

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO

TRABALHO DE FORMATURA

**MODELO DE PROGRAMAÇÃO
DA PRODUÇÃO PARA INDÚSTRIAS
INTERMITENTES REPETITIVAS**

ANDRÉ MARCEL DRIZUL HAVRENNE

REINALDO PACHECO DA COSTA

1999

*FF-1999
H-299m*

Meus agradecimentos a todos aqueles que contribuíram de alguma forma na elaboração deste trabalho:

A meus pais, por terem se dedicado a mim e aos meus irmãos durante toda a nossa vida.

ao Prof. Reinaldo Pacheco, pela sua valiosa colaboração como orientador deste trabalho,

e a Prof. Débora Ronconi, pois sem seus conselhos e a sua orientação creio que este trabalho não poderia ser feito com o mesmo brilho e este perderia muito em qualidade.

Resumo

O objetivo deste trabalho é construir um modelo de programação da produção para indústrias intermitentes repetitivas. O trabalho detalha todas as características relevantes ao modelo proposto, tais como as premissas envolvidas, as restrições, além de critérios de avaliação do modelo.

Testamos o modelo em um estudo de caso real, numa manufatura de materiais de escritório, onde pudemos comprovar que o modelo gera uma programação sensivelmente melhor em relação à programação que é atualmente utilizada na empresa.

Índice

Resumo	3
Capítulo 1 – CARACTERIZAÇÃO DO PROBLEMA.....	6
1.1. Introdução	6
1.1.1. Apresentação e análise da fábrica escolhida.....	7
1.2. Descrição do problema	8
1.3. Fluxo de Informações para um pedido de compra	10
1.4. Fluxo de Materiais dentro da fábrica.....	11
1.5. Características do processo de maior relevância.....	13
Capítulo 2 – REVISÃO DA LITERATURA.....	14
2.1. Introdução	14
2.2 O “Flexible Flow Shop Problem”.....	14
2.3. Critérios de Avaliação.....	16
2.3. Hipóteses Simplificadoras.....	19
2.4. Descrição do método de busca.....	20
2.5 Heurísticas Construtivas	21
2.6. Balanço das heurísticas construtivas	24
2.7. Busca Tabu.....	25
2.8. Conclusões Teóricas	29
Capítulo 3 – DEFINIÇÃO DO MODELO	31
3.1. Introdução	31
3.2. Considerações gerais.....	31
3.3. Sequências consideradas.....	32
3.3.2. Sequência de prioridade de fabricação dos processos	34
3.3.3. Sequência de prioridade de fabricação dos produtos em carteira	36
3.4. Avaliação do desempenho das soluções	36
3.5. Busca de soluções.....	38
3.5.1. Solução Inicial.....	38
3.5.2. Busca Tabu.....	39
3.6. Hipóteses Simplificadoras.....	46
3.6.1. Hipóteses com relação as ordens.....	46
3.6.2. Hipóteses com relação aos tempos de processamento e outros atributos.....	47
3.6.3. Hipóteses sobre as máquinas.....	48
3.7. Tabelas criadas	50
3.7.1. Tabela da Carteira de Pedidos.....	50
3.7.2. Tabela de processos de fabricação.....	51
3.7.3. Tabela de operações dos processos de fabricação	52
3.7.4. Tabela de Máquinas	54
3.7.5. Tabela de Checkpoints.....	54
3.7.6. Tabela de soluções geradas	55
3.8. Rotinas desenvolvidas.....	56
3.8.1. Levantamento da Matriz de Tempos de Processamento.....	56
3.8.2. Ordenação dos produtos e dos processos de fabricação	59

3.8.3. Alocação das operações nas máquinas	60
3.8.4. Busca Tabu.....	67
Capítulo 4 – LEVANTAMENTO DE DADOS	70
4.1. Dados relevantes para o problema.....	70
4.1.1 Árvore do Produto	70
4.1.2. Ficha do Processo de Fabricação.....	71
4.1.3. Carteira de Produtos	73
4.1.4. Cadastro de Máquinas e Equipamentos	74
Capítulo 5 – Análise de Resultados	75
5.1. Descrição do Primeiro Teste	75
5.2. Descrição do Segundo Teste	83
5.3. Análise dos resultados dos testes.....	93
Capítulo 6 – CONCLUSÕES	94
6.1. Autocrítica do Modelo Adotado.....	94
6.2. Conclusões	95
BIBLIOGRAFIA	96
ANEXO 1.....	97
ANEXO 2.....	107

Capítulo 1 – CARACTERIZAÇÃO DO PROBLEMA

1.1. Introdução

O trabalho visa criar um modelo de programação da produção para indústrias intermitentes repetitivas. Devido à complexidade em termos de número de produtos e máquinas dentro de uma indústria, a solução fornecida pelo modelo não deverá ser a solução ótima para o problema e, sim, uma solução de boa qualidade dada por um algoritmo heurístico.

Os itens seguintes se dedicam a apresentar a fábrica em questão e suas características mais relevantes para a programação a ser feita. No capítulo 2 apresentamos uma revisão da teoria da programação de produção.

O Capítulo 3 irá detalhar a forma como foi projetado o modelo, além de características específicas utilizadas, tais como o algoritmo adotado. Deverão ser também relatadas aqui as adaptações necessárias do modelo teórico para o caso real estudado.

O Capítulo 4 tratará do levantamento de dados a serem utilizados no modelo.

O capítulo 5 mostra os resultados obtidos em dois testes comparativos das soluções geradas pelo modelo com as soluções geradas pela programação atualmente feita na empresa.

O capítulo 6 apresenta as conclusões do trabalho, faz críticas ao modelo construído e sugere pontos de melhoria para o modelo.

1.1.1. Apresentação e análise da fábrica escolhida

A fábrica escolhida se chama PlastPark e fabrica agendas, porta-cartões, pastas de plástico e cadernos em geral. É uma empresa pequena, com menos de 100 funcionários, que ainda se enquadra dentro da filosofia das empresas atendidas pelo SEBRAE. Fica localizada na cidade de Sorocaba a cerca de 95km da capital. Possui um arranjo funcional, dividido em quatro grandes setores, a saber: almoxarifado, preparação de materiais, fabricação e expedição. A empresa fabrica os seus produtos em lotes, os quais são armazenados no estoque e de lá são enviados diretamente aos seus clientes. Esta foi a solução encontrada para atender aos seus três ou quatro clientes preferenciais, responsáveis por grande parte do faturamento mensal da empresa. Estes clientes são grandes atacadistas de São Paulo e demais capitais, tais como a Kalunga, Wal-Mart entre outros, que apenas aceitam a pronta-entrega dos seus pedidos ou então dão um prazo curtíssimo para o recebimento. Todos os pedidos feitos por estes clientes fazem parte do portfolio de produtos da empresa, porem existem pedidos que não são produtos de linha.

A fábrica funciona normalmente de 2^a feira a Sábado, em dois turnos de trabalho, com uma parada de 1h para a refeição em cada um deles. O primeiro começa às 6h da manhã e vai até às 15h; o segundo, das 15h até aproximadamente às 22h. Porem, em períodos de picos de demanda, a fábrica pode funcionar ininterruptamente durante a semana.

A empresa comercializa cerca de 500 produtos, 200 dos quais são por ela fabricados. Fora estes 500 produtos, a empresa também aceita pedidos especiais, considerados como produtos fora de linha, que recebem um tratamento especial na etapa de programação da produção. O restante dos produtos é praticamente todo importado de

outros países. Este é um detalhe importante, pois a importação destes produtos implica em um tempo de entrega relativamente grande (de 45 até 90 dias), o que obriga a fábrica a manter altos estoques e fazer novos pedidos constantemente, quando o nível de estoques atinge um patamar baixo. Esta checagem dos níveis dos estoques da fábrica é feita mensalmente, e só no final de cada período são emitidas novas ordens de serviço para os produtos de linha com estoques baixo. O novo patamar a ser atingido pelos estoques é definido através de uma previsão de demanda baseada nos dados históricos dos produtos.

Por causa da natureza dos seus produtos, como grande parte deles são materiais escolares, a sua demanda possui sazonalidades, com três períodos nitidamente diferentes. O período de maior demanda se concentra do final de Outubro até o mês de Março. Em seguida existe um período de pequena demanda que dura de Abril a Junho, e depois mais um período de demanda intermediária que dura do início do mês de Julho até o final de Outubro.

1.2. Descrição do problema

Os modelos de programação da produção são função do tipo de processo envolvido: uma máquina e várias tarefas, várias máquinas e várias tarefas, etc, além de várias alternativas de critérios de avaliação: minimizar o tempo total de fabricação, maximizar a utilização de recursos, minimizar atrasos, etc. , sendo que a maior parte dos trabalhos desenvolvidos se referem a problemas com uma ou duas máquinas, fazendo com que isto não seja aplicável a situações reais.

Em matéria de complexidade, os modelos que mais se aproximam de casos reais são o "Flow Shop Problem" e o "Job Shop Problem". Infelizmente o problema de "Flow

Shop” simples possui diversas limitações, fazendo com que seja pouco adequado para a nossa empresa. No nosso caso, a fábrica possui características de certa forma parecidas com o modelo do “Flow Shop Problem”, porém com alguns aspectos distintos. O problema é chamado pela literatura como “Flexible Flow Shop Problem”. Como preconizado neste modelo, o processo de produção dentro da empresa é caracterizado por ser unidirecional, isto é, o material processado segue obrigatoriamente uma única direção, começando em uma operação inicial, passando por várias máquinas diferentes em sequência, e terminando em uma outra operação final. Porém, nem todos os produtos seguem o mesmo roteiro de produção. Existem produtos que podem pular estágios de produção sem passar por certas máquinas existentes no meio do caminho. O número de máquinas é outro parâmetro importante que aumenta de forma drástica a complexidade do problema. No nosso caso, o número de máquinas é moderado, o que complica a identificação de uma solução ótima, de característica combinatória. Existem para alguns estágios da produção mais de uma única máquina, fazendo com que seja necessário designar a ordem de serviço deste estágio em fila a uma das máquinas existentes dentro de um setor. Esta é uma fábrica que possui diversas máquinas semelhantes distribuídas em vários centros produtivos. A grande maioria dos produtos segue o mesmo caminho, porém existem aqueles que não passam por algumas das etapas, ou então centros produtivos existentes.

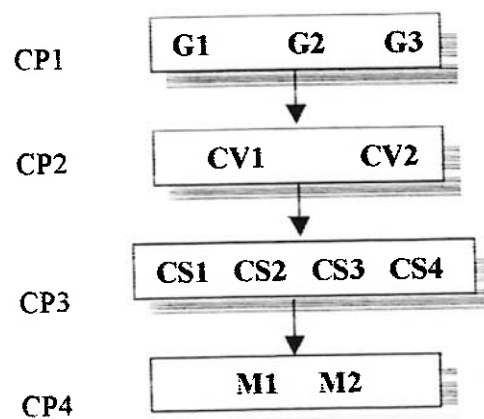


Figura – Centros Produtivos em arranjo funcional
Fonte: Autor

,onde,

CPi – Centros Produtivos,

Gi – Guilhotinas,

CVi – Corte e Vinco,

CSi – Corte e Solda,

Mi – Montagem.

1.3. Fluxo de informações para um pedido de compra

Os novos pedidos de fabricação seguem um roteiro preestabelecido que possui algumas diferentes alternativas, descritas a seguir:

A PlastPark possui cerca de 42 representantes em todo o território nacional, os quais são os responsáveis pela emissão de novos pedidos, feitos por via fax. Estes novos pedidos são cadastrados dentro de um banco de dados, e, em seguida, o setor de expedição verifica se os pedidos estão disponíveis em estoque. Caso afirmativo, os pedidos são separados e aguardam na expedição para serem despachados para o cliente. Caso contrário, a programação da produção define a sua posição dentro de uma fila. A última

etapa é a emissão de uma Nota Fiscal, que é anexada ao pedido, e segue com ele para o cliente.

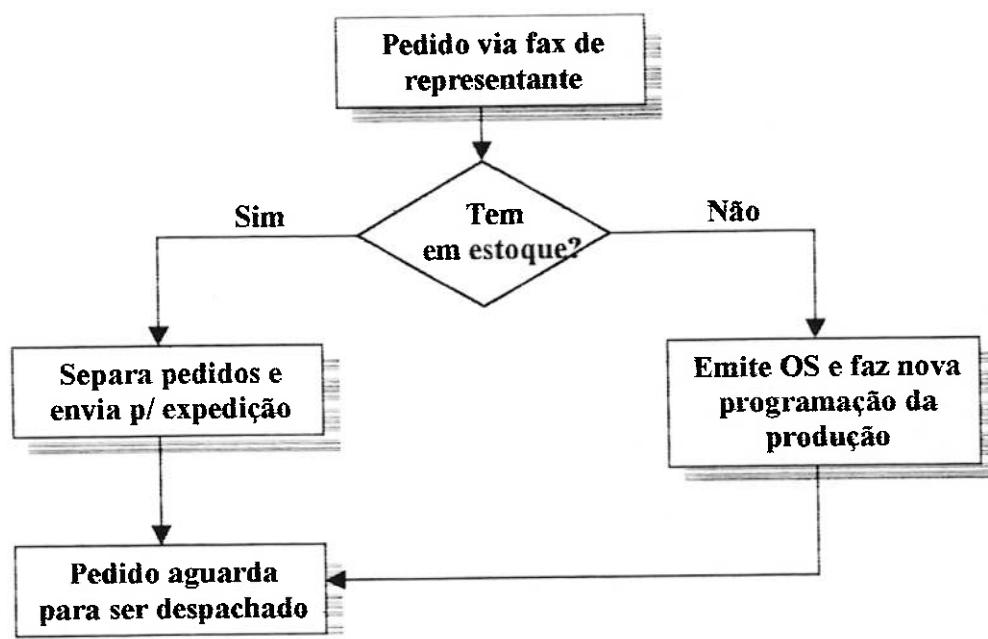


Figura – Roteiro geral do fluxo de informações dentro da fábrica
Fonte: Autor

1.4. Fluxo de Materiais dentro da fábrica

O fluxo de materiais dentro da fábrica segue uma única direção por entre os centros produtivos, a saber: do almoxarifado os materiais seguem para a preparação de materiais; de lá para a fabricação propriamente dita, e, em seguida, vão para o setor de estoques/expedição. Os principais materiais consumidos dentro da fábrica são: o PVC, o polietileno, que chegam em bobinas de 1,40m de comprimento e as chapas de papelão, que chegam em um tamanho padrão, porém em diferentes espessuras. Os demais materiais são compostos por parafusos, arruelas, elásticos, garras para fichário; enfim, acessórios que fazem parte de cadernos, pastas e fichários.

Há uma etapa de pré-corte de materiais, seguido por corte sob medida, caso haja essa necessidade. Em seguida cada componente segue seu próprio roteiro de fabricação. As máquinas seguintes são: Corte e vinco, corte e solda, e no final do roteiro de produção existe normalmente uma pequena montagem ou acabamento. De lá os produtos seguem diretamente para o setor de expedição, onde ficam armazenados em estoque.

Conforme dito anteriormente, os produtos são normalmente produzidos em lote, exceto aqueles considerados especiais. As ordens de serviço seguem uma fila. Deste modo, a data de entrega para estes produtos tem que ser combinada com o cliente. Esta data de entrega chega a ser de até 7 dias depois de feito o pedido. Porém, esta fila pode sofrer alterações na sua programação, caso algum pedido dos seus clientes preferenciais chegue e não exista estoque suficiente para atendê-lo. Estes pedidos são considerados urgentes e vão direto para o começo da fila dos produtos.

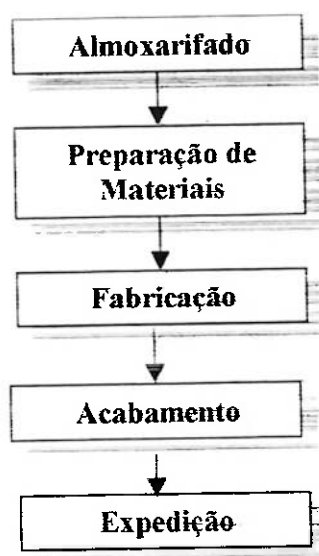


Figura – Roteiro Geral do fluxo de materiais dentro da fábrica
Fonte: Autor

1.5. Características do processo de maior relevância

Como geralmente acontece com pequenas empresas, existe uma certa falta de infraestrutura e falta de controle, e isto influirá muito no modelo de programação desenvolvido.

A posição destes produtos dentro da fila depende basicamente de três critérios: primeiro, se o pedido existe em estoque, segundo, se é feito por um dos seus clientes preferenciais, e terceiro, se é um produto de linha ou não. Se o pedido é especial, ele é programado segundo uma regra de seqüenciação FIFO, ou seja, o primeiro produto a chegar é o primeiro a ser produzido.

Caso os produtos do pedido não existam em estoque e sejam de cliente preferencial, eles seguem diretamente para a produção e sua produção é considerada prioritária.

Infelizmente a fábrica não possui em seus bancos de dados os tempos de preparação de máquinas. Segundo depoimento do gerente da indústria, a questão do tempo de preparação de máquinas poderá ser revisada em uma etapa posterior afim de melhorar o modelo.

Capítulo 2 – REVISÃO DA LITERATURA

2.1. Introdução

Grande parte dos algoritmos desenvolvidos para sequenciar ordens se referem a fábricas com no máximo duas máquinas de um mesmo tipo. Com problemas de mais de duas máquinas, ou então, máquinas de diferentes tipos dentro de uma fábrica, somos obrigados, na maioria das vezes, a recorrer a métodos heurísticos para solucionar o problema de programação da produção.

Outra dificuldade é a de estabelecermos critérios para se avaliar o modelo escolhido aplicado a um caso real. Normalmente a avaliação de um modelo envolve mais de um objetivo e com diferentes pesos para cada um dos objetivos escolhidos. Escolhidos os critérios de avaliação e definidos os pesos para cada um deles, devemos transformar os resultados de cada uma das parcelas do objetivo final em um denominador comum, pois, em vários casos, as diferentes parcelas envolvem medidas com unidades diferentes, tais como, por exemplo, tempo e dinheiro.

O objetivo deste capítulo é de apresentar o segmento da teoria da programação da produção que diz respeito ao problema real tratado, e explicar suas características e as suas dificuldades.

2.2 O “Flexible Flow Shop Problem”

Em várias empresas fabris podemos encontrar um roteiro de fabricação mais ou menos comum à maioria dos produtos. Existem operações que devem ser repetidas em todos os produtos da fábrica e sempre na mesma ordem. O problema flexível dos roteiros em fluxo é um sistema de produção com um número de s estágios em série. Ele difere do

“Flow Shop Problem” mais simples, pois em cada um destes s estágios podemos encontrar um número de m máquinas idênticas dispostas em paralelo. Neste caso, uma operação deve ser feita em qualquer uma das máquinas dentro do centro produtivo. Pode ou não haver alguma restrição sobre a quantidade de estoque intermediário entre dois estágios sucessivos. Isto ocorre quando os produtos em questão ocupam um espaço físico grande dentro da fábrica, causando interrupções e bloqueios de passagem acima de um certo nível de estoque intermediário. Caso contrário, quando os produtos não ocupam grande espaço físico, podemos ter uma quantidade de estoque intermediário teoricamente ilimitada. Seguindo estas características, podemos montar um esquema genérico que representa o roteiro de produção da Plastpark, tal como mostrado abaixo.

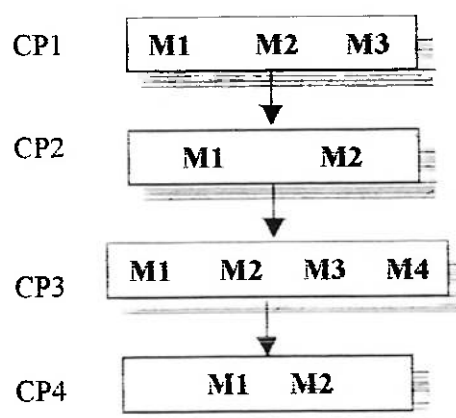


Figura – Centros Produtivos
Fonte: Autor

, onde,

CP_i – Centro Produtivo i

M_i – Máquinas i .

No esquema anterior, os materiais e componentes dos produtos passariam por qualquer uma das máquinas do primeiro até o último centro produtivo, dentro dos quais estariam

m máquinas idênticas dispostas em paralelo. As máquinas podem não operar necessariamente de acordo com a regra de seqüenciação FIFO. A seqüência destas ordens de produção que aguardam para serem fabricadas pode ser alterada afim de resultar em um menor tempo total de fabricação.

Relatamos a seguir as peculiaridades de um modelo tradicional de programação da produção relevantes para o caso aqui estudado.

2.3. Critérios de Avaliação

Os critérios de avaliação são utilizados não só para avaliar as soluções para o problema tratado, mas também são parte fundamental para a geração de novas soluções a partir de soluções antigas, como é normalmente feito no caso em que se adotam algoritmos heurísticos. Os algoritmos heurísticos geralmente incluem condições de interrupção da procura por novas soluções, e são baseadas na avaliação das melhores soluções geradas em um passo anterior.

Os critérios de avaliação normalmente envolvem pontos relacionados como: velocidade de fluxo das ordens nos sistema produtivo, tempo total de fabricação de todas as ordens de serviço, nível de atendimento ou atraso das datas de entrega das ordens de serviço, níveis de utilização de recursos físicos na produção e estoques, além de custos ou margens de contribuição a serem recebidas no decorrer do tempo.

Normalmente se considera mais de um objetivo para se avaliar um modelo de programação, e cada um dos objetivos pode ter uma importância diferente. Como decorrência deste fato, podemos ter parcelas deste objetivo final com diferentes unidades, tais como no caso tempo e dinheiro. Neste caso, normalmente adotamos premissas para o modelo afim de transformar todas as parcelas em unidades monetárias.

O primeiro dos itens relacionados acima é a questão do tempo total de fabricação. Depois de adotada uma sequência de ordens de serviço que irão passar pelas máquinas da fábrica, o momento em que a última ordem de serviço deverá ficar pronta fica automaticamente definido. Porém, o que normalmente ocorre, é que se criam tempos mortos de fabricação entre dois diferentes estágios de produção, devido a restrições de mão-de-obra, máquinas e também porque as tarefas devem necessariamente obedecer a uma certa ordem de precedência. Imediatamente, se conseguirmos eliminar ou então reduzir todos estes tempos mortos existentes, reduziremos o tempo total de fabricação destas ordens de serviço.

Porém, os pedidos em carteira geralmente possuem datas de entrega pré-determinadas pelos clientes e pela gerência da fábrica. Caso estes prazos não sejam obedecidos pela produção, o que muitas vezes ocorre devido à sobrecarga do sistema, implicará à empresa fabril uma multa em dinheiro ou, então, perdas do pedido, do cliente, ou simplesmente perda de prestígio. Caso estes atrasos sejam realmente importantes, ou implicarem em perdas graves, isto deverá ser considerado dentro do modelo de programação.

Existe outro tipo de critério normalmente levado em conta dentro dos modelos de programação, que diz respeito ao nível de utilização dos recursos físicos dentro da fábrica, tais como equipamentos e mão-de-obra. Estabelecido um horizonte desta programação, que normalmente vai desde o início da fabricação de uma carteira de produtos até o momento final da sua fabricação, é possível calcular os níveis de utilização dos recursos dentro de uma fábrica. Normalmente a utilização de equipamentos e da mão-de-obra (MOD) está diretamente ligada ao aparecimento dos tempos mortos entre as operações, e, portanto, pode-se concluir que as mesmas medidas

tomadas para se reduzir o tempo total de fabricação servem para aumentar os níveis de utilização de recursos dentro de uma fábrica.

Finalmente, queremos destacar os critérios que se referem ao lado econômico da sequencição de ordens de produção. Como cada um dos produtos tem diferentes custos variáveis de fabricação. Se considerarmos que os produtos podem ser vendidos aos clientes logo a partir do momento em que eles ficam prontos, teremos uma certa entrada (ou saída, no caso dos custos) de recursos financeiros no caixa da empresa diretamente ligada à programação da produção da fábrica. Isto resultará, portanto, em ganhos financeiros para a empresa, que estão diretamente ligados ao momento em que os pedidos dos clientes ficam prontos. Porém, mesmo que estes produtos não possam ser diretamente vendidos para os clientes no momento em que ficam prontos, pode ser do interesse da empresa que eles permaneçam aguardando em estoque, ao invés da empresa continuar armazenando os seus materiais no depósito de materiais. Em um outro caso mais complexo, os produtos só serão faturados na data de entrega estipulada pelo cliente, o que representa, como consequência, capital empatado da empresa, se decidir estrategicamente fabricar com antecedência e manter o pedido do cliente em estoque.

Como se pode ver, a discussão sobre os objetivos de um modelo de programação da produção é ampla, cada um com suas peculiaridades. A empresa pode ter múltiplos objetivos e com pesos variados, e por isso cabe a empresa adotar a melhor estratégia, escolhendo os seus objetivos, e, se necessário, uma relação entre os graus de importância de cada um deles. Esta relação de pesos entre os objetivos funciona como diferentes graus de importância para cada um dos objetivos que a empresa quer alcançar. Os graus de importância dentro de uma empresa diferem caso a caso, e, ainda por cima, variam na maioria das vezes em função do tempo. Por fim, deve-se tomar

cuidado na adoção de pesos para os objetivos, pois como no caso entre o nível de utilização de recursos e no tempo total de fabricação, ambos afetam da mesma maneira a programação da produção.

2.3. Hipóteses Simplificadoras

As hipóteses simplificadoras, encontradas na maioria das pesquisas com modelos de seqüenciação deste tipo, são as seguintes:

1. Não são permitidos roteiros alternativos de fabricação;
2. Não é permitido o cancelamento de ordens em processamento;
3. Todos os insumos necessários para a fabricação dos produtos escolhidos estão disponíveis desde o início da fabricação;
4. Não se consideram refugos e retrabalhos;
5. As ordens são normalmente indivisíveis e não podem sofrer o “overlapping”, isto é, não podem ser processadas simultaneamente em mais de uma máquina;
6. As datas de entrega são consideradas fixas;
7. Cada centro produtivo opera independentemente dos outros centros;
8. As máquinas têm disponibilidade fixada no horizonte de programação, não se considerando interrupções por quaisquer motivos;
9. As máquinas operam com uma ordem, no máximo;
10. Normalmente quando considerados os tempos de preparação são independentes da seqüência de ordens que são fabricadas em uma máquina;
11. Outra questão simplificadora diz respeito ao processo de chegada dos pedidos.

No nosso caso, a fábrica trabalha com uma carteira de produtos fixa desde o

~~12.~~ início, sendo que esta incluirá os pedidos para reporem estoques, além de pedidos encomendados por clientes e com data fixa.^[1]

2.4. Descrição do método de busca

A literatura sobre *flow shop flexível* é relativamente escassa e muito trabalho ainda precisa ser feito. Estudos aprofundados são necessários para se determinar métodos de resolução para problemas de médio e grande porte. Quase todos os trabalhos realizados utilizam como objetivo do problema a questão de minimização do *makespan*, que é justamente o tempo consumido para se fazer todas as tarefas desejadas.

Escolheremos um método heurístico de busca para encontrar uma solução de boa qualidade para o problema de programação, devido a sua complexidade e ao enorme universo amostral de soluções. Só para se ter uma idéia o número de possíveis soluções para um problema de n tarefas e m máquinas é igual a $(n!)^m$, ou seja, é possível fazer a permutação de n tarefas para cada um das m máquinas presentes no trabalho. A idéia que está por trás dos métodos exatos mais conhecidos, tais como o *branch & bound*, é fazer uma “varredura” dentro das possíveis respostas para se determinar a solução ótima, o que consome muito tempo, além de muito esforço computacional.

A vantagem dos algoritmos heurísticos é que eles são capazes de gerar soluções de boa qualidade em tempo razoavelmente curto, que vai ser proporcional ao número de produtos e de máquinas existentes no caso real. A principal desvantagem é que eles não garantem encontrar a solução ótima para o problema. Uma heurística bastante utilizada e com largas aplicações é a busca local ou busca em vizinhança. A sua idéia é a seguinte: gerada uma solução inicial para o problema, também chamada de solução “semente”, o método gera novas soluções a partir de uma solução anterior, também

[1] Santoro, Miguel Cezar, “Modelo de Programação para produção intermitente com composição de produtos variável no tempo”, São Paulo, 1982

chamada de vizinhança da primeira solução. Seleciona-se uma nova solução na vizinhança e repetimos o processo de geração de soluções vizinhas até que uma condição de parada seja satisfeita. A melhor solução é chamada de ótimo local, pois não há como afirmar que esta seja o ótimo global, uma vez que não verificamos todas as possíveis soluções do problema. Para não se correr o risco de ficar estacionado sobre uma vizinhança que apresenta um ótimo local, surgem as metaheurísticas, que trabalham como guias da busca local direcionando os seus movimentos para a geração de outras vizinhanças em diferentes regiões do espaço de soluções.

Porém, todos os métodos de busca necessitam de uma solução inicial para explorar as soluções vizinhas. Para tanto, muitos dos algoritmos de busca utilizam conjuntamente um método heurístico construtivo para dar início a sua execução. O tópico a seguir trata da revisão bibliográfica sobre as heurísticas construtivas.

2.5 Heurísticas Construtivas

Heurísticas construtivas são aquelas que constroem uma solução progressivamente. O algoritmo sempre começa com uma solução parcial nula, e a cada passo um elemento é adicionado à solução parcial. No final do processo, tem-se uma solução para o problema.

Todas as heurísticas desenvolvidas para o *flow shop* flexível podem ser classificadas em três grupos:

2.5.1. Seqüenciamento seguido de alocação

Primeiro o algoritmo encontra uma seqüência, considerando apenas um *flow shop* de permutação e com apenas uma máquina por estágio. Depois, o algoritmo faz a alocação

para as máquinas paralelas em cada estágio, segundo algum critério aplicado à sequência encontrada no primeiro passo. Por *flow shop* de permutação se deve entender uma programação da produção em que as tarefas são alocadas na mesma ordem em todos os estágios que compõem o problema.

Ding e Kittichartphayak (1994), desenvolveram 2 variações da heurística de Campbell (1970). Guinet e Solomon (1996) desenvolveram heurísticas para vários estágios de processamento semelhantes às propostas por Ding e Kittichartphayak (1994).

Sacchi (1994) realizou vários testes com as heurísticas construtivas tratadas neste texto acima, e destacou a boa qualidade de solução gerada pelas heurísticas de Ding e Kittichartphayak. Além disso, como esta heurística é de fácil implementação, será útil para a geração da solução inicial do nosso problema.

2.5.2. Alocação seguida de seqüenciamento

Na primeira fase o algoritmo determina quais máquinas serão responsáveis pelo processamento de cada tarefa. Na segunda fase o algoritmo determina a sequência de processamento das tarefas em cada máquina.

Wittrock (1985 e 1988) realizou trabalhos desenvolvendo heurísticas. A solução é baseada na decomposição do problema em três subproblemas a serem resolvidos seqüencialmente:

- Alocação de máquinas (determinando quais tarefas visitam cada máquina individual em cada estágio)
- Seqüenciamento das tarefas (especificando a ordem em que as tarefas devem entrar na linha de processamento)
- Especificação dos tempos em que as tarefas devem entrar na linha de processamento.

A solução proposta por Wittrock (1985) é mais simples por considerar a programação de um pequeno conjunto de tarefas que é repetida de forma periódica. Ambos os trabalhos tratam até 51 tarefas em 3 estágios de processamento que não possuem mais que três máquinas paralelas.

2.5.3. Alocação e seqüenciamento simultaneamente

A medida que as tarefas passam a fazer parte do programa, ficam determinadas as alocações e as seqüências em cada uma das máquinas pelas quais as tarefas passam.

Sawik (1993) propôs uma heurística construtiva na qual a cada iteração a rota completa de uma tarefa ao longo do *flow shop* flexível é determinada. A seleção da tarefa e a rota a seguir no sistema são obtidas baseadas na programação parcial obtida até o momento. As decisões em cada iteração são tomadas usando um procedimento de otimização local, baseado na minimização do tempo total ocioso ao longo da rota da tarefa escolhida. O método considera um “buffer” limitado, ou seja, uma restrição da quantidade de componentes entre dois estágios de fabricação. Sawik (1995) usa o mesmo método, porém considerando buffer zero. Chegam a ser considerados até 5 estágios, 100 tarefas e 4 máquinas por estágio.

Ding e Kittichartphayak (1994) desenvolveram uma heurística similar a de Sawick (1993). Os testes tratam até 20 tarefas, 8 estágios e um número aleatório entre 1 e 3 de máquinas paralelas.¹²¹

Hunsucker e Shah (1994) aplicaram várias regras de prioridade ao *flow shop* flexível: LPT (*longest processing time*), SPT (*shortest processing time*), MWR (*most work remaining*), LWR (*least work remaining*), FIFO (*first in, first out*) e LIFO (*last in first out*). Além do *makespan*, os tempos de fluxo médio e máximo são utilizados como

[2] Sacchi, Luís H., “Algoritmos genéticos para minimização de *makespan* em um *flow shop* flexível”, Campinas, 1997

critérios de otimização. Em seu trabalho, limitam o número de tarefas que pode coexistir dentro do sistema. Até 16 estágios, 5 máquinas por estágio e 320 tarefas são tratados.

Guinet e Solomon (1996) desenvolveram uma heurística para vários estágios de processamento equivalente ao trabalho de Sawik (1993), além de estudarem o critério de máximo atraso. Os testes computacionais envolvem no máximo 30 tarefas, 5 estágios e 5 máquinas por estágio.

2.6. Balanço das heurísticas construtivas

No nosso modo de ver, as heurísticas construtivas estudadas na teoria podem até serem aplicadas a um caso real, porém este exige certas adaptações para que a solução inicial seja de fato boa para alimentar o algoritmo heurístico que normalmente se segue em um modelo de programação. Uma das principais dificuldades vistas diz respeito à questão da ordem de precedência entre as "tarefas". Neste ponto é preciso se discutir um pouco a respeito do significado das tarefas utilizadas pelas heurísticas construtivas. Não é possível utilizar as tarefas como sinônimo de um produto ou subconjunto que possui um processo de fabricação, pois para acharmos uma sequência de produção viável é preciso respeitar a ordem de precedência entre os subconjuntos e os produtos. Antes de passar a fabricar um produto, é preciso que todos os seus subconjuntos sejam previamente fabricados e disponíveis no momento da fabricação do produto. O mesmo ocorre quando pensamos em um subconjunto. Neste caso, é possível que outros subconjuntos menores façam parte do processo de fabricação e, por este motivo, têm de ser previamente fabricados e estarem disponíveis.

2.7. Busca Tabu

Escolhemos a Busca Tabu como método heurístico para solucionar o problema, pela relativa simplicidade de implantação, e pelos excelentes resultados conseguidos normalmente em problemas de programação da produção. Esta metaheurística inicia sua busca através de uma solução inicial e prossegue iterativamente de uma solução para outra através de operações chamadas movimentos, até que uma condição de parada seja satisfeita. Para diminuir o risco de ficar parado sobre uma vizinhança de soluções que apresenta um ótimo local, este método também aceita soluções piores do que a solução atual. A Busca Tabu é baseada na premissa de que a resolução de um problema, para ser classificada como inteligente, deve incorporar uma memória adaptativa, e boas estratégias de busca, utilizando o histórico das informações da busca afim de direcionar a busca para regiões ainda não exploradas. Esta memória adaptativa pode ser de curto ou de longo prazo. O princípio básico da utilização da memória adaptativa de curto prazo é o de restringir a busca através da proibição de certos movimentos, armazenando informações reativas às vizinhanças recentemente visitadas.^[3]

Normalmente, após cada geração de soluções vizinhas, uma regra de proibição relativa a um atributo da solução gerada é armazenada. As regras de proibição evitam que alguns movimentos sejam realizados. Armazenam-se estas regras por um certo número de iterações, conhecido como duracão tabu, que pode ser um valor fixo, ou então, pode ser alterado dinamicamente. Um movimento classificado como tabu pode ser executado caso seja considerado atrativo de acordo com o critério de aspiração. Dentre os vários critérios de aspiração mais utilizados destaca-se o movimento tabu, que resulta em uma

solução como valor superior ao da melhor solução encontrada até o momento. O critério de parada deste método pode ter várias formas, tais como um certo número de iterações sem melhoria da busca, um número fixo de iterações, ou um determinado tempo de execução do algoritmo. Este critério é estabelecido a partir do problema e do objetivo de sua aplicação.

O modelo heurístico proposto consiste de uma busca iterativa de novas vizinhanças orientado por uma Busca Tabu. Denomina-se como versão básica das heurísticas a utilização de Busca Tabu somente com a memória de curto prazo. Para tanto, o próximo passo consiste em definir a solução inicial a ser utilizada e os componentes básicos da Busca. Dentre estes se destacam: a estratégia de geração de vizinhança, escolha da próxima solução vizinha, atributos, regras de proibição, duração tabu, critério de aspiração e critério de parada.

2.7.1. Estratégia de Geração de vizinhança

Várias podem ser as estratégias de geração de vizinhança utilizadas pelos métodos heurísticos. Determinando-se uma sequência de ordens de serviço a serem alocadas nas suas respectivas máquinas, fica automaticamente definido o aspecto da programação da produção gerada. Deste modo, o nosso problema se reduz a como gerarmos novas sequências a partir de uma solução anterior. Nesse caso, um dos métodos mais utilizados é a troca de posição entre duas ordens de serviço nas posições x e y . Uma vizinhança inteira seria formada pela totalidade de sequências contendo trocas entre duas tarefas, sendo a primeira delas fixa e a segunda variável. Deste modo, o algoritmo de geração de vizinhança fixa uma determinada tarefa x , localizada na posição $p(x)$, e

experimenta todas as trocas possíveis entre x e as outras ordens de serviço. O tamanho da vizinhança gerada é de $(n/2) * (n-1)$, onde n é o número de tarefas da fila de espera. Outro método bastante simples de ser implementado consiste em inserir uma determinada tarefa x , localizada na posição $p(x)$, em uma posição $p(z)$ qualquer. Esta estratégia gera uma vizinhança de tamanho $(n-1)^2$, sendo n o número de tarefas existentes para serem realizadas.

Uma outra estratégia derivada desta última, consiste em se fixar a posição que devemos inserir as tarefas e variarmos a tarefa escolhida. Neste caso, por exemplo, selecionaríamos uma tarefa x , localizada em uma posição $p(x)$, e a colocariamos na última posição da fila de espera. Em seguida, voltamos a solução que possuíamos de início e selecionaríamos uma outra tarefa. O método é repetido até que tenhamos percorrido todas as tarefas.

2.7.2. Escolha da próxima solução

A escolha da próxima solução que é registrada no histórico de soluções geradas e se torna a “semente” geradora de uma solução. A nova solução é escolhida cada vez que temos uma nova vizinhança gerada. Dentre os métodos mais utilizados podemos destacar:

- A seleção da primeira solução que melhora a solução atual, ou;
- A seleção da melhor solução encontrada dentro da vizinhança (ou melhor movimento).

2.7.3. Seleção dos atributos do movimento e regras de proibição

Estas são regras específicas para cada método de geração de vizinhança, e visa proibir que o método heurístico retorne às vizinhanças já exploradas anteriormente. Dentre as

regras de proibição conhecidas podemos destacar aquela que proíbe a inserção, ou troca de duas tarefas; enfim, o(s) último(s) movimento(s) que foi(foram) dado(s) pelo algoritmo.

2.7.4. Duração Tabu

A duração Tabu faz parte da memória que acompanha um método de Busca Tabu. A duração Tabu estabelece um determinado período dentro do qual as regras de proibição continuam a ser armazenadas, proibindo que o algoritmo assuma como soluções vizinhas aquelas que infringem as regras de proibição. A Duração Tabu depende da regra de proibição especificada e pode ser atualizada de forma dinâmica. Como exemplo das formas de atualização da Duração Tabu, podemos ter:

- 1. Forma Aleatória.** A cada iteração gera-se um novo valor para a duração da proibição;
- 2. Forma Ajustada.** A cada vinte iterações, por exemplo, gera-se um novo valor para a duração de proibição, e a cada iteração é realizado um ajuste neste valor. Este ajuste é feito de acordo com o valor da solução selecionada com relação à solução que gerou o movimento (solução atual). Caso o valor da solução selecionada seja maior que o da solução atual, a duração da proibição associada à solução selecionada é aumentada de uma unidade. Caso o valor solução selecionada seja menor que o da solução atual, a duração é diminuída de uma unidade.

2.7.5. Critério de Aspiração

O Critério de Aspiração pode ser entendido como uma condição na qual se permite que o algoritmo infrinja alguma das regras de proibição e gere uma nova solução atual. Um

dos casos mais utilizados é o de se permitir que o algoritmo gere uma nova solução semente, mesmo que este movimento seja proibido por alguma regra de proibição, caso a sua avaliação seja melhor do que o valor da melhor solução encontrada até o momento pela busca.

2.7.6. Critério de Parada

O critério de parada do algoritmo determina o momento em que cessa a execução (de uma nova iteração) do algoritmo. Ele pode ser dado por uma determinada duração de tempo em que não é encontrada solução melhor do que a atual, ou então, pode ser estabelecido um certo número de iterações, após o qual o algoritmo pára. Este número de iterações pode ser determinado dinamicamente ou de modo fixo (antes do início de funcionamento do algoritmo). Em outras palavras, podemos determinar um número máximo de iterações do algoritmo a partir do momento em que uma nova solução é encontrada, ou então, a partir do momento em que o algoritmo começa a funcionar.

2.8. Conclusões Teóricas

Neste capítulo foram apresentadas algumas características dos modelos de programação da produção. Dentre estas características se destacam as limitações e dificuldades para simular o problema real, além dos pontos fortes e das possibilidades de melhoria dentro de uma programação e de uma sequenciação de ordens bem feita.

Após isto, apresentamos a Busca Tabu, um algoritmo heurístico para determinar a melhor, ou então, uma solução de boa qualidade dentro de um vasto campo de soluções existentes para o problema enfrentado. Foram descritos também as principais

características do modelo e os pontos que devem ser definidos antes da implementação de um modelo para o caso real.

Capítulo 3 – DEFINIÇÃO DO MODELO

3.1. Introdução

Após a revisão bibliográfica feita para o problema em questão, iremos desenvolver neste capítulo a modelagem do problema. Características tais como as premissas adotadas no início do problema, a função objetivo para se avaliar a qualidade das soluções geradas, as restrições envolvidas dentro do problema, limitações do modelo, além do algoritmo adotado para a construção de soluções viáveis devem fazer parte deste capítulo. Todos estes aspectos são explicados em detalhes e devem perseguir a concepção de um modelo adequado para as necessidades da Plastpark, que represente adequadamente a realidade que ocorre dentro da fábrica.

3.2. Considerações gerais

O objetivo do modelo é responder as principais questões que envolvem a programação da produção: o que fazer, aonde fazer e quando fazer? Para tanto devemos encontrar sequências de ordens de serviço, relatando em que máquina as ordens deverão ser feitas e contendo os seus respectivos tempos de início de fabricação e de término de produção. Na produção intermitente as máquinas são compartilhadas durante o processo de fabricação dos produtos. No nosso caso, o problema da programação é estabelecer a sequência de produção de cada lote em cada máquina, estando restrito às precedências impostas pelos roteiros de fabricação e às disponibilidades dos recursos de produção, de maneira a otimizar algum objetivo mensurável.

Vários podem ser os objetivos adotados para verificar a qualidade de uma solução geralmente escolhidos através do compromisso entre a conveniência computacional e as exigências ditadas pela natureza dos produtos fabricados.

3.3. Sequências consideradas

As sequências de fabricação devem ser divididas em três diferentes níveis, do ponto de vista mais focado para o mais global:

1. Sequência de operações dentro de cada processo de produção;
2. Sequência de prioridade de fabricação dos processos de produção de um mesmo produto;
3. Sequência de prioridade de fabricação dos produtos da carteira de produtos.

É importante dizer que as duas últimas sequências devem ser consideradas apenas em caso de desempate para a alocação de operações nas máquinas. O modelo funciona do seguinte modo: designada a ordem de prioridade de fabricação dos produtos em carteira, cada um dos subconjuntos recebe um número indicando a ordem de prioridade de fabricação de cada um. Quando existe duas ou mais operações que querem ser alocadas em uma mesma máquina, a sequência de prioridades de fabricação dos processos (segunda sequência) é consultada e a operação que pertence ao processo prioritário é alocada enquanto que a outra operação aguarda na fila.

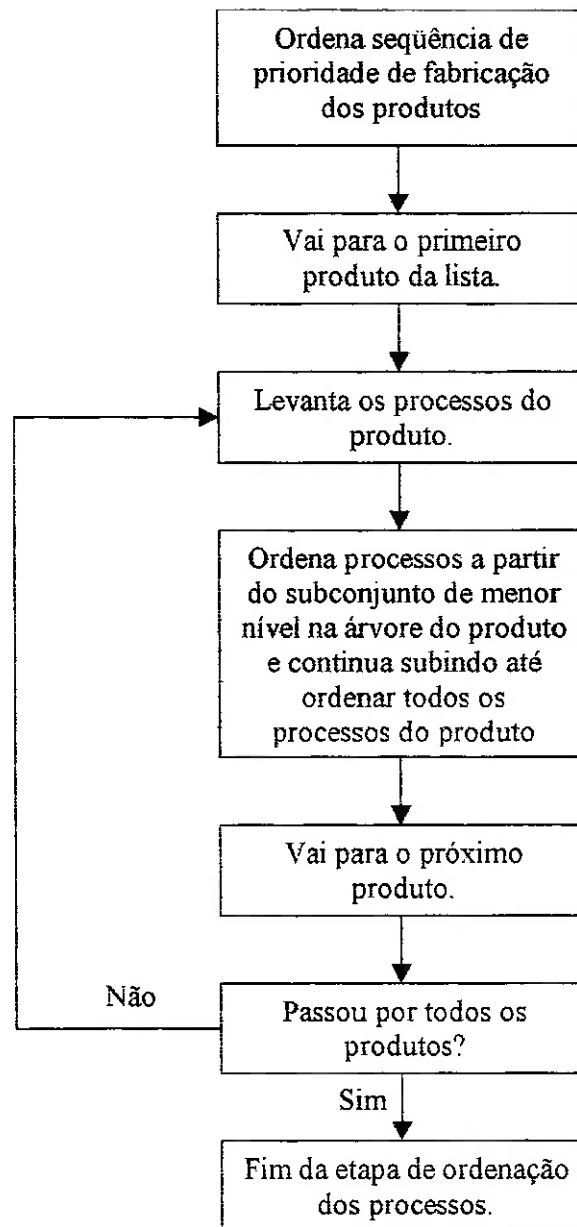


Figura – Esquema da ordenação dos processos de fabricação
Fonte: Autor

3.3.1. Seqüências das operações do processo de fabricação

A fabricação dos produtos/subconjuntos é dividida em operações e estas normalmente possuem uma seqüência lógica. Em cada uma das operações existem tempos de mão-de-obra e tempos de máquina associados. Esta ordem de precedência entre as operações, esquematizada na figura abaixo, está definida nas fichas de processo de fabricação dos produtos, que a empresa estudada já possui cadastrada. Portanto, as seqüências consideradas no nosso modelo da programação da produção só podem ser consideradas viáveis caso respeitem esta precedência entre as operações necessárias para se fabricar os produtos.

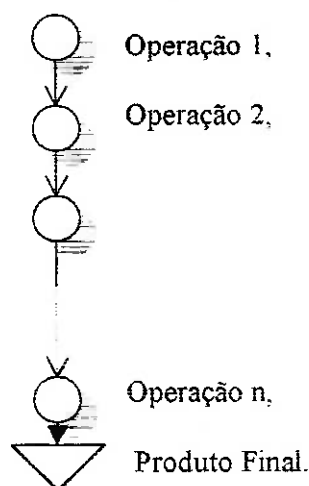


Figura – Relação de operações do processo de fabricação e a precedência entre elas
Fonte: Autor

3.3.2. Seqüência de prioridade de fabricação dos processos

A ficha do processo de fabricação de um produto qualquer contém também os subconjuntos que fazem parte do mesmo. Por sua vez, estes subconjuntos possuem normalmente um ou mais processos de fabricação, que pode inclusive conter outros subconjuntos de menor nível. Conseqüentemente, é necessária a fabricação dos subconjuntos localizados nos níveis mais baixos da árvore do produto, subindo-se aos

poucos para os níveis mais elevados dentro da árvore do produto antes do início da execução das operações referentes ao processo do produto. Esta ordenação debaixo para cima será a ordem padrão de fabricação dos subconjuntos.

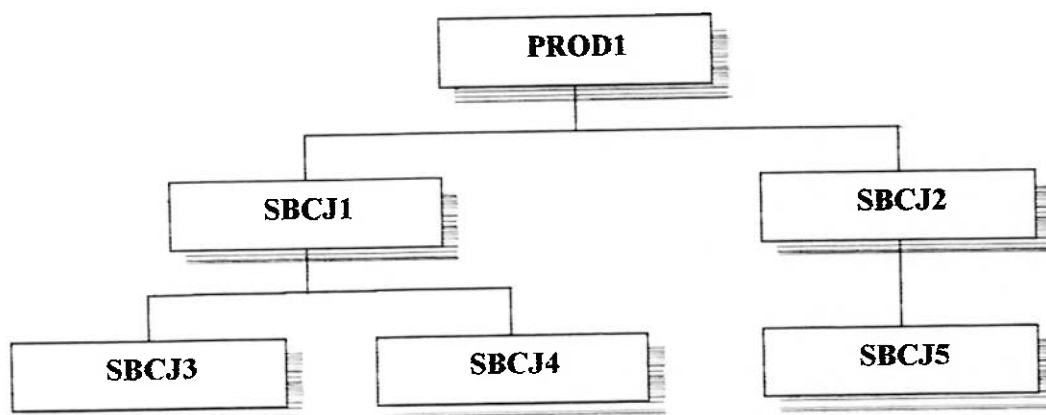


Figura – Árvore do Produto
Fonte: Autor

, onde,

PROD1 – Produto Final,

SBCJ1 e 2 – Subconjuntos de 1^a. ordem,

SBCJ3, 4 e 5 – Subconjuntos de 2^a. ordem.

Cabe lembrar aqui que o modelo deve permitir que a fabricação dos subconjuntos existentes pode ser intercambiável, desde que todos os subconjuntos precedentes (abaixo) estejam fabricados. Por exemplo, na figura acima, podemos notar que entre os subconjuntos SBCJ2, SBCJ3 e SBCJ4, não existe qualquer ordem de precedência de fabricação entre os mesmos. Isto pode ser incluído no modelo permitindo que o próprio usuário do modelo defina a ordem de fabricação dos subconjuntos do modelo.

3.3.3. Sequência de prioridade de fabricação dos produtos em carteira

Esta é a sequência mais importante para o modelo, e deve ser estabelecida com relação aos próprios produtos que fazem parte da carteira de produtos. Esta ordem de precedência é a que guiará a ordem em que são fabricados os subconjuntos e as operações. Por sua vez, as operações do processo prioritário têm preferência sobre as operações dos demais processos pertencentes a produtos de menor urgência de fabricação em caso de disputa por uma vaga em uma mesma máquina.

3.4. Avaliação do desempenho das soluções

O procedimento para se avaliar as soluções foi definido através de uma função de objetivos múltiplos, dentro da qual poderão ser atribuídos pesos específicos para cada uma das duas parcelas da função.

No nosso caso, a programação da produção deve reduzir ou solucionar os problemas do aparecimento dos tempos mortos para que a sua produtividade aumente e penalizar de um certo montante a entrega dos produtos fora do prazo especificado. A programação da produção deve também estabelecer datas para que se possa fazer o cálculo de necessidades de materiais, e, ao mesmo tempo, fazer com que a fábrica possa obter seus ganhos o quanto antes possível, privilegiando a fabricação dos produtos de maior margem de contribuição para a fábrica.

Sugerimos à PlastPark incluir duas parcelas na função objetivo: as penalidades em caso de atraso de fabricação de seus produtos e a questão da data de entrega ponderada pelas margens de contribuição dos seus produtos. Esta segunda parcela privilegiaria a fabricação dos produtos que tivessem a maior margem de contribuição entre os produtos cadastrados em carteira.

Desta forma, teremos uma função objetivo como explicitada abaixo:

$$\text{Min } \sum (P_i * (DDT_i - CDT_i) + MC_i * x\% * CDT_i) \quad (p/ i = 1 \dots n)$$

, onde,

P_i Penalidade em unidades monetárias por dia de atraso do produto i

DDT_i Data de Entrega do Produto i (Due Date),

CDT_i Data de Término de Fabricação de i (Completion Date),

MC_i Margem de Contribuição do produto i ,

$x\%$ Ganho Financeiro Diário,

n Número de produtos da carteira de produtos.

Cabe lembrar que a penalidade por dia de atraso é completamente personalizável, isto é, deixaremos a cargo do usuário completar este campo para cada um dos produtos da carteira de produtos da PlastPark.

Quanto à segunda parcela da função objetivo, a taxa i , que representa o ganho financeiro diário, fica a cargo da taxa de desconto considerada pelo próprio usuário.

Em resumo, para calcular o custo variável dos produtos, o modelo levanta o consumo de todos os materiais, a quantidade de mão-de-obra, horas-máquina, serviços de terceiros, além do custo de cada uma destas partes envolvidas no processo de fabricação dos produtos através de coeficientes técnicos cadastrados. Feito isto, é possível calcularmos os custos variáveis de produção e a estes se somam as despesas variáveis de venda, isto é, os impostos de venda dos produtos. Desta forma, temos um custo para cada um dos produtos. Consideramos os preços dos produtos em carteira e, em seguida, calculamos as margens de cada um dos produtos. Finalmente, estas margens serão levadas em

conta na hora de calcularmos o valor da função objetivo das seqüências envolvidas no modelo.

3.5. Busca de soluções

As soluções para o problema serão dadas por dois algoritmos: uma heurística construtiva para se determinar a solução inicial do problema, e em segundo lugar, a Busca Tabu.

3.5.1. Solução Inicial

A solução que buscamos é determinada por uma seqüência de operações alocadas em cada uma das máquinas. Esta seqüência pode ser definida por uma ordem de precedência dada para os processos que fazem parte de cada um dos produtos. Esta ordenação será fundamental para definir disputas de operações para serem alocadas em uma mesma máquina.

Para encontrarmos uma solução inicial, utilizaremos uma regra de seqüenciação para determinarmos a ordem em que os produtos devem ser fabricados. Como a nossa função objetivo diz respeito a datas de entrega dos produtos e a margens de contribuição de cada um dos produtos, foram consideradas inicialmente três diferentes regras de seqüenciação, dentre as quais escolheremos apenas uma:

1. Menor data de entrega,
2. Maior margem de contribuição, e
3. Maior relação (margem de contribuição/dias restantes para a data de entrega) dos produtos.

$$\uparrow C_j = \frac{M C \uparrow}{D R \downarrow}$$

Como os pesos utilizados para cada uma das duas parcelas da função objetivo pode variar, não seria muito útil definirmos a regra para a solução inicial como sendo a primeira alternativa, ou então, a segunda. Portanto, escolhemos a ordem de precedência entre os produtos da carteira de produtos dada pela relação (margem de contribuição do produto/dias restantes para a data de entrega do produto).

No momento inicial da programação, são liberados os processos localizados no menor nível das árvores de todos os produtos da carteira de pedidos, porém as operações dos processos menos urgentes só poderão serem alocados nas máquinas caso não haja nenhuma operação de um produto de fabricação mais prioritária que utilize determinada máquina.

3.5.2. Busca Tabu

A Busca Tabu é uma Metaheurística, que busca uma solução de boa qualidade para o problema proposto, explorando um conjunto de soluções localizados na vizinhança de uma solução inicial. Para o seu correto funcionamento, a Busca Tabu requer a definição de alguns tópicos, tais como a geração de soluções vizinhas, a escolha da próxima solução, as regras de proibição, a Duração Tabu, o critério de aspiração e o critério de parada da busca. Esta seção visa apresentar como estes tópicos foram definidos dentro do nosso modelo de programação da produção.

3.5.2.1. Geração de soluções vizinhas

Os critérios para a definição da estratégia de geração de vizinhança do modelo são: primeiro, a eficiência do método e, segundo, o tamanho da vizinhança gerada. Baseando-se nestes critérios e voltando aos métodos de geração de vizinhança

pesquisados na revisão bibliográfica do estudo, cabe escolher um entre os dois métodos: de inserção de tarefas e a troca de tarefas.

Quanto a eficiência do método, não existe qualquer estudo que comprove uma maior eficiência da parte de qualquer um destes dois métodos. Porém, existe uma diferença básica no tocante ao tamanho da vizinhança gerada de cada um dos métodos (vide tabela a seguir). É presumível que quando investigamos uma vizinhança menor, o método precisa realizar um maior número de movimentos para explorar o mesmo tamanho de “área” (de possíveis soluções), em comparação com um método que gera vizinhanças maiores. Portanto, trata-se de uma questão de estratégia: ou realizar um maior número de iterações, ou então, explorar uma área maior a cada movimento. A tabela a seguir mostra a diferença de tamanho da vizinhança gerada em cada um dos métodos.

	Métodos de geração de vizinhança	
	Inserção de Tarefas	Troca de Tarefas
Tamanho da vizinhança	$(n-1)^2$	$n/2*(n-1)$
5 tarefas	16	10
10 tarefas	81	45
15 tarefas	196	105
20 tarefas	361	190
25 tarefas	576	300
30 tarefas	841	435

Tabela – Tamanho da vizinhança para métodos de geração de soluções vizinhas
 Fonte: Autor

Do ponto de vista do esforço computacional, é muito trabalhoso lidar com um número tão grande de soluções vizinhas, como no caso da inserção de tarefas, e quando o número de tarefas é relativamente grande. Portanto, preferimos utilizar o método de troca de tarefas, explicado a seguir.

A troca de tarefas consiste na troca de uma tarefa i , localizada na posição $p(i)$, que é trocada com a tarefa j , localizada na posição $p(j)$. A cada movimento realizamos todas as trocas possíveis de uma determinada tarefa i com as demais $(n-1)$ tarefas; em seguida trocamos a antiga tarefa i pela tarefa seguinte, localizada na posição $(i+1)$. Repete-se o processo até a chegada da tarefa localizada exatamente no meio da sequência.

3.5.2.2. Escolha da próxima solução

Dada uma solução inicial, a busca prossegue com a avaliação de cada solução vizinha, e, posteriormente, a escolha de uma nova solução a partir da qual iremos gerar novas soluções vizinhas. Entre os métodos de escolha de novas soluções, podemos escolher a primeira solução (vizinha) que melhora a solução atual (primeiro movimento), ou então, escolher a melhor solução encontrada entre as soluções vizinhas (melhor movimento). Não existe comprovação de que um método possa obter soluções melhores do que o outro. Deste modo, preferimos escolher a melhor solução entre as soluções vizinhas existentes.

3.5.2.3. Seleção de Atributos do movimento e Regras de proibição

Escolhida uma próxima solução que irá gerar uma nova vizinhança é interessante que a Busca não volte para trás, na direção da última solução encontrada. Para tanto, é preciso que a Busca contenha uma memória guardando os dados referentes aos últimos movimentos dados. Estes dados fazem com que certos movimentos sejam proibidos de serem feitos pela Busca Tabu.

Como os movimentos são feitos a partir de soluções anteriores, é preciso ter em mente que estes movimentos dizem respeito à sequência de fabricação dos produtos da carteira de pedidos.

Foram selecionadas as opções mais conhecidas ou as que apresentam melhores resultados. Da opção mais restrita à menos restrita, temos:

1. **ij** – a tarefa *i* e a tarefa *j* não podem ser movidas;
2. **i** – a tarefa *i* não pode ser movida;
3. **mi** – a tarefa *i* não pode ser movida para posições *k*, onde $k \leq \pi(i)$.

Sem subsídio teórico para esta questão e sem consenso sobre qual das opções é a melhor, decidimos, portanto, utilizar a terceira opção, pois um número menor de movimentos é feito em cada iteração.

3.5.2.4. Duração Tabu

É de interesse do método de Busca proibir certas mudanças na sequência de tarefas, a fim de que o método não volte para regiões que já foram exploradas anteriormente. Porém, estas restrições devem ser válidas somente durante um determinado período de tempo. Este período de tempo, geralmente dado por um certo número de iterações, é justamente a Duração Tabu. Para que o método guarde as restrições durante um determinado período de tempo, é preciso que a Busca contenha um banco de dados que funciona como um histórico dos últimos movimentos realizados do problema.

As formas de atualização desta memória caracterizam a Duração Tabu. Normalmente, a atualização da Duração Tabu é feita de forma dinâmica, isto é, recalculada após cada iteração do problema. Estas formas de atualização podem ser:

1. **Aleatória** – a cada iteração gera-se um novo valor para a duração da proibição;
2. **Ajustada** – a cada vinte iterações se gera um novo valor para a proibição, e a cada iteração é realizado um ajuste neste valor. Este ajuste é feito de acordo com o valor da solução selecionada com relação à solução que gerou o

movimento (solução atual). Caso o valor da solução selecionada seja maior que o da solução atual, a duração da proibição associada à solução selecionada é aumentada de uma unidade. Caso o valor da solução selecionada seja menor que o da solução atual, a duração é diminuída de uma unidade. Estes valores são delimitados entre limites inferiores e superiores, de acordo com o número de tarefas existentes dentro do problema. A tabela abaixo mostra os limitantes deste intervalo de geração.

<u>Movimento</u>	<u>Atributo</u>	<u>Regra</u>	<u>Limitante</u>	<u>Limitante</u>
			<u>Inferior</u>	<u>Superior</u>
Troca de Tarefas	Mi	i não pode ser movido para posições k, onde $k \leq p(i)$	N/2	3N/2

Tabela – Limitantes de geração de lista Tabu
Fonte: Autor

,onde,

N – número de tarefas,

P(i) – posição da tarefa i,

Mi – método utilizado para troca de tarefas.

No nosso modelo iremos utilizar a Duração Tabu Ajustada, pois esta forma de atualização é uma das regras que apresentam os melhores resultados, tal como comprovado em testes feitos por Glover & Laguna (1993).

3.5.2.5. Critérios de Aspiração

Os critérios de aspiração podem ser entendidos como uma condição de “relaxamento”. É possível aceitar uma dada sequência para achar nova vizinhança caso ela satisfaça os critérios de aspiração, mesmo infringindo as regras de proibição. Existem muitos critérios de aspiração, porém o mais utilizado diz que só aceitaremos uma sequência como base para uma nova solução semente caso o valor da solução desta seja melhor do que a melhor solução atual encontrada até o momento pela busca.

3.5.2.6. Critério de Parada

O critério de parada estabelece a condição para o término de funcionamento do método de Busca. Ele pode ser definido pelo tempo de funcionamento, por um número fixo de iterações, ou então, por uma certa quantidade de iterações em que não é possível se achar uma solução melhor do que a melhor solução atual. Como o tamanho do problema é variável, ditado pela carteira de pedidos inserida pelo cliente, não achamos razoável estabelecer um tempo de funcionamento como critério de parada. Por causa do mesmo motivo, descartamos a opção de se fazer um número fixo de iterações. Utilizaremos no nosso modelo a terceira opção de condição de parada, dentro da qual é estipulado que se a Busca não conseguir achar uma solução melhor do que a atual, cessa-se o funcionamento da busca. O número certo de iterações desta condição deve ser definido após alguns testes realizados, para verificarmos a eficiência do método. A princípio, iremos testar a busca com uma condição de parada de 20 iterações, aumentando o número de iterações progressivamente para se avaliar se há melhoria nas soluções.

3.5.2.7. Resumo das principais características

Neste item discutimos e definimos as principais características do método de Busca Tabu que será adotado, em conjunto com a heurística construtiva DK1. Na tabela a seguir, podemos ver as características do modelo em forma de resumo.

Item	Descrição
Geração de solução inicial	Utilização da heurística construtiva própria
Geração de soluções vizinhas	Troca de Produtos
Escolha da próxima solução	Melhor solução da vizinhança
Seleção de Atributos do m_i – o produto i não pode ser movido para posições movimento e Regras de k , onde $k \leq \pi(i)$ proibição	
Duração Tabu	Duração Tabu Ajustada – a cada vinte iterações, aumenta-se ou se diminui de uma unidade a duração das regras de proibição.
Crítérios de Aspiração	Aceta-se nova solução caso ela seja melhor do que a atual
Condição de Parada	20 iterações sem encontrar solução melhor do que a atual

Tabela – Resumo das principais características do modelo
Fonte: Autor

3.6. Hipóteses Simplificadoras

O desenvolvimento de um modelo de programação tem como principal finalidade criar um instrumento para tomada de decisão no mundo real, além de auxiliar o conhecimento dos sistemas físico e de informações.

Dado um problema real e definidos os objetivos a serem atingidos, podemos construir vários modelos diferentes, que permitem sua obtenção. A qualidade das decisões obtidas depende, entre outras coisas, da “aderência” do particular modelo escolhido em relação à realidade sobre a qual pretendemos agir. Esta aderência depende diretamente das hipóteses implícita ou explicitamente nele consideradas. Essas hipóteses normalmente restringem a aplicabilidade do modelo e diminuem o crédito nas decisões obtidas, mesmo em modelos de otimização, pois estamos interessados na melhor decisão para o problema real, e não para o modelo. Parece-nos fundamental, portanto, que, afim de se ter uma idéia da abrangência e limitação do nosso trabalho, listemos as hipóteses consideradas no nosso modelo.

3.6.1. Hipóteses com relação às ordens

1. Cada ordem é processada de acordo com uma sequência de máquinas ou roteiro pré-estabelecido, não sendo permitido roteiro alternativo;
2. Cada operação, presente no roteiro de uma ordem, só pode ser executada por uma máquina determinada, não sendo permitida a existência de máquinas alternativas;

3. Uma vez iniciada uma operação de determinada ordem numa máquina, deve ser executada até o seu final, antes que outra ordem inicie seu processamento nessa máquina;
4. Uma vez iniciado o processamento de uma ordem, deve seguir até o término, não sendo permitido cancelamento;
5. Cada ordem é considerada uma entidade indivisível, mesmo que composta de várias unidades. Não se permite, portanto, a divisão ou junção de várias ordens, mesmo que formadas por lotes de itens;
6. Cada ordem não pode ser processada simultaneamente por mais do que uma máquina (Não é permitido o "overlapping" de operações);
7. As ordens podem esperar pelas operações entre máquinas, caso não haja disponibilidade de máquina;
8. Somente as ordens cadastradas, no momento inicial, dentro de uma carteira de produtos serão alvo da programação da produção.

3.6.2. Hipóteses com relação aos tempos de processamento e outros atributos

1. Os tempos de processamento são finitos e conhecidos;
2. Os coeficientes técnicos representam uma relação do tempo de fabricação médio de uma unidade de um componente ou produto, considerando-se o lote médio de fabricação ou lote representativo do que ocorre na realidade;
3. Os coeficientes técnicos incluem eventuais tempos de colocação e retirada da ordem na máquina, além de tempos de transporte entre máquinas;

4. Todos os insumos necessários para a fabricação dos produtos escolhidos estão disponíveis desde o início da fabricação;
5. Não se consideram refugos e retrabalhos;
6. Todos os atributos das ordens, além dos já considerados, são finitos e conhecidos (receita, custo, margem, custo de atraso diário, data de entrega solicitada, grupo e código de identificação).

3.6.3. Hipóteses sobre as máquinas

1. As máquinas têm disponibilidade fixada no horizonte de programação, não se considerando interrupções por quaisquer motivos;
2. Não são considerados os tempos de preparação de máquina;
3. Cada máquina opera independentemente das outras;
4. Todas as máquinas estão prontas para operação, continuamente, durante o tempo determinado pelas disponibilidades úteis de cada uma das máquinas consideradas;
5. As máquinas não ficam paradas se existe uma operação pronta para ser fabricada (mesmo que esteja mal colocada na lista de precedências);
6. Não se consideram paradas por quaisquer motivos, como, por exemplo, falta de energia, de operadores, quebras e manutenção;

As hipóteses consideradas, em sua maioria, são comuns aos trabalhos na área de sequenciação, e não são tão restritivas como podem parecer a primeira vista.

Acreditamos que, embora possam delimitar o campo de aplicação do trabalho, ainda permitem boa generalidade para os principais objetivos do problema considerado. Outras considerações sobre o modelo proposto e pontos a serem melhorados no futuro são considerados no item autocrítica do modelo adotado, no final do trabalho.

3.7. Tabelas criadas

Os dados anteriores foram divididos em cinco diferentes tabelas criadas para a programação da produção, que serão explicadas a seguir:

1. Tabela da Carteira de pedidos;
2. Tabela de processos de fabricação;
3. Tabela de operações dos processos de fabricação;
4. Tabela de máquinas e equipamentos;
5. Tabela de Checkpoints;
6. Tabela de soluções geradas;

3.7.1. Tabela da Carteira de Pedidos

Esta tabela contém a relação todos os produtos que fazem parte das carteiras de pedidos da fábrica, além de informações relevantes para o modelo de programação da produção. Esta tabela é consultada durante as etapas de levantamento das operações envolvidas na produção e na etapa de avaliação da programação da produção. Os campos da tabela são listados a seguir:

- DATA – Data do cadastro da carteira de pedidos;
- DESCRIÇÃO – Descrição breve da carteira de pedidos;
- GRUPO – Grupo do produto que faz parte da carteira de pedidos
- CÓDIGO– Código do produto que faz parte da carteira de pedidos;
- DESCRIÇÃO – Descrição do produto;

- QUANTIDADE – Quantidade de produtos vendidos;
- PREÇO – Preço (total) da venda do produto;
- DATA DE ENTREGA – Data de entrega combinada do produto;
- PENALIDADE – Penalidade diária por dia de atraso da entrega do produto;
- COND. PAGTO – Condição de Pagamento da venda do produto;
- MARGEM – Margem de contribuição total desta venda;
- ORDEM – Ordem ou prioridade na lista de seqüências de fabricação dos produtos.

3.7.2. Tabela de processos de fabricação

Esta tabela foi criada para armazenar os processos de produção que fazem parte dos produtos da carteira de pedidos-alvo da programação da produção, além de informações relevantes para o nosso problema. Esta é, sem dúvida, a tabela mais importante do modelo, pois é utilizada em todas as etapas da programação da produção, desde o levantamento das informações, a ordenação das seqüências de fabricação envolvidas, até a alocação das operações nas máquinas da fábrica. Deve-se lembrar que esta tabela armazena não apenas os processos de fabricação dos produtos, mas também os processos de fabricação dos subconjuntos destes produtos. Os campos da tabela são mostrados a seguir:

Informações sobre a carteira que o processo pertence

- DATA – Data da carteira de produtos ao qual o processo faz parte;
- DESCRIÇÃO – Descrição da carteira de pedidos;
- GRUPO_PROD – Grupo do produto ao qual o processo pertence;
- COD_PROD – Código do produto ao qual o processo pertence;

Informações sobre o elemento pai do processo

- TIPO_PAI – (Produto ou subconjunto) Tipo do processo pai do processo;
- GRUPO_PAI – Grupo do produto/subconjunto pai do processo registrado;
- COD_PAI – Código do produto/subconjunto pai do processo registrado;

Informações sobre o processo de fabricação

- TIPO – Tipo do processo registrado;
- GRUPO – Grupo do processo registrado;
- CÓDIGO – Código do processo registrado;
- ORDEM – Ordem ou prioridade na sequência de processos de fabricação;
- READY_TIME – Instante inicial em que o processo está pronto para ser fabricado;
- COMPLETION_TIME – Instante de término de fabricação do processo;
- LIBERADO – Este campo avisa se o processo de fabricação esta pronto para ser fabricado;
- PRODUZIDO – Avisa se a fabricação deste processo foi terminada.

3.7.3. Tabela de operações dos processos de fabricação

Esta tabela especifica as operações que fazem parte dos processos de fabricação da carteira de pedidos, além de informações relevantes para a parte da programação da produção. Esta tabela será manipulada somente durante a parte da alocação das operações nas máquinas da fábrica. Os campos da tabela são mostrados a seguir:

Informações sobre a carteira que as operações pertencem

- DATA – Data da carteira de produtos ao qual o processo faz parte;
- DESCRIÇÃO – Descrição da carteira de pedidos;
- GRUPO_PROD – Grupo do produto ao qual a operação registrada pertence;
- COD_PROD – Código do produto ao qual o processo a operação registrada pertence;

Informações sobre o elemento pai das operações

- TIPO_PAI – (Produto ou subconjunto) Tipo do processo pai do processo da operação registrada;
- GRUPO_PAI – Grupo do produto/subconjunto pai do processo da operação registrada;
- COD_PAI – Código do produto/subconjunto pai do processo da operação registrada;

Informações sobre o processo de fabricação das operações

- TIPO – Tipo do processo ao qual a operação registrada pertence;
- GRUPO – Grupo do processo ao qual a operação registrada pertence;
- CÓDIGO – Código do processo ao qual a operação registrada pertence;

Informações sobre a operação registrada

- OPERAÇÃO_NO – Sequência da operação no processo de fabricação;
- PRODUZIDO – Indica se a operação já foi produzida ou não;
- MÁQUINA – Máquina utilizada para alocar a operação;
- READY_TIME – Instante inicial da fabricação;

- **PROCESSING_TIME** – Tempo de Processamento da operação;
- **COMPLETION_TIME** – Instante final de fabricação da operação;

3.7.4. Tabela de Máquinas

Esta tabela contém as informações sobre as máquinas da fábrica e informações para nos auxiliar nas etapas de alocação das operações. A tabela é utilizada na etapa de levantamento das informações do problema e na etapa de alocação das operações. Os seus campos estão mostrados a seguir:

- **MAQUINA** – Código da máquina;
- **NOME_MAQUINA** – Código da máquina + índice da máquina. Este campo é fundamental para diferenciarmos as máquinas que fazem parte da fábrica. As máquinas idênticas precisam receber um índice para que nós possamos diferenciá-las. Assim se possuímos três máquinas corte-vinco, as máquinas são cadastradas como CV1, CV2 e CV3, por exemplo;
- **ESTADO** – (Vazia ou Ocupada) Indica o estado em que a máquina se encontra;
- **INSTANTE_ESVAZIA** – Indica o próximo instante em que a máquina se esvaziará;
- **CAPACIDADE** – Dada em horas/dia;

3.7.5. Tabela de Checkpoints

Esta tabela registra os momentos em que ou uma máquina fica vazia novamente e que podemos fazer uma nova checagem para alocação de operações nas máquinas. Os instantes de verificação são consultados cronologicamente, como se houvesse um

relógio que indica em que momento temos de tomar uma decisão para alocar as operações. Os seus campos são mostrados a seguir.

- AÇÃO – (Checkpoint, ou Tarefa alocada ou tarefa torna-se candidata);
- CHECADO – Sim ou Não;
- INSTANTE – Idem;

3.7.6. Tabela de soluções geradas

Esta tabela registra as soluções geradas para a programação e os valores da função objetivo para dada solução. É preciso lembrar que a solução para dada programação é dada pela sequência de prioridade dos produtos da carteira, conforme discutido anteriormente. Ela é utilizada somente na etapa de avaliação da solução gerada. Os campos da tabela são mostrados a seguir:

- SEQUÊNCIA – Sequência de prioridade dos produtos da carteira;
- PENALIDADE – (R\$) Penalidade total pelo atraso na entrega dos produtos da carteira;
- MARGEM CESSANTE – (R\$) Margem de contribuição cessante da carteira de produtos;
- FC_OBJETIVO- Valor da função objetivo da solução gerada. Soma dos dois últimos campos;

3.8. Rotinas desenvolvidas

As rotinas desenvolvidas para a programação da produção são reflexo das cinco grandes etapas do modelo de programação da produção: levantamento das informações, ordenação dos produtos e processos de fabricação, alocação das operações, avaliação da solução gerada, além da busca tabu.

3.8.1. Levantamento da Matriz de Tempos de Processamento

Cadastramos a carteira de pedidos;

Cadastramos todos os produtos que fazem parte da carteira de pedidos e suas informações;

Vai para o Primeiro Produto (Registro);

Enquanto não passar por todos os registros (Loop)

Começa

Enquanto não passar por todas as operações

Pesquisa todos os processos de subconjuntos e produtos que fazem parte do processo do produto:

Grava processo de subconjuntos na tabela de processos de fabricação;

Pesquisa operações do processo atual

Enquanto não passar por todas as operações

Começa

Cadastra Operação e suas informações pertinentes na tabela de operações;

Se a máquina utilizada ainda não foi cadastrada **então**

Cadastra máquina utilizada na operação:

Próxima Operação;

Término da rotina

Próximo Produto;

Termino da Rotina

Figura – Esquema ilustrativo da rotina de levantamento das informações para programação

Fonte: Autor

As primeiras rotinas desenvolvidas visam completar as tabelas anteriores com todas as informações que necessitamos para começarmos a fazer a programação da produção. Depois de reunidas todas as informações, reduziremos o problema a achar uma programação da produção de boa qualidade a partir de uma matriz de tarefas x máquinas, onde são dados os tempos de processamento de cada operação e de uma listagem dos processos prioritários.

A rotina desenvolvida deve levantar todos os processos que fazem parte da carteira de produtos, além das operações envolvidas e seus tempos de processamento. Isto deve ser feito dentro de cada um dos processos que fazem parte dos produtos de uma carteira de produtos especificada pelo usuário. Neste ponto precisaremos de todos os tempos dos processos dos produtos e dos subconjuntos.

Outro ponto importante a se salientar é a dúvida que surge em se pegar os tempos relacionados à mão-de-obra ou então os tempos de máquina em cada uma das operações, visto que um pode diferir do outro. Optamos por levantar apenas os tempos de máquina em cada uma das operações, deixando de lado os coeficientes técnicos da mão-de-obra envolvida, a não ser que haja uma operação que possui tempos de mão-de-obra. Isto ocorre pois o tempo de mão-de-obra nem sempre é adequado, visto que um funcionário pode cuidar ao mesmo tempo de várias máquinas, ou, por outro lado, dois ou mais funcionários podem estar trabalhando na mesma máquina. Por outro lado, caso haja uma operação que não utilize nenhuma máquina, e sim apenas mão-de-obra, então associaremos os tempos de mão-de-obra à operação.

	Processo 1	Processo 2	Processo 3	Processo n
Máquina 1	10	0	4	0
Máquina 2	5	3	0	0
Máquina m	0	2	0	2

Tabela – Tabela de tempos de processamento de tarefas x máquinas
Fonte: Autor

$$RV = \frac{MC}{DR}$$

3.8.2. Ordenação dos produtos e dos processos de fabricação

Calcula relação (MC/dias restantes para entrega) para todos produtos;

Enquanto todos os produtos não tiverem uma ordem

Começa

Acha o menor valor da relação entre os produtos que ainda não possuem uma ordem;

Atribui valor para o campo ORDEM para a tabela de produtos em carteira;

Término da rotina

Acha o produto de menor ordem;

Vai para o último processo na árvore do produto atual;

Enquanto todos os processos não tiverem uma ordem

Começa

Atribui valor para o campo ORDEM da tabela de processos;

Vai para o próximo processo (sobe um pouco na árvore do produto);

Se todos os processos do produto atual possuem ordem **então**

Passa para o próximo produto;

Término da rotina

Figura – Esquema da rotina de ordenação dos produtos e dos seus respectivos processos de fabricação

Fonte: Autor

A próxima etapa do processo é atribuir um número que indica a prioridade de cada um dos produtos. A partir desta ordenação são então ordenados os processos pertencentes aos produtos. A primeira ordenação dos produtos prioriza os produtos que possuem os menores valores da relação margem de contribuição/dias restantes para sua entrega. A partir desta ordem dos produtos, ordenamos os processos a partir do processo de menor nível na árvore do produto. A rotina prossegue no sentido debaixo para cima na árvore

dos produtos até que todos os processos tenham um número de prioridade. Repete-se, então o processo para todos os produtos da carteira.

3.8.3. Alocação das operações nas máquinas

O processo de alocação das operações nas máquinas é dividida em três etapas principais: a seleção das operações que já podem ser produzidas em determinado instante, a checagem da disponibilidade de máquinas, e a alocação da operação e preparação para a alocação das operações em sequência.

3.8.3.1. Rotina de seleção das operações candidatas

Início da Programação

Vai para o primeiro processo de fabricação:

Enquanto não passar por todos os processos

Começa

*Se o processo não possui nenhum subconjunto dentro do processo **então***

Libera processo para ser produzido:

Completa campo READY TIME da primeira operação:

Vai para o próximo processo:

Fim da rotina

No meio da programação

Se não acharmos nenhuma operação sem ser produzida **então**

Começa

*Se houver processo pai **então***

Enquanto não passarmos por todos os processos filho deste

Começa

*Se o processo já foi produzido **então***

Vai para o próximo processo filho:

Senão

Para a rotina:

Fim da rotina

Processo pai do processo atual é liberado para começar a ser produzido

Fim da rotina

Figura – Esquema de rotina de seleção das operações candidatas

Fonte: Autor

A rotina de seleção das operações candidatas para serem alocadas pergunta a cada momento de decisão quais são as tarefas que podem ser alocadas. No início da programação se pergunta a cada processo se ele possui algum subconjunto dentro dele.

Caso não haja nenhum subconjunto na sua árvore do produto, então este processo é liberado para produção no momento inicial da programação. Deste modo as suas operações são liberadas para serem produzidas, porém, em qualquer momento, só serão operações candidatas para serem alocadas as primeiras operações dentre as operações que não foram produzidas. No meio da rotina de alocação das operações existe outra rotina pequena que pergunta se existe alguma operação dentro do processo que ainda não foi produzida. Caso não haja nenhuma operação que não foi produzida, faz-se uma nova consulta no processo pai para sabermos se todos os seus subconjuntos já foram produzidos.

3.8.3.2. Checagem de máquinas disponíveis

Vai para processo inicial;

Enquanto *não passar por todos os processos*

Começa

Se *existe operação que não foi produzida em processo liberado p: produção* **então**

Começa

Se *existe máquina do tipo pedido c/ campo ESTADO = "VAZIA"* **então**

Aloca Máquina;

Fim da rotina

Vai para o próximo processo;

Fim da rotina

Figura – Esquema da rotina de checagem das máquinas

Fonte: Autor

A rotina pergunta se a máquina que a operação candidata utiliza está vazia, ou então, se existe alguma máquina daquele tipo vazia no momento. Caso haja máquinas vazias, então a operação é alocada na máquina com o menor índice, isto é, se a máquina

número 1 está vazia, então a operação é alocada nesta, senão consultamos a máquina 2 e assim por diante. Caso contrário, a operação não seja alocada e a sua alocação depende da disponibilidade de máquinas no próximo instante de checagem.

3.8.3.3. Rotina de alocação das tarefas

Início da Programação

Se existe operação que não foi produzida em processo liberado p/ produção **então**

Começa

Se existe máquina do tipo pedido c/ campo ESTADO = "VAZIA" **então**

Se o READY_TIME < instante **então**

Preenche READY_TIME, COMPLETION_TIME da operação;

Guarda COMPLETION_TIME

Vai para a próxima operação;

Preenche READY_TIME da operação com o COMPLETION_TIME da anterior;

Muda ESTADO da máquina para "OCUPADA";

Preenche campos INST_ESVAZIA da máquina alocada com
COMPLETION_DATE;

Cadastra tabela de Checkpoints com COMPLETION_DATE;

Fim da rotina

Figura – Rotina de alocação de tarefas

Fonte: Autor

Esta rotina faz parte de um contexto maior da qual também faz parte a rotina anterior, de checagem das máquinas vazias. A rotina de alocação de operações começa a partir do instante em que está decidido que determinada operação deve ser alocada em uma

determinada máquina vazia. Para tanto, a rotina leva em conta o estado da máquina disputada, a ordem de precedência dos processos e a lista das operações candidatas para serem alocadas. Das ordens preferenciais para as demais ordens menos importantes, a rotina aloca a operação candidata caso a máquina disputada esteja vazia. Em caso de disputa de mais de uma operação por uma mesma máquina, é consultada a ordem de preferência dos processos dos quais fazem parte. É importante perceber que toda a alocação bem sucedida de máquina deve registrar o momento em que ela acaba na tabela de máquinas, na tabela de operações e na tabela de checkpoints. Quando o instante atual avançar para o instante em que a operação acaba de ser produzida (através da consulta a tabela Checkpoints), a máquina é liberada e a próxima operação do processo fica liberada para ser produzida.

Outro aspecto importante diz respeito à sequência das operações. Apenas uma única operação de cada processo pode ser alocada a cada instante, se houver esta possibilidade. Se o campo `READY_TIME` da operação estiver com um valor superior ao do instante atual, a operação não é alocada, pois isto significa que a operação anterior do mesmo processo ainda está sendo fabricada. Se, por outro lado, não houver máquinas vazias, a operação terá de aguardar nova consulta quando o instante atual tiver avançado para o próximo Checkpoint.

3.8.3.4. Rotina de programação

Pesquisa processos liberados para produção;

Vai para o primeiro processo;

Enquanto *houver processo não fabricado*

Começa

Enquanto *não passar por todos os processos*

Esvazia máquinas ocupadas e c/ INST_ESVAZIA = instante atual;

Começa

Se *há operações não produzidas então*

Aloca a primeira operação e anota COMPLETION_DATE nas tabelas necessárias;

Senão

Se *processo pai tiver todos subconjuntos produzidos então*

Libera processo pai para produção;

Vai para o próximo processo na lista de prioridades de processos;

Fim da rotina

Vai para o próximo instante da tabela Checkpoint;

Fim da rotina

Calcula Valor da Função Objetivo;

Grava na tabela de soluções geradas

Figura – Esquema da rotina de programação da produção

Fonte: Autor

A rotina de programação começa consultando a tabela Checkpoint para saber o primeiro instante a ser checado. Neste momento, primeiro ele esvazia as máquinas ocupadas que acabam de fabricar a ordem alocada. Depois ele consulta os processos não fabricados e um a um, na ordem da lista de prioridades de processos e tenta alocar as suas operações não produzidas. Depois que ele passa por todos os processos, consulta o próximo Checkpoint até que todos os processos de fabricação sejam produzidos e que não haja

mais nenhum Checkpoint. Só então a função objetivo é calculada a partir dos instantes em que termina a fabricação dos produtos em carteira. Enfim, a solução gerada é gravada na tabela de soluções geradas pelo programa.

3.8.4. Busca Tabu

Esta rotina tem como objetivo fazer uma busca de novas soluções para o problema de programação da produção através do re-ordenamento das seqüências dos produtos em carteira e dos seus processos de fabricação. Mexendo na sua ordem, estabeleceremos novas prioridades para a alocação de operações em máquinas disponíveis e teremos soluções geradas completamente diferentes. É preciso lembrar que o algoritmo da Busca Tabu parte de uma solução inicial e a partir desta começamos a procurar novas soluções. As etapas da Busca Tabu e o modo como configuramos o algoritmo está descrito no item 4.4.2. e pode ser resumido através da tabela seguinte.

Item	Descrição
Geração de solução inicial	Utilização da heurística construtiva própria
Geração de soluções vizinhas	Troca de Produtos
Escolha da próxima solução	Melhor solução da vizinhança
Seleção de Atributos do movimento e Regras de proibição	do m_i – o produto i não pode ser movido para posições de k , onde $k \leq \pi(i)$
Duração Tabu	Duração Tabu Ajustada – a cada vinte iterações, aumenta-se ou se diminui de uma unidade a duração das regras de proibição.
Crêterios de Aspiraço	Aceta-se nova soluço caso ela seja melhor do que a atual
Condiço de Parada	20 iteraço sem encontrar soluço melhor do que a atual

Tabela – Resumo da Busca Tabu
Fonte: Autor

Definidos os parâmetros e os itens que fazem parte desta Busca Tabu, a próxima etapa é montar a rotina para buscar novas soluções.

Enquanto (*iteração atual – iteração da melhor solução*) ≤ 20

Começa

Solução “semente” recebe melhor solução da iteração anterior(ou solução inicial);

Enquanto não testar troca da tarefa *i* com a última tarefa

Começa

Fixa nova tarefa i localizada em p(i);

Pega solução semente e fixa nova tarefa j localizada em p(j);

Pega solução semente e realiza Troca de Tarefas i e j;

Se Movimento não é Tabu **então**

Aloca Operações;

Calcula Função Objetivo;

Se Valor da nova solução é melhor do que valor da melhor solução da iteração

então

Guarda nova solução gerada;

Guarda valor da função objetivo da nova solução e iteração da melhor solução;

Registra Regra de Proibição;

*Registra nova posição máxima de i = **mi**;*

Vai para próxima tarefa j;

Termina Rotina

Figura – Rotina da Busca Tabu

Fonte: Autor

A rotina da Duração Tabu começa com uma solução “semente” gerada anteriormente e uma tarefa i localizada na posição $p(i)$. Enquanto o critério de parada não for satisfeito, isto é, a rotina continua selecionando uma tarefa i , fixada a cada iteração, e outra tarefa j , que varia a cada movimento. Em seguida, é feita a alocação das operações nas máquinas segundo a ordem de tarefas prioritárias gerada após a troca de i com j . Calcula-se o valor da função objetivo e esta é comparada com o valor da melhor solução encontrada até o momento. Caso a nova solução seja melhor, esta é guardada e a rotina prossegue até que a última tarefa j da sequência seja testada. Então, voltamos ao início da rotina e a solução inicial recebe a melhor sequência gerada até o momento. Fixamos nova tarefa $i+1$, localizada na posição $p(i+1)$ e prosseguimos com o teste de novas sequências de tarefas. Isto ocorre até que o critério de parada seja satisfeito.

Capítulo 4 – LEVANTAMENTO DE DADOS

4.1. Dados relevantes para o problema

Neste capítulo serão descritas as principais informações utilizadas para se alimentar o nosso modelo de programação da produção adotado.

Os principais tipos de informação para se começar um problema de programação e de seqüenciação da produção são: a árvore dos produtos, a ficha do processo de fabricação dos mesmos, a carteira de pedidos que se deseja tratar, além das máquinas e equipamentos que fazem parte da fábrica.

4.1.1 Árvore do Produto

As árvores dos produtos da fábrica estudada contêm todos os materiais e subconjuntos que compõem os seus produtos. Quanto aos materiais, uma das premissas em que se baseia os modelos de programação da produção e que é normalmente adotada nestes problemas diz que os materiais são considerados como disponíveis desde o início da fabricação. Neste caso, o nosso problema não apresentara quaisquer restrições sobre disponibilidade de insumos para dar início a fabricação de determinado produto ou subconjunto.

Para ilustrarmos este item, temos a seguir um exemplo:

Produto PST 104 -- Pasta c/ 50 envelopes finos v.

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	COL	Colchetes	10,40 kg	1000U
Material	DIV	CXP	Caixa p/ Embalagens CTS	30,00 U	1000U
Material	DIV	ETI	Etiquetas para visor	1,00 mil	1000U
Material	DIV	FGO	Fita Gomada	0,31 kg	1000U
Material	PAP	P40	Papelão 40	93,00 kg	1000U
Material	PVC	C10	Cristal 0,10	1,10 kg	1000U
Material	PVC	PVC	PVC Preto 0,15	68,00 kg	1000U
Subconjunto	AC	LMBCL	Lombada estreita preta	1,00 mil	1000U
Subconjunto	AC	S06	Sacos finos furados	50,00 ml	1000U

Figura – Exemplo de árvore do produto

4.1.2. Ficha do Processo de Fabricação

Um processo de fabricação de um produto é composto por diversos materiais, subconjuntos, operações e serviços de terceiros. A cada operação estão ligados uma determinada quantidade de mão-de-obra e/ou uma determinada quantidade de tempos de máquina. Esta quantidade é, na verdade, explicitada por uma relação entre dois componentes: o componente pai e o componente filho e dada por coeficientes técnicos. Estes coeficientes técnicos são a principal informação contida dentro da ficha de fabricação dos produtos da Plastpark, da qual podemos retirar os tempos de processamento necessários para cada uma das etapas de fabricação.

Um aspecto importante é que não estão cadastrados e a fábrica não possui dados referentes aos tempos de preparação de máquinas. Os tempos de “setup” não possuem uma importância significativa dentro desta fábrica, pois estes são tempos são consumidos normalmente para se fazer pequenos (e rápidos) ajustes no caso da maioria das máquinas. Além disto, o próprio gerente da Plastpark disse que podemos fazer este tipo de simplificação neste momento. Portanto, é importante dizer que ao ignorarmos os tempos de preparação de máquinas nós não invalidamos o nosso modelo de programação da produção.

Ademais, não criticaremos os coeficientes técnicos das ficha de fabricação da fábrica já cadastrados, pois a fábrica já possui este cadastro há alguns anos e o revê continuamente, atualizando-o quando existe alguma mudança em algum processo de fabricação. Esta tarefa não se encaixa dentro do escopo do trabalho, mas pode ser feita para se averiguar se os dados das fichas de fabricação condizem com a realidade e se estão atualizados.

Para ilustrar este item mostramos a seguir um exemplo:

Produto FMC 111 – Sacos Plásticos finos

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	B06	Bobina Grossa Polietileno	3,70 kg	1000U
Material	DIV	CPO	Caixa p/ Polietileno	1,00 U	1000U
Material	DIV	FGO	Fita Gomada	0,01 kg	1000U
Operação	P	CS	Corte na FMC		
MAQUI		CES	Corte e Solda na FMC	6,25min	1000U
MAQUI		COP	Compressores	6,25min	1000U
M.Obra	P	45	Operador de Csolda FMC	6,25min	1000U
M.Obra	P	47	Aux. De Csolda FMC	12,05min	1000U

Figura – Exemplo de Processo de Fabricação

4.1.3. Carteira de Produtos

A carteira de produtos é a base para se começar a fazer a programação da produção e o sequenciamento das ordens de serviço dentro da fábrica. Será montado um cadastro de pedidos estipulando:

1. Produtos que fazem parte da carteira,
2. Quantidades que devem ser produzidas,
3. Preço total da venda combinado,
4. Penalidade diária (em R\$) por dia de atraso da entrega,
5. Data de entrega combinada com o cliente, ou então a data programada para o pedido ser encaminhado aos estoques,
6. Condição de Pagamento utilizada para se fazer a venda.

Estes dados serão utilizados tanto na hora da geração de uma solução inicial, quanto no momento de se avaliar o modelo de programação de produção.

No caso da PlastPark, a sua carteira de pedidos deverá conter não só os produtos demandados pelos seus clientes, bem como os produtos que devem ser repostos no estoque da fábrica, ambos com datas para o término de sua fabricação. Porém, a reposição destes produtos para estoque não pode ser tratada com penalidades tão severas quanto com pedidos sendo fabricados para os clientes, pois aqueles representam apenas uma venda em potencial, ao passo que os pedidos de clientes já estão certos. Por outro lado, a reposição do estoque é sinônimo de segurança para o atendimento dos pedidos de pronta entrega feitos e pode ser mais primordial do que o atendimento das ordens dos clientes menores da PlastPark, dependendo dos níveis dos estoques no momento. A conclusão que se tira é que o modelo a ser adotado deve, portanto estabelecer graus de importância para o atendimento de cada uma das ordens de serviço que fazem parte da carteira de pedidos.

4.1.4. Cadastro de Máquinas e Equipamentos

A informação principal para a programação da produção que o cadastro de máquinas deve conter é o número de máquinas de cada tipo existentes e a capacidade de produção em horas por dia. Neste ponto surge uma outra simplificação do modelo decorrente de uma de nossas premissas. As máquinas de um mesmo tipo são consideradas idênticas entre si, como se fossem todas iguais e pertencentes a um centro produtivo(CP) (ex: CP de máquinas corte-solda, corte-vinco, montagem, etc..).

O cadastro das máquinas da Plastpark é mostrado no Anexo 2.

Capítulo 5 – Análise de Resultados

Este capítulo apresenta dois problemas simples para averiguar a eficiência do modelo de programação da produção e comparar os seus resultados com os resultados que seriam obtidos utilizando a programação atual da fábrica.

5.1. Descrição do Primeiro Teste

O primeiro teste propõe uma programação para uma carteira de produtos composta por três dos produtos mais vendidos na fábrica. Estes produtos são três dos produtos mais vendidos com relação a quantidade e foram escolhidos a partir do histórico de vendas da fábrica, que compreende o período de Abril/98 a Março/99. São eles:

1. **Produto1:** FMC 113 – Sacos Plásticos super Grossos
 - a. Quantidade: 100mil unidades;
 - b. Preço Total: R\$3320,00(R\$ 12,06/mil)(Margem de Contrib. = ~~R\$2494,00~~);
 - c. Estrutura de Impostos: s/ impostos;
 - d. Data de Entrega: 3 dias;
 - e. Penalidade por dia de atraso: R\$ 1660,00 (50% preço)
2. **Produto2:** FMC 111 - Sacos plásticos Finos
 - a. Quantidade: 100mil unidades;
 - b. Preço Total: R\$1206,00 (R\$12,06/mil) (Margem de Contrib. = ~~R\$570,00~~);
 - c. Estrutura de Impostos: s/ impostos;
 - d. Data de Entrega: 3 dias;
 - e. Penalidade por dia de atraso: R\$120,60 (10% preço)
3. **Produto3:** FMC 798 - Sacos Plásticos Grossos
 - a. Quantidade: 100mil unidades;

- b. Preço Total: R\$2556,00 (R\$25,56/mil) (Margem de Contrib. = R\$1590,00);
- c. Estrutura de Impostos: s/ impostos;
- d. Data de Entrega: 3 dias;
- e. Penalidade por dia de atraso: R\$255,60 (10% preço)

A ficha de processo de fabricação dos produtos é mostrada a seguir.

Produto FMC 113 – Sacos Plásticos super grossos

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	B15	Bobina Grossa Polietileno	9,50 kg	1000U
Material	DIV	CPO	Caixa p/ Polietileno	2,50 U	1000U
Material	DIV	FGO	Fita Gomada	0,02 kg	1000U
Operação	P	CS	Corte na FMC		
MAQUI		CES	Corte e Solda na FMC	4,54min	1000U
MAQUI		COP	Compressores	4,54min	1000U
M.Obra	P	45	Operador de Csolda FMC	4,54min	1000U
M.Obra	P	47	Aux. De Csolda FMC	9,08min	1000U

Tabela- Ficha de Processo de Fabricação dos plásticos super grossos

Fonte: Autor

Produto FMC 111 – Sacos Plásticos finos

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	B06	Bobina Grossa Polietileno	3,70 kg	1000U
Material	DIV	CPO	Caixa p/ Polietileno	1,00 U	1000U
Material	DIV	FGO	Fita Gomada	0,01 kg	1000U
Operação	P	CS	Corte na FMC		
MAQUI		CES	Corte e Solda na FMC	6,25min	1000U
MAQUI		COP	Compressores	6,25min	1000U
M.Obra	P	45	Operador de Csolda FMC	6,25min	1000U
M.Obra	P	47	Aux. De Csolda FMC	12,05min	1000U

Tabela- Ficha de Processo de Fabricação dos plásticos finos

Fonte: Autor

Produto FMC 798 – Sacos Plásticos grossos

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	B12	Bobina 0,12	8,00kg	1000U
Material	DIV	CPO	Caixa p/ Polietileno	2,00U	1000U
Material	DIV	FGO	Fita Gomada	0,02kg	1000U
Operação	P	CS	Corte na FMC		
MAQUI		CES	Corte e Solda na FMC	4,54min	1000U
MAQUI		COP	Compressores	4,54min	1000U
M.Obra	P	45	Operador de Csolda FMC	4,54min	1000U
M.Obra	P	47	Aux. De Csolda FMC	9,08min	1000U

Tabela- Ficha de Processo de Fabricação dos plásticos grossos

Fonte: Autor

Outro item importante é o número das máquinas envolvidas em cada um destes processos de produção.

Código	Descrição	Quantidade
CES	Corte e Solda FMC	2
COP	Compressores	3

Tabela- Máquinas envolvidas na fabricação da carteira de produtos

Fonte: Autor

O processo de fabricação destes produtos podem ser vistos anexados ao final do trabalho.

Iremos assumir as seguintes premissas para este caso:

1. Consideramos que o primeiro produto foi pedido por um dos clientes preferenciais da fábrica;
2. Admitimos que não há estoques de nenhum destes produtos;
3. Todas as máquinas estão vazias no momento inicial da programação ;
4. A fábrica funciona em um turno único, que começa às 6:00h e terminando às 19:00h;
5. A fabricação destes produtos começa no início do 1º. turno do dia 1;
6. A Taxa de Desconto utilizada para avaliar as soluções é de 0,09% ao dia (juros simples equivalentes a taxa de 2% ao mês e considerando 22 dias úteis por mês);

Calculando a função objetivo para a programação acima, temos:

$$MC_Cessante = (4 * R\$2494,00 + 3 * R\$570,00 + 4 * R\$1590,00) * 0,09\% = R\$14,66$$

$$Penalidades \text{ por atraso} = 1 * R\$120,60 + 1 * R\$255,60 = R\$376,20$$

$$\text{Função Objetivo} = R\$14,66 + R\$376,20 = R\$390,85$$

2ª. Situação: Produto3 chega antes do Produto2

Ordem de prioridade de fabricação: 1º. Produto1 (FMC-113)

2º. Produto3 (FMC-798)

3º. Produto2 (FMC-111),

o que gera a seguinte programação:

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES1	Dia 1	06:00	15:08	Dia 2	08:08
FMC	113	FMC	113	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16
FMC	111	FMC	111	1	CES1	Dia 2	08:08	20:50	Dia 3	15:58
FMC	111	FMC	111	2	COP1	Dia 3	15:58	20:50	Dia 5	10:48
FMC	798	FMC	798	1	CES2	Dia 1	06:00	15:08	Dia 2	08:08
FMC	798	FMC	798	2	COP2	Dia 2	08:08	15:08	Dia 3	10:16

Tabela – Programação da Produção para a seq.: P1-P3-P2

Fonte: Autor

Calculando a função objetivo para a programação acima, temos:

$$MC_Cessante = (3 * R\$2494,00 + 5 * R\$570,00 + 3 * R\$1590,00) * 0,09\% = R\$15,48$$

$$Penalidades \text{ por atraso} = 2 * R\$120,60 = R\$241,20$$

$$\text{Função Objetivo} = R\$13,73 + R\$241,20 = R\$254,93$$

Utilizando a relação (MC/dias restantes para a entrega do pedido), temos a seguinte ordem de prioridade de fabricação:

1°. Produto1 (FMC-113)

2°. Produto3 (FMC-798)

3°. Produto2 (FMC-111),

gerando a seguinte programação:

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES1	Dia 1	06:00	15:08	Dia 2	08:08
FMC	113	FMC	113	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16
FMC	111	FMC	111	1	CES1	Dia 2	08:08	20:50	Dia 3	15:58
FMC	111	FMC	111	2	COP1	Dia 3	15:58	20:50	Dia 5	10:48
FMC	798	FMC	798	1	CES2	Dia 1	06:00	15:08	Dia 2	08:08
FMC	798	FMC	798	2	COP2	Dia 2	08:08	15:08	Dia 3	10:16

Tabela – Programação da produção para a seq. : P1-P3-P2

Fonte: Autor

, e a função objetivo:

MC_Cessante = R\$13,73

Penalidades = R\$241,20

Função Objetivo = R\$254,93

Em seguida a solução inicial é utilizada na Busca Tabu, gerando a seguinte vizinhança:

Seqüência	MC_Cessante (R\$)	Penalidades (R\$)	Fç. Objetivo(R\$)
P3-P1-P2*	13,73	241,20	254,93
P2-P3-P1	15,48	1780,60	1796,08
P1-P2-P3	14,66	376,20	390,86

Tabela – Resultados da 1a. iteração da Busca Tabu

Fonte: Autor

** Solução geradora de nova vizinhança*

A Busca Tabu seleciona a solução P3-P1-P2 como a geradora da nova vizinhança, que gerará a seguinte vizinhança:

Seqüência	MC_Cessante (R\$)	Penalidades (R\$)	Fç. Objetivo(R\$)
P2-P1-P3*	14,66	376,20	390,86
P3-P2-P1	15,48	1780,60	1796,08

Tabela – Resultados da 2a. iteração da Busca Tabu

Fonte: Autor

** Solução geradora de nova vizinhança*

Depois da 2ª. iteração, o modelo já avaliou todas as possíveis soluções através das trocas de tarefas. Ele prossegue a Busca Tabu selecionando as melhores soluções para serem geradoras de novas vizinhanças até que a condição de parada seja satisfeita. Porém, não é possível achar solução melhor do que a solução inicial P1-P3-P2 (, cujo resultado é igual a P3-P1-P2).

A solução encontrada pelo modelo é P1-P3-P2 igual a R\$254,93, em contraste R\$390,68 da 1ª.situação e R\$254,93 da 2ª. situação, sendo que neste caso o algoritmo heurístico encontra a solução ótima para o problema. A programação da produção para cada uma das seis possíveis soluções é apresentada no Anexo 1.

5.2. Descrição do Segundo Teste

No segundo teste constam três dos produtos mais rentáveis da Plastpark. O primeiro produto inclui subconjuntos dentro do seu processo de fabricação, o que representa uma dificuldade a mais neste caso. São eles:

1. **Produto1:** PST 104 – Pasta c/ 50 envelopes finos v.

- a. Quantidade: 1mil unidades;
- b. Preço Total: R\$1720,00 (MC = R\$1133,38);
- c. Estrutura de Impostos: s/ impostos;
- d. Data de Entrega: 3 dias;
- f. Penalidade por dia de atraso: R\$860,00 (50% do preço total);

377

2. **Produto2:** FMC 113 – Sacos Plásticos super Grossos

- a. Quantidade: 100mil unidades;
- b. Preço Total: R\$1206,00 (MC = R\$207,3);
- c. Estrutura de Impostos: s/ impostos;
- d. Data de Entrega: 1 dia;
- g. Penalidade por dia de atraso: R\$120,60 (10% do preço total);

207

3. **Produto3:** CVN 342 – Divisoria 10 posições

- a. Quantidade: 3mil unidades;
- b. Preço Total: R\$1660,00 (MC = R\$689,59);
- c. Estrutura de Impostos: s/ impostos;
- d. Data de Entrega: 1 dia;
- h. Penalidade por dia de atraso: R\$166,00 (10% do preço total);

689

A ficha de processo de fabricação dos produtos é mostrada a seguir.

Produto PST 104 – Pasta c/ 50 envelopes finos v.

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	COL	Colchetes	10,40 kg	1000U
Material	DIV	CXP	Caixa p/ Embalagens CTS	30,00 U	1000U
Material	DIV	ETI	Etiquetas p/ Visor	1,00 mil	1000U
Material	DIV	FGO	Fita Gomada	0,32 kg	1000U
Material	PAP	P40	Papelão 40	93,00 kg	1000U
Material	PVC	C10	Cristal 0,10	1,10 kg	1000U
Material	PVC	PVC	PVC Preto 0,15	68,00 kg	1000U
Subconjunto	AC	LMBCL	Lombada Estr. Preta	1,00 mil	1000U
Subconjunto	AC	SO6	Sacos Finos Furados	50,00 mil	1000U
Operação	A	ACAB	Acabamento de Produtos		
M.Obra	A	OPC	Acabamento Final	190 min	1000U
Operação	F	SOLDA	Solda de Material		
MAQUI		COP	Compressores	440 min	1000U
MAQUI		SDE	Solda Eletrônica PQ	160 min	1000U
MAQUI		SDG	Solda Gde. SKWA	280 min	1000U
M.Obra	F	OPB	Operador de Solda "B"	440 min	1000U
Operação	P	CORTE	Corte de Material		
MAQUI		CES	Corte e solda FMC 1	22 min	1000U
MAQUI		COP	Compressores	22 min	1000U
MAQUI		GUI	Guilhotina	12 min	1000U
MAQUI		RCO	Risco e Corte	25 min	1000U
M.Obra	P	45	Op. de Corte e solda FMC	22 mn	1000U
M.Obra	P	47	Aux. de corte e solda FMC	22 min	1000U
M.Obra	P	OPB	Op. de corte "B"	37 min	1000U

Tabela- Ficha de Processo de Fabricação da pasta c/ 50 envelopes finos v.

Fonte: Autor

Os processos dos subconjuntos que fazem parte do processo de PST 104 são mostrados a seguir:

Subconjunto AC LMBCL– Lombada Estreita Preta

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	PVC	PVC	PVC Preto 0,15	18,00kg	1000U
Operação	A	ACAB	Acabamento de Produtos		
M.Obra	A	OPC	Acabamento Final	45 min	1000U
Operação	F	SOLDA	Solda de Material		
MAQUI		COP	Compressores	53 min	1000U
MAQUI		SDG	Solda Grande 8kW	53 min	1000U
M.Obra	F	OPB	Operador de solda “B”	53 min	1000U
Operação	P	CORTE	Corte de Material		
MAQUI		GUI	Guilhotina	40 min	1000U
M.Obra	P	OPB	Operador de Corte “B”	40 min	1000U

Tabela- Ficha de Processo de Fabricação da Lombada Estreita Preta

Fonte: Autor

Subconjunto AC S06– Sacos Finos Furados

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	B06	Bobina Fina Polietileno	3,50kg	1000U
Operação	P		Corte de Material		
MAQUI		CES	Corte e Solda FMC	7 min	1000U
MAQUI		COP	Compressores	7 min	1000U
M.Obra	P	45	Op. de Corte e solda FMC	7 min	1000U
M.Obra	P	47	Aux. de corte e solda FMC	14 min	1000U

Tabela- Ficha de Processo de Fabricação dos Sacos Finos Furados

Fonte: Autor

Produto FMC 113 – Sacos Plásticos super grossos

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	B15	Bobina Grossa Polietileno	9,50 kg	1000U
Material	DIV	CPO	Caixa p/ Polietileno	2,50 U	1000U
Material	DIV	FGO	Fita Gomada	0,02 kg	1000U
Operação	P	CS	Corte na FMC		
MAQUI		CES	Corte e Solda na FMC	4,54min	1000U
MAQUI		COP	Compressores	4,54min	1000U
M.Obra	P	45	Operador de Csolda FMC	4,54min	1000U
M.Obra	P	47	Aux. De Csolda FMC	9,08min	1000U

Tabela- Ficha de Processo de Fabricação dos plásticos super grossos

Fonte: Autor

Produto CVN 342 – Divisória de 10 Posições

Componente	Cod. Grupo	Código	Descrição	Coef. Filho	Coef. Pai
Material	DIV	ADIV	Adesivo P/ Divisória	1,00 mil	1000U
Material	DIV	CXP	Caixa p/ Embalagem CTS	8,00 U	1000U
Material	DIV	FGO	Fita Gomada	0,07 kg	1000U
Material	DIV	ISF	Índice separ. p/ fichário	1,00 mil	1000U
Material	DIV	SEM	Sacos p/ Embalagens	4,20 kg	1000U
Material	PVC	R XO	Rigex Opaco p/ divisória	199,00 kg	1000U
Operação	A	ACAB	Acabamento de produtos		
MAQUI		S01	Termosolda	11,00 min	1000U
M.Obra	A	OPC	Acabamento Final	11,00 min	1000U
Operação		CORTE	Corte de Material		
MAQUI		CES	Corte e Solda na FMC	15,00 min	1000U
MAQUI		COP	Compressores	15,00 min	1000U
MAQUI		CVI	Corte e Vinco	150,00 min	1000U
MAQUI		GUI	Guilhotina	48,00 min	1000U
M.Obra	P	45	Operador de Csolda FMC	15,00 min	1000U
M.Obra	P	47	Aux. De Csolda FMC	30,00 min	1000U
M.Obra	P	OPB	Operador de Corte "B"	198,00min	1000U

Tabela- Ficha de Processo de Fabricação dos plásticos super grossos

Fonte: Autor

Outro item importante é o número das máquinas envolvidas em cada um destes processos de produção.

Código	Descrição	Quantidade
CES	Corte e Solda FMC	2
COP	Compressores	3
CVI	Corte e Vinco	1
GUI	Guilhotina	2
RCO	Risco e Corte	1
RFI	Refiladeira	1
SDE	Solda Eletrônica pq.	1
SDG	Solda Grande 8kW	13

Tabela- Máquinas envolvidas na fabricação da carteira de produtos

As premissas da programação da produção são as mesmas do teste passado.

Considerando o primeiro pedido feito por um cliente preferencial, e de acordo com a ordem de chegada dos demais pedidos, teríamos as seguintes situações pela lógica de programação feita na Plastpark:

1ª. Situação: Produtos 2, 3 chegam nesta ordem

Ordem de prioridade de fabricação: 1º. Produto 1 (PST - 104)

2º. Produto2 (FMC - 113)

3º. Produto3 (CVN - 342).

gerando a sequência de prioridade de fabricação: P1 - P2 - P3, e a seguinte programação da produção:

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
PST	104	PST	104	1	OPC	Dia 1	17:40	03:10	Dia 2	07:50
PST	104	PST	104	2	COP1	Dia 2	07:50	07:20	Dia 2	15:10
PST	104	PST	104	3	SDE1	Dia 2	15:10	02:40	Dia 2	17:50
PST	104	PST	104	4	SDG1	Dia 2	17:50	04:40	Dia 3	09:30
PST	104	PST	104	5	CES1	Dia 3	09:30	07:20	Dia 3	16:50
PST	104	PST	104	6	COP1	Dia 3	16:50	00:22	Dia 3	17:12
PST	104	PST	104	7	GUI1	Dia 3	17:12	00:22	Dia 3	17:34
PST	104	PST	104	8	RCO1	Dia 3	17:34	00:25	Dia 3	17:59
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES1	Dia 1	06:00	05:50	Dia 1	11:50
PST	104	AC	SO6	2	COP2	Dia 1	11:50	05:50	Dia 1	17:40
FMC	113	FMC	113	1	CES2	Dia 1	06:00	07:24	Dia 1	13:24
FMC	113	FMC	113	2	COP1	Dia 1	07:38	07:24	Dia 1	15:02
CVN	342	CVN	342	1	CES1	Dia 1	11:50	00:33	Dia 1	12:23
CVN	342	CVN	342	2	SO1	Dia 1	12:23	00:33	Dia 1	12:56
CVN	342	CVN	342	3	CES1	Dia 1	12:56	00:33	Dia 1	13:29
CVN	342	CVN	342	4	COP1	Dia 1	13:29	00:45	Dia 1	14:14
CVN	342	CVN	342	5	CVI1	Dia 1	14:14	07:30	Dia 2	08:44

Tabela – Programação da sequência P1-P2-P3

Fonte: Autor

Calculando a função objetivo para a programação acima, temos:

$$MC_Cessante = (3 * R\$1133,38 + 1 * R\$207,3 + 2 * R\$689,59) * 0,09\% = R\$4,53$$

$$Penalidades \text{ por atraso} = 1 * R\$166,00 = R\$166,00$$

$$Função \text{ Objetivo} = R\$4,53 + R\$166,00 = R\$170,53$$

2ª. Situação: Produtos 3, 2 chegam nesta ordem

Ordem de prioridade de fabricação: 1º. Produto1 (PST – 104)

2º. Produto3 (CVN - 342)

3º. Produto3 (FMC -113),

gerando a sequência de prioridade de fabricação: P1 - P3 - P2, e a seguinte programação da produção:

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
PST	104	PST	104	1	OPC	Dia 1	17:40	03:10	Dia 2	07:50
PST	104	PST	104	2	COP1	Dia 2	07:50	07:20	Dia 2	15:10
PST	104	PST	104	3	SDE1	Dia 2	15:10	02:40	Dia 2	17:50
PST	104	PST	104	4	SDG1	Dia 2	17:50	04:40	Dia 3	09:30
PST	104	PST	104	5	CES1	Dia 3	09:30	07:20	Dia 3	16:50
PST	104	PST	104	6	COP1	Dia 3	16:50	00:22	Dia 3	17:12
PST	104	PST	104	7	GUI1	Dia 3	17:12	00:22	Dia 3	17:34
PST	104	PST	104	8	RCO1	Dia 3	17:34	00:25	Dia 3	17:59
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES1	Dia 1	06:00	05:50	Dia 1	11:50
PST	104	AC	SO6	2	COP2	Dia 1	11:50	05:50	Dia 1	17:40
FMC	113	FMC	113	1	CES2	Dia 1	06:33	07:24	Dia 1	13:57
FMC	113	FMC	113	2	COP1	Dia 1	13:57	07:24	Dia 2	08:21
CVN	342	CVN	342	1	CES2	Dia 1	06:00	00:33	Dia 1	06:33
CVN	342	CVN	342	2	SO1	Dia 1	06:33	00:33	Dia 1	07:06
CVN	342	CVN	342	3	CES2	Dia 1	07:06	00:33	Dia 1	07:39
CVN	342	CVN	342	4	COP1	Dia 1	07:39	00:45	Dia 1	08:24
CVN	342	CVN	342	5	CVI1	Dia 1	08:24	07:30	Dia 1	15:54

Tabela – Programação da ~sequência P1-P3-P2

Fonte: Autor

Calculando a função objetivo para a programação acima, temos:

$$MC_Cessante = (3 \cdot R\$1133,38 + 2 \cdot R\$207,3 + 1 \cdot R\$689,59) \cdot 0,09\% = R\$4,09$$

$$\text{Penalidades por atraso} = 1 \cdot R\$120,60;$$

$$\text{Função Objetivo} = R\$124,69$$

, e utilizando o modelo de programação da produção, temos:

Solução Inicial

A sequência da solução inicial é dada por P3 - P1 - P2, dada pela relação MC/dias restantes p/ fabricação, conforme mostra a tabela abaixo:

No.	Produto	Código	MC/Dias Restantes
P1	PST	104	377,79
P2	FMC	113	207,30
P3	CVN	342	689,59

Tabela – Relação (MC Dias Restantes)

Fonte: Autor

, e a programação da produção seria:

Código Gr.	Código Produto	Código Gr.	Código Processo	Operação	Máquina	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
PST	104	PST	104	1	OPC	Dia 1	17:40	03:10	Dia 2	07:50
PST	104	PST	104	2	COP	Dia 2	07:50	07:20	Dia 2	15:10
PST	104	PST	104	3	SDE	Dia 2	15:10	02:40	Dia 2	17:50
PST	104	PST	104	4	SDG	Dia 2	17:50	04:40	Dia 3	09:30
PST	104	PST	104	5	CES	Dia 3	09:30	07:20	Dia 3	16:50
PST	104	PST	104	6	COP	Dia 3	16:50	00:22	Dia 3	17:12
PST	104	PST	104	7	GUI	Dia 3	17:12	00:22	Dia 3	17:34
PST	104	PST	104	8	RCO	Dia 3	17:34	00:25	Dia 3	17:59
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES2	Dia 1	06:00	05:50	Dia 1	11:50
PST	104	AC	SO6	2	COP	Dia 1	11:50	05:50	Dia 1	17:40
FMC	113	FMC	113	1	CES1	Dia 1	06:33	07:24	Dia 1	13:57
FMC	113	FMC	113	2	COP2	Dia 1	13:57	07:24	Dia 2	08:21
CVN	342	CVN	342	1	CES1	Dia 1	06:00	00:33	Dia 1	06:33
CVN	342	CVN	342	2	SO1	Dia 1	06:33	00:33	Dia 1	07:06
CVN	342	CVN	342	3	CES	Dia 1	07:06	00:33	Dia 1	07:39
CVN	342	CVN	342	4	COP1	Dia 1	07:39	00:45	Dia 1	08:24
CVN	342	CVN	342	5	CVI	Dia 1	08:24	07:30	Dia 1	15:54

Tabela – Programação da sequência P3-P1-P2

Fonte: Autor

Calculando a função objetivo para a programação anterior, temos:

$$MC_Cessante = (3 * R\$1133,38 + 2 * R\$207,3 + 1 * R\$689,59) * 0,09\% = R\$4,09$$

$$Penalidades \text{ por atraso} = 1 * R\$120,60 = R\$120,60;$$

$$Função \text{ Objetivo} = R\$4,09 + R\$120,60 = R\$124,69;$$

Em seguida a solução inicial é utilizada na Busca Tabu, gerando a seguinte vizinhança:

Seqüência	MC_Cessante (R\$)	Penalidades (R\$)	Fç. Objetivo(R\$)
P1-P3-P2*	4,09	120,60	124,69
P2-P1-P3	4,53	166,00	170,53
P3-P2-P1	4,72	286,60	291,32

Tabela – Resultados da 1a. iteração da Busca Tabu

Fonte: Autor

** Solução geradora de nova vizinhança*

A Busca Tabu seleciona a solução P1-P3-P2 como a geradora da nova vizinhança, que gerará a seguinte vizinhança:

Seqüência	MC_Cessante (R\$)	Penalidades (R\$)	Fç. Objetivo(R\$)
P2-P1-P3*	4,53	166,00	170,53
P1-P2-P3	4,53	166,00	170,53

Tabela – Resultados da 2a. iteração da Busca Tabu

Fonte: Autor

** Solução geradora de nova vizinhança*

Como há um empate entre as duas soluções, a Busca adota a primeira solução como a geradora de novas vizinhanças.

Com esta solução, o modelo gera uma nova vizinhança, cujos resultados são mostrados a seguir:

Seqüência	MC_Cessante (R\$)	Penalidades (R\$)	Fç. Objetivo(R\$)
P3-P1-P2(Tabu)	4,09	120,60	124,69
P2-P3-P1*	4,09	120,60	124,69

Tabela – Resultados da 2a. iteração da Busca Tabu

Fonte: Autor

** Solução geradora de nova vizinhança*

Como o primeiro movimento ainda é considerado como Tabu, pois esta armazenado na memória adaptativa, a solução geradora de nova vizinhança será P2-P3-P1.

Depois da 3^a. iteração, o modelo já avaliou todas as possíveis soluções através das trocas de tarefas. Ele prossegue a Busca Tabu selecionando as melhores soluções para serem geradoras de novas vizinhanças até que a condição de parada seja satisfeita. Porém, não é possível achar solução melhor do que a solução inicial P1-P3-P2 (, cujo resultado é igual a P2-P3-P1).

A solução encontrada pelo modelo é P1-P3-P2 igual a R\$124,69, em contraste R\$170,53 da 1^a.situação e R\$124,69 da 2^a. situação, sendo que neste caso o algoritmo heurístico encontra a solução ótima para o problema. A programação da produção para cada uma das seis possíveis soluções é apresentada no Anexo 2.

5.3. Análise dos resultados dos testes

Podemos concluir pelos resultados dos testes que o modelo gera soluções tão boas ou melhores do que as soluções obtidas pela fábrica estudada.

Teste1	1a. Situação	2a. Situação	Média
Fc_Objetoivo	390,86	254,93	322,90
Modelo			254,93

Tabela- Comparação entre os resultados do modelo e da fábrica do primeiro teste

Fonte: Autor

Teste2	1a. Situação	2a. Situação	Média
Fc_Objetoivo	170,53	124,69	147,61
Modelo			124,69

Tabela- Comparação entre os resultados do modelo e da fábrica do primeiro teste

Fonte: Autor

Como visto acima, podemos perceber que em média as soluções da Plastpark são piores do que as soluções geradas pelo modelo. Nos testes anteriores, o modelo indicou a existência de duas solução diferentes, que possuíam o melhor resultado encontrado.

Capítulo 6 – CONCLUSÕES

6.1. Autocrítica do Modelo Adotado

Este item tem por objetivo apontar possíveis falhas ou pontos a serem melhorados em um novo modelo de programação da produção

O primeiro ponto é a questão dos tempos de preparação de máquinas. Estes tempos são relevantes no momento de se fazer a alocação das tarefas nas máquinas e podem alterar as soluções obtidas pelo modelo, principalmente caso estes tempos sejam relativamente longos ou em grande número. Uma consequência da inclusão dos tempos de preparação de máquinas é a alteração dos custos reais de fabricação dos produtos, visto que a preparação de máquinas implica muitas vezes em um custo fixo de mão-de-obra e/ou ferramentas antes do início da operação. No nosso modelo, consideramos os custos variáveis de fabricação para um lote médio de fabricação, pois este é o tipo de lote considerado no momento da inclusão dos coeficientes técnicos na ficha do processo de fabricação.

Porém, existem casos em que o tempo de preparação de uma determinada máquina é dependente da última operação que foi realizada nesta máquina. Desta forma, teríamos que incluir uma matriz de tempos de preparação para cada uma das máquinas, além de um controle sobre os dados da fábrica que não possuímos atualmente na fábrica estudada.

Um outro ponto a ser discutido seria possibilitar a divisão de um determinado produto em diversos lotes menores, criando a possibilidade de utilizarmos mais de uma máquina fabricando dois ou mais lotes de um mesmo produto simultaneamente.

6.2. Conclusões

O objetivo do trabalho foi alcançado, pois definimos um modelo de programação da produção que obteve bons resultados, e, ao mesmo tempo, representa bem a forma como se realiza a programação da produção em indústrias intermitentes repetitivas, bem como no caso real estudado. Quanto aos resultados obtidos, pudemos verificar que nos dois testes realizados o modelo chegou a uma solução igual ou melhor do que as soluções que poderíamos chegar utilizando a lógica da fábrica estudada.

Com relação a representatividade do modelo, chegamos a conclusão junto com os professores e com o gerente da fábrica que o modelo é aplicável ao caso real estudado. Porém, visando atender a uma maior gama de indústrias intermitente repetitivas, apontamos no último capítulo algumas críticas sobre o modelo, além de pontos de melhoria a serem estudados para um próximo modelo.

BIBLIOGRAFIA

1. Morton, Thomas E. e Pentico, David W., "Heuristic Scheduling Systems. With Applications to Production Systems and Project Management", John Wiley & Sons, Inc, 1993.
2. Baker, Kenneth R., "Elements of Sequencing and Scheduling", 1995.
3. Pinedo, Michael, "Scheduling – Theory, Algorithms and Systems", Prentice Hall, 1995.
4. Santoro, Miguel Cezar, "Modelo de Programação para produção intermitente com composição de produtos variável no tempo", São Paulo, 1982.
5. Santoro, Miguel Cezar, "Apostila de Planejamento, Programação e Controle da Produção – Planejamento Agregado", São Paulo, 1997.
6. Santoro, Miguel Cezar, "Apostila de PRO – 760 – Sistemas de Produção II", São Paulo, 1996.
7. Schoueri, Ricardo, "Ambiente de Desenvolvimento de Algoritmos Genéricos para Programação da Produção Intermitente", São Paulo, 1998.
8. Ronconi, Débora P., "Uma contribuição para o estudo do problema de flowshop com buffer ilimitado e zero para minimizar a soma dos atrasos", Campinas, 1985.
9. Sacchi, Luis H., "Algoritmos genéticos para minimização de makespan em um flow shop flexível", Campinas, 1997.

ANEXO 1

Programação da Produção

Do

Primeiro Teste

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES2	Dia 1	06:00	15:08	Dia 2	08:08
FMC	113	FMC	113	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16
FMC	111	FMC	111	1	CES1	Dia 1	06:00	20:50	Dia 2	13:50
FMC	111	FMC	111	2	COP2	Dia 2	13:50	20:50	Dia 4	08:40
FMC	798	FMC	798	1	CES2	Dia 2	08:08	15:08	Dia 3	10:16
FMC	798	FMC	798	2	COP1	Dia 3	10:16	15:08	Dia 4	12:24

Seqüência P2/P1/P3

MC_Cessante 14,66

Penalidade 376,20

FcObjetivo 390,86

Tabela – Solução da seq. P2-P1-P3

Fonte: Autor

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES2	Dia 2	08:08	15:08	Dia 3	10:16
FMC	113	FMC	113	2	COP1	Dia 3	10:16	15:08	Dia 4	12:24
FMC	111	FMC	111	1	CES1	Dia 1	06:00	20:50	Dia 2	13:50
FMC	111	FMC	111	2	COP2	Dia 2	13:50	20:50	Dia 4	08:40
FMC	798	FMC	798	1	CES2	Dia 1	06:00	15:08	Dia 2	08:08
FMC	798	FMC	798	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16

Seqüência P2/P3/P1

MC_Cessante 15,48

Penalidade 1.780,60

FcObjetivo 1.796,08

Tabela – Solução da seq. P2-P3-P1

Fonte: Autor

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES1	Dia 1	06:00	15:08	Dia 2	08:08
FMC	113	FMC	113	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16
FMC	111	FMC	111	1	CES2	Dia 1	06:00	20:50	Dia 2	13:50
FMC	111	FMC	111	2	COP2	Dia 2	13:50	20:50	Dia 4	08:40
FMC	798	FMC	798	1	CES1	Dia 2	08:08	15:08	Dia 3	10:16
FMC	798	FMC	798	2	COP1	Dia 3	10:16	15:08	Dia 4	12:24

Seqüência P1/P2/P3

MC_Cessante 14,66

Penalidade 376,20

FcObjetivo 390,86

Tabela – Solução da seq. P1-P2-P3

Fonte: Autor

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES1	Dia 2	08:08	15:08	Dia 3	10:16
FMC	113	FMC	113	2	COP2	Dia 3	10:16	15:08	Dia 4	12:24
FMC	111	FMC	111	1	CES2	Dia 1	06:00	20:50	Dia 2	13:50
FMC	111	FMC	111	2	COP1	Dia 2	13:50	20:50	Dia 4	08:40
FMC	798	FMC	798	1	CES1	Dia 1	06:00	15:08	Dia 2	08:08
FMC	798	FMC	798	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16

Seqüência P3/P2/P1

MC_Cessante 15,48

Penalidade 1.780,60

FcObjetivo 1.796,08

Tabela – Solução da seq. P3-P2-P1

Fonte: Autor

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES1	Dia 1	06:00	15:08	Dia 2	08:08
FMC	113	FMC	113	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16
FMC	111	FMC	111	1	CES1	Dia 2	08:08	20:50	Dia 3	15:58
FMC	111	FMC	111	2	COP1	Dia 3	15:58	20:50	Dia 5	10:48
FMC	798	FMC	798	1	CES2	Dia 1	06:00	15:08	Dia 2	08:08
FMC	798	FMC	798	2	COP2	Dia 2	08:08	15:08	Dia 3	10:16

Seqüência P1/P3/P2

MC_Cessante 13,73

Penalidade 241,20

FcObjetivo 254,93

Tabela – Solução da seq. P1-P3-P2

Fonte: Autor

Codigo Gr. Produto	Codigo Produto	Codigo Gr. Processo	Codigo Processo	Operação	Máquina Alocada	Ready Date	Ready Time	Processing Time (h)	Completion Date	Completion Time
FMC	113	FMC	113	1	CES2	Dia 1	06:00	15:08	Dia 2	08:08
FMC	113	FMC	113	2	COP2	Dia 2	08:08	15:08	Dia 3	10:16
FMC	111	FMC	111	1	CES1	Dia 2	08:08	20:50	Dia 3	15:58
FMC	111	FMC	111	2	COP1	Dia 3	15:58	20:50	Dia 5	10:48
FMC	798	FMC	798	1	CES1	Dia 1	06:00	15:08	Dia 2	08:08
FMC	798	FMC	798	2	COP1	Dia 2	08:08	15:08	Dia 3	10:16

Seqüência P3/P1/P2

MC_Cessante 13,73

Penalidade 241,20

FcObjetivo 254,93

Tabela – Solução da seq. P3-P1-P2

Fonte: Autor

ANEXO 2

Programação da Produção

Do

Segundo Teste

1-2-3-

Codigo Gr.	Codigo	Codigo Gr.	Codigo	Operação	Máquina	Ready	Ready	Processing	Completion	Completion
Produto	Produto	Processo	Processo		Alocada	Date	Time	Time (h)	Date	Time
PST	104	PST	104	1	OPC	Dia 1	17:40	03:10	Dia 2	07:50
PST	104	PST	104	2	COP1	Dia 2	07:50	07:20	Dia 2	15:10
PST	104	PST	104	3	SDE1	Dia 2	15:10	02:40	Dia 2	17:50
PST	104	PST	104	4	SDG1	Dia 2	17:50	04:40	Dia 3	09:30
PST	104	PST	104	5	CES1	Dia 3	09:30	07:20	Dia 3	16:50
PST	104	PST	104	6	COP1	Dia 3	16:50	00:22	Dia 3	17:12
PST	104	PST	104	7	GUI1	Dia 3	17:12	00:22	Dia 3	17:34
PST	104	PST	104	8	RCO1	Dia 3	17:34	00:25	Dia 3	17:59
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES1	Dia 1	06:00	05:50	Dia 1	11:50
PST	104	AC	SO6	2	COP2	Dia 1	11:50	05:50	Dia 1	17:40
FMC	113	FMC	113	1	CES2	Dia 1	06:00	07:24	Dia 1	13:24
FMC	113	FMC	113	2	COP1	Dia 1	07:38	07:24	Dia 1	15:02
CVN	342	CVN	342	1	CES1	Dia 1	11:50	00:33	Dia 1	12:23
CVN	342	CVN	342	2	SO1	Dia 1	12:23	00:33	Dia 1	12:56
CVN	342	CVN	342	3	CES1	Dia 1	12:56	00:33	Dia 1	13:29
CVN	342	CVN	342	4	COP1	Dia 1	13:29	00:45	Dia 1	14:14
CVN	342	CVN	342	5	CVI1	Dia 1	14:14	07:30	Dia 2	08:44

MC_Cessante

4,53

P1

3

Penalidade

166,00

P2

1

FcObjetivo

170,53

P3

2

1-3-2

Codigo Gr.	Codigo	Codigo Gr.	Codigo	Operação	Máquina	Ready	Ready	Processing	Completion	Completion
Produto	Produto	Processo	Processo		Alocada	Date	Time	Time (h)	Date	Time
PST	104	PST	104	1	OPC	Dia 1	17:40	03:10	Dia 2	07:50
PST	104	PST	104	2	COP1	Dia 2	07:50	07:20	Dia 2	15:10
PST	104	PST	104	3	SDE1	Dia 2	15:10	02:40	Dia 2	17:50
PST	104	PST	104	4	SDG1	Dia 2	17:50	04:40	Dia 3	09:30
PST	104	PST	104	5	CES1	Dia 3	09:30	07:20	Dia 3	16:50
PST	104	PST	104	6	COP1	Dia 3	16:50	00:22	Dia 3	17:12
PST	104	PST	104	7	GUI1	Dia 3	17:12	00:22	Dia 3	17:34
PST	104	PST	104	8	RCO1	Dia 3	17:34	00:25	Dia 3	17:59
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES1	Dia 1	06:00	05:50	Dia 1	11:50
PST	104	AC	SO6	2	COP2	Dia 1	11:50	05:50	Dia 1	17:40

FMC	113	FMC	113	1	CES2	Dia 1	06:33	07:24	Dia 1	13:57
FMC	113	FMC	113	2	COP1	Dia 1	13:57	07:24	Dia 2	08:21
CVN	342	CVN	342	1	CES2	Dia 1	06:00	00:33	Dia 1	06:33
CVN	342	CVN	342	2	SO1	Dia 1	06:33	00:33	Dia 1	07:06
CVN	342	CVN	342	3	CES2	Dia 1	07:06	00:33	Dia 1	07:39
CVN	342	CVN	342	4	COP1	Dia 1	07:39	00:45	Dia 1	08:24
CVN	342	CVN	342	5	CVI1	Dia 1	08:24	07:30	Dia 1	15:54

MC_Cessante 4,09
 Penalidade 120,60
 FcObjetivo 124,69

P1 3
 P2 2
 P3 1

2-1-3

Codigo Gr.	Codigo	Codigo Gr.	Codigo	Operação	Máquina	Ready	Ready	Processing	Completion	Completion
Produto	Produto	Processo	Processo		Alocada	Date	Time	Time (h)	Date	Time
PST	104	PST	104	1	OPC	Dia 1	17:40	03:10	Dia 2	07:50
PST	104	PST	104	2	COP1	Dia 2	07:50	07:20	Dia 2	15:10
PST	104	PST	104	3	SDE1	Dia 2	15:10	02:40	Dia 2	17:50
PST	104	PST	104	4	SDG1	Dia 2	17:50	04:40	Dia 3	09:30
PST	104	PST	104	5	CES1	Dia 3	09:30	07:20	Dia 3	16:50
PST	104	PST	104	6	COP1	Dia 3	16:50	00:22	Dia 3	17:12
PST	104	PST	104	7	GUI1	Dia 3	17:12	00:22	Dia 3	17:34
PST	104	PST	104	8	RCO1	Dia 3	17:34	00:25	Dia 3	17:59
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES2	Dia 1	06:00	05:50	Dia 1	11:50
PST	104	AC	SO6	2	COP1	Dia 1	11:50	05:50	Dia 1	17:40
FMC	113	FMC	113	1	CES1	Dia 1	06:00	07:24	Dia 1	13:24
FMC	113	FMC	113	2	COP2	Dia 1	13:24	07:24	Dia 1	20:48
CVN	342	CVN	342	1	CES2	Dia 1	11:50	00:33	Dia 1	12:23
CVN	342	CVN	342	2	SO1	Dia 1	12:23	00:33	Dia 1	12:56
CVN	342	CVN	342	3	CES2	Dia 1	12:56	00:33	Dia 1	13:29
CVN	342	CVN	342	4	COP3	Dia 1	13:29	00:45	Dia 1	14:14
CVN	342	CVN	342	5	CVI1	Dia 1	14:14	07:30	Dia 2	08:44

MC_Cessante 4,53
 Penalidade 166,00
 FcObjetivo 170,53

P1 3
 P2 1
 P3 2

2-3-1

Codigo Gr.	Codigo	Codigo Gr.	Codigo	Operação	Máquina	Ready	Ready	Processing	Completion	Completion
Produto	Produto	Processo	Processo		Alocada	Date	Time	Time (h)	Date	Time
PST	104	PST	104	1	OPC	Dia 1	18:13	03:10	Dia 2	08:23
PST	104	PST	104	2	COP	Dia 2	08:23	07:20	Dia 2	15:43
PST	104	PST	104	3	SDE	Dia 2	15:43	02:40	Dia 2	18:23
PST	104	PST	104	4	SDG	Dia 2	18:23	04:40	Dia 3	10:03

PST	104	PST	104	5	CES	Dia 3	10:03	07:20	Dia 3	17:23
PST	104	PST	104	6	COP	Dia 3	17:23	00:22	Dia 3	17:45
PST	104	PST	104	7	GUI	Dia 3	17:45	00:22	Dia 3	18:07
PST	104	PST	104	8	RCO	Dia 3	18:07	00:25	Dia 3	18:32
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES2	Dia 1	06:33	05:50	Dia 1	12:23
PST	104	AC	SO6	2	COP1	Dia 1	12:23	05:50	Dia 1	18:13
FMC	113	FMC	113	1	CES1	Dia 1	06:00	07:24	Dia 1	13:24
FMC	113	FMC	113	2	COP2	Dia 1	13:24	07:24	Dia 2	07:48
CVN	342	CVN	342	1	CES2	Dia 1	06:00	00:33	Dia 1	06:33
CVN	342	CVN	342	2	SO1	Dia 1	06:33	00:33	Dia 1	07:06
CVN	342	CVN	342	3	CES2	Dia 1	07:06	00:33	Dia 1	07:39
CVN	342	CVN	342	4	COP	Dia 1	07:39	00:45	Dia 1	08:24
CVN	342	CVN	342	5	CVI	Dia 1	08:24	07:30	Dia 1	15:54

MC_Cessante	4,09	P1	3
Penalidade	120,60	P2	2
FcObjetivo	124,69	P3	1

3-1-2

Codigo Gr.	Codigo	Codigo Gr.	Codigo	Operação	Máquina	Ready	Ready	Processing	Completion	Completion
Produto	Produto	Processo	Processo	Alocada	Date	Time	Time (h)	Date	Time	Time
PST	104	PST	104	1	OPC	Dia 1	17:40	03:10	Dia 2	07:50
PST	104	PST	104	2	COP	Dia 2	07:50	07:20	Dia 2	15:10
PST	104	PST	104	3	SDE	Dia 2	15:10	02:40	Dia 2	17:50
PST	104	PST	104	4	SDG	Dia 2	17:50	04:40	Dia 3	09:30
PST	104	PST	104	5	CES	Dia 3	09:30	07:20	Dia 3	16:50
PST	104	PST	104	6	COP	Dia 3	16:50	00:22	Dia 3	17:12
PST	104	PST	104	7	GUI	Dia 3	17:12	00:22	Dia 3	17:34
PST	104	PST	104	8	RCO	Dia 3	17:34	00:25	Dia 3	17:59
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES2	Dia 1	06:00	05:50	Dia 1	11:50
PST	104	AC	SO6	2	COP	Dia 1	11:50	05:50	Dia 1	17:40
FMC	113	FMC	113	1	CES1	Dia 1	06:33	07:24	Dia 1	13:57
FMC	113	FMC	113	2	COP2	Dia 1	13:57	07:24	Dia 2	08:21
CVN	342	CVN	342	1	CES1	Dia 1	06:00	00:33	Dia 1	06:33
CVN	342	CVN	342	2	SO1	Dia 1	06:33	00:33	Dia 1	07:06
CVN	342	CVN	342	3	CES	Dia 1	07:06	00:33	Dia 1	07:39
CVN	342	CVN	342	4	COP1	Dia 1	07:39	00:45	Dia 1	08:24
CVN	342	CVN	342	5	CVI	Dia 1	08:24	07:30	Dia 1	15:54

MC_Cessante	4,09	P1	3
-------------	------	----	---

Penalidade	120,60	P2	2
FcObjetivo	124,69	P3	1

3-2-1

Codigo Gr.	Codigo	Codigo Gr.	Codigo	Operação	Máquina	Ready	Ready	Processing	Completion	Completion
Produto	Produto	Processo	Processo		Alocada	Date	Time	Time (h)	Date	Time
PST	104	PST	104	1	OPC	Dia 1	18:13	03:10	Dia 2	08:23
PST	104	PST	104	2	COP	Dia 2	08:23	07:20	Dia 2	15:43
PST	104	PST	104	3	SDE	Dia 2	15:43	02:40	Dia 2	18:23
PST	104	PST	104	4	SDG	Dia 2	18:23	04:40	Dia 3	10:03
PST	104	PST	104	5	CES	Dia 3	10:03	07:20	Dia 3	17:23
PST	104	PST	104	6	COP	Dia 3	17:23	00:22	Dia 3	17:45
PST	104	PST	104	7	GUI	Dia 3	17:45	00:22	Dia 3	18:07
PST	104	PST	104	8	RCO	Dia 3	18:07	00:25	Dia 3	18:32
PST	104	AC	LMBCL	1	OPC	Dia 1	06:00	00:45	Dia 1	06:45
PST	104	AC	LMBCL	2	COP1	Dia 1	06:45	00:53	Dia 1	07:38
PST	104	AC	LMBCL	3	SDG1	Dia 1	07:38	00:53	Dia 1	08:31
PST	104	AC	LMBCL	4	GUI1	Dia 1	08:31	00:40	Dia 1	09:11
PST	104	AC	SO6	1	CES1	Dia 1	06:33	05:50	Dia 1	12:23
PST	104	AC	SO6	2	COP2	Dia 1	12:23	05:50	Dia 1	18:13
FMC	113	FMC	113	1	CES2	Dia 1	06:00	07:24	Dia 1	13:24
FMC	113	FMC	113	2	COP1	Dia 1	12:56	07:24	Dia 2	07:20
CVN	342	CVN	342	1	CES1	Dia 1	06:00	00:33	Dia 1	06:33
CVN	342	CVN	342	2	SO1	Dia 1	06:33	00:33	Dia 1	07:06
CVN	342	CVN	342	3	CES1	Dia 1	12:23	00:33	Dia 1	12:56
CVN	342	CVN	342	4	COP1	Dia 1	12:56	00:45	Dia 1	13:41
CVN	342	CVN	342	5	CVI1	Dia 1	13:41	07:30	Dia 2	08:11

MC_Cessante	4,72	P1	3
Penalidade	286,60	P2	2
FcObjetivo	291,32	P3	2

ANEXO 3

Cadastro

De

Máquinas

Cadastro de Máquinas

Empresa Empresa Teste Ltda.

Pág.1

Data:18/11/99

		Preço	Eficiência	Produtividade	Vida Util	Quantidade
219	SOLDA AUTOMATICA	70.000,00	100%	4,32 h	15	1
220	SOLDA TAVOLA REDONDA	65.000,00	100%	0,92 h	20	2
BOT	BOTONEIRA	2.400,00	80%	8 h	1	2
CES	CORTE E SOLDA FMC I	85.000,00	100%	7,57 h	20	2
CHI	REBITADEIRA MECANICA	12.000,00	100%	0,34 h	10	2
COP	COMPRESSORES	4.000,00	100%	20,9 h	10	3
CVI	CORTE E VINCO	20.000,00	100%	1,84 h	10	1
Célula A	Máquina 220, 221.	70.000,00	100%	8 h	20	2
Célula B	Máquinas 202, 206, 208, 209, 210, 211, 217, 218.	14.000,00	80%	8 h	10	8
Célula C	Máquinas 219.	70.000,00	100%	8 h	20	1
Célula D	Máquinas 201, 207, 212.	14.000,00	80%	8 h	15	3
Célula E	Máquinas 214, 215, 216.	14.000,00	80%	8 h	15	3
FUR	FURADEIRA	1.000,00	10%	8 h	5	2
GUI	GUILHOTINA	11.000,00	100%	4,4 h	10	2
MES	MESA DE CORTE	150,00	100%	8 h	10	1
MTD	MANTEADEIRA	5.000,00	100%	0,2 h	10	1
RCO	RISCO E CORTE	5.000,00	100%	3,07 h	5	1
REB	REBITADEIRA PNEUMATICA	500,00	100%	0,21 h	10	2
RES	FECHADEIRA DE ESTOJO	100,00	100%	6 h	25	2
RFI	REFILADEIRA	2.500,00	100%	3,9 h	10	1
SDE	SOLDA ELETRONICA PQ	15.000,00	100%	5,18 h	10	1
SDG	SOLDA GRANDE 8KWA	17.000,00	100%	2,53 h	10	13
SO1	TERMOSOLDA	2.000,00	100%	3,36 h	10	1