sum_{i=0}^I \Delta_{a[i,j]} = 1 \quad j=1..I \quad i \neq j$$

$$\text{VIII.b)} \quad \sum_{i=0}^I \Delta_{b[i,j]} = 1 \quad j=1..I$$

$$\text{AT}[\text{TIPO}[i],b]=1$$

$$\text{AT}[\text{TIPO}[j],b]=1, \quad i \neq j$$

$$\text{VIII.c)} \quad \sum_{i=0}^I \Delta_{c[i,j]} = 1 \quad j=1..I$$

$$\text{AT}[\text{TIPO}[i],c]=1$$

$$\text{AT}[\text{TIPO}[j],c]=1, \quad i \neq j$$

3.4.3.9. Restrição classe IX - Integridade de $\Delta_{a[i,j]}$ (III)

Tem a função de garantir a existência de um lote inicial na sequência, ou seja, um lote que sucede o lote fictício $i = 0$ na previsão da programação. Tem-se, portanto :

$$\text{IX.a)} \quad \sum_{j=1}^I \Delta_{a[0,j]} = 1$$

$$\text{IX.b)} \quad \sum_{j=1}^I \Delta_{b[0,j]} = 1 \quad \text{AT}[\text{TIPO}[j],b]=1$$

$$\text{IX.c)} \quad \sum_{j=1}^I \Delta_{c[0,j]} = 1 \quad \text{AT}[\text{TIPO}[j],c]=1$$

3.4.3.10 Restrição classe X - Integridade de $\Delta_{i,j}$ (IV)

Tem a função de garantir que exista no máximo um lote j sucedendo um lote i na sequência de produção em uma linha. Tem-se:

$$\text{X.a) } \sum_{j=1}^I \Delta_{a,i,j} \leq 1 \quad i = 1..I, \quad i \neq j$$

$$\text{X.b) } \sum_{j=1}^I \Delta_{b,i,j} \leq 1 \quad i = 1..I \quad i \neq j$$

$$\text{AT[TIPO}[i],b]=1$$

$$\text{AT[TIPO}[j],b]=1$$

$$\text{X.c) } \sum_{j=1}^I \Delta_{c,i,j} \leq 1 \quad i = 1..I \quad i \neq j$$

$$\text{AT[TIPO}[i],c]=1$$

$$\text{AT[TIPO}[j],c]=1$$

Observação : Na revisão desta dissertação foi observado o fato de que as variáveis $\Delta_{i,j}$ e $\Delta_{i,j}$ na forma como foram utilizadas na restrição Classe III, poderiam ser substituídas por uma única variável booleana, uma vez que a expressão $(\Delta_{i,j} - \Delta_{i,j})$ só pode resultar em valores iguais a 0 ou 1.

3.5. RESUMO

Apresentou-se neste capítulo os modelos desenvolvidos para uma abordagem heurística e matemática do problema formulado.

Foram apresentadas as entradas comuns às duas abordagens, o tratamento dos dados necessário para a resolução, o algoritmo heurístico e o modelo matemático desenvolvido para o problema da célula piloto.

No próximo capítulo serão apresentadas as formas como os modelos foram implementados, os resultados computacionais obtidos através de alguns exemplos, e uma análise comparativa entre os métodos.

4. Capítulo 4 - Implementação dos Modelos e Resultados

4.1. INTRODUÇÃO

São apresentados, neste capítulo, os sistemas computacionais desenvolvidos e implementados a partir dos modelos descritos anteriormente.

São considerados alguns exemplos de utilização definidos para a configuração da célula piloto. Finalmente, é efetuada uma análise comparativa entre os métodos empregados a partir dos resultados obtidos.

4.2. IMPLEMENTAÇÃO DO MODELO HEURÍSTICO

A implementação do método heurístico se deu ao longo do desenvolvimento do sistema SEQMAN.

Este sistema foi projetado, a nível de características operacionais, objetivando um funcionamento dedicado a um grupo pequeno de células de manufatura, programando uma célula de cada vez. A operação do sistema foi organizada para ser efetuada pelo próprio supervisor de produção, ou por um programador de produção no próprio ambiente produtivo.

O sistema SEQMAN permite configurar a célula a partir da especificação do fluxo de produção. Esta configuração é feita através da determinação do número existente de níveis de produção e da alocação das linhas dentro de cada nível.

Como características ambientais de implantação, definiu-se pela implantação em ambiente PCXT, em função da flexibilidade e baixo custo decorrentes, tendo sido desenvolvido um módulo (LE_MRP), específico para a Equipamentos Clark, que faz a aquisição de dados de demanda de cada célula a partir de arquivos gerados pelo sistema de planejamento da produção (PRGP II), o qual por sua vez é executado em ambiente IBM 4341.

O sistema SEQMAN, bem como o módulo LE_MRP, foi codificado em PASCAL. A configuração fornecida à Equipamentos Clark Ltda permite sequenciamento de até 85 lotes com até 16 tipos diferentes de peças em um horizonte de programação de até 9 dias. A célula pode ter até 10 linhas distribuídas em até 6 níveis.

Em termos de testes, o sistema já resolveu satisfatoriamente problemas de sequenciamento de 30 lotes de produção com 6 tipos diferentes de peças em uma célula de 5 linhas distribuídas em 2 níveis. Os resultados obtidos foram factíveis, uma vez que todos os resultados do SEQMAN podem ser posteriormente editados.

Em termos de problemas reais, dentro da configuração da célula piloto, foi efetuada uma simulação de 20 dias de produção, com dados reais, sendo elaborada uma programação dia a dia da produção através do software SEQMAN. Os resultados foram considerados satisfatórios pelo PCP da empresa ao serem comparados com a situação obtida pelo método manual convencional. A base de comparação foi a quantidade de peças acumuladas nos estoques intermediários. Em termos médios, o uso do SEQMAN propiciou uma diminuição teórica de aproximadamente 30% do volume de peças em estoques intermediários.

O sistema SEQMAN tem como função fornecer subsídios aos supervisores de produção das células de manufatura, auxiliando-os nas tomadas de decisões. Esta característica do tipo "computer-aided" deve ser ressaltada, uma vez que o sistema não pretende substituir o supervisor na tarefa de programação, mas sim, sugerir uma programação a partir da heurística desenvolvida, permitindo edições sobre os resultados obtidos. Isto é muito importante uma vez que, como já foi dito anteriormente, apenas as variáveis mais relevantes ao processo produtivo são levadas em consideração pelo sistema, ficando a cargo do supervisor a consideração das demais variáveis a nível de edição dos resultados obtidos via sistema.

O sistema permite armazenamento da solução editada em disco, possibilitando uma posterior reprogramação da produção, a partir do instante em que esta se fizer necessária.

O método heurístico também se caracteriza por sempre gerar um resultado, mesmo que este seja considerado infactível sob o ponto de vista do modelo matemático (por exemplo: atraso no prazo de entrega, rompimento dos volumes mínimos ou máximos de estoque, etc.).

No Manual do Usuário do SEQMAN (CTI, 1988) estão descritos os módulos existentes e as formas de operação do sistema de maneira detalhada. Os módulos do sistema SEQMAN são os seguintes :

- Edição dos Dados de Entrada
- Visualização dos Dados Existentes
- Layout da Célula
- Sequenciamento da Produção
- Solução Armazenada
- Visualização dos Outputs do Sistema
- Edição dos Resultados
- Reprogramação da Produção

4.3. IMPLEMENTAÇÃO DO MODELO MATEMÁTICO

A implementação do modelo desenvolvido foi efetuada em duas partes. A primeira trata da introdução de dados de entrada e da geração do modelo matemático a partir de uma interface homem-máquina, gerada em ambiente de microcomputador PCXT. A segunda trata da resolução matemática do modelo gerado, tendo sido desenvolvida em ambiente VAX 11/785.

4.3.1. Geração do Modelo Matemático

Optou-se pela geração automática do modelo a partir dos dados de parâmetros, demanda e estado da célula. Para isso, utilizamos uma interface homem-máquina já desenvolvida e utilizada (protótipo do sistema SEQMAN para a célula piloto da Equipamentos Clark Ltda), onde são efetuadas as entradas de dados referentes à célula e aos lotes que se deseja produzir.

A partir destes dados, são gerados automaticamente a função objetivo do problema, a matriz das restrições, os limitantes das restrições e as demais considerações levadas a efeito na modelagem do problema. Estas informações são armazenadas em arquivos sequenciais no PCXT, os quais são posteriormente transferidos para o VAX 785 através do programa KERMIT e em arquivos padronizados, que possibilitam uma visualização do problema modelado. A figura 4.1 apresenta alguns padrões de arquivos sequenciais utilizados.

O programa utilizado nesta fase de geração foi desenvolvido em PASCAL (*Borland Turbo Pascal 5.0*).

4.3.2. Resolução do Problema Matemático

Conforme já apresentado anteriormente, o modelo matemático gerado é de programação linear inteira mista, sendo que as variáveis inteiras, do tipo booleanas, representam as decisões de precedência (Δt_j) e as indicações para controle de estoques (Alfa_j , Beta_j , Gamma_j). Apenas os instantes de início de fabricação dos lotes nas linhas são expressos por variáveis reais (ISEQ_j).

Dentro deste contexto, o método selecionado para a resolução do problema foi o método *Branch and Bound*.

A primeira tentativa de resolução se deu através do programa LINDO (*Linus Schrage, The Scientific Press, 1985*), desenvolvido para microcomputadores compatíveis com o PCXT, que resolve problemas de programação inteira mista. Logo nos primeiros testes verificou-se que o tamanho do problema estava muito além da capacidade computacional característica dos PCXTs.

Partiu-se então para a resolução do problema em ambiente VAX 785, efetuando-se adaptações em um programa de *Branch and Bound* desenvolvido no Instituto de Automação do CTI para este ambiente.

Este programa, codificado em FORTRAN 77, tem como base o método *Simplex*, utilizando-se da rotina SPLP do pacote SLATEC (Hanson, R.J.; Hiebert, K.L. [SNA]).

Entretanto, para o problema em questão o SLATEC não apresentou bom desempenho, apresentando resultados infactíveis para exemplos onde, através da heurística, se chegou a soluções factíveis. Não se chegou a uma conclusão definitiva sobre a origem deste mau comportamento. O mais provável é que isto se deva ao fato da matriz dos coeficientes ser mal condicionada, criando situações de condicionamento numérico não contempladas pelo pacote. Da mesma forma, em testes realizados, o pacote MINOS v. 2.5 (Murtagh, B.A.; Saunders, M.A.) não se mostrou adequado.

O sistema de resolução do *Simplex* adotado finalmente foi o pacote H01ADF (NAG) que, apesar de ter um baixo desempenho relativo ao tempo computacional, apresentou confiabilidade suficiente na resolução dos problemas.

Além das alterações necessárias no programa de *Branch and Bound* para adequá-lo especificamente a este problema, foi efetuada uma alteração no critério para seleção da variável booleana a ser desmembrada, utilizando-se das características do modelo em questão.

Foram testados três critérios na escolha da variável inteira relaxada a ser fixada :

- a) fixação da variável que mais se aproxima do ponto médio, ou seja, do valor 0.5.
- b) fixação da variável que mais se aproxima do valor 0.0.
- c) fixação da variável que mais se aproxima do valor 1.0.

Convém observar que este teste não foi efetuado a nível de obtenção de uma garantia científica de melhor desempenho, tendo sido efetuado em apenas um exemplo para que decisões fossem tomadas, quanto ao procedimento a ser seguido ao longo dos testes. O critério selecionado e adotado foi o de fixação da variável inteira relaxada que mais se aproxima do valor 1.0. Acreditamos que seria interessante uma investigação futura mais pormenorizada quanto a este aspecto, relacionando o critério de escolha à performance da resolução do problema.

Uma outra consideração feita com o intuito de provocar uma aceleração na resolução, foi o estabelecimento, na geração do modelo, de um teste de possibilidade de rompimento de estoque para cada tipo de peça em cada nível da célula. Este teste consiste em verificar se a soma dos tamanhos dos lotes de cada tipo excede a diferença entre o volume mínimo e máximo admissível em cada nível. Caso isto não ocorra, não é necessária a geração das restrições de controle de estoques correspondentes. Neste caso, também não são geradas as variáveis Alfa_i, Beta_i e Gama_i, uma vez que estas são usadas exclusivamente com a finalidade de controle sobre o inventário, o mesmo ocorrendo com as restrições de garantia de integridade destas variáveis.

4.4. EXEMPLOS DE UTILIZAÇÃO DAS ABORDAGENS HEURÍSTICA E MATEMÁTICA.

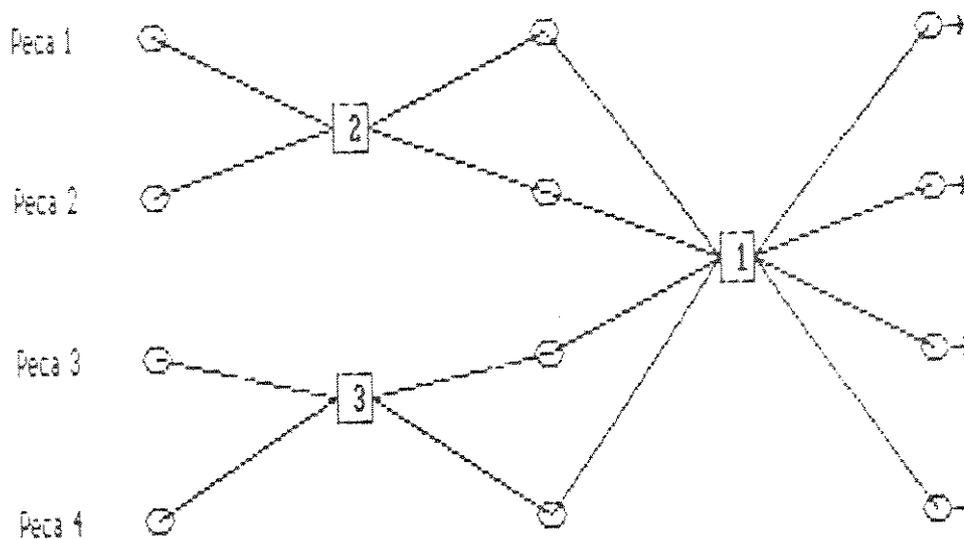
Esta seção apresenta os resultados obtidos a partir de exemplos de problemas, considerando sempre a configuração da célula piloto.

Ao longo destes exemplos foram mantidas as mesmas características de parâmetros da célula. Portanto, se mantiveram inalterados : a configuração de layout da célula, os fluxos de produção dos tipos de peças, os volumes mínimos e máximos de estoques em cada nível, as velocidades de produção e as matrizes dos tempos de preparação. Também se manteve inalterado o estado inicial da célula, ou seja, a montagem inicial e o volume inicial dos estoques no instante zero de programação.

O que efetivamente variou de um exemplo para outro foi a demanda de produção, isto é, o número, o tamanho e o tipo dos lotes a serem produzidos.

4.4.1. Descrição dos Parâmetros da Célula

.Configuração de layout



. Fluxo de produção

tipo	nivel 2	nivel 1
Peça 1	linha 2 →	linha 1
Peça 2	linha 2 →	linha 1
Peça 3	linha 3 →	linha 1
Peça 4	linha 3 →	linha 1

. Velocidades de produção (unidade : peças/hora)

tipo	nivel 2	nivel 1
1	37.50	50.00
2	25.00	50.00
3	25.00	25.00
4	25.00	37.50

. Estoques mínimos e máximos (unidade : peças)

tipo	nivel 2	nivel 1
1	0/400	0/400
2	0/400	0/400
3	0/400	0/400
4	0/400	0/400

. Matrizes de tempo de preparação (unidade : horas)

Nível 1

		tipo (peça)			
		1	2	3	4
tipo (peça)	para de				
	1	0	1.6	0.8	0.8
	2	0.8	0	1.6	0.8
	3	0.8	0.8	0	0.8
4	1.6	1.6	0.8	0	

Nível 2

		tipo (peça)			
		1	2	3	4
tipo (peça)	para de				
	1	0	3.2	—	—
	2	4.0	0	—	—
	3	—	—	0	4.8
4	—	—	3.2	0	

Obs. : No nível 2, as Peças 1 e 2 compartilham a linha 2 e as Peças 3 e 4 compartilham a linha 3 Não estão definidos os tempos de preparação entre peças que compartilham linhas diferentes.

4.4.2. Descrição do Estado Inicial

. Volume inicial dos estoques (unidade : peças)

tipo	nível 2	nível 1
1	0	0
2	0	0
3	0	0
4	0	0

. Montagem inicial no instante zero

Linha 1	→	Peça 4
Linha 2	→	Peça 1
Linha 3	→	Peça 4

4.4.3. Demandas de Produção Consideradas

São aqui apresentadas as demandas dos 7 exemplos que serão considerados. Cada lote de demanda é caracterizado por três parâmetros: tipo (Peça 1, Peça 2, etc.), tamanho do lote (número de peças) e prazo de entrega (unidade : dias)

	Exemplos	1	2	3	4	5	6	7
Lotes								
1	Tipo	1	1	1	1	1	1	1
	Tamanho	100	200	100	100	100	100	200
	Prazo	1.0	3.0	2.0	2.0	1.0	1.0	2.0
2	Tipo	4	2	2	2	2	1	2
	Tamanho	100	100	100	220	100	100	200
	Prazo	2.0	2.0	2.0	3.0	2.0	2.0	4.0
3	Tipo		4	4	4	4	2	3
	Tamanho		200	100	220	100	100	200
	Prazo		2.0	1.0	2.0	1.0	2.3	3.3
4	Tipo			4	4	4	4	4
	Tamanho			100	220	100	100	200
	Prazo			2.0	3.0	2.0	2.0	4.0
5	Tipo					4	4	4
	Tamanho					200	200	390
	Prazo					2.5	2.3	5.0

4.4.4. Resultados obtidos.

Esta seção apresenta os resultados heurísticos e matemáticos dos exemplos.

A saída obtida no modelo heurístico é tal como se apresenta no sistema SEQMAN, sendo que não foi efetuado qualquer tipo de edição sobre os resultados. Estes são aqui apresentados, para cada exemplo, na forma de lista de programação e gráfico de *Gantt*.

Nos exemplos em que o método heurístico provocou rompimento de patamares de estoques, são também apresentados os gráficos de evolução de estoques das peças onde isto ocorreu. A saída obtida no modelo matemático é organizada em termos dos valores específicos das variáveis que se encontravam na base na última interação do método *Simplex* da solução ótima inteira.

Para facilidade de entendimento, os resultados obtidos na resolução matemática são colocados na mesma forma adotada na resolução heurística. Para gerar esta saída, nós nos utilizamos do módulo de edição dos resultados do sistema SEQMAN.

Resultado Matemático

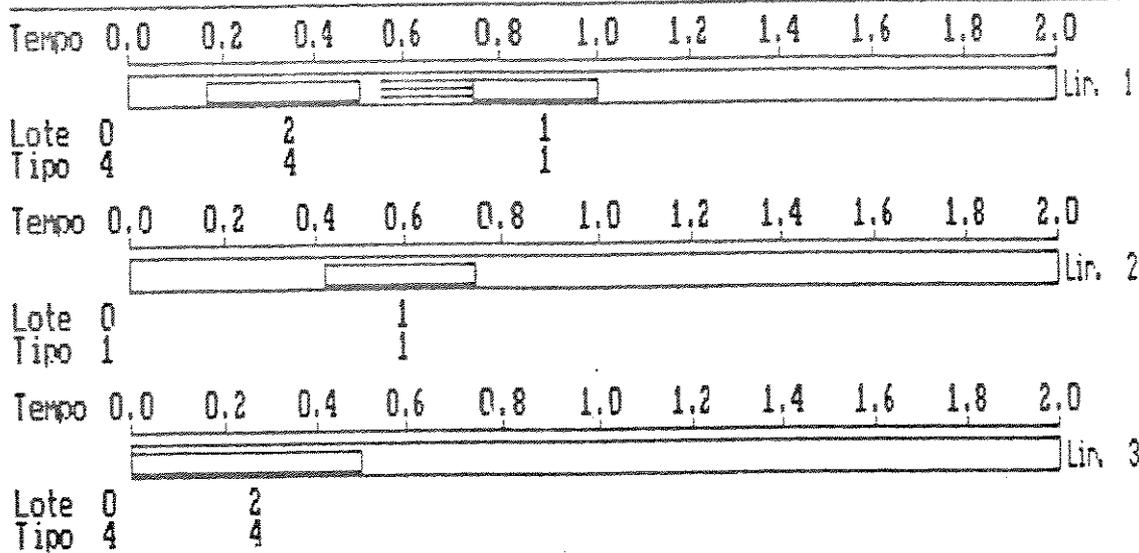
Lista de Programação

Nivel 1		Linha 1		Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	2	Peca 4	100	0.00	0.17	0.50	2.00
2	1	Peca 1	100	0.20	0.75	1.00	1.00

Nivel 2		Linha 2		Set Up inicial Peca 1			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.00	0.42	0.75	1.00

Nivel 2		Linha 3		Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	2	Peca 4	100	0.00	0.00	0.50	2.00

Análise via Gráfico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	
	Esc volta menu anterior

4.4.4.2. Exemplo 2.

Resultado Heurístico

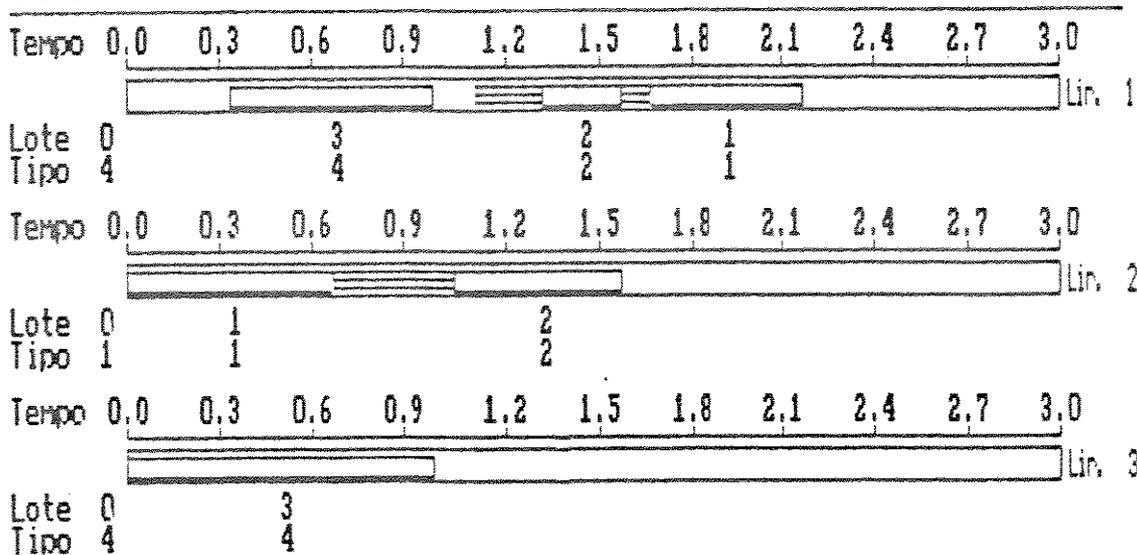
Lista de Programacao

		Nivel 1		Linha 1		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	200	0.00	0.33	1.00	2.00
2	2	Peca 2	100	0.20	1.32	1.57	2.00
3	1	Peca 1	200	0.10	1.67	2.17	3.00

		Nivel 2		Linha 2		Set Up inicial Peca 1	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	200	0.00	0.00	0.67	3.00
2	2	Peca 2	100	0.40	1.07	1.57	2.00

		Nivel 2		Linha 3		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	200	0.00	0.00	1.00	2.00

Analise via Grafico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	Esc volta menu anterior

Resultado Matemático

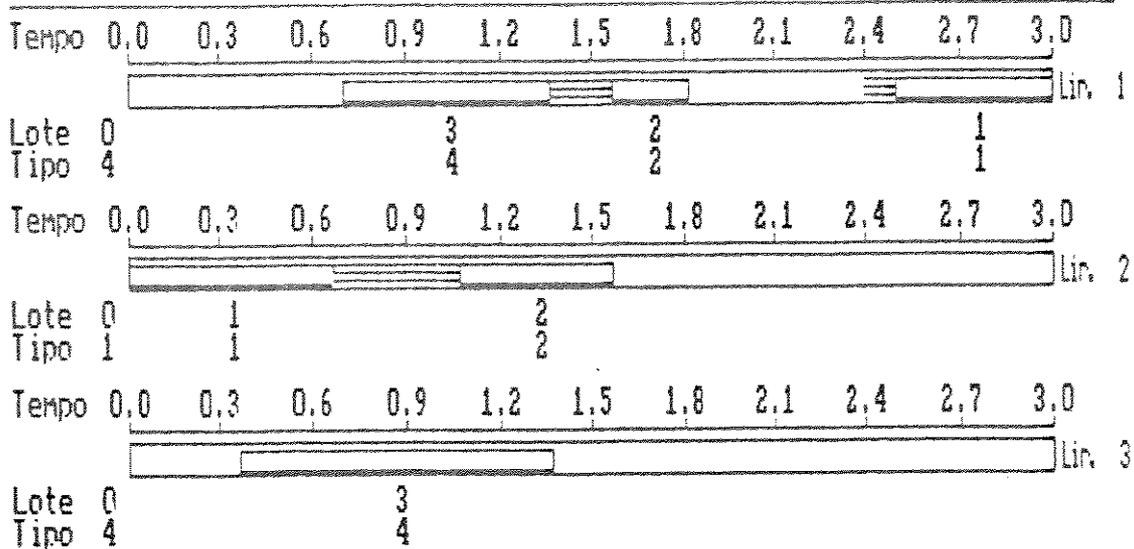
Lista de Programacao

Nivel 1		Linha 1		Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	200	0.00	0.70	1.37	2.00
2	2	Peca 2	100	0.20	1.57	1.82	2.00
3	1	Peca 1	200	0.10	2.50	3.00	3.00

Nivel 2		Linha 2		Set Up inicial Peca 1			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	200	0.00	0.00	0.67	3.00
2	2	Peca 2	100	0.40	1.07	1.57	2.00

Nivel 2		Linha 3		Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	200	0.00	0.37	1.37	2.00

Analise via Grafico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	Esc volta menu anterior

4.4.4.3. Exemplo 3.

Resultado Heurístico

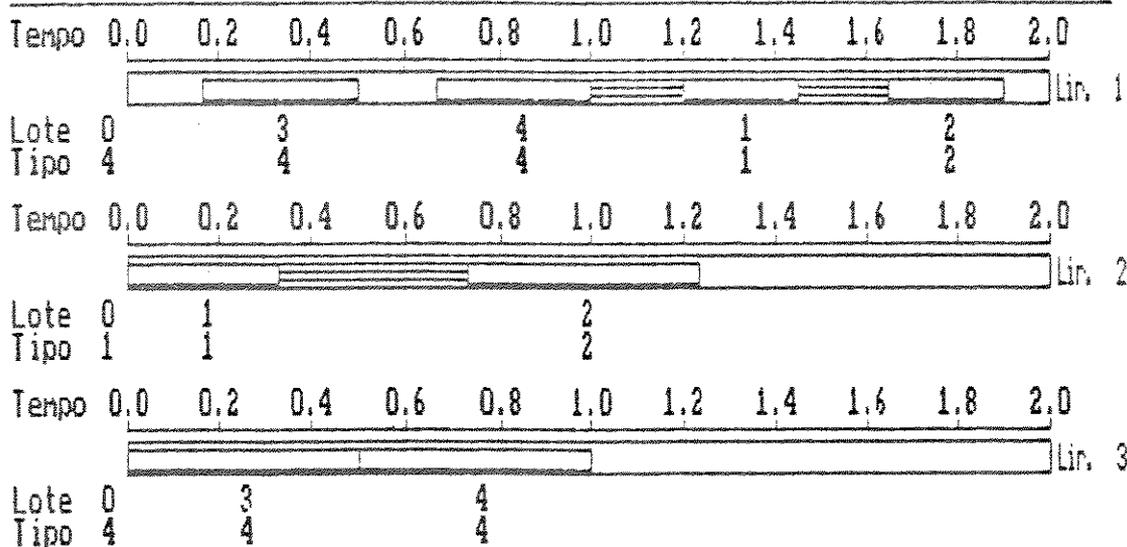
Lista de Programacao

		Nivel : 1	Linha : 1	Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	100	0.00	0.17	0.50	1.00
2	4	Peca 4	100	0.00	0.67	1.00	2.00
3	1	Peca 1	100	0.20	1.20	1.40	2.00
4	2	Peca 2	100	0.20	1.65	1.90	2.00

		Nivel : 2	Linha : 2	Set Up inicial Peca 1			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.00	0.00	0.33	2.00
2	2	Peca 2	100	0.40	0.73	1.23	2.00

		Nivel : 2	Linha : 3	Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	100	0.00	0.00	0.50	1.00
2	4	Peca 4	100	0.00	0.50	1.00	2.00

Analise via Grafico de Gantt



Relacao tipo -> codigo		↑ - ↓ scroll
1-Peca 1		
2-Peca 2		
3-Peca 3		
4-Peca 4		
		Esc volta menu anterior

Resultado Matemático

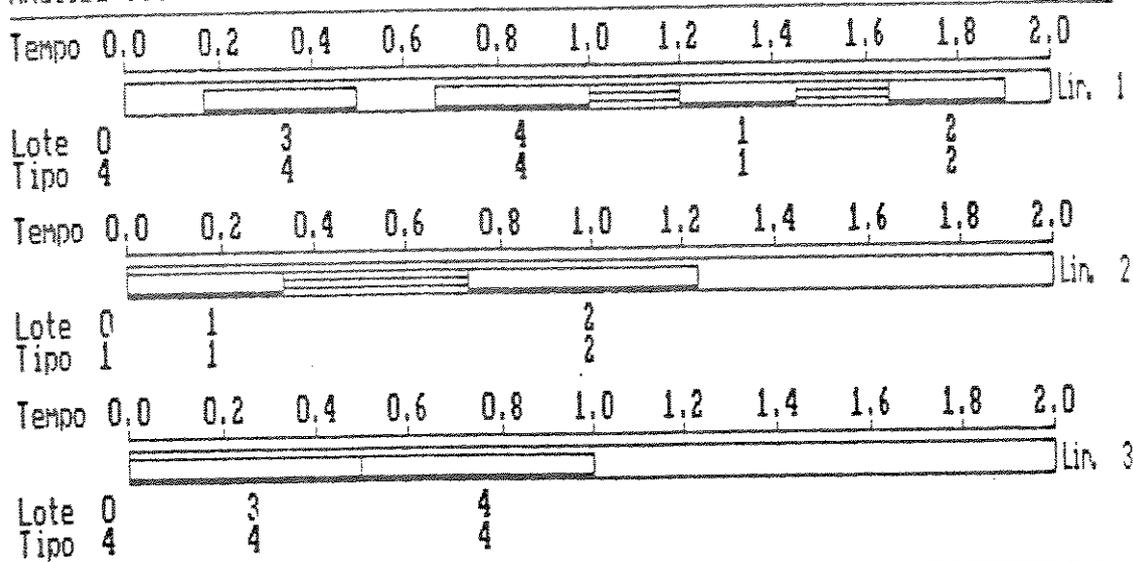
***** Lista de Programacao *****

Pos. Lote		Nivel 1	Linha 1	Set Up inicial	Peca 4		Prazo
		Tipo	Tamanho	Set Up	Inicio	Fim	
1	3	Peca 4	100	0.00	0.17	0.50	1.00
2	4	Peca 4	100	0.00	0.67	1.00	2.00
3	1	Peca 1	100	0.20	1.20	1.45	2.00
4	2	Peca 2	100	0.20	1.65	1.90	2.00

Pos. Lote		Nivel 2	Linha 2	Set Up inicial	Peca 1		Prazo
		Tipo	Tamanho	Set Up	Inicio	Fim	
1	1	Peca 1	100	0.00	0.00	0.33	2.00
2	2	Peca 2	100	0.40	0.73	1.23	2.00

Pos. Lote		Nivel 2	Linha 3	Set Up inicial	Peca 4		Prazo
		Tipo	Tamanho	Set Up	Inicio	Fim	
1	3	Peca 4	100	0.00	0.00	0.50	1.00
2	4	Peca 4	100	0.00	0.50	1.00	2.00

Analise via Grafico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	Esc volta menu anterior

4.4.4.4. Exemplo 4.

Resultado Heurístico

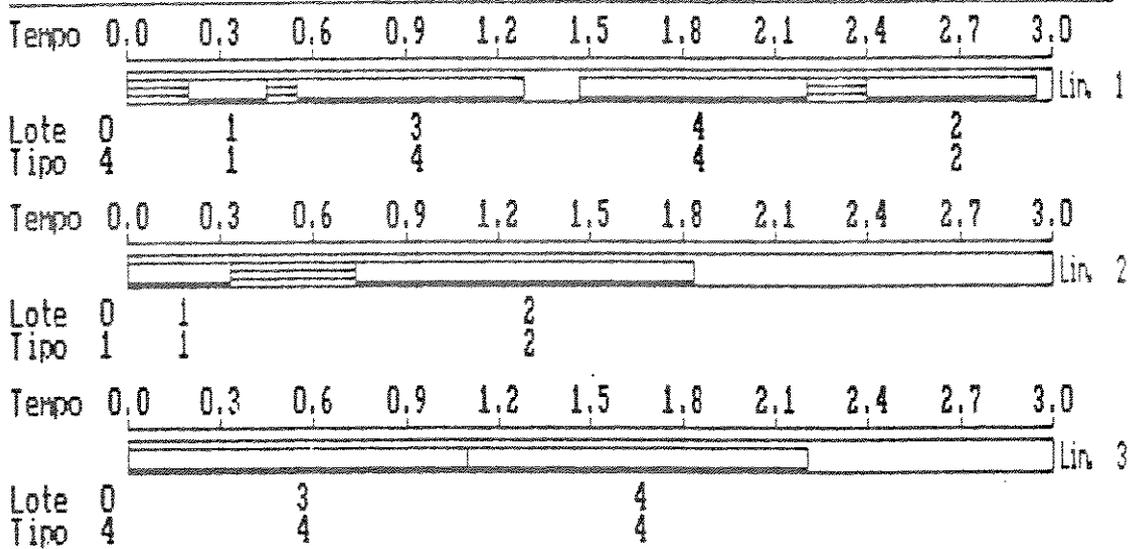
Lista de Programacao

		Nivel 1		Linha 1		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.20	0.20	0.45	2.00
2	3	Peca 4	220	0.10	0.55	1.26	2.00
3	4	Peca 4	220	0.00	1.47	2.20	3.00
4	2	Peca 2	220	0.20	2.40	2.95	3.00

		Nivel 2		Linha 2		Set Up inicial Peca 1	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.00	0.00	0.33	2.00
2	2	Peca 2	220	0.40	0.70	1.63	3.00

		Nivel 2		Linha 3		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	220	0.00	0.00	1.10	2.00
2	4	Peca 4	220	0.00	1.10	2.20	3.00

Analise via Grafico de Gantt



Relacao tipo -> codigo		↑ - ↓	scroll
1-Peca 1			
2-Peca 2			
3-Peca 3			
4-Peca 4			
		Esc	volta menu anterior

Resultado Matemático

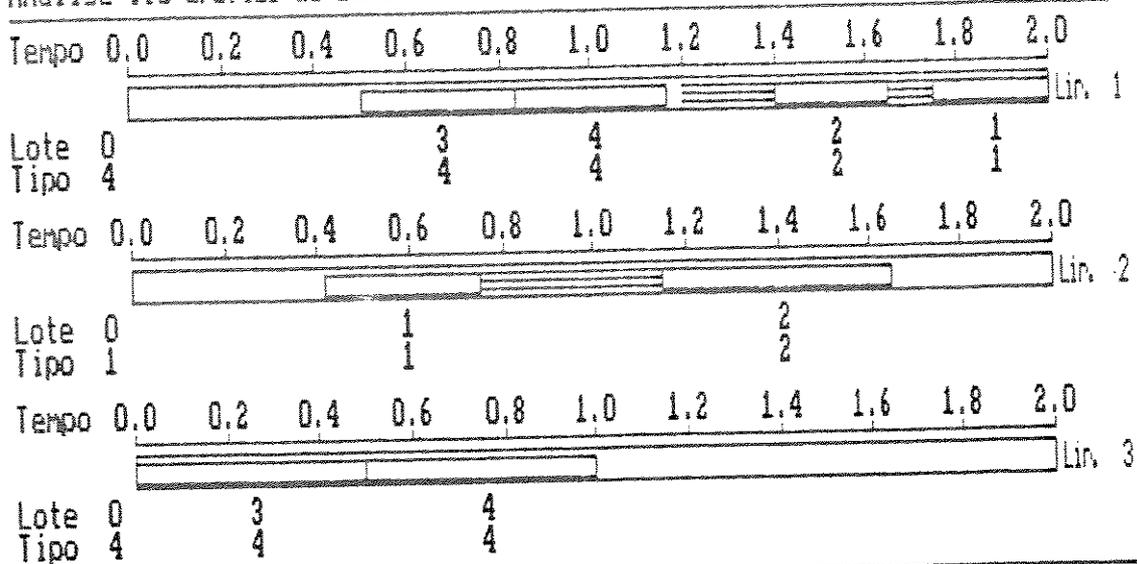
Lista de Programacao

		Nivel 1		Linha 1		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	100	0.00	0.50	0.83	1.00
2	4	Peca 4	100	0.00	0.83	1.17	2.00
3	2	Peca 2	100	0.20	1.40	1.65	2.00
4	1	Peca 1	100	0.10	1.75	2.00	2.00

		Nivel 2		Linha 2		Set Up inicial Peca 1	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.00	0.42	0.75	2.00
2	2	Peca 2	100	0.40	1.15	1.65	2.00

		Nivel 2		Linha 3		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	100	0.00	0.00	0.50	1.00
2	4	Peca 4	100	0.00	0.50	1.00	2.00

Analise via Grafico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	
	Esc volta menu anterior

4.4.4.5. Exemplo 5.

Resultado Heurístico

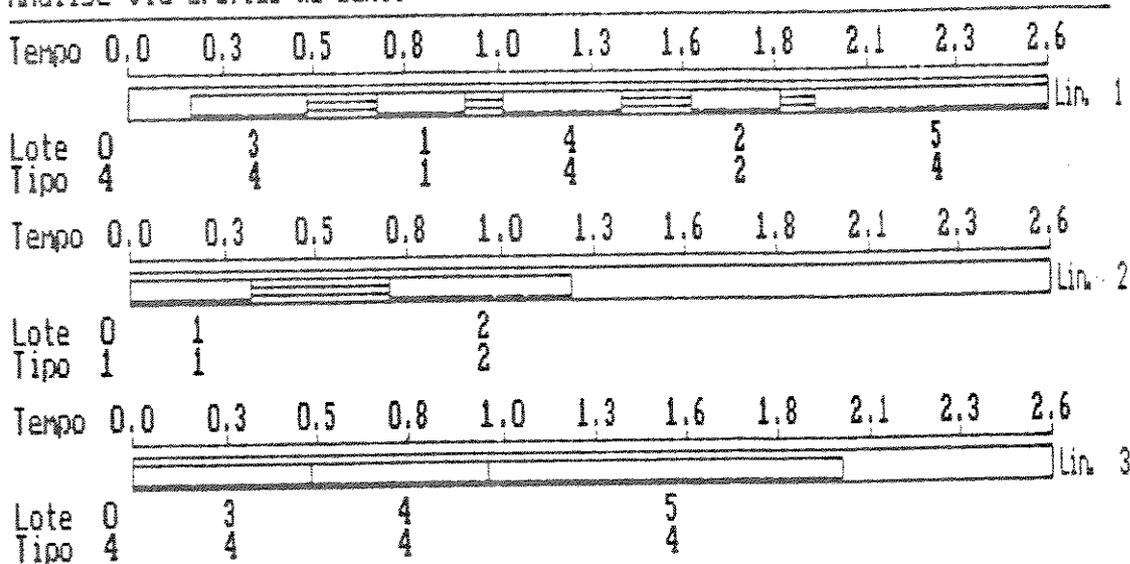
***** Lista de Programacao *****

Pos. Lote		Nivel 1	Linha 1.	Set Up inicial	Peca 4	Set Up	Inicio	Fim	Prazo
		Tipo	Tamanho						
1	3	Peca 4	100	0.00		0.17	0.50	1.00	
2	1	Peca 1	100	0.20		0.70	0.95	1.00	
3	4	Peca 4	100	0.10		1.05	1.30	2.00	
4	2	Peca 2	100	0.20		1.50	1.80	2.00	
5	5	Peca 4	200	0.10		1.90	2.60	2.50*	

Pos. Lote		Nivel 2	Linha 2	Set Up inicial	Peca 1	Set Up	Inicio	Fim	Prazo
		Tipo	Tamanho						
1	1	Peca 1	100	0.00		0.00	0.33	1.00	
2	2	Peca 2	100	0.40		0.73	1.23	2.00	

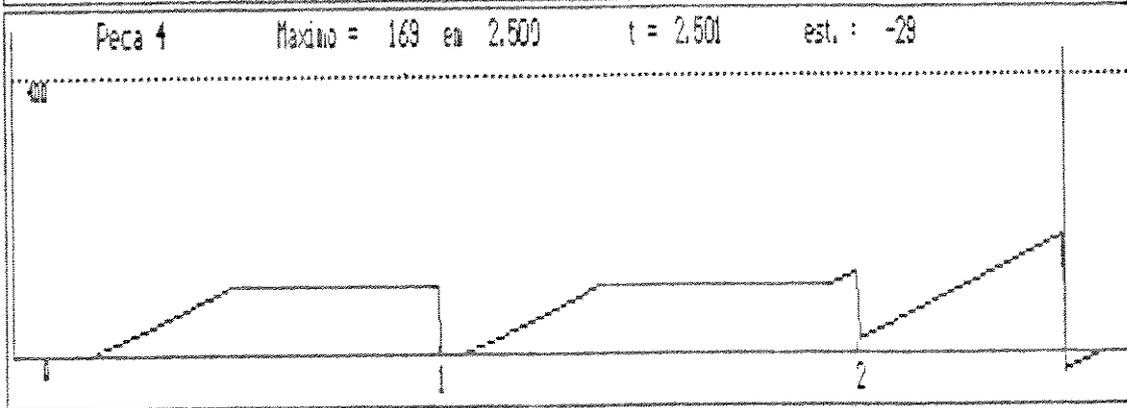
Pos. Lote		Nivel 2	Linha 3	Set Up inicial	Peca 4	Set Up	Inicio	Fim	Prazo
		Tipo	Tamanho						
1	3	Peca 4	100	0.00		0.00	0.50	1.00	
2	4	Peca 4	100	0.00		0.50	1.00	2.00	
3	5	Peca 4	200	0.00		1.00	2.00	2.50	

Analise via Grafico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	Esc volta menu anterior

	Evolução de estoques			Nível : 1
	0.0	1.0	2.0	
Peca 1	0	0	0	← - rápido
Peca 2	0	0	0	Ctrl ← - lento
Peca 3	0	0	0	F1 - volta
* Peca 4	0	0	19	



Resultado Matemático

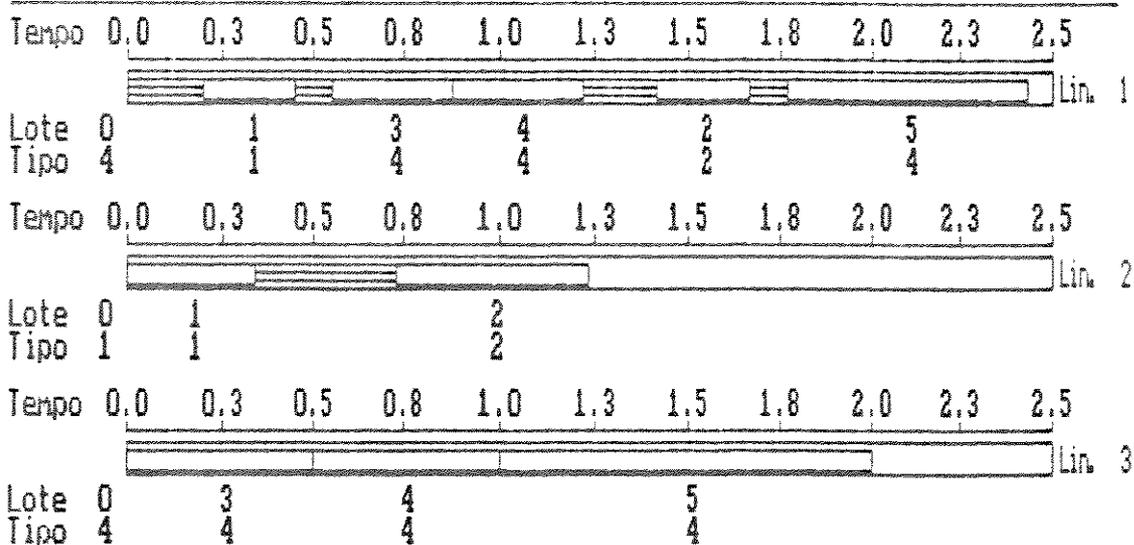
***** Lista de Programacao *****

		Nivel 1	Linha 1	Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.20	0.20	0.45	1.00
2	3	Peca 4	100	0.10	0.55	0.88	1.00
3	4	Peca 4	100	0.00	0.86	1.22	2.00
4	2	Peca 2	100	0.20	1.42	1.67	2.00
5	5	Peca 4	200	0.10	1.77	2.43	2.50

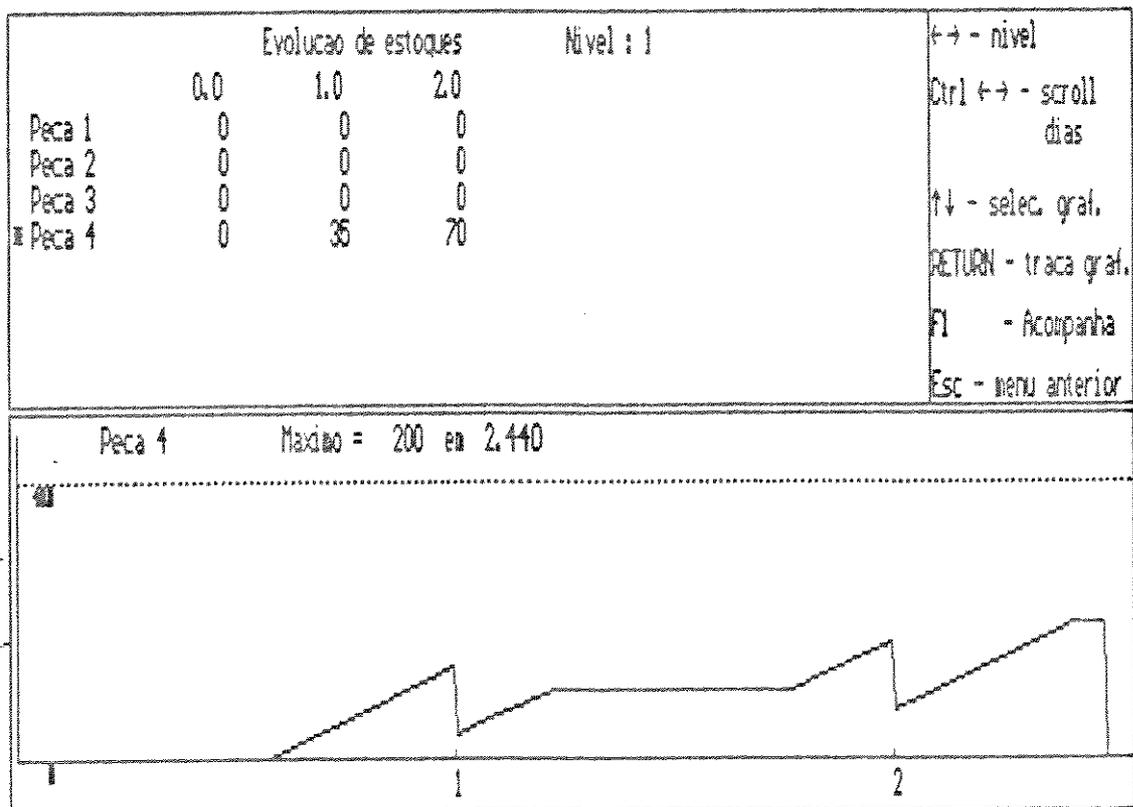
		Nivel 2	Linha 2	Set Up inicial Peca 1			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.00	0.00	0.33	1.00
2	2	Peca 2	100	0.40	0.73	1.23	2.00

		Nivel 2	Linha 3	Set Up inicial Peca 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	3	Peca 4	100	0.00	0.00	0.50	1.00
2	4	Peca 4	100	0.00	0.50	1.00	2.00
3	5	Peca 4	200	0.00	1.00	2.00	2.50

Analise via Grafico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	Esc volta menu anterior



4.6. Exemplo 6

Resultado Heurístico

***** Lista de Programacao *****

Nivel 1 Linha 1 Set Up inicial Peca 4

Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	4	Peca 4	100	0.00	0.17	0.50	2.00
2	1	Peca 1	100	0.20	0.70	0.95	1.00
3	2	Peca 1	100	0.00	0.95	1.20	2.00
4	3	Peca 2	100	0.20	1.40	1.65	2.30
5	5	Peca 4	200	0.10	1.75	2.42	2.30*

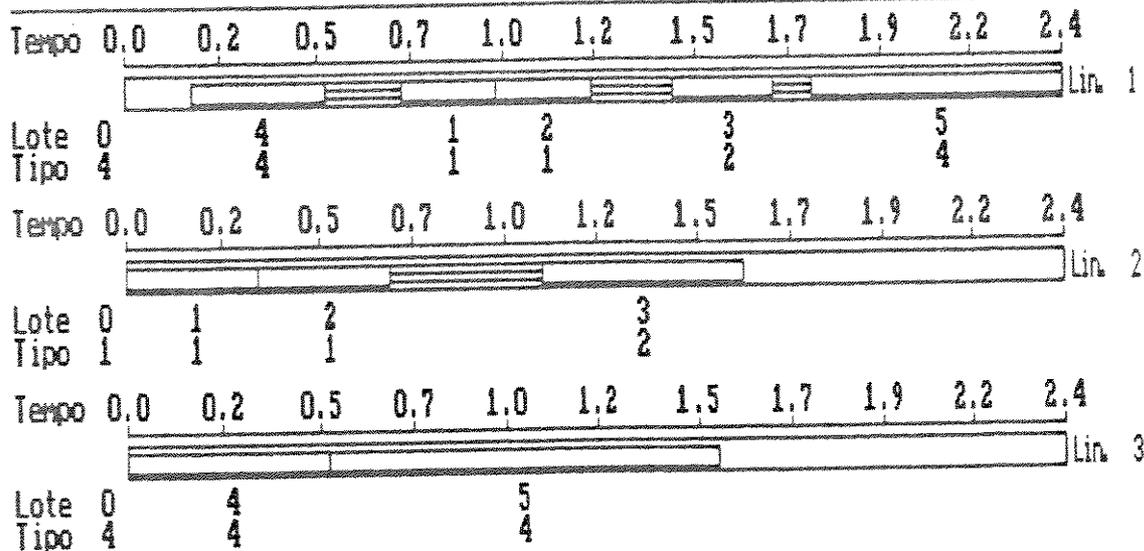
Nivel 2 Linha 2 Set Up inicial Peca 1

Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	1	Peca 1	100	0.00	0.00	0.33	1.00
2	2	Peca 1	100	0.00	0.33	0.67	2.00
3	3	Peca 2	100	0.40	1.07	1.57	2.30

Nivel 2 Linha 3 Set Up inicial Peca 4

Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fim	Prazo
1	4	Peca 4	100	0.00	0.00	0.50	2.00
2	5	Peca 4	200	0.00	0.50	1.50	2.30

Análise via Grafico de Gantt

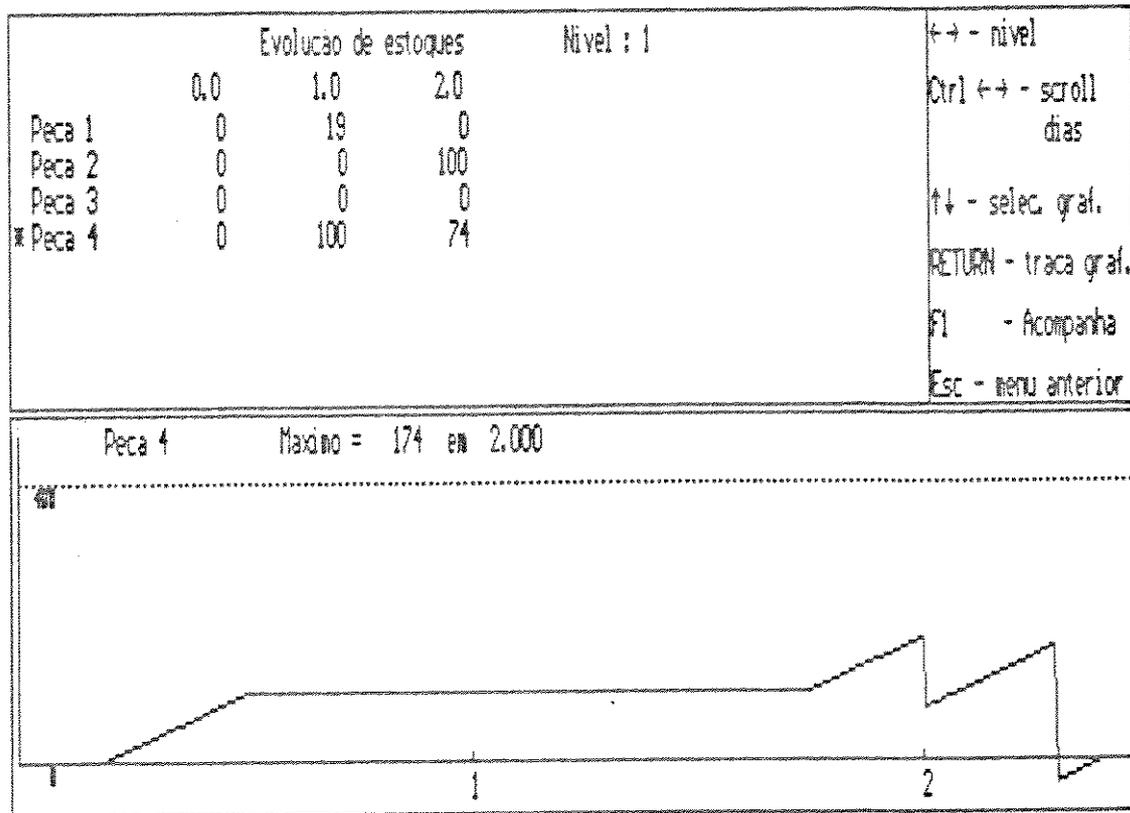


Relacao tipo -> codigo

1-Peca 1
2-Peca 2
3-Peca 3
4-Peca 4

↑ - ↓ scroll

Esc volta menu anterior



Resultado Matemático

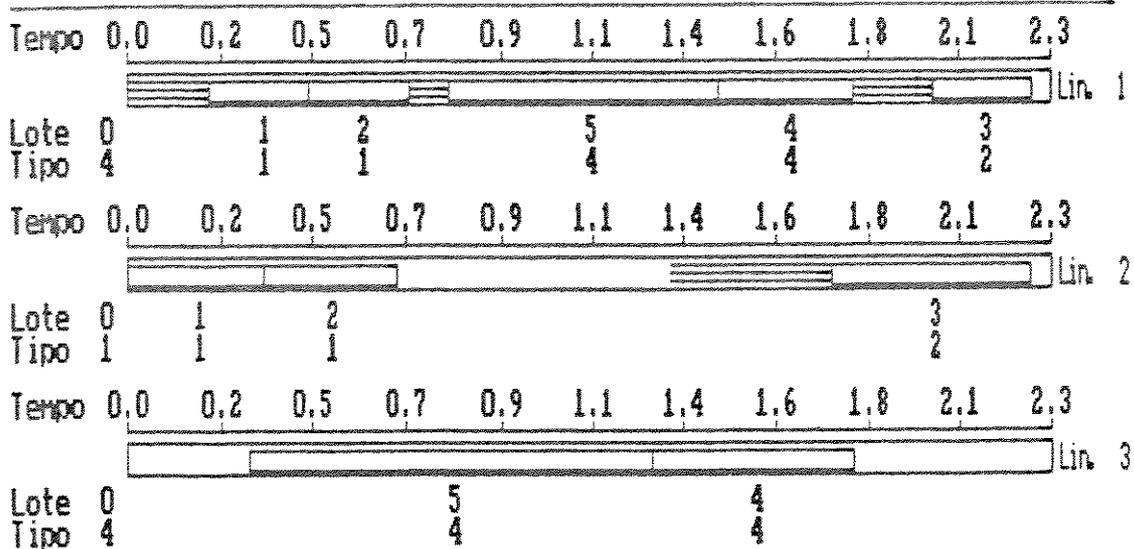
Lista de Programação

		Nível 1		Linha 1		Set Up inicial: Peça 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Início	Fim	Prazo
1	1	Peça 1	100	0.20	0.20	0.45	1.00
2	2	Peça 1	100	0.00	0.45	0.70	2.00
3	5	Peça 4	200	0.10	0.80	1.47	2.30
4	4	Peça 4	100	0.00	1.47	1.80	2.00
5	3	Peça 2	100	0.20	2.00	2.25	2.30

		Nível 2		Linha 2		Set Up inicial: Peça 1	
Pos.	Lote	Tipo	Tamanho	Set Up	Início	Fim	Prazo
1	1	Peça 1	100	0.00	0.00	0.33	1.00
2	2	Peça 1	100	0.00	0.33	0.67	2.00
3	3	Peça 2	100	0.40	1.75	2.25	2.30

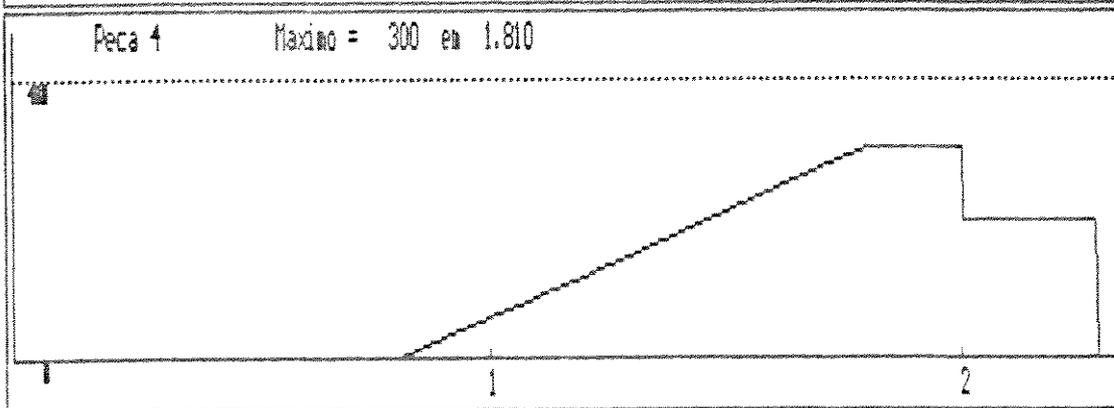
		Nível 2		Linha 3		Set Up inicial: Peça 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Início	Fim	Prazo
1	5	Peça 4	200	0.00	0.30	1.30	2.30
2	4	Peça 4	100	0.00	1.30	1.80	2.00

Análise via Gráfico de Gantt



Relação tipo -> código	↑ - ↓
1-Peça 1	
2-Peça 2	
3-Peça 3	
4-Peça 4	

Evolução de estoques				Nível : 1	← → - nível
	0.0	1.0	2.0		Ctrl ← → - scroll
Peca 1	0	100	0		dias
Peca 2	0	0	0		↑ ↓ - selec. graf.
Peca 3	0	0	0		RETURN - traca graf.
* Peca 4	0	59	200		F1 - Acompanha
					Esc - menu anterior



4.4.4.7. Exemplo 7.

Resultado Heurístico

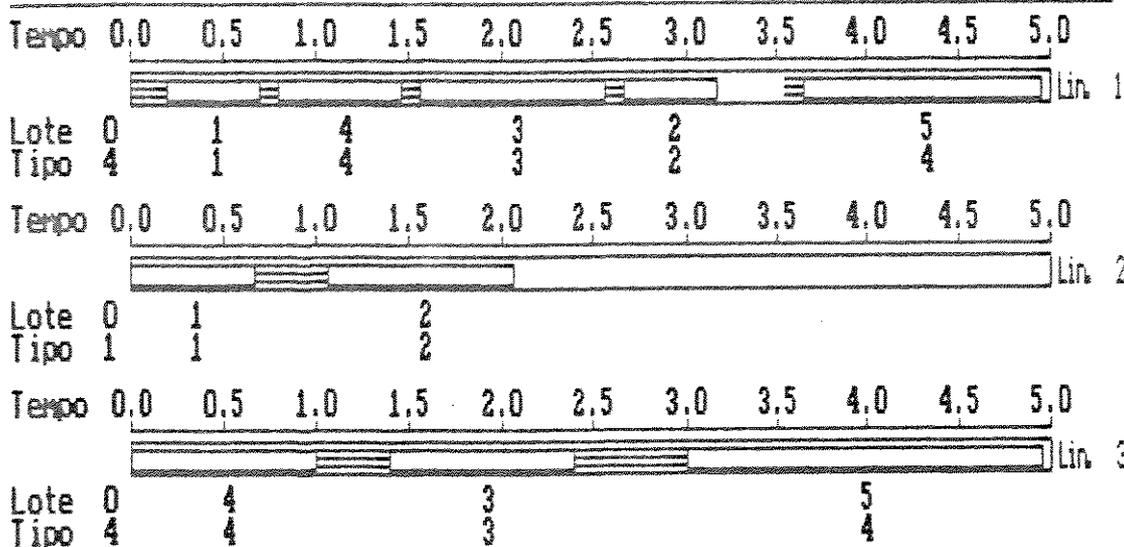
***** Lista de Programação *****

Nível 1		Linha 1		Set Up inicial: Peça 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Início	Fim	Prazo
1	1	Peça 1	200	0.20	0.20	0.70	2.00
2	4	Peça 4	200	0.10	0.80	1.47	4.00
3	3	Peça 3	200	0.10	1.57	2.57	3.30
4	2	Peça 2	200	0.10	2.67	3.17	4.00
5	5	Peça 4	390	0.10	3.65	4.95	5.00

Nível 2		Linha 2		Set Up inicial: Peça 1			
Pos.	Lote	Tipo	Tamanho	Set Up	Início	Fim	Prazo
1	1	Peça 1	200	0.00	0.00	0.67	2.00
2	2	Peça 2	200	0.40	1.07	2.07	4.00

Nível 2		Linha 3		Set Up inicial: Peça 4			
Pos.	Lote	Tipo	Tamanho	Set Up	Início	Fim	Prazo
1	4	Peça 4	200	0.00	0.00	1.00	4.00
2	3	Peça 3	200	0.40	1.40	2.40	3.30
3	5	Peça 4	390	0.60	3.00	4.95	5.00

Análise via Gráfico de Gantt



1-Peça 1 2-Peça 2 3-Peça 3 4-Peça 4	Relação tipo -> código ↑ - ↓ scroll Esc volta menu anterior
--	---

Resultado Matemático

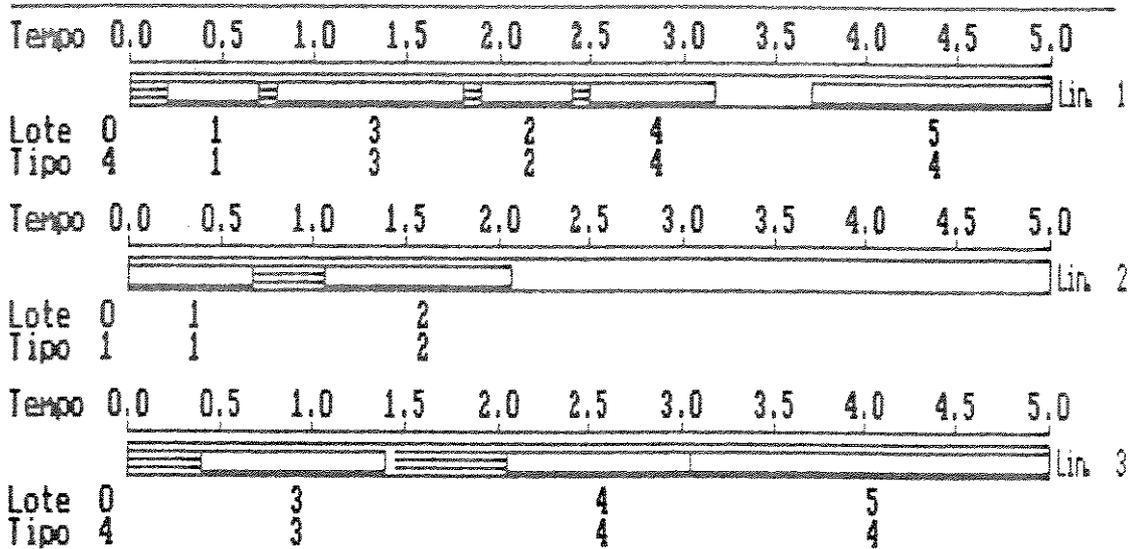
Lista de Programacao

		Nivel 1		Linha 1		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fin	Prazo
1	1	Peca 1	200	0.20	0.20	0.70	2.00
2	5	Peca 3	200	0.10	0.80	1.00	3.30
3	2	Peca 2	200	0.10	1.90	2.40	4.00
4	4	Peca 4	200	0.10	2.50	3.17	4.00
5	5	Peca 4	390	0.00	3.70	5.00	5.00

		Nivel 2		Linha 2		Set Up inicial Peca 1	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fin	Prazo
1	1	Peca 1	200	0.00	0.00	0.67	2.00
2	2	Peca 2	200	0.40	1.07	2.07	4.00

		Nivel 2		Linha 3		Set Up inicial Peca 4	
Pos.	Lote	Tipo	Tamanho	Set Up	Inicio	Fin	Prazo
1	3	Peca 3	200	0.40	0.40	1.40	0.30
2	4	Peca 4	200	0.60	2.00	3.00	4.00
3	5	Peca 4	390	0.00	3.00	5.00	5.00

Analise via Grafico de Gantt



Relacao tipo -> codigo	↑ - ↓ scroll
1-Peca 1	
2-Peca 2	
3-Peca 3	
4-Peca 4	
	Esc volta menu anterior

4.5. ANÁLISE COMPARATIVA ENTRE OS MÉTODOS

A idéia aqui não é promover uma comparação a nível de desempenho computacional, uma vez que a implementação do método matemático foi efetuada de maneira tradicional, sem maiores considerações sobre formas de acelerações da resolução do problema. O que foi aqui efetivamente considerado foi a capacidade de obtenção de uma boa solução pelo método heurístico, em relação à solução ótima gerada pelo modelo matemático. No entanto, são registrados os tempos computacionais obtidos, e é feita uma comparação em termos das soluções obtidas pelos métodos.

4.5.1. Desempenho computacional

Nos exemplos assinalados com * foi efetuado o teste prévio de controle de estoques mencionado anteriormente na seção 4.3.2.

Exemplo	Métodos	
	Heurístico	Matemático
1	< 1 seg.	42.36 seg.
2	< 1 seg.	742.33 seg.
3	< 1 seg.	1773.21 seg.
4	< 1 seg.	4000.72 seg.
5	aprox. 1 seg.	41743.62 seg.
6	aprox. 1 seg.	5685.92 seg. *
7	aprox. 1 seg.	35023.50 seg. *

4.5.2. Análise dos resultados

Os resultados observados na página seguinte dizem respeito aos valores da função objetivo. O desempenho médio do método heurístico encontrado nos exemplos é igual a **96.14%**

Nos exemplos 5 e 6, apesar da heurística conseguir desempenho de 100% em relação aos valores da função objetivo do modelo matemático, houve rompimentos na restrição de patamar mínimo de estoques, como pode ser observado nos diagramas de evolução de estoques correspondentes. Isto se deve ao fato de que a heurística empregada sempre chega a uma solução, mesmo que tenha que romper algumas restrições.

Exemplo	Métodos		%Desempenho Matemático
	Heurístico	Matemático	
1	0.2 dia	0.2 dia	100%
2	0.7 dia	0.7 dia	100%
3	0.7 dia	0.7 dia	100%
4	0.9 dia	0.7 dia	78%
5	1.0 dia	1.0 dia	100%
6	0.9 dia	0.9 dia	100%
7	2.0 dia	1.9 dia	95%

4.6. RESUMO

Foram apresentadas neste capítulo as formas adotadas na implementação dos modelos propostos em termos de ambientes computacionais, conteúdo e funcionamento dos sistemas. Foram apresentados também, alguns exemplos com a finalidade de aferição do desempenho e de uma comparação entre os métodos.

No próximo capítulo é elaborada a conclusão deste trabalho, analisando-se os resultados gerais obtidos, as implementações adicionais pretendidas para o método heurístico e algumas alternativas de aceleração e melhorias a serem efetuadas futuramente no modelo e no método de resolução matemático.

5. Capítulo 5 - Conclusão

5.1. INTRODUÇÃO

São apresentados neste capítulo uma avaliação dos resultados obtidos com este trabalho e algumas considerações quanto a evoluções futuras a serem feitas tanto no modelo heurístico quanto no modelo matemático.

5.2. AVALIAÇÃO DOS RESULTADOS GLOBAIS OBTIDOS

Uma vez não encontrada na bibliografia consultada uma solução que se adequasse à resolução do problema apresentado no capítulo 2, foi desenvolvida uma heurística que atendeu satisfatoriamente, como ferramenta de programação da produção a nível de chão de fábrica. Esta heurística posteriormente serviu de base para o desenvolvimento de um sistema computacional o qual, para validação final, vem sendo colocado às empresas cujos perfis de manufatura são adequados a este tipo de aplicação.

Em termos de aplicação, a abrangência da utilização do sistema desenvolvido não se mostrou restrita ao tamanho da empresa, mas sim à organização do sistema produtivo em estruturas celulares, devido à simplificação do fluxo produtivo decorrente deste tipo de estrutura.

A utilização prática do programa SEQMAN depende muito da qualidade da base de dados de processos da empresa e da qualidade do trabalho de transformação da estrutura convencional em estrutura celular. Este verdadeiro "saneamento básico" do sistema produtivo deve ser feito antes de se pensar em utilizar uma ferramenta computacional deste tipo. Não existe sentido em iniciar um processo de racionalização pelo problema de sequenciamento de produção. Se os dados introduzidos no sistema não tiverem crédito, a programação resultante também não terá.

Em paralelo ao desenvolvimento heurístico, foi desenvolvido um modelo matemático para validação da heurística em um caso específico, obtendo-se resultados para exemplos de pequeno porte. A implementação do modelo matemático se deu de forma convencional, a partir do método *Branch and Bound* conjuntamente com o método *Simplex* da programação linear. A preocupação neste caso era a validação a nível do valor da função objetivo, considerando-se fora do escopo deste trabalho uma análise mais profunda sobre metodologias de aceleração do processo de obtenção de resultados computacionais. Apesar disso, os tempos computacionais observados na seção 4.5.1. tornam proibitiva a sua utilização prática nos dias de hoje. Mesmo considerando as inúmeras possibilidades de aceleração do método *Branch and Bound* e as diversas configurações de arquiteturas computacionais conhecidas, a relação custo-benefício da programação matemática, nesta classe de problema, contraindica a sua utilização.

Focalizando os enfoques mais gerais desta dissertação, os resultados obtidos foram bastante positivos, uma vez que a proposição de se obter um método heurístico eficaz se cumpriu. Além disso, o desenvolvimento de um modelo matemático aplicado à validação do modelo heurístico confirmou o fraco desempenho computacional de ferramentas de programação matemática em utilizações práticas para este tipo de problema de sequenciamento de produção. Isto não invalida o desenvolvimento teórico desta linha de pesquisa.

5.3. TRABALHOS A SEREM DESENVOLVIDOS NO FUTURO

Em termos de continuação da linha de trabalho aqui considerada, está prevista uma série de complementações tanto na abordagem heurística quanto na matemática.

No que diz respeito à abordagem heurística de sequenciamento, algumas complementações a nível de modelagem poderão ser desenvolvidas e implementadas, visando permitir maior flexibilidade na sua utilização. Os principais pontos passíveis de ser incorporados são os seguintes:

Indicação de montagem ou submontagem : Permitirá ao sistema configurações de maior espectro, podendo o sistema no futuro, a partir da Lista de Materiais e do Plano Mestre de Produção do produto final, atuar como um módulo inteligente de MRP,

Indicação de fluxos alternativos de peças : Hoje o sistema atua com fluxo rígido, atribuindo uma etapa do processo de um determinado item a uma única linha de processamento. A idéia é criar opções de processo dando ao planejador a decisão sobre qual linha, em determinadas circunstâncias de carga, efetuará uma determinada etapa do processo,

Refluxo de peças : Atualmente um item não pode retornar a uma linha anterior (por onde ele já passou, em etapa anterior de seu fluxo), estando o fluxo orientado rigidamente. Esta é uma limitação em determinadas situações de utilização, existindo portanto, com menor prioridade, a preocupação em saná-la.

Todas estas evoluções a nível de modelagem de fluxo e configuração exigirão tecnologias mais sofisticadas que a atual para representação em rede do sistema produtivo.

Em função do desenvolvimento do sistema SEQMAN já existe um trabalho em paralelo em andamento no CTI, procurando desenvolver um módulo de sistema auxiliar que tratará estocasticamente variáveis de tempo de paralização de máquinas da célula e de demanda, no sentido de estabelecer tecnicamente os patamares mínimos e máximos de estocagem a serem adotados em cada nível da célula.

Outra implementação previsível é a realimentação das informações de produção via interface homem-máquina do sistema, permitindo atualização automática dos dados de estado do sistema produtivo. Com isso, é possível a atualização automática do sequenciamento da produção, tornando o sistema realmente *on line*.

A nível de modelagem matemática, o caminho óbvio é a generalização do modelo de forma a permitir uma configuração flexível da célula, tal como no sistema SEQMAN. Este caminho ainda não foi trilhado devido ao fato de que o sistema de representação do processo produtivo ainda deverá ser aperfeiçoado para o atendimento mais genérico do problema de célula de manufatura.

Uma melhoria possível a curto prazo, em termos de aproximação de enfoque do modelo matemático em relação ao SEQMAN, é a questão de factibilidade do problema. No SEQMAN é possível se obter soluções onde os patamares de estoques mínimos e máximos são rompidos, o

mesmo ocorrendo quanto aos prazos de entrega. Isto ocorre nas situações em que o sistema não encontra uma solução melhor via a heurística implementada.

Já no estado atual do modelo matemático, os patamares de estoque e os prazos são tratados a nível de restrições fortes, ou seja, não existe a possibilidade de atrasos dos prazos ou, por exemplo, o estoque exceder o patamar máximo estabelecido.

Uma alternativa relativamente simples de implementação seria a incorporação dos atrasos e dos rompimentos dos limites de estoques à função objetivo. Isto seria efetuado de forma ponderada, dando baixa prioridade à minimização de *setups*, média prioridade à manutenção dos níveis de estoques e alta prioridade à manutenção dos prazos de entrega.

Por exemplo, considerando a restrição Classe I (manutenção dos prazos de entrega e indiretamente do nível mínimo de estoque) representada pela expressão :

$$Ax \leq B$$

onde Ax representa a expressão da restrição e B representa o prazo em um determinado nível.

Se acrescentarmos a esta expressão a possibilidade de ocorrência de atrasos, ela poderia ser transformada em :

$$Ax - \text{Atraso} \leq B$$

$$\text{Atraso} \geq 0$$

Da mesma forma, a restrição Classe III (manutenção do estoque máximo) ora representada genericamente por :

$$A'x' \leq B'$$

acrescida da possibilidade de excesso de estoque em dado nível, poderia ser expressa na forma:

$$A'x' - \text{Excesso} \leq B'$$

$$\text{Excesso} \geq 0$$

Desta forma, a função objetivo, ora representada aqui por :

$$\text{f.o.} = \text{Min } Cy$$

poderia incorporar os atrasos e excessos de forma ponderada, de maneira a priorizar a não ocorrência de atrasos, em segundo lugar a não ocorrência de excessos e finalmente a minimização dos tempos de *setup*. Desta forma poderíamos ter, por exemplo :

$$\text{f.o.} = \text{Min } 1*Cy + 1000*\text{Excesso} + 1000000*\text{Atraso}$$

Como o objetivo é uma minimização, o sistema, antes de assumir uma solução com excessos ou atrasos diferentes de zero, deverá esgotar todas as demais possibilidades.

Com relação ao método em si, seria interessante um estudo profundo quanto a formas de aceleração do *Branch and Bound*, de forma a , em um futuro não distante, viabilizar a sua utilização prática nesta classe de problemas. De fato, com o aumento progressivo de capacidade e performance, e redução de custos dos equipamentos computacionais, certamente chegará o dia em que a programação matemática poderá ser utilizada como poderosa ferramenta de auxílio às tarefas de programação e controle do chão de fábrica. A inclusão de heurísticas apropriadas à abordagem matemática, juntamente com a possibilidade de exploração de arquiteturas paralelas na execução dos algoritmos, deverão viabilizar o uso de abordagens híbridas no tratamento de problemas de sequenciamento *on line*.

6. Referências Bibliográficas

BAKER, K. [1974]. *Introduction to Sequencing and Scheduling*. John Wiley & Sons, Inc. USA.

BLACK, J. T. [1986]. *Unburn Universitys Approach to Manufacturing Systems Engineering*. Autofact'86 Conference Proceedings, November, 12 - 14, Michigan, USA.

BURBIDGE, J. L. [1978]. *Planejamento e Controle da Produção*. Editora Atlas S. A., São Paulo, edição 1988.

CHOOBINEH, F. [1984]. *Optimum Loading for GT/MRP Manufacturing Systems*. Comput. & Indust. Engineering, vol. 8, no. 3/4.

GRAVES, S. [1981]. *A Review of Production Scheduling*. Operations Research, vol. 19, no. 4

HITOMI, K. ; HAM, I. [1977a]. *Group Scheduling Technique for Multiproduct, Multistage Manufacturing Systems*. Journal of Engineering for Industry, August.

HITOMI, K. ; HAM, I. [1977b]. *Machine Loading for Group Technology Applications*. Annals of the CIRP, vol. 25/1/1977.

HITOMI, K. [1979]. *Manufacturing Systems Engineering*. Taylor & Francis Ltd. UK.

LEE, J.L. ; MIRCHANDANI, P.B. [1986]. *Scheduling with Setups on a Two-Machine FMS*. Proceedings 1986 IEEE International Conference on Robotics and Automation, vol. 3/1, April.

MANNE, A.S. [1960]. *On the Jobshop Scheduling Problem*. Operations Research, vol. 8.

MOSIER, C.T. ; ELVERS, D.A. ; KELLY, D. [1984]. *Analysis of Group Technology Scheduling Heuristics*. International Journal of Production Research, vol. 12, no. 5.

NORBIS, M.I. ; SMITH, J.M. [1986]. *Two Level Heuristic for the Resource Constrained Scheduling Problem*. International Journal of Production Resource, vol. 24, no. 5.

OZDEN, M. ; EGBELU, P.J. ; IYER, A.V. [1985]. *Job Scheduling in a Group Technology Environment for a Single Facility*. Comput. & Indust. Engineering, vol. 9, no. 1.

POURBABAI, B. [1986]. *An Operation Model for Automated Batch Manufacturing Systems*. Proceedings 1986 IFAC Information Control Problems in Manufacturing Technology, Suzdal, USSR.

RENTES, A.F. ; SAWADA, F.S. ; SAKODA, M. ; CARVALHO, D.T. ; GIACHETTA, N. [1988]. *Sequenciamento da Produção em Células de Manufatura Baseadas em Tecnologia de Grupo*. Anais do 8o. Seminário de Comando Numérico no Brasil, SOBRACON, vol. 1.

RENTES, A.F.; SAKODA, M.; SAWADA, F. S. [1988]. *Manual do Usuário do Sistema SEQMAN*. CTI, Campinas - SP.

SÉRIO, L.C. [1988]. *Apostila do Curso "Tecnologia de Grupo no Planejamento de Sistemas Produtivos"*, SOBRACON, 1988.

TEIXEIRA, E. ; STANGE, P. ; FERNANDES, C.A. ; HIRAMA, M. ; ASANUMA, K. [1987]. *A Model for the Problem of Workpieces Sequencing in a Flexible Manufacturing Cell*. Proceedings of the Intern. AMSE Conference "Modelling & Simulation", Pomona, California, vol. 4.

YAMAMOTO, M. ; NOF, S.Y. [1985]. *Scheduling/Rescheduling in the Manufacturing Operating System Environment*. International Journal of Production Research, 23(4)1985.

7. Bibliografia

ADIRI, I.; AMIT, N. [1983]. Route-Dependent Open-Shop Scheduling. IIE Transactions, September.

ADSHEAD, N.S.; PRICE, H.R. [1986]. Experiments with Stock Control Policies and Leadtime Rules, Using an Aggregate Planning Evaluation Model of a Make-for-stock Shop. Int. J. Prod. Res., vol 24, no. 5.

ANDERSON, E.J. [1981]. A New Continuous Model for Job-Shop Scheduling. Int. J. Systems Sci., vol 12, no. 12.

BENSANA, E.; CORREGGE, M.; BEL, G.; DUBOIS, D. [1986]. An Expert System Approach to Industrial Job-Shop Scheduling.. IEEE International Conference on Robotics and Automation.

BERTRAND, J.W.M. [1986]. Balancing Production Level Variations and Inventory Variations in Complex Production Systems. Int. J. Prod. Res., vol 24, no. 5.

BUNNAG, P.; SMITH, S.B. [1985]. A Multifactor Priority Rule for Jobshop Scheduling Using Computer Search. IIE Transactions, June.

CHOO, Y.; SAHNI, S. [1981]. Preemptive Scheduling of Independent Jobs with Release and Due Times on Open, Flow and Job Shops. Operations Research, vol 29, no. 3, May-June.

DALLERY, Y. [1986]. A Queueing Network Model of Flexible Manufacturing Systems Consisting of Cells. IEEE International Conference on Robotics and Automation.

DANIL'CHENKO, A.M.; LEVCHENKO, S.N.; PANISHEV, A.V. [1985]. An Approximate Algorithm for the Three-Machine Problem. Automation and Remote Control, vol 46, no. 7.

DAY, J.E.; HOTTENSTEIN, M.P. [1975]. The Impact of Advancing Due Dates in a Pure Job Shop. Int. J. Prod. Res., vol 13, no. 6.

DODIN, B. [1985]. Scheduling N Products on a Single Facility with Allowed Backordering. Int. J. Prod. Res., vol 23, no. 2.

ERSCHLER, J.; FONTAN, G.; MERCE, C.; ROUBELLAT, F. [1982] Applying New Dominance Concepts to Job Scheduling Optimization. European Journal of Operational Research, vol 11, September.

FISHER, M.L.; LAGEWEG, B.J.; LENSTRA, J.K.; KAN, A.H.G.R. [1983]. Surrogate Duality Relaxation for Job Shop. Discrete Applied Mathematics, vol 5.

FISHER, M.L.; KRIEGER, A.M. [1984]. Analysis of a Linearization Heuristic for Single-Machine Scheduling to Maximize Profit. Mathematical Programming, vol 28.

- GAPP, W.; MANKEKAR, P.S.; MITTEN, L.G. [1965]. Sequencing Operations to Minimize In-Process Inventory Costs. *Management Science*, vol 11, no. 3, January.
- GENS, G.V.; LEVNER, E.V. [1981]. Fast Approximation Algorithm for Job Sequencing with Deadlines. *Discrete Applied Mathematics*, vol 3.
- GOODWIN, J.S.; WEEKS, J.K. [1986]. Evaluating Scheduling Policies in a Multi-Level Assembly System. *Int. Jour. Prod. Res.*, vol 24, no 2.
- HARIRI, A.M.A.; POTTS, C.N. [1983]. An Algorithm for Single Machine Sequencing with Release Dates to Minimize Total Weighted Completion Time. *Discrete Applied Mathematics*, vol 5.
- HUANG, P.Y.; MOORE, L.J.; RUSSELL, R.S. [1984]. Workload VS Scheduling Policies in a Dual-Resource Constrained Job Shop. *Comput. & Ops. Res.*, vol 11, no. 1.
- JOHRI, P.K. [1987]. A Linear Programming Approach to Capacity Estimation of Automated Production Lines with Finite Buffers. *Int. J. Prod. Res.*, vol 25, no. 6.
- KANET, J.J. [1981]. Minimizing the Average Deviation of Job Completion Times About a Common Due Date. *Naval Research Logistics Quarterly*, vol 28, no. 4.
- KIRAN, A.S.; SMITH, M.L. [1984]. Simulation Studies in Job Shop Scheduling II - Performance of Priority Rules. *Comput. & Indus. Engng.*, vol 8, no. 2.
- KUMARA, S.R.T.; JOSHI, S.; KASHIYAP, R.L.; MOODIE, C.L.; CHANG, T.C. [1986]. Expert Systems in Industrial Engineering. *Int. J. Prod. Res.*, vol 24, no. 5.
- LAMBRECHT, M.R.; VANDERVEKEN, H. [1979]. Heuristic Procedures for the Single Operation, Multi-Item Loading Problem. *AIIE Transactions*, December.
- LARSON, R.E.; DESSOUKY, M.I.; DEVOR, R.E. [1985]. A Forward- Backward Procedure for the Single Machine Problem to Minimize Maximum Lateness. *IIE Transactions*, September.
- LEBEER, C. [1985]. Priority Planning of a Dynamic Complex Product Mix: Elements for a Computerized Interface Between Delivery Times and Plant Utilization. *Europ. J. Operat. Res.*, vol 20.
- LEMOINE, A.J. [1986]. A Stochastic Network Formulation for Complex Sequential Processes. *Naval Research Logistics Quarterly*, vol 33.
- LEUNG, J.Y.T. [1982]. On Scheduling Independent Tasks with Restricted Execution Times. *Operations Research*, vol 30, no.1, January-February.

LEZIN, V.I.; NOVIKOV, S.G.; TAL', A.A., TVIL'NEVA, G.K. [1980]. Formalized Method for Intra-Factory On-Line Scheduling of Multi- Item Production. Automation and Remote Control, vol 41, no. 12, December.

LIN, K.S. [1983]. Hybrid Algorithm for Sequencing with Bicriteria. Journal of Optimization Theory and Applications, vol 39, no. 1, January.

LIU, C.Y.; BULFIN, R.L. [1985]. On Complexity of Preemptive Open-Shop Scheduling Problems. Operations Research Letters, vol 4, no. 2, July.

MAXWELL, W.L.; SINGH, H. [1983]. The Effect of Restricting Cycle Times in the Economic Lot Scheduling Problem. IIE Transactions, September.

NARASIMHAN, S.L.; PANWALKAR, S.S. [1984]. Scheduling in a Two- Stage Manufacturing Process. Intern. J. Operat. Res., vol 22, no. 4, July-August.

NELSON, R.T.; SARIN, R.K.; DANIELS, R.L. [1986]. Scheduling with Multiple Performance Measures: The One-Machine Case. Management Science, vol 32, no. 4, April.

NOWICKI, E.; ZDRSALKA, S. [1984]. Scheduling Jobs with Controllable Processing Times as an Optimal Control Problem. Int. J. Control, vol 39, no. 4.

PANWALKAR, S.S.; SMITH, M.L. [1982]. Common Due Date Assignment to Minimize Total Penalty for the One Machine Scheduling Problem. Operations Research, vol 30, no. 2, March-April.

POSNER, M.E. [1986]. A Sequencing Problem with Release Dates and Clustering Jobs. Management Science, vol 32, no. 6, June.

SCHUSSEL, G.; PRICE, S. [1970]. A Case History in Optimum Inventory Scheduling. Operations Research, January-February.

SELEN, W.J.; HOTT, D.D. [1986]. A Mixed-Integer Goal- Programming Formulation of the Standard Flow-Shop Scheduling Problem. J. Opl. Res. Soc., vol 37, no. 12.

SHIMOYASHIRO, S.; ISODA, K.; AWANE, H. [1984]. Input Scheduling and Load Balance Control for a Job Shop. Int. J. Prod. Res., vol 22, no. 4, July-August.

SZWARC, W. [1983]. The Flow-Shop Problem with Mean Completion Time Criterion. IIE Transactions, vol 15, no. 2.

WEMMERLOV, U. [1986]. A Time-Phased Order-Point in Environments with and without Demand Uncertainty: A Comparative Analysis of Non-Monetary Performance Variables. Int. J. Prod. Res., vol 24, no. 2.

YAO, D.D.; BUZACOTT, J.A. [1986]. Models of Flexible Manufacturing Systems with Limited Local Buffers. *Int. J. Prod. Res.*, vol 24, no. 1.