

THAIS SAYURI NAKAGAWA

THIAGO ALESSIO

*nota final*

*9.4 (nove e quatro)*

*Nay*

OTIMIZAÇÃO DE LINHA DE PRODUÇÃO DE CABINAS POR MEIO DE  
HEURÍSTICA APLICADA EM SISTEMA ACCESS COM LINGUAGEM  
VISUAL BASIC

Monografia apresentada à Escola  
Politécnica da Universidade de São Paulo  
para obtenção de Título em Engenharia

São Paulo

2004

THAIS SAYURI NAKAGAWA

THIAGO ALESSIO

OTIMIZAÇÃO DE LINHA DE PRODUÇÃO DE CABINAS POR MEIO DE  
HEURÍSTICA APLICADA EM SISTEMA ACCESS COM LINGUAGEM  
VISUAL BASIC

Monografia apresentada à Escola  
Politécnica da Universidade de São Paulo  
para obtenção de Título em Engenharia

Área de concentração:  
Engenharia Mecatrônica

Orientador: Prof. Dr.  
Marcos Ribeiro Pereira Barreto

São Paulo  
2004

## **RESUMO**

Os sistemas de manufatura vêm se apresentando cada vez mais como sendo uma das peças mais importantes para a obtenção de processos e grau de atendimento a clientes cada vez mais elevados. Eles desejam que o produto encomendado esteja pronto no dia programado, pois o atraso, por exemplo, de um componente de uma geladeira provoca prejuízos que dificilmente conseguirão ser recuperados, pois ela deixará de ser produzida naquele momento. Além disso, a diminuição do tempo de fabricação de um produto acarreta na diminuição de custos do próprio produtor e a otimização na utilização de seu maquinário provoca um desgaste muito menor nas ferramentas e também diminui a quantidade de *setups* das máquinas, o que acarreta menores riscos aos trabalhadores e um tempo de produção ainda menor. Tendo isto em mente, neste trabalho, propõe-se a obtenção da melhor seqüência de montagem em uma linha de produção com mais de um produto, diversas máquinas e roteiros de montagem diferentes. Os objetivos desta linha de produção será o do atendimento dentro do prazo com o menor tempo de fabricação possível. Para tanto, serão utilizadas técnicas de scheduling visando à maximização da produção e a minimização do tempo de fabricação. À partir da técnica de scheduling mais apropriada ao problema, propõe-se o projeto e a posterior implementação de um sistema, utilizando o MS Access e a linguagem de programação visual basic, que ordena a produção de forma a atingir os objetivos descritos anteriormente.

## SUMÁRIO

1. INTRODUÇÃO .....	1
2. OBJETIVO .....	2
3. FAMILIARIZAÇÃO COM O ACCESS .....	3
3.1. Tabelas.....	3
3.2. Consultas .....	3
3.3. Formulários.....	4
3.4. Relatórios.....	4
3.5. Macros .....	4
3.6. Módulos.....	4
3.6.1. Módulo classe.....	4
3.6.2. Módulo padrão .....	5
4. ESPECIFICAÇÃO DE SISTEMA.....	6
4.1. Montagem Bruta: .....	7
4.2. Pintura: .....	8
4.3. Montagem Final: .....	9
5. TÉCNICAS DE SCHEDULING.....	11
5.1. O que é scheduling? .....	11
5.1.1. Uma idéia geral .....	11
5.1.2. O que é um bom scheduling? .....	12
5.2. Modelo Dinâmico de Uma Máquina.....	12
5.2.1. Estrutura e Modelo de Entradas .....	12
5.2.2. Procedimento Geral de Scheduling .....	13
5.2.3. Medidas de Saída Primárias .....	14
5.2.4. Funções de Objetivo Comuns .....	15
5.3. Agrupando atividades.....	18
5.4. Agrupando recursos.....	18
5.4.1. Recursos Múltiplos: Routing .....	19
5.5. Reconfiguração de scheduling .....	20
5.6. Questões de sistemas de informação .....	21
5.7. Resumo .....	22
5.8. Níveis de Scheduling.....	23
5.8.1. Introdução .....	23
5.8.2. Nível 1: Planejamento em longo prazo.....	24
5.8.3. Nível 2: Planejamento em médio prazo.....	26
5.8.4. Nível 3: Planejamento em curto prazo.....	26
5.8.5. Nível 4: Scheduling .....	28
5.8.6. Nível 5: Scheduling/controle reativo.....	28
5.9. Tipos de Ambientes de Scheduling.....	29
5.9.1. Linha de trabalhos (job shop) clássica.....	29
5.9.2. Linha de trabalhos (job shop) aberta .....	30
5.9.3. Batch shop.....	31
5.9.4. Flow shop.....	31
5.9.5. Batch/flow shops .....	32

5.9.6. Célula de manufatura.....	32
6. SCHEDULING DE FLOW SHOPS: MAKESPAN.....	34
6.1. Makespan genérico.....	34
6.1.1. Visão geral .....	34
6.1.2. Heurísticas para m máquinas .....	34
6.1.3. Problema de gargalo de uma máquina.....	36
6.2. Tabu Search .....	37
6.2.1. Introdução .....	37
6.2.2. Utilização da memória.....	38
6.2.3. Funcionamento .....	40
6.2.4. Tabu Search Canônico .....	41
7. MODELAGEM UML.....	43
7.1. Fases do desenvolvimento de um sistema em UML .....	44
7.1.1. Análise de requisitos.....	44
7.1.2. Análise .....	44
7.1.3. Design (Projeto) .....	45
7.1.4. Programação.....	45
7.1.5. Testes .....	45
8. PROJETO.....	46
8.1. Análise de requisitos .....	46
8.1.1. Metodologia de sequenciamento.....	46
8.1.2. Adequação do Tabu Search.....	47
8.1.3. Casos de uso do sistema.....	48
8.2. Análise .....	49
8.3. Design.....	50
8.3.1. Telas.....	51
8.3.2. Funções / Transições .....	52
8.3.3. Tabelas.....	52
8.3.4. Relatórios .....	53
8.3.5. Código.....	53
9. RESULTADOS .....	54
10. CONCLUSÃO.....	55
11. BIBLIOGRAFIA .....	56
ANEXO A.....	57
ANEXO B.....	58

## **LISTA DE FIGURAS**

Figura 1 – Estruturas de memória do Tabu Search .....	39
Figura 2 – Seqüência inicial de produtos.....	47
Figura 3 – Seqüências resultantes de permutação .....	48
Figura 4 – Diagrama de casos de uso .....	49
Figura 5 – Diagrama de classes.....	50
Figura 6 – Diagrama de seqüência .....	51

## **LISTA DE TABELAS**

Tabela 1 – Classificação de níveis de scheduling .....	24
--	----

## 1. INTRODUÇÃO

A fabricação de bens manufaturados em uma indústria pode ser feita por dois tipos básicos de fabricação. Em uma delas, os postos de trabalhos e máquinas estão dispostos de forma distribuída pela fábrica e os produtos passam pelas máquinas de acordo com as suas necessidades. Na outra, as máquinas e equipamentos estão dispostos de maneira linear, de forma que todos os produtos passam por todas as máquinas e estas se readaptam, por exemplo, com troca de ferramentas, de forma a se adaptar ao produto e para que se consiga produzir todos os tipos de produtos nesta linha de produção.

Os produtos podem ainda ter que passar, por linhas de produção diferentes, e estas diferentes linha podem ter diferentes tempos de fabricação. Nestes casos tem-se a necessidade de se planejar a produção de forma a produzir a maior quantidade no menor tempo possível de forma a atender aos requisitos da produção.

Desta forma, o planejamento da produção tem que ser feita da maneira ótima e para isto, existem diversas técnicas que visam o sequenciamento da produção. E este é o foco do projeto, que tem como objetivo a obtenção da melhor seqüência de produção de acordo com as restrições da linha. Para a implementação deste projeto, serão utilizados um banco de dados (MS Access) e a técnica de sequenciamento que melhor se aplicar ao caso que será estudado como exemplo.

## **2. OBJETIVO**

Elaboração de um software com o auxilio de um banco de dados (MS-Access) que realize automaticamente o sequenciamento da produção. Este seqüência de produção será feita de acordo com a técnica de sequenciamento escolhida e dos dados da linha a ser estudada.

### **3. FAMILIARIZAÇÃO COM O ACCESS**

Escolheu-se o MS-Access como a base para o desenvolvimento do programa. Por isto a necessidade de se analisar e conhecer seus recursos.

O MS-Access pode ser dividido em:

#### **3.1. Tabelas**

É a base do banco de dados já que nelas ficam armazenadas as informações que serão armazenadas. O banco de dados pode ter várias tabelas, cada uma armazenando o dado e o tipo de informação conforme o que for definido pelo usuário.

#### **3.2. Consultas**

Utilizadas para a filtragem e organização dos dados contidos nas tabelas para que se obtenha somente os dados desejados. As consultas podem obter dados de uma única tabela ou de várias tabelas relacionadas.

Quando os dados são obtidos de uma única tabela, geralmente utiliza-se somente um filtro, como por exemplo, quantas vendas foram efetuadas com mais de 10 itens. Já com dados relacionados de duas tabelas, utiliza-se a vinculação de dados para se obter dados mais completos sobre o dados desejado, como por exemplo, quantos itens, qual vendedor, em que dia, em qual caixa a venda foi efetuada.

As consultas também podem efetuar cálculos com os dados obtidos, adicionar, excluir ou atualizar os dados em tabelas.

No MS-Access, as consultas podem ser desenvolvidas tanto em ambiente de programação SQL quanto em um ambiente mais familiar, com a visualização das tabelas e dados que podem ser obtidos de maneira mais “amigável” e com o próprio programa transformando a consulta em linguagem SQL.

### **3.3. Formulários**

Os formulários são utilizados principalmente para se realizar a comunicação entre programa e usuário. Ele obtém os dados que o usuário quer e abre consultas e ou relatórios com os dados obtidos de acordo com o desejado pelo usuário. Também pode ser utilizado como uma maneira mais amigável para a inserção, edição e exclusão de registros (dados).

### **3.4. Relatórios**

Apresentam os dados obtidos das consultas na forma de um relatório. Estes relatórios podem apresentar os dados na forma de texto, gráfico ou com vários sub-relatórios.

### **3.5. Macros**

Utilizadas principalmente para a automatização das tarefas, as macros podem, entre outras funções, abrir relatórios, executar códigos (módulos), exibir caixas de mensagem, abrir outros formulários entre outros eventos.

### **3.6. Módulos**

Os módulos são utilizados para a manipulação dos dados e também para a automatização de tarefas. Podem abrir consultas e tabelas para manipulação dos dados, assim como executar novas consultas a partir de SQLs criadas exclusivamente para eles. Muito mais poderosos que as macros, utilizam-se da linguagem de programação VB (Visual Basic) e podem ser divididos em dois tipos:

#### **3.6.1. Módulo classe**

Utilizado dentro dos formulários e relatórios para realizar tarefas quando de eventos como a abertura e formatação do formulário ou relatório. Pode

realizar tarefas como alteração de cor do relatório conforme o dados que foi obtido ou a execução de um evento.

### 3.6.2. Módulo padrão

O módulo padrão contém procedimentos de uso geral que não estão associados a nenhum outro objeto e procedimentos utilizados com freqüência que podem ser executados a partir de qualquer local do seu banco de dados. A principal diferença entre um módulo padrão e um módulo classe que não está associada a nenhum objeto específico é o escopo e a vida útil. O valor de qualquer variável ou constante que seja declarada ou exista em um módulo classe sem um objeto associado está disponível para uso somente enquanto esse código estiver sendo executado e apenas a partir desse objeto ao contrário do módulo padrão, onde variáveis podem estar disponíveis para qualquer outro módulo desde que declarada como variável global.

#### **4. ESPECIFICAÇÃO DE SISTEMA**

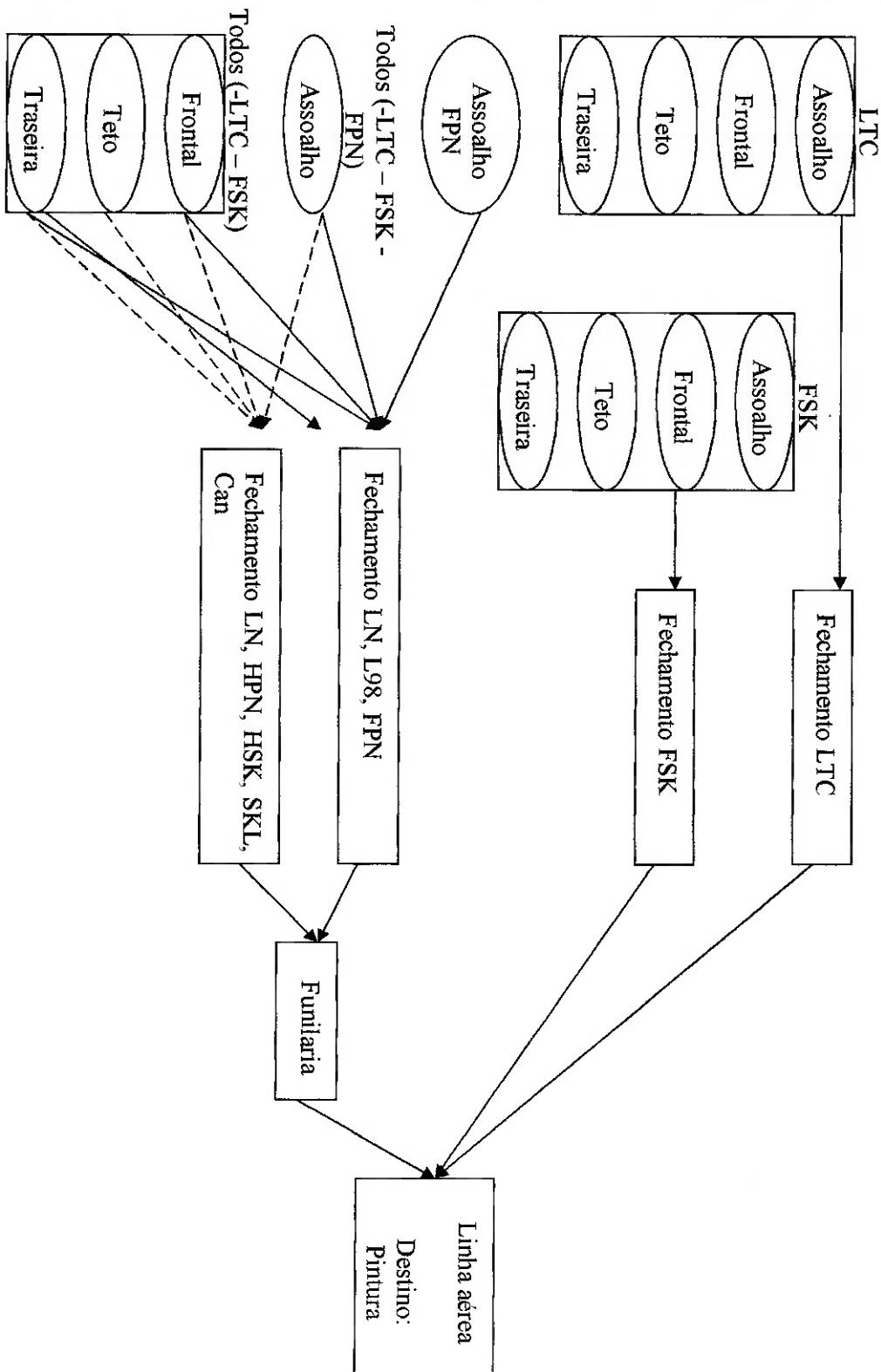
Foi estudado como um modelo de produção a linha de fabricação de cabinas da empresa DaimlerChrysler do Brasil Itda.. Este modelo servirá como a base para o desenvolvimento do produto.

Neste estudo, dividimos os produtos fabricados nas seguintes famílias de acordo com a proximidade de manufatura dentre os produtos.

- LN
- L98
- FPN
- HPN
- HSK
- SKL
- CANAVIEIRO
- FSK
- LTC

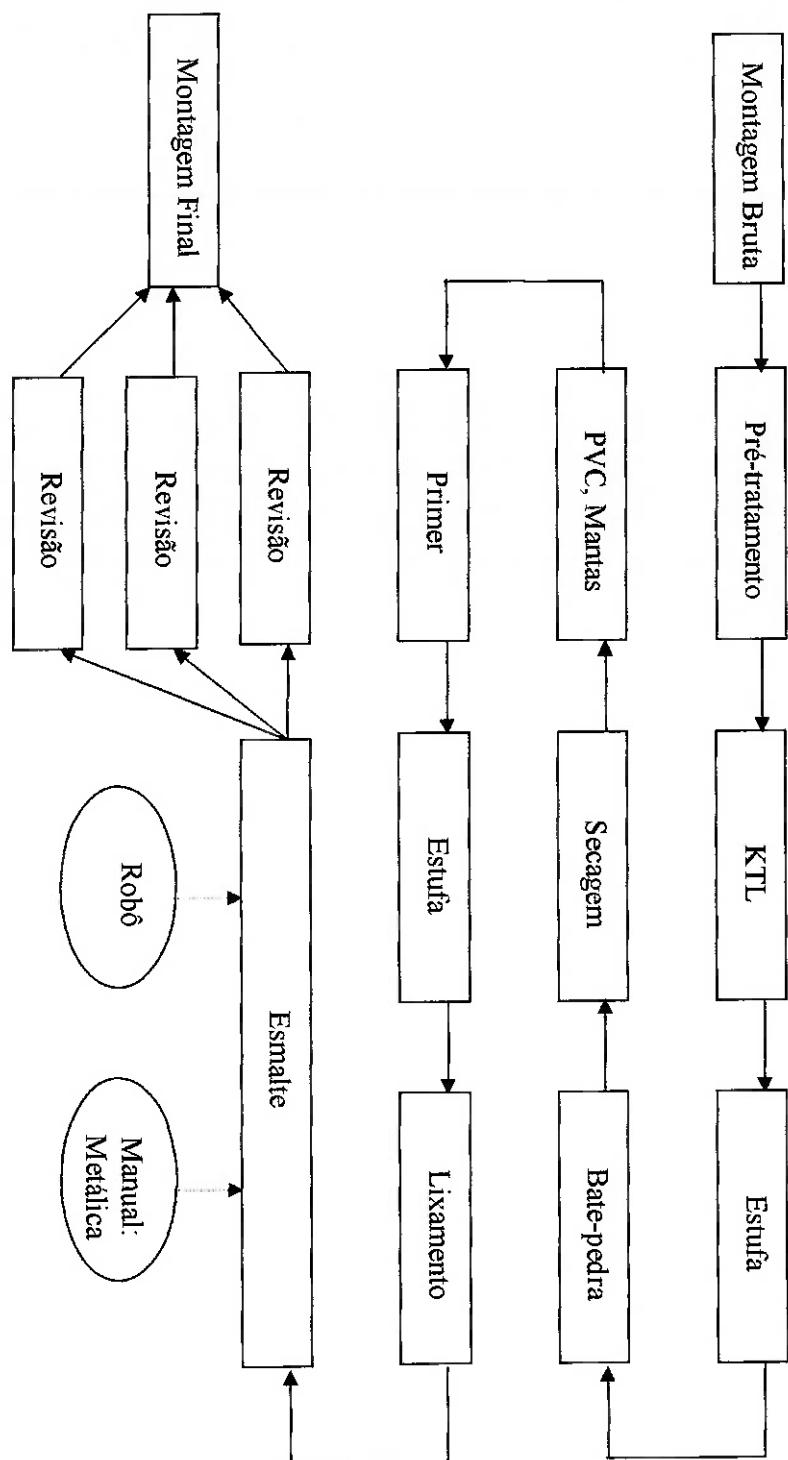
E estes modelos seguem o seguinte percurso de fabricação.

#### 4.1. Montagem Bruta:

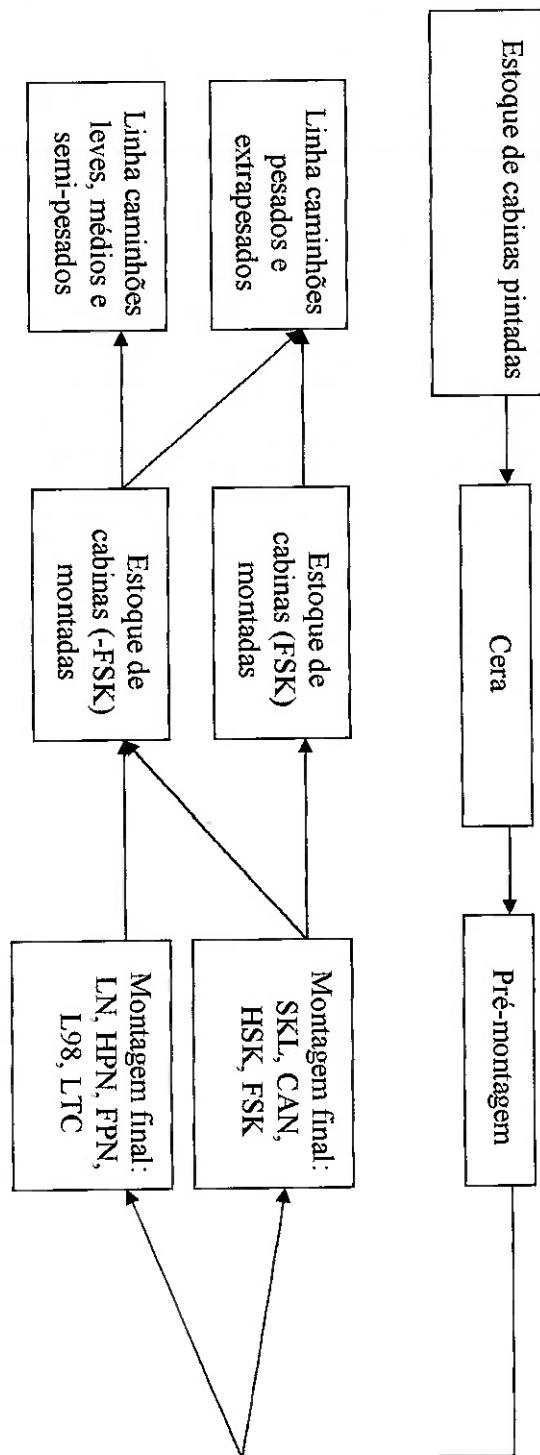


#### 4.2. Pintura:

Todos os modelos.



#### 4.3. Montagem Final:



Temos como objetivo da linha, o fornecimento de cabines para as linhas de montagem de caminhões (duas) e para exportação de cabines avulsas. Para tanto é necessário diminuir o tempo de montagem total que hoje gira em torno de três dias e consiga fornecer as cabines necessárias a serem montadas no dia.

Para efetuarmos o sequenciamento, utilizaremos os tempos de produção de cada família de cabines listada em cada posto de montagem ao longo da linha de produção. Por questão de sigilo, os tempos obtidos serão utilizados na simulação do funcionamento do sistema, porém não serão publicados.

Assim, definimos a análise do sistema a ser estudado e dos objetivos a serem alcançados pelo programa. Com isto em mãos, é possível encontrar a técnica de sequenciamento que melhor se adeque ao nosso sistema. Para isto, é feita uma análise das técnicas de Scheduling disponíveis a seguir.

## 5. TÉCNICAS DE SCHEDULING

### 5.1. O que é scheduling?

#### 5.1.1. Uma idéia geral

Scheduling ocorre em uma vasta gama de atividades econômicas. Sempre envolve realizar inúmeras funções que atam vários recursos por períodos de tempo. Os recursos são limitados. As funções a serem realizadas podem ser chamadas “jobs” (trabalhos) ou “projects” (projetos) ou “assignments” e são compostas de partes elementares chamadas “activities” (atividades) ou “operations” e “delays”. Cada atividade requer certos montantes de recursos específicos por um tempo especificado chamado de “process time”. Os recursos também possuem partes elementares, chamadas “machines” (máquinas), “cells” (células), “transport”, “delays” e assim por diante.

Problemas de scheduling são geralmente complicados por um grande número de variáveis relacionando atividades umas com as outras, recursos com atividades e com si mesmos, e também recursos ou atividades com eventos externos ao sistema. Por exemplo, pode haver variáveis de precedência conectando atividades, que especificam que atividades precedem outras, e qual a diferença de tempo entre elas, ou se pode haver sobreposição. Ou duas atividades específicas interferir entre si e não poderem utilizar o mesmo recurso simultaneamente durante certas partes do dia ou na mesma atividade. Ou um recurso pode não ficar disponível durante intervalos específicos dados por manutenção ou uso fora do sistema. Como essas inter-relações complexas podem dificultar uma solução exata ou mesmo aproximada de um problema de scheduling amplo, é natural tentar resolver primeiro uma versão mais simples do problema para ganhar percepção. Então se pode testar a susceptibilidade dessa solução para a complexidade desejada e encontrar soluções aproximadas para problemas difíceis onde a complexidade é fundamental.

### 5.1.2. O que é um bom scheduling?

Achar um bom objetivo a maximizar ou minimizar num problema de scheduling pode ser difícil por muitas razões. Primeiro, objetivos importantes como satisfação do cliente com qualidade ou prontidão são difíceis de quantificar e não figuram nos números contábeis. Segundo, uma linha geralmente lida com três tipos de objetivos, que são um tanto diferentes:

- a) Maximizar a produção num determinado período de tempo;
- b) Satisfazer a necessidade dos clientes por qualidade e prontidão;
- c) Minimizar os custos adicionais.

Várias abordagens são possíveis:

- a) Resolver problemas com um objetivo por vez;
- b) Resolver com comprometimento de objetivos;
- c) Combinar objetivos por designar valores às necessidades do cliente e à não utilização.

## 5.2. Modelo Dinâmico de Uma Máquina

Lidando com atributos de trabalhos e atividades para o modelo dinâmico de uma máquina, nós precisamos distinguir entre informações previamente conhecidas (ou previsões) e informações que são geradas como resultado de decisões de scheduling. Informações previamente conhecidas servem como dados de entrada para o procedimento do scheduling.

### 5.2.1. Estrutura e Modelo de Entradas

I1 – A máquina processa atividades uma de cada vez serialmente.

I2 – O recurso está disponível no intervalo de scheduling  $t_s$  a  $t_e$  (geralmente assumimos  $t_s=0$ ).

I3 –  $n$  trabalhos (atividades) de uma operação chegam durante o intervalo (cedo o bastante para todos terminarem em  $t_e$ ).

I4 – Entrada da atividade: tempo de processamento  $p_j$ .

I5 – Entrada da atividade: ready time  $r_j$  ( $=0$  para o caso estático).

I6 – Entrada da atividade: data de entrega  $d_j$ .

I7 – Escolha de modelo de prevenção: (a) sem prevenção, atividades terminam sem interrupção; (b) prevenção, atividades podem ser interrompidas sem penalizações.

I8 – Escolha de função de objetivo: (a) regular, todas as atividades preferem terminar antes do que depois; (b) não regular, tipos remanescentes de funções de objetivos.

Alguns pontos sobre o modelo e suas entradas.

- Assumimos que a máquina fica disponível durante o intervalo e deve terminar todos os trabalhos no final do intervalo.
- Assumimos que as entradas são determinísticas, ou ao menos terem sido previstas satisfatoriamente, e não dependem da seqüência de processamento. (Atividades podem ter setup, contanto que não dependam de atividades processadas previamente. O tempo de processamento dado é entendido como incluindo o tempo de processamento direto e o tempo de setup da atividade.)
- Datas de entrega devem ser absolutas, ou as datas de entrega podem ser violadas pelo pagamento de uma penalidade. A ausência de datas de entrega pertinentes é facilmente resolvida; por exemplo, as datas de entrega podem simplesmente serem movimentadas à direita do intervalo ou à esquerda como for apropriado.

Agora que temos a estrutura e as entradas do problema, nos voltamos para a real mecânica do scheduling do problema, e então para medidas de saída simples para ajudar a avaliarmos o scheduling.

#### 5.2.2. Procedimento Geral de Scheduling.

Todos scheduling possíveis para o problema dinâmico com ou sem prevenção e com ou sem objetivos regulares podem ser formados por repetirmos os seguintes passos:

- S1 – Escolher o tempo em que vamos remover a atividade corrente do recurso.
- S2 – Escolher a próxima atividade a ser processada.
- S3 – Escolher o tempo em que colocaremos essa atividade no recurso.
- S4 – Repetir os passos S1 a S3 até todas as atividades estarem completas.

Seguir esses passos produz um scheduling completo para os processos posteriores. Logicamente, algumas condições devem estar satisfeitas.

- C1 – Uma atividade não pode ser processada antes de sua chegada.
- C2 – Uma atividade não pode ser processada depois de sua finalização.
- C3 – Duas atividades não podem ser processadas ao mesmo tempo.

Note que no caso sem prevenção elimina a alternativa S1; a remoção de um trabalho  $j$  é sempre depois de  $p_j$  unidades de tempo de seu início.

### 5.2.3. Medidas de Saída Primárias

- P1 – Tempo de finalização  $C_j$  – tempo em que a atividade  $j$  é processada.
- P2 – Tempo de fluxo  $F_j$  – montante de tempo que a atividade  $j$  gasta no sistema.

$$F_j = C_j - r_j$$

- P3 – Demora  $L_j$  – montante de tempo (positivo ou negativo) pelo qual a finalização da atividade  $j$  excede sua data de entrega.

$$L_j = C_j - d_j$$

- P4 – Atraso  $T_j$  – a demora da atividade  $j$  se esta é positiva; se a demora da atividade  $j$  não for positiva, o atraso é zero.

$$T_j = \max\{0, L_j\}$$

- P5 – Adiantamento  $E_j$  – o negativo da demora da atividade  $j$  se a demora for negativa; se a demora é positiva, o adiantamento é zero.

$$E_j = \max\{0, -L_j\}$$

Qual é o significado dessas medidas? P1, tempo de finalização, é por si só a única saída necessária do scheduling, já que todas as outras são determinadas por ele.

P2, tempo de fluxo, mede a resposta do sistema à demanda individual por serviço. O tempo de fluxo é o montante de tempo entre a chegada e a saída do trabalho no sistema. Claramente o tempo de fluxo de uma atividade está intimamente relacionada com seu custo de work-in-process (WIP).

P3, demora, mede a conformidade do scheduling com uma data de entrega. A demora recompensa trabalhos por estarem adiantados como também pune por estarem atrasados.

P4, atraso, reflete o fato que, em muitas situações, penalidades distintas e outros custos estarão associados com atraso positivo, mas nenhuma penalidade ou benefício serão associados com atraso negativo.

P5, adiantamento, reflete o fato de que em algumas situações pode haver penalidades por terminar as atividades antes do tempo devido.

Scheduling é geralmente avaliado por quantidades agregadas que resumem o desempenho do scheduling para todos os trabalhos. Três tipos de agregações são as mais comuns:

- Uma média ponderada das medidas primárias de saída.
- O máximo ou mínimo de alguma das medidas primárias.
- Alguma combinação de (a) e (b).

A média ponderada é a ponderação da importância relativa da atividade: um peso  $w_j$  para o trabalho  $j$  implica um peso  $w_j$  na função de objetivo. Estamos preparados para verificar funções de objetivo comuns, que usam as saídas primárias acima.

#### 5.2.4. Funções de Objetivo Comuns

$$F1 - \text{Makespan:} \quad C_{\max} = \max_j \{C_j\}$$

$$F2 - \text{Fluxo de tempo ponderado:} \quad F_{wt} = \sum_j w_j F_j$$

$$F3 - \text{Demora ponderada:} \quad L_{wt} = \sum_j w_j L_j$$

$$F4 - \text{Atraso ponderado:} \quad T_{wt} = \sum_j w_j T_j$$

$$F5 - \text{Máximo fluxo de tempo: } F_{\max} = \max_j \{F_j\}$$

$$F6 - \text{Máxima demora: } L_{\max} = \max_j \{L_j\}$$

$$F7 - \text{Máximo atraso: } T_{\max} = \max_j \{T_j\}$$

F8 – Número ponderado de

$$\text{trabalhos atrasados: } N_{wt} = \sum_j w_j \delta(T_j)$$

onde  $\delta(x) = 1$  se  $x > 0$

$\delta(x) = 0$  do contrário

F9 – Adiantamento ponderado

$$\text{mais atraso ponderado: } ET_{wt} = \sum_j (w_{Ej} E_j + w_{Tj} T_j)$$

Precisamos dar algumas explicações sobre os objetivos acima.

Makespan: O scheduling com o menor makespan freqüentemente é tido como o substituto para o scheduling com a mais alta utilização. A idéia intuitiva é que terminar antes a série de atividades que se tem a executar vai permitir o início de outras atividades mais cedo. Essa abordagem é uma ótima aproximação para o caso de uma máquina e uma primeira etapa útil para outros problemas. No entanto, em problemas com múltiplos processamentos e múltiplos recursos, o makespan não dá crédito a recursos que finalizam bem antes que os outros.

Fluxo de tempo ponderado e demora ponderada: Fluxo de tempo ponderado e demora ponderada são ambos objetivos muito populares por uma variedade de razões. Primeiro, nós veremos que ambos são muito fáceis de se usar; suas soluções ótimas são idênticas e muito intuitivas. (Por isso que não fizemos distinção de seus pesos, usando  $w_j$  para os dois.) Segundo, eles são robustos no sentido que scheduling que são ótimos para eles freqüentemente produzem bons scheduling para problemas com objetivos algo diferentes.

Atraso ponderado / Outras medidas de atraso: Em muitas situações o atraso ponderado é um bom objetivo, mas problemas que usam esse objetivo são muito

difícis de ser resolvidos exatamente. A abordagem da dinâmica de gargalo provê fortes heurísticas para esse objetivo. O objetivo de adiantamento-atraso ponderados é também muito importante quando os clientes não querem trabalhos atrasados mas não aceitam entregas antes da hora. Esse objetivo é o único não-regular que apresentamos até agora. O objetivo de número ponderado de trabalhos atrasados reflete a situação de quando o cliente simplesmente se recusa a receber trabalhos atrasados, perdendo-se assim o pedido. Note que os pesos para fluxo de tempo ( $w_j$ ), atraso ( $w_{Tj}$ ), adiantamento ( $w_{Ej}$ ) e número de atrasos ( $w_{Nj}$ ) são distintos.

**Máximos Fluxo de Tempo, Demora, Atraso:** Minimizar o máximo atraso é importante quando os clientes toleram pequenos atrasos mas se tornam rapidamente mais irritados para atrasos maiores. Minimizar a máxima demora, é primariamente importante porque é um problema relativamente fácil e pode ser usado como ajuda para resolver outros problemas. Por exemplo, minimizar o máximo fluxo de tempo pode ser conseguido por definir as datas de entrega igual aos tempos de chegada e minimizando a máxima demora. Minimizar o máximo atraso pode ser similarmente conseguido por minimizar a máxima demora e truncar para zero se o valor for negativo. Finalmente, existe um uso técnico para o objetivo da máxima demora para resolver problemas de múltiplos recursos com um objetivo de makespan. Se nós definirmos uma data de entrega arbitrária para todos os trabalhos do problema, então minimizar a máxima demora vai minimizar o makespan.

**Pesos ( $w_j$ ):** Algumas vezes, por falta de bons dados, todos os pesos são definidos como sendo  $1/n$ . Para indicar esse caso, nós usaremos, por exemplo,  $F_{av}$  ao invés de  $F_{wt}$ . Infelizmente, essa prática, apesar de ocasionalmente necessária, tende a obscurecer as recompensas e penalidades econômicas fundamentais inerentes a uma situação de produção. A dificuldade em estimar  $w_j$  não denigre a urgência de estimá-lo o melhor possível. Usando  $F_{av}$  assumimos que todas as penalidades são iguais, que é uma declaração muito forte.

O peso  $w_j$  representa o custo de atrasarmos a atividade  $j$  por, digamos, um dia. Esse peso deveria se relacionar com o valor da atividade (ou o valor adicionado); também deveria se relacionar com a importância do cliente. Alguns acham que

estimar  $w_j$  é mais fácil se for escrito como  $w_j = D_j A_j$ , onde  $D_j$  é o valor monetário da atividade e  $A_j$  é o fator de prioridade do cliente.

**Pesos ( $w_{Ej}$ ,  $w_{Tj}$ ):** Note que existem dois pesos para cada trabalho no problema de adiantamento/atraso ponderado: um para o custo diário de finalizar adiantado e outro para o custo diário de finalizar atrasado. Algumas suposições simplificadoras podem ser úteis, como assumir que a razão entre peso de adiantamento e de atraso é o mesmo para todo trabalho.

### 5.3. Agrupando atividades

É natural perguntar: Quando um grupo de atividades é um job ou project e quando são vários? Da mesma forma, quando uma série de máquinas é um único recurso e quando são vários? Para responder à primeira questão, existem algumas atividades de especial interesse: aquelas cuja finalização permite a saída de um produto completo da linha e assim traz um benefício monetário. Nós podemos chamar essa atividade de “finalização de produto” ou de “finalização de projeto” ou de “marco de projeto”. Outras atividades são de interesse indireto: sua conclusão permite o início de outra atividade, a qual finalmente pode ter a conclusão do produto. São chamadas de “atividades intermediárias”. A conclusão do produto, junto com todas suas atividades intermediárias de apoio, são chamadas job. Similarmente, uma conclusão de projeto e suas atividades de apoio são chamadas projeto ou, se parte de um projeto maior, um “marco”.

### 5.4. Agrupando recursos

Definir recursos simples ou múltiplos é um pouco mais complicado. O recurso simples básico é definido por não haver decisões internas a ser feitas e uma única fila de atividades em processo, atividades aguardando na fila, e atividades ainda por vir. Para ser considerado um recurso simples, as únicas decisões a serem feitas ao longo do tempo são:

- a) A próxima atividade a ser iniciada;

- b) O tempo de início da atividade no recurso;
- c) O tempo de conclusão da atividade para liberar o recurso.

Para ilustrar a idéia, uma única fresa é um recurso por definição. Dado uma série de atividades na espera, a próxima atividade a ser iniciada é uma decisão de sequenciamento. Normalmente, o tempo de conclusão é simplesmente o tempo do final da última atividade, depois da qual o material sendo usinado é removido. Assim, existe uma única decisão: “a próxima atividade a ser iniciada”.

Agora considere duas fresas idênticas. Se existem duas filas, e cada máquina é programada separadamente, então existem dois recursos. Mas se existe uma única fila com o controle que o próximo trabalho vai na primeira máquina disponível, isso reduz a situação novamente a um problema de apenas um recurso, com a única decisão a ser feita é o sequenciamento dos trabalhos em espera.

Se muitos recursos estão envolvidos na conclusão de uma dada atividade mais ou menos simultaneamente, então, para aquela atividade, precisamos determinar o tempo inicial e final de cada recurso e seu tempo de conclusão resultante. Isso pode se tornar bem complicado, principalmente se uma atividade requer tempos e montantes diferentes nos diversos recursos ou se atividades distintas são variadas quanto às suas necessidades de recursos.

#### 5.4.1. Recursos Múltiplos: Routing

No mais geral sistema multi-recursos, uma atividade ou série de atividades pode ter várias escolhas de recursos para realizar sua conclusão. Por exemplo, uma polia pode ser fresada numa nova célula FMS, ou pode ser torneada mais devagar numa máquina manual mais geral. Três escolhas diferentes para uma atividade são chamadas três “rotas”; achar a melhor rota é chamado de “problema de roteamento”. O problema de roteamento pode também existir para um grupo de atividades como um todo: por exemplo, um rolo de papel pode ser feito numa linha antiga ou numa linha nova.

### 5.5. Reconfiguração de scheduling

Até o momento demos uma descrição geral do problema de scheduling: atividades e recursos pré-definidos, com talvez períodos de disponibilidade para cada atividade e recurso, e um objetivo envolvendo conclusão no prazo e custos operacionais razoáveis. Esse é o assunto mais tratado em livros e pesquisas sobre scheduling no passado; nós podemos chamá-lo de “problema clássico de scheduling”.

Mas, na prática, existem muitas outras decisões que interfaceiam com o problema clássico. Se nós adicionarmos algumas dessas decisões de interfaces no nosso modelo podemos falar do “problema estendido de scheduling”. Uma dessas decisões é mudar o montante e a configuração interna e/ou externa dos recursos enquanto o problema de scheduling avança no sentido de compensar a variante carga no sistema. Por exemplo, uma máquina pode ser capaz de acelerar num período muito movimentado para de alguma maneira equilibrar a carga, pagando com o aumento do uso da ferramenta. Ou uma máquina pode ser fisicamente conectada para adicionar capacidade a uma parte do sistema. Ou pessoal extra pode ser alocado para diferentes partes do sistema quando necessário. Ou recursos extras podem ser alugados, ou uma linha obsoleta pode ser ligada temporariamente. Também é possível que recursos sejam compartilhados por dois sistemas – com alguma definição de intervalos de tempo de disponibilidade para cada ou intervalos programados para manutenção. Uma decisão pode ser a de aumentar o intervalo de um recurso. Nós resumimos esse tipo de decisões como “reconfiguração de recursos” ou simplesmente “configuração”.

No mesmo estilo nós podemos reconfigurar atividades, trabalhos e projetos. Por exemplo, pode haver diferentes escolhas de tecnologia para realizar uma atividade mesmo em curto prazo, requerendo montantes diferentes dos recursos, e requerendo como resultado mais ou menos tempo de processamento (chamamos de comprometimento recurso/tempo). Esta é uma reconfiguração interna. Uma reconfiguração externa seria negociar prazos e custos excedentes baseado na nossa projeção de carga da linha e clientes concorrentes. Ou pode ser possível aumentar o intervalo alternativamente por transferir o processamento para um sistema ou

transporte anterior, ou por planejar transferir processamento num sistema ou transporte posterior.

### 5.6. Questões de sistemas de informação

Até agora, nós falamos sobre as decisões clássicas primárias do scheduling, que são de três tipos: sequenciamento, temporização e roteamento. Também falamos de decisões relacionadas um tanto maiores, que devem ser consideradas freqüentemente, como reconfiguração de recursos ou de atividades. Existem também um número de decisões de gerenciamento de sistemas de informação (MIS – management information system) que necessitam ser tomadas para sustentar as decisões primárias mencionadas acima. Essas incluem previsão (forecasting), classificação (labeling), agrupamento (grouping), agregação (aggregating) e desagregação (disaggregating). Vamos discutir brevemente cada uma delas nos parágrafos seguintes.

Nós imaginamos três tipos de previsão para scheduling: externa, interna e distribuída. Uma previsão tipicamente externa é tentar estimar os requisitos do cliente nos próximos três meses. Tal previsão é normalmente uma combinação de pedidos conhecidos, pedidos em negociação e estimação estatística de pedidos ainda desconhecidos. Nós normalmente assumimos uma série de demandas e prazos conhecidos e não estacionários, sabendo que serão revisadas freqüentemente.

Previsões internas, contudo, são normalmente desenvolvidas pela própria heurística do scheduling como parte do processo de solução. Por exemplo, o programa pode querer fazer uma estimativa grosseira o tempo de conclusão de vários trabalhos na linha. Os prazos conhecidos fora da linha permitem estimar o tempo que falta depois da operação atual. Uma heurística simples dará prioridade mais alta ao trabalho com o prazo mais curto. Dinâmicas de gargalo fazem muito uso dessas previsões, tempos restantes, custos de atraso de trabalho, recursos críticos, futuros recursos críticos e futuros tempos de chegada.

Sistemas de scheduling extensos podem ser distribuídos. Por exemplo, pode haver um subsistema para cada parte da linha, com um sistema de planejamento os controlando. E um subsistema anterior pode suprir tempos de chegada projetados

para o sistema atual. Da mesma forma, o sistema de planejamento pode mandar uma vista geral dos recursos críticos, enquanto um subsistema de interesse retorna correções com base no seu conhecimento detalhado de sua área.

Localizar na fábrica onde está um pedido de um cliente é claramente importante e aparentemente simples. Contudo, se uma dada parte de um WIP (work-in-process) tem múltiplas utilizações, aparece a questão de classificação. Geralmente existe a necessidade de uma flexibilidade considerável nesse respeito. Por exemplo, o programa de scheduling pode classificar uma parte de um trabalho em processamento ou em estoque como necessária para um trabalho que tem alta prioridade. Se aparece uma nova necessidade com maior prioridade, ou se as prioridades mudam, então o sistema classifica novamente este WIP com um propósito diferente. Se isso acontece freqüentemente, os clientes podem se queixar. Particularmente, uma maneira de lidar com estoques de segurança é por associar o estoque com os clientes e relacionar os estoques de segurança com os tempos de segurança ao invés de ao estoque.

O termo agrupamento pode ter dois significados ligeiramente diferentes. Se nosso modelo detalhado de scheduling nos informa a executar uma série de atividades junto, esse é um problema de temporização e tamanho de lote. Embora nós possamos dizer numa linguagem casual que “agrupamos” os itens, nós não usamos o conceito de agrupamento. Agrupamento sempre significa um tipo de agregação heurística, que pode causar perda de acurácia em troca de conveniência e fácil computação. Por exemplo, se decidimos que uma série de itens têm características semelhantes e sempre os mantivermos juntos através da linha com uma prioridade interna fixa, de fato agregamos essas atividades em um grupo. A questão de agregação de atividades como um todo é altamente complexa.

### **5.7. Resumo**

Resumindo, o que é scheduling? Um sistema de scheduling faz decisões de forma dinâmica sobre atividades e recursos semelhantes para finalizar trabalhos e projetos que necessitam essas atividades, numa maneira rápida e com qualidade, simultaneamente maximizando a produção e minimizando custos diretos de produção. Tipos de decisões básicas de scheduling são:

- a) Sequenciamento;
- b) Temporização/Liberação;
- c) Roteamento.

Decisões adicionais para modelos de scheduling extensos incluem:

- d) Reconfiguração de recursos;
- e) Reconfiguração de atividades.

Adicionalmente, um número de decisões MIS de suporte são geralmente necessárias, incluindo previsão, classificação, agrupamento, agregação e desagregação.

## 5.8. Níveis de Scheduling

### 5.8.1. Introdução

Deve ter ficado claro pela discussão na seção anterior que se um grupo de bens é considerado um recurso único ou não para o scheduling depende se decisões internas estão sendo consideradas no nosso nível atual de abstração ou são simplesmente dadas por “regras” fixas que são transparentes a nós. Assim uma dada linha de trabalhos pode ser considerada como sendo um número de recursos quando modelados detalhadamente; um nível acima, contudo, nós podemos apenas modelar a decisão do que liberar para a linha em seguida e quando. Isto é, no próximo nível de abstração a linha é considerada como um único recurso. Isso pode ser repetido várias vezes, levando a uma classificação de problemas de manufatura em vários níveis. A maioria desses esquemas tem quatro ou cinco níveis. Apresentamos um com cinco níveis na Tabela 1.

Nível	Exemplos de problemas	Horizonte
1. Planejamento em longo prazo	Expansão da fábrica, layout da fábrica, projeto da fábrica.	2-5 anos
2. Planejamento em médio prazo	Suavização da produção, logística.	1-2 anos
3. Planejamento em curto prazo	Plano de requerimentos, carga de linha, definição de prazos.	3-6 meses
4. Scheduling	Roteamento de trabalhos de linha, balanceamento de linha de montagem, dimensionamento de lista de processamento.	2-6 semanas
5. Scheduling / controle reativo	Trabalhos importantes, máquinas inativas, material atrasado.	1-3 dias

Tabela 1 – Classificação de níveis de scheduling

Enquanto todos esses níveis podem ser considerados scheduling, pois todos eles têm problemas de sequenciamento, temporização, roteamento, reconfiguração, previsão, classificação, agrupamento, agregação e desagregação, essa similaridade formal não pode ser levada muito adiante. Existe uma variedade de modelos de scheduling no nível 4. Na tentativa de modelar um nível diferente, é útil reconhecer as similaridades, mas as diferenças são também cruciais. Transferências descuidadas de modelos de similaridades intactos para um outro nível são geralmente desastrosas. Discutimos questões de modelamento em cada nível com algum detalhe, dando atenção especial para similaridades e diferenças com o scheduling do nível 4. Esses níveis diferem especialmente quanto às previsões.

#### 5.8.2. Nível 1: Planejamento em longo prazo

No nível de planejamento em longo prazo, tipos de trabalhos e projetos incluem localização, dimensionamento e projeto e fábricas, armazéns, departamentos, linhas de transferência e células de FMS, expansão de fábricas e armazéns, layout e projeto. Atividades são vários pedaços menores desses projetos, enquanto recursos incluem capacidade financeira, capacidade de engenharia e capacidade de gerenciamento. Com vários projetos e muitas atividades concomitantes, questões de sequenciamento e temporização são claramente

relevantes. Roteamento é relacionado com essas questões como fontes alternativas de financiamento e fornecedores de maquinário alternativos.

No entanto, essas similaridades formais com scheduling são freqüentemente superadas na prática pela grande quantidade de questões de previsão e necessidades relacionadas para atividade extensiva e reconfiguração de recursos. Qual será a demanda daqui a cinco anos? Que tipos de cenários podem ser visualizados? Que tecnologia estará disponível? Qual é o risco de comprar uma linha de montagem de última geração que pode estar obsoleta em três anos? Que tipo de fábrica é mais segura? As fábricas devem estar dispersas geograficamente?

Não apenas os problemas de previsão dominam o nível 1, mas métodos atuais de previsão são muito ineficientes. Um método comum é calcular um número de possíveis cenários, cada um baseado talvez numa previsão determinística com ou sem estoque de segurança. Bons modelos estocásticos são claramente necessários, mas pareceria que os métodos atuais não são adequados, já que modelos com médias e variâncias conhecidas não relacionadas de tempos em tempos simplesmente não capturam a situação.

Como um exemplo do perigo de aplicar modelos entre níveis, considere o problema de temporização de sucessivas expansões da fábrica para equilibrar a economia de escala na expansão e custos de dinheiro parado investido em capacidade excessiva. Esses problemas são freqüentemente atacados por modelos de dimensionamento de lote desenvolvidos geralmente para o nível 4. Infelizmente esses modelos abstraem as questões de reconfiguração e previsão e não têm sido muito bem sucedidos.

Também existe muitos trabalhos aplicando modelos de scheduling de projetos a esses tipos de projeto de longo e longuíssimo prazo. Essas aplicações são um tanto mais bem sucedidas, embora a questão da previsão permaneça extremamente difícil. Particularmente, correções reativas e re-projeto de tais programas não são bem compreendidos.

### 5.8.3. Nível 2: Planejamento em médio prazo

No nível de planejamento em médio prazo, métodos de scheduling têm sido mais bem sucedidos. Suavização de produção tem muito a ver com reconfiguração de recursos ao longo do tempo para prover capacidade balanceada em face de cargas de demanda sazonais. Mudanças de recursos externos incluem contratação, demissão, dispensa e subcontrato; mudanças internas incluem horas extras, abertura de linhas e mudar recursos entre linhas. Um scheduling de produção é certamente necessário, então questões de sequenciamento, temporização e roteamento são relevantes diretamente. No passado, versões de atividades de produção altamente agregadas e recursos disponíveis tiravam o sabor do scheduling. Com o avanço do poder computacional, e heurísticas mais acuradas, é agora prático, por exemplo, fazer o scheduling de oito linhas de produto em vinte recursos mensalmente num período de dezoito meses.

Problemas de previsão continuam bem difíceis. Modelos de suavização de produção e modelos de logística são freqüentemente lidados como determinísticos com ajustes para estoque de segurança. Em modelos de logística, métodos que incorporam efeitos de congestionamento deterministicamente dentro de modelos de rede parecem promissores.

Tanto modelos estocásticos como determinísticos têm sido empregados com sucesso no nível 2. Embora problemas de previsão sejam importantes e não foram totalmente resolvidos, problemas de agrupamento, agregação e desagregação prevalecem.

### 5.8.4. Nível 3: Planejamento em curto prazo

Para planejamento em curto prazo, modelos de interesse incluem o planejamento de requerimentos de material (MRP – Material Requirements Planning), carga de linha, definição de prazos. MRP aceita como entrada o plano de nivelamento de produção do nível 2, pedidos conhecidos, pedidos em negociação e previsões estatísticas dos pedidos restantes no horizonte de 3 a 6 meses para produzir um “master schedule”, que é uma previsão ao longo do tempo sobre as necessidades

da produção como tipo de produto, quantidade e período de tempo. Cada parte da previsão final necessita em algum momento ser “explodido”, ou seja, quebrado em componentes que são compensados para trás por um prazo de entrega padrão para montagens finais. Cada componente é então quebrado e compensado para trás novamente, e assim por diante. Quando componentes se transformam em matérias primas devidas numa certa data, isso é traduzido como um pedido a um fornecedor. Nesse primeiro passo, a capacidade dos recursos não é levada em conta. Temporização é resolvida de uma maneira simples, compensando para trás a partir vencimento por prazos de entrega registrados. Então é feito o scheduling da linha numa base primeiro a chegar primeiro a servir, deixando óbvios tanto o sequenciamento como a temporização. Escolhas de rotas não são consideradas. Um procedimento tão simples freqüentemente produz uma carga de linha errônea. O MRP estima um perfil grosso do uso de recursos, ajusta o master schedule e tenta novamente. Isso pode ser repetido.

Modelos de carga de linha e de definição de prazos permitem o balanceamento da linha num estilo diferente do que repetidamente reajustar o master schedule. Preços mais altos e entregas mais demoradas podem ser negociados para a linha sobrecarregada; preços mais baixos e entregas rápidas podem ser oferecidos quando há uma folga na linha. Assim essas abordagens são primariamente atividades de reconfiguração por natureza. No passado, modelos de carga de linha e de definição de prazos eram muitas vezes feitos confiando na própria experiência do negociador. Discutimos métodos que usam uma representação dinâmica de scheduling das atuais e próximas linhas para sugerir que tipos de preços e condições de entrega são apropriados.

Num horizonte de três a seis meses, o master schedule é uma combinação de pedidos conhecidos, pedidos em negociação e pedidos desconhecidos. Os últimos são estimados com técnicas como a suavização exponencial. Na prática, sistemas determinísticos com estoques de segurança são usualmente empregados para planejamento de pedidos de material, enquanto modelos de carga de linha e de definição de prazos podem ser tanto determinísticos como probabilísticos.

#### 5.8.5. Nível 4: Scheduling

Com scheduling, temos um horizonte de duas a seis semanas, dependendo se os tempos de processamento das atividades são mais da ordem de um dia ou uma semana. O sistema MRP de nível superior organiza uma previsão das chegadas no sistema de scheduling, dando tempos de chegada, prazos de entrega e quantidades de cada trabalho, junto com as atividades para completá-lo. O scheduling completo para, digamos, o período de seis semanas pode ser desenvolvido em uma semana, usando entradas e status da linha atualizados atualmente. Isso normalmente é uma atividade principal mesmo para a heurística e pode tomar oito horas de um mainframe. Numa semana, o scheduling é resolvido de novo para que apenas a primeira semana de cada scheduling é realmente usada. O restante é só ter certeza que o planejamento desta semana se encaixa nas seguintes. Freqüentemente haverá correções necessárias no scheduling entre os reschedulings semanais. Mas um processamento maciço de oito horas não é prático várias vezes por semana, não só pelo custo, mas também porque o sistema resultante não será em tempo real. Esse problema é tratado abaixo, no nível 5.

O nível 4 scheduling opera usando um master schedule razoavelmente acurado de trabalhos a realizar, prioridades, e datas de entrega para as próximas semanas.

#### 5.8.6. Nível 5: Scheduling/controle reativo

No nível mais detalhado de scheduling e controle reativos, questões centrais são emergências e falhas, como quebra de máquinas, falha de material ou chegadas atrasadas. Controles detalhados de atividades e recursos que foram esquecidos no nível 4 precisam ser tratados. Muita atividade está envolvida em corrigir o scheduling do nível 4 para essas emergências, falhas e detalhes. Até o scheduling reativo no meio da semana para corrigir quebras e atrasos de entrega, manuais antigos de sistemas de scheduling eram robustos. Como eles usavam regras simples de prioridades de despacho, que eram apenas executadas “na mosca” durante a semana, mudanças poderiam ser incorporadas por aplicar as mesmas prioridades para

a linha mudada. Ou uma ou mais prioridades poderiam ser mudadas, e assim o scheduling poderia prosseguir. Sistemas matemáticos de scheduling que sucederam não permitiam uma mudança no scheduling de maneira tão fácil. A falta de praticidade de se fazer um processamento de muitas horas a cada pequena falha ou mudança nos tempos de chegada levou na prática ao completo controle manual durante a semana. Isso trouxe a tendência de baixar a confiança nos sistemas automáticos e até mesmo ao seu desuso.

A abordagem de dinâmica de gargalo tenta incorporar o melhor dos dois mundos. Um processamento de muitas horas ainda é feito semanalmente. Esse processamento estabelece prioridades dinâmicas, que em troca permite o sistema a fazer o scheduling durante a semana, incorporando as mudanças rodando uma simulação de baixo custo (talvez 15 minutos). Um grande número de tais atualizações podem ser feitas sem perder a característica de tempo real do sistema.

Problemas de previsão para o nível 5 têm sido discutidos. Métodos para prever com mais acurácia emergências e falhas são importantes, mas não são consideradas aqui.

### **5.9. Tipos de Ambientes de Scheduling**

Na última seção vimos o scheduling em diferentes níveis de horizonte de tempo e agregação. Agora nós fixamos nossa atenção nos ambiente do scheduling no nível 4, que varia de uma linha de trabalhos pequena, complexa e customizada para linha de transferência de alta velocidade e pequena variedade de produtos, de manufatura discreta de peças a fluxo de produção contínua.

#### **5.9.1. Linha de trabalhos (job shop) clássica**

Um típico exemplo de um job shop clássico é uma companhia de ferramentas de fresa para máquinas de pesquisa. Cada pedido é único e tem uma rota única. Operações são realizadas seqüencialmente num lote único de partes, que viajam juntas pela linha. Não há inventário de fábrica que não seja relacionado com uma atividade. Scheduling é muito complicado e não se repete de maneira simples. O job

shop clássico é chamado de fechado, pois pedidos são distintos e WIP normalmente não podem ser emprestados de um trabalho para outro.

Contudo, pode não ser prontamente parente, vários ambientes de produção muito diferentes podem ser identificados como tendo características de um job shop fechado. Por exemplo, qualquer projeto único customizado, desde o projeto e construção de uma casa ao desenvolvimento de um protótipo de nave espacial à papelada padrão que flui sobre uma escrivaninha, compartilham muitas características de um job shop. Conhecimento sobre como fazer o scheduling de uma fábrica de ferramentas para fresamento nos dá muita percepção desses outros problemas. Contudo, também há muitas diferenças que não podem ser ignoradas. Muitas dessas se centram na escala de tempo. Com maiores horizontes de tempo e mudanças de tecnologia concomitantes, uma variedade de novas questões vem à pauta. Como freqüentes pedidos de mudança de engenharia e outros reprojetos afetam o scheduling? Como estimar a duração e o progresso das atividades?

#### 5.9.2. Linha de trabalhos (job shop) aberta

Tradicionalmente, uma linha que produz para o inventário final e não para um pedido é chamada de linha aberta. Nós usamos aqui o termo de forma similar, mas mais geral onde deve haver muitos clientes com demanda para os mesmos produtos para que faça sentido manter um inventário final, ou grande WIP, ou para desviar atividades ou trabalhos que eram de um cliente para outro com maior prioridade. Problemas de scheduling são similares aos de linha fechada, exceto que a classificação de trabalhos parcialmente ou totalmente completos de acordo com clientes e prazos de entrega se tornam mais complexos e dinâmicos. Um procedimento cuidadoso para reclassificação se torna necessário. Um exemplo comum é uma empresa que fornece peças de reposição a várias montadoras.

O fluxo de produção da nave espacial se encaixaria em algum lugar entre as linhas aberta e fechada. Produção de mainframes é uma combinação de pedidos especiais e pedidos comuns. O construtor de um conjunto habitacional é um excelente exemplo de um job shop aberto. Horizontes de tempo extremamente longos e altas incertezas e revisões não são um problema, mas ainda são importantes.

### 5.9.3. Batch shop

Um batch shop é basicamente uma linha aberta para qual a duplicação de WIP e produção final entre clientes se torna tão grande que o processamento de grandes remessas é típico para se ter vantagem econômicas de escala em processar peças similares. Fluxo através da linha não é totalmente linear, mas é normalmente menos complexo que para job shops abertos e fechados. Um exemplo de um batch shop discreto é a indústria têxtil. Outro exemplo é um fabricante que fornece peças para outros fabricantes.

Um exemplo de batch shop contínuo pode ser uma refinaria de petróleo. Outro pode ser uma indústria química.

### 5.9.4. Flow shop

Um flow shop é basicamente um batch shop com fluxo linear. Fluxo pode ser discreto, contínuo ou semi-contínuo. Agrupamento e loteamento tendem a ser importantes. No caso mais simples cada trabalho na mesma série de atividades a serem realizadas na mesma seqüência na mesma série de máquinas para todos os trabalhos. Isto é, existe uma máquina que faz a primeira atividade de cada trabalho, outra que faz a segunda, e assim por diante.

Até agora, descrevemos um flow shop simples. Num flow shop composto, cada máquina na série pode ser trocada por um grupo de máquinas em paralelo que podem ser iguais ou bem diferentes. Cada trabalho vai para uma máquina do primeiro grupo, depois para uma do segundo e assim prossegue. Outra variante, chamada “reentrant flow” envolve usar a mesma máquina ou grupo mais de uma vez. Suponha que um cano é extrudado cada vez com diâmetros menores. Depois de cada extrusão, o cano pode ir sempre para o mesmo forno para ser aquecido. Podemos chamar de flow shop reentrante. Em outras variantes, algumas máquinas podem ser puladas por algum trabalho. Se re-trabalho é importante, algumas atividades devem ser repetidas quando a inspeção revela falhas.

Existem poucos flow shops absolutamente puros, mas muitos flow shops compostos com pequenas variações. Companhias de engarrafamento, algumas editoras e prensas de aço são alguns exemplos.

#### 5.9.5. Batch/flow shops

Um surpreendentemente grande número de processos de produção se dividem em duas metades: a primeira é um grande processo batch, onde as matérias primas e os materiais intermediários são tratados. A segunda metade é uma típica flow shop composta. Uma companhia de processamento de alimentos, por exemplo, primeiro literalmente cozinha os produtos comestíveis em caldeiras ligadas por tubulações e assim por diante; geralmente esse sistema não é totalmente linear. A segunda metade envolve o acondicionamento da comida em várias linhas paralelas de alta velocidade. Por exemplo, três linhas podem ser antigas e idênticas, uma linha pode ser antiga e usada apenas para potes grandes, uma linha é nova e rápida e capaz de lidar com os novos produtos bem como com os antigos.

#### 5.9.6. Célula de manufatura

A célula de manufatura tenta combinar a flexibilidade do job shop com o baixo custo e baixa variedade do flow shop. Basicamente, itens a serem produzidos no job shop são agrupados em diversas famílias similares. Cada família é servida por uma célula, que é normalmente um manipulador automatizado cercado por um grupo de máquinas.

As máquinas também são eletronicamente agendadas e sincronizadas. Elas são geralmente flexíveis nas atividades que podem realizar. O fato de todos os trabalhos de um grupo serem similares torna praticável que sete ou oito máquinas realizem todas as atividades necessárias. O fato de que as máquinas podem ser acessadas em qualquer ordem com quase a velocidade e eficiência de um flow shop permitem que antigos itens de job shops sejam produzidos com a eficiência dos flow shops.

Células são ainda relativamente experimentais, mas muitas empresas de trabalho em metal têm pelo menos projetos piloto.

## 6. SCHEDULING DE FLOW SHOPS: MAKESPAN

### 6.1. Makespan genérico

#### 6.1.1. Visão geral

Problemas de makespan ficam muito difíceis quando existem mais de duas máquinas. Branch-and-bound tem tido algum sucesso em resolver com exatidão problemas pequenos, talvez com três ou quatro máquinas e 10 a 15 trabalhos pelo critério de makespan. Heurísticas do tipo de Johnson, que são baseadas na heurística de diferenças generalizada para o caso com duas máquinas têm sido estudadas extensivamente. Mais recentemente, várias heurísticas computacionais para problemas de flow shop se tornaram disponíveis.

#### 6.1.2. Heurísticas para $m$ máquinas

O algoritmo exato de Johnson para o problema de makespan de duas máquinas com chegadas estáticas e as várias heurísticas de diferença associadas para o mesmo problema se baseiam em encontrar um trabalho com a maior tendência em progredir de tempos curtos a tempos longos na seqüência de operações. Nós podemos esperar explorar essa mesma ideia para problema com chegadas estáticas de  $m$  máquinas; nesta seção nós estudamos três heurísticas que fazem exatamente isso.

#### Heurísticas clássicas

A heurística de diferença para  $m = 2$  calcula um índice  $\pi_j = -p_{j1} + p_{j2}$  e ordena o trabalho com o maior  $\pi_j$  primeiro. Isto sugere para  $m = 3$  um índice  $\pi_j = -2p_{j1} + 0p_{j2} + 2p_{j3}$ , que finalmente leva à seguinte heurística.

#### Heurística de Palmer

Para um flow shop com  $m$  máquinas e chegadas estáticas e com critério makespan, calcula a prioridade abaixo:

$$\pi_j = -(m-1)p_{j1} - (m-3)p_{j2} - (m-5)p_{j3} + \dots + (m-3)p_{j,m-1} + (m-1)p_{jm}$$

e ordena o trabalho com maior prioridade primeiro.

Note que uma analogia a razão heurística pode facilmente ser construída por substituir  $\log(p_{jk})$  por  $p_{jk}$  na heurística de Palmer.

Gupta procurou uma regra de prioridade semelhante à de Palmer, mas baseada diretamente na regra de Johnson, tal que produziria schedules perfeitos para  $m = 2$ , perto de ótimo para  $m = 3$ , e bons para o caso geral. Ele notou que quando a regra de Johnson era ótima para o caso de três máquinas, ela pode ser modelada como um esquema de prioridade onde,

$$\pi_j = [e_j] / [\min\{p_{j1} + p_{j2}, p_{j2} + p_{j3}\}]$$

e onde definimos  $e_j = 1$  se  $p_{j1} < p_{j2}$ , e  $e_j = -1$  se  $p_{j1} \geq p_{j2}$ .

Generalizando esta estrutura, Gupta obteve a seguinte heurística.

### Heurística de Gupta

Para um flow shop de  $m$  máquinas com critério de makespan e chegadas estáticas, calcule as prioridades

$$\pi_j = [e_j] / \left[ \min_{k=1,m-1} \{p_{jk} + p_{j,k+1}\} \right]$$

onde nós definimos  $e_j = 1$  se  $p_{j1} < p_{jm}$ , e  $e_j = -1$  se  $p_{j1} \geq p_{jm}$  e ordenamos o trabalho de maior prioridade primeiro.

Num estudo computacional, Gupta percebeu que sua heurística gerava melhores makespans do que Palmer na maioria substancial dos casos.

Talvez a mais precisa extensão do método de Johnson para o problema com  $m$  máquinas de chegadas estáticas é o método de Campbell, Dudek e Smith (CDS). Também é algo mais perto do nosso propósito geral de métodos de heurística para problemas de flow shop, em que ela gera um número de soluções e seleciona a melhor.

### Heurística de CDS

Resolve para  $m$  diferentes schedules em  $m$  diferentes iterações. Para a iteração  $i$  define

$$P_{j1} = \sum_{k=1,j} p_{jk} \quad \text{e} \quad P_{j2} = \sum_{k=1,j} p_{j,m-k+1}$$

(isto é,  $P_{j1}$  é a soma de tempos para o trabalho  $j$  nas primeiras  $i$  máquinas e  $P_{j2}$  é a soma para as últimas  $i$  máquinas). Resolve o problema de duas máquinas de Johnson resultante, e salva a seqüência correspondente e o makespan  $M_i$ . Finalmente, pegue o makespan como  $M = \min\{M_i\}$  e usa o schedule correspondente da iteração  $i^*$ .

#### 6.1.3. Problema de gargalo de uma máquina

O par  $ji$  representa a operação  $i$  do trabalho  $j$  e assim representa uma atividade particular. Se queremos especificar uma ordem de prioridade para atividades na máquina  $k$ , nós especificaríamos  $ji(k,1)$  como a primeira atividade a entrar em  $k$ ,  $ji(k,2)$  como a segunda, e assim por diante.

O *head time* da operação  $ji$  é o tempo mínimo antes do trabalho  $j$  poder começar se nenhuma operação anterior precisa esperar, isto é,

$$HT_{ji} = \sum_{i'=1,i-1} p_{ji'}$$

O *tail time* da operação  $ji$  é o tempo mínimo antes do trabalho  $j$  acabar depois de  $ji$  terminar se nenhuma operação subsequente precisa esperar, isto é,

$$TT_{ji} = \sum_{i'=i+1,q} p_{ji'}$$

O *make time* do trabalho  $j$  é o tempo total preciso para completar  $j$  se não há espera, isto é,

$$MT_j = \sum_{i'=1,q} p_{ji'}$$

O *span time* da série de trabalhos é o tempo máximo para completar todos os trabalhos se não há espera, isto é,

$$ST = \max_j \{MT_j\}$$

Supomos que todos os recursos exceto o que estamos interessados estão com folga, não formando fila nas suas entradas. Assim, atividades antes e depois dessa máquina não têm nenhuma espera, e o tempo de chegada da operação  $ji$  na máquina atual  $k$  é apenas o *head time*, isto é:

$$r_{ji} = HT_{ji}$$

Agora, se a data de entrega final do trabalho  $j$  é  $d_j$ , e a operação  $j$  requer exatamente o *tail time* depois de terminar essa operação até que o trabalho termine, então o prazo de entrega efetivo para a operação  $ji$  é:

$$d_{ji} = d_j - TT_{ji}$$

Finalmente, alguns trabalhos podem não utilizar esta máquina. Como esses trabalhos não têm que esperar em outra máquina, eles vão ter um tempo fixo finalização igual ao *make time*, isto é:

$$C_j = MT_j$$

Um problema menor é que para makespan não há datas de entrega. Uma formulação alternativa do problema do mínimo makespan pode facilmente ser construído com datas de entrega. Definimos todas as datas de entrega igual a um tempo arbitrário, digamos, o *span time*.

## 6.2. Tabu Search

### 6.2.1. Introdução

A palavra Tabu (ou Taboo) tem sua origem na língua Tongan, da Polinésia, e é usada para referir algo que é proibido por ser sagrado. No contexto da Tabu Search este significado transforma-se na imposição de restrições para guiar uma procura, ajudando-a a não seguir caminhos tabu.

A engenharia e tecnologia vêm continuamente provendo exemplos de problemas de difícil otimização. A Tabu Search não tem provas claras de convergência, mas na prática mostrou uma notável eficiência em vários problemas.

As raízes do Tabu Search remontam aos anos 70. Ele foi apresentado na sua forma atual primeiramente por Fred Glover, professor de economia na Universidade do Colorado em Boulder. Uma definição usada por Glover para descrever essa meta-heurística é: “Tabu Search é uma heurística computacional de procura conhecida por geralmente superar o problema da convergência local em problemas de otimização”.

Os métodos clássicos de resolução de problemas de otimização apresentam resultados limitados à medida que os problemas aumentam em complexidade. A Tabu Search surge como uma solução visto que mudou dramaticamente a taxa de sucesso na resolução de problemas tão complexos. O uso dessa meta-heurística oferece por vezes resultados de maior qualidade do que os métodos até então utilizados. Vários experimentos computacionais mostraram que o Tabu Search se tornou uma técnica de otimização bem estabelecida que pode competir com muitas das técnicas tradicionais e que – pela sua flexibilidade – supera muitos procedimentos clássicos.

Essa heurística é utilizada em diversos campos como a Engenharia, Economia, Ciência e Negócios, como por exemplo:

- Planejamento de recursos (horários, Job Shop, Flow Shop, sequenciamento, etc);
- Desenho (desenho auxiliado por computador, problemas de recorte irregular, etc);
- Lógica e Inteligência Artificial (integridade de dados, redes neurais, etc);
- Tecnologia (distribuição de eletricidade, construção de estações espaciais, etc);
- Telecomunicações (call routing, etc).

#### 6.2.2. Utilização da memória

As estruturas de memória do Tabu Search são compostas por quatro dimensões:

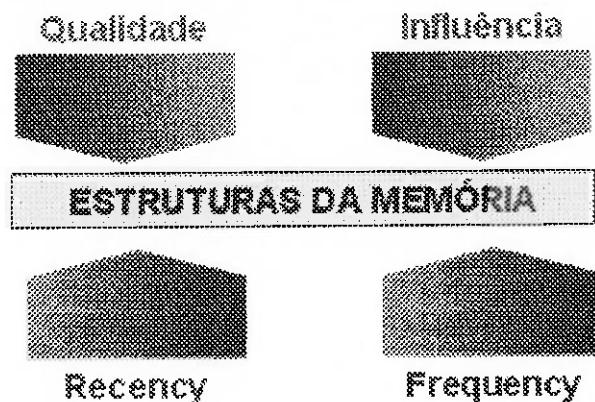


Figura 1 – Estruturas de memória do Tabu Search

### Qualidade

Baseia-se na habilidade de diferenciar o mérito das soluções visitadas durante a procura. Permite que a memória possa ser usada para identificar elementos que são comuns a soluções consideradas como boas, ou caminhos que conduzem a tais soluções. A qualidade é o ponto alto onde ações que levem a boas soluções são incentivadas, enquanto que penaliza ou desencoraja aquelas que levam a más soluções. A flexibilidade dessas estruturas de memória permite que a procura possa ser feita num ambiente multi-objetivo, ou seja, a qualidade de um determinado caminho pode ser calculada a partir de mais do que uma função.

### Influência

Considera o impacto das decisões feitas durante a procura, não só em termos de qualidade, mas também em termos de estrutura. Ao guardar informação acerca da influência de escolhas de determinados elementos de uma solução em particular, incorpora um nível de aprendizagem adicional.

### Recência e Freqüência

A conciliação ou utilização conjunta destas duas dimensões da memória da Tabu Search possibilita a definição de componentes que definem critérios mais longos de aplicação. Elas medem respectivamente quão recentemente passamos por aqui e quão freqüentemente passamos por aqui.

### 6.2.3. Funcionamento

O Tabu Search funciona por iterações. A cada iteração é encontrada a melhor solução não tabu. O número de iterações é escolhido em conformidade com o problema. Seguem abaixo algumas ferramentas do Tabu Search.

#### Critério de Aspiração

Um critério de aspiração é utilizado para determinar quando uma restrição tabu pode ser ignorada. Eles são freqüentemente utilizados para ignorar a restrição tabu sempre que o movimento proibido origine uma solução melhor do que a melhor solução encontrada até o momento. A aplicação adequada desses procedimentos é fundamental para se atingir altos níveis de performance na Tabu Search. Alguns critérios de aspiração são:

- O melhor até agora (best so far)
- O melhor na vizinhança (best in neighborhood)
- Diferente de solução existente (diversification)
- Semelhante a solução existente (intensification)

#### Função Objeto

Define como é escolhida a solução entre todas as possíveis. É escolhida conforme a necessidade do problema estudado. A função objeto tem que tornar a procura computacionalmente eficaz.

#### Critérios de Parada

São os critérios adotados para que o programa pare de procurar a solução ótima. Alguns dos critérios de parada podem ser:

- Solução ótima determinada;
- Efetuados x números de iterações sem melhoria da solução encontrada;
- Tempo limite atingido.

O passo genérico de um procedimento iterativo consiste em construir, a partir de uma solução atual  $i$ , uma próxima solução  $j$  e checar se podemos parar com as iterações ou precisamos fazer um novo passo. Métodos de procura de vizinhança são

procedimentos iterativos nos quais uma vizinhança  $N(i)$  é definida para solução factível  $i$ , e a próxima solução  $j$  é procurada entre as soluções em  $N(i)$ .

Para melhorar a eficiência do processo de exploração, precisamos manter registro não apenas de informações locais (como o valor atual da função objeto) mas também algumas informações relacionadas com o processo de exploração. Esse uso sistemático da *memória* é uma característica essencial do Tabu Search. Enquanto a maioria dos métodos de exploração mantém na memória essencialmente o valor da melhor solução visitada até o momento, o Tabu Search também mantém informações sobre o itinerário das últimas soluções visitadas. Essas informações serão usadas para guiar a movimentação de  $i$  para a próxima solução  $j$  a ser escolhida em  $N(i)$ .

Mais precisamente, vamos notar que a estrutura de vizinhança  $N(i)$  de uma solução  $i$  será variável de iteração a iteração. Portanto, seria mais apropriado incluir o TS numa classe de procedimentos chamada *técnicas de procura de vizinhança dinâmica*.

#### 6.2.4. Tabu Search Canônico

O algoritmo Tabu Search pode tomar várias formas, dependendo de para que aplicações ele é utilizado. Mas temos um formato genérico que todos devem seguir. Segue abaixo uma rotina canônica, rotina essa que é a base para nossa rotina de busca da melhor solução.

```

define comprimento da lista tabu, t
gera aleatoriamente solução inicial
até critério de parada é satisfeito
{
    de vizinhança i = 1 a N
    {
        vai para uma solução vizinha
        avalia
        salva em ordem rankeada, j = 1 a N
        i ++
    }
    de j = 1 a N
    {
        se solução j não é tabu, vai para 100
    }
}
```

```
senão
    se solução j é melhor que a melhor até agora, vai
    para 100
    j ++
}
100    adiciona solução j à lista tabu
    se tamanho da lista tabu > t, remove a solução mais antiga
    se solução j é melhor que a melhor até agora
        melhor até agora = solução j
}
retorna melhor até agora
```

## 7. MODELAGEM UML

Para o desenvolvimento de um software, é necessário que se conheça os objetivos a serem alcançados pelo software. Tendo o objetivo a vista, faz-se necessária a estruturação do programa para que se organize e estruture os dados a fim de se atingir os objetivos desejados da melhor maneira e da forma mais organizada possível.

A UML é a tentativa se padronizar a modelagem de sistemas que utilizam como base a modelagem orientada a objetos e, caso seja utilizado corretamente, o sistema modelado através dela possui muitas vantagens em relação aos demais sistemas que utilizam outras padronizações como base. Dentre as vantagens podemos dizer, por exemplo:

- Facilidade de comunicação com outras aplicações;
- Simples de ser atualizado;
- Sistema totalmente documentado;
- Utilizar os métodos conceituais para implementar o sistema,
- Unificação de desenvolvimento e metodologias.

Podemos dizer que a maior vantagem do uso da UML está no fato de ser uma modelagem orientada a objetos. Isso ocorre principalmente devido ao fato de que este tipo de abordagem gera sistemas com estruturas conhecidas, o que nos permite criar, implementar e utilizar sistemas totalmente reutilizáveis e flexíveis. Mas para isto ser feito, ele requer um método que integre o processo de desenvolvimento e a linguagem de modelagem com a construção de técnicas e ferramentas adequadas. Neste ponto entra a UML.

Existem diversas fases para o desenvolvimento de um sistema e a UML pode ser utilizada em praticamente todas elas. Desde a especificação de análise até a implantação total do sistema. Para tanto, utilizam-se os diversos diagramas que fazem parte desta modelagem no seu desenvolvimento. Estes diagramas auxiliam em todas as etapas e tornam o processo mais rápido e eficaz.

## 7.1. Fases do desenvolvimento de um sistema em UML

Para auxiliar no desenvolvimento dos sistemas, o desenvolvimento é dividido em diversas fases. Isso facilita a administração do projeto e facilita a correção do sistema em caso de alteração de necessidades ou de melhorias no projeto. Ou seja, o projeto entra em um processo de realimentação fazendo com que o resultado do desenvolvimento seja de alta qualidade e performance. Para o desenvolvimento, pode-se nomear as seguintes fases.

### 7.1.1. Análise de requisitos

Responsável pela análise preliminar do sistema, com o levantamento do que o cliente deseja e com a análise dos dados que serão necessários para chegar ao resultado final. Tendo em mãos os dados necessários, desenvolve-se através dos chamados casos de uso, as atividades que o sistema terá que atuar com a intervenção do usuário.

Nos casos de uso, especifica-se as condições em que o usuário terá que atuar além de suas pré e pós condições. Assim, obtemos toda a funcionalidade do sistema, porém sem que se leve em consideração a maneira que será implementada.

### 7.1.2. Análise

Esta fase começa a se importar com a maneira de implementação será realizada. Nela, começamos com as primeiras separações das classes e objetos do sistema. Utilizando os diagramas de classes, podemos visualizar o sistema com suas classes e os relacionamentos existentes nelas. Apesar disto, só são modeladas as classes que pertençam ao domínio principal do problema e pontos como interfaceamento permanecem de fora deste diagrama. Além do diagrama de classes, existe uma série de outros diagramas, neste caso, utilizaremos também o diagrama de seqüência e o diagrama de casos de uso. O detalhamento dos casos de uso serão colocados como anexo a este trabalho.

### 7.1.3. Design (Projeto)

Na fase de design, acrescentamos às classes modeladas na fase de análise as classes de infra-estrutura do sistema, como as classes de interface e gerenciamento de banco de dados. Assim, obtemos o detalhamento de todas as especificações necessárias do sistema e pode-se partir para a fase de programação.

### 7.1.4. Programação

Aqui, as classes desenvolvidas na parte de design são transformadas para o código de programação. Não se pode simplesmente tentar traduzir o que foi escrito durante as fases anteriores pois as classes foram modeladas para dar o entendimento da estrutura do sistema, neste fase é necessário que se adeque o que foi escrito para a real condição de programação. Assim sendo os diagramas não mais demonstram a atual condição do sistema.

### 7.1.5. Testes

Todos os testes possíveis são utilizados para verificar se o sistema está atendendo todas as funcionalidades que foram solicitadas. Dentre os testes que são realizados, podemos destacar:

1. Teste de unidade – utilizado pelo próprio programador para verificar a integridade das classes desenvolvidas ou até de grupos de classes.
2. Teste de integração – Verificação através de testes com as classes e componentes integrados para confirmar se as classes estão cooperando umas com as outras como especificado nos modelos desenvolvidos.
3. Teste de aceitação – testa o sistema como se fosse uma caixa preta e verifica se todos os casos de uso especificados estão funcionando conforme o especificado.
4. Teste do usuário – o usuário final testa o sistema para verificar se os resultados estão de acordo com o que foi requisitado.

## 8. PROJETO

Para o desenvolvimento deste projeto, foram seguidas as seguintes etapas:

### 8.1. Análise de requisitos

Foram levantados os dados e especificações do sistema e os resultados estão demonstrados no item 5, Especificação de Sistemas, que vai das páginas 12 a 16. De posse destes dados, foram levantadas as funcionalidades que o sistema deveria apresentar além da metodologia de sequenciamento que será utilizada de acordo com o que propomos como sendo o objetivo da nossa linha de produção.

#### 8.1.1. Metodologia de sequenciamento

Tendo como base os dados de linha e produtos levantados e levando em conta suas características, escolhemos a melhor metodologia para o sequenciamento. Para isso tivemos de fazer algumas considerações iniciais.

Tratamos o problema como um flow shop, pois se trata de uma linha com uma alta especificidade, tendo poucas variações dos produtos nela produzidos. Para tanto, a melhor abordagem a ser feita é a de ter como objetivo otimizar o makespan, que é o tempo de produção. Isso porque, para linha de produção como a nossa, esse é o objetivo mais estudado e que obtém melhores resultados além de resultados mais precisos. O nosso flow shop é convencional, mas será considerada no nosso sistema com um skip shop, apesar de essa variação de denominação não alterar as características tanto do sistema como do diagrama.

Para resolver o problema é necessário gerar o schedule, isto é, gerar a ordem em que os trabalhos vão entrar na linha de produção. Existe muitos métodos, chamados heurísticas, para se gerar o schedule de uma linha, vários destes descritos neste trabalho. Nosso problema é o de um flow shop com o tempo de produção (makespan) como objetivo. Outro requisito do nosso problema é que a seqüência é um fator decisivo. A seqüência final do scheduling não pode diferir grandemente da seqüência inicial. Como a linha de caminhões apresenta bifurcações, o melhor

método de abordagem do scheduling é o Tabu Search, que faz uma busca por vizinhança antes de procurar o mínimo global.

### 8.1.2. Adequação do Tabu Search

Nesta seção vamos discutir como adequamos o nosso problema com a metodologia do Tabu Search.

Sabemos que o Tabu Search é uma meta-heurística que faz uma busca por máximos locais antes de buscar um máximo global, ou seja, faz uma busca por vizinhança e depois “sobe o nível”, fazendo uma busca global a partir dos resultados de cada vizinhança.

Também é sabido que o nosso problema consiste no sequenciamento da produção de uma linha de cabinas de caminhão. O objetivo da linha é a diminuição do makespan, ou seja, do tempo de produção. Como entrada recebemos uma relação de produtos numa ordem definida pela linha que monta todo o caminhão. Essa seqüência nos traz uma restrição. Não podemos alterar significativamente essa ordem. Para isso temos que limitar o número de iterações. Portanto, o número de iterações é o nosso critério de parada. Também não utilizaremos critérios de aspiração, ou seja, critérios para se considerar uma restrição Tabu, pois senão entraremos num loop infinito no nosso programa, como veremos adiante.

Para efeito de demonstração, vamos nos utilizar de uma seqüência de 5 produtos, como a que segue abaixo.

P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>4</sub>	P <sub>5</sub>
----------------	----------------	----------------	----------------	----------------

Figura 2 – Seqüência inicial de produtos

Consideramos uma seqüência como essa como sendo uma vizinhança, na verdade a menor vizinhança tratada no programa. Dizemos que é uma vizinhança, e não um resultado único, porque a linha tem algumas bifurcações, e cada uma delas pode ter um tempo de processamento diferente.

Uma vez calculado o melhor resultado nessa vizinhança, partimos para a consideração de outras vizinhanças, estas relacionadas com a inicial, apenas com a alteração de uma posição dos produtos. Assim, criamos quatro novas vizinhanças, cada uma resultante de uma permutação, como segue ilustrado abaixo.

P <sub>2</sub>	P <sub>1</sub>	P <sub>3</sub>	P <sub>4</sub>	P <sub>5</sub>
P <sub>1</sub>	P <sub>3</sub>	P <sub>2</sub>	P <sub>4</sub>	P <sub>5</sub>
P <sub>1</sub>	P <sub>2</sub>	P <sub>4</sub>	P <sub>3</sub>	P <sub>5</sub>
P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>5</sub>	P <sub>4</sub>

Figura 3 – Seqüências resultantes de permutação

Temos então cinco vizinhanças primárias, cada uma com um resultado ótimo. Essas cinco vizinhanças formam um outro nível de vizinhança, que também tem um resultado ótimo, na verdade o melhor resultado dentre as cinco. Esse nível consideramos como uma iteração.

Depois de uma iteração, verificamos qual o melhor resultado. Se esse resultado for a seqüência inicial, podemos finalizar o programa, pois senão entraremos num loop infinito. Se o melhor resultado não for da seqüência original, fazemos outra iteração, alterando a seqüência inicial para a com o melhor resultado. Fazemos isso até chegarmos ao limite máximo de iterações, que é dado pelo administrador do sistema.

A cada iteração o resultado dela é comparado com o melhor resultado então. Se o resultado for melhor, substituímos o melhor resultado e passamos para a próxima iteração. Se for pior, paramos o programa e já temos o melhor resultado.

#### 8.1.3. Casos de uso do sistema

De acordo com o que foi levantado como sendo os dados necessários para o desenvolvimento do sistema anteriormente descritos, analisam-se os casos de uso necessários. O detalhamento dos casos de uso podem ser encontrados no Anexo A. A seguir está representado o diagrama de casos de uso:

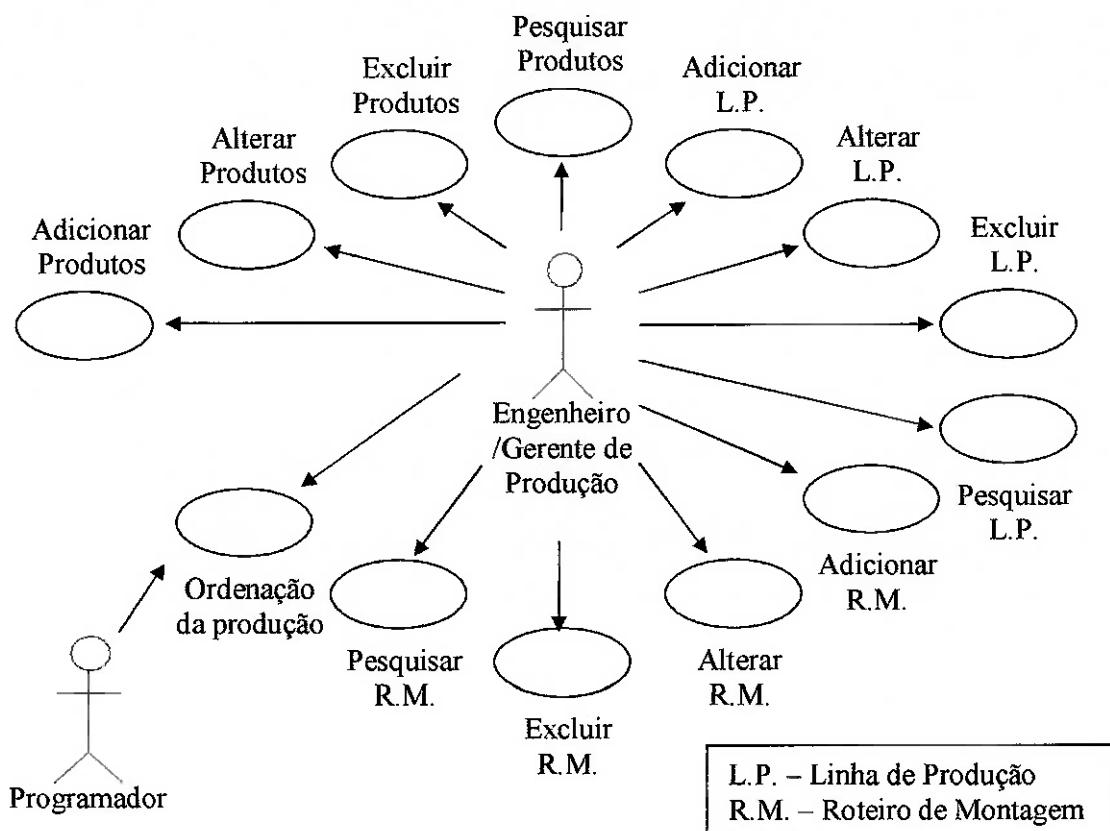


Figura 4 – Diagrama de casos de uso

## 8.2. Análise

Após o levantamento dos requisitos e dos casos de uso do sistema, pode-se montar o primeiro diagrama de classes principais do sistema com suas classes e relacionamentos. Chega-se então à seguinte situação:

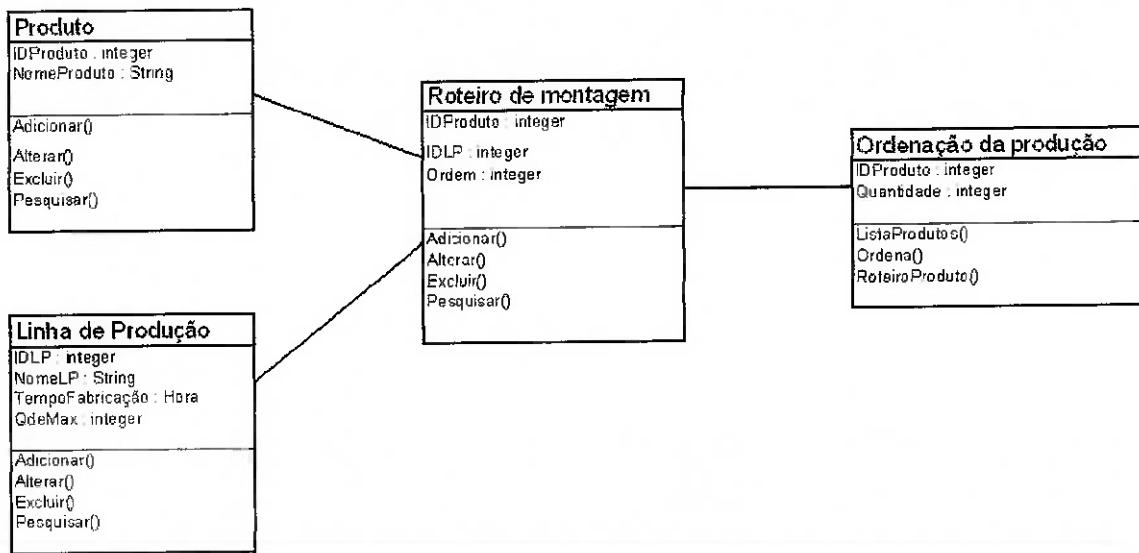


Figura 5 – Diagrama de classes

Neste ponto, cabe uma re-análise do sistema, onde se analisa se os objetivos do projeto estão ao não sendo atendidos e de que forma. O como programar ainda não entra no projeto já que esta parte será desenvolvida posteriormente.

Além disto, tratamos o algoritmo selecionado como sendo uma “caixa-preta” na representação pelo diagrama de classes. Este só será implementado na fase de programação do sistema.

### 8.3. Design

Neste ponto, acrescentamos as camadas de estrutura do sistema às classes que foram estruturadas na etapa anterior. Como camada de estrutura, entende-se toda a parte do sistema que se comunica com o ambiente externo, interface com o usuário, gerenciamento do banco de dados, entre outros.

Para tanto, analisa-se a seqüência de eventos que o sistema deve realizar para conseguir atingir os objetivos desejados. Esta seqüência de eventos pode ser representada através de um diagrama de seqüência. Abaixo está representado o diagrama de seqüência do evento adicionar produto, como os demais diagramas

seguem basicamente os mesmos eventos só mudando o destino (produto, linha de montagem, entre outros) somente este diagrama será apresentado neste trabalho.

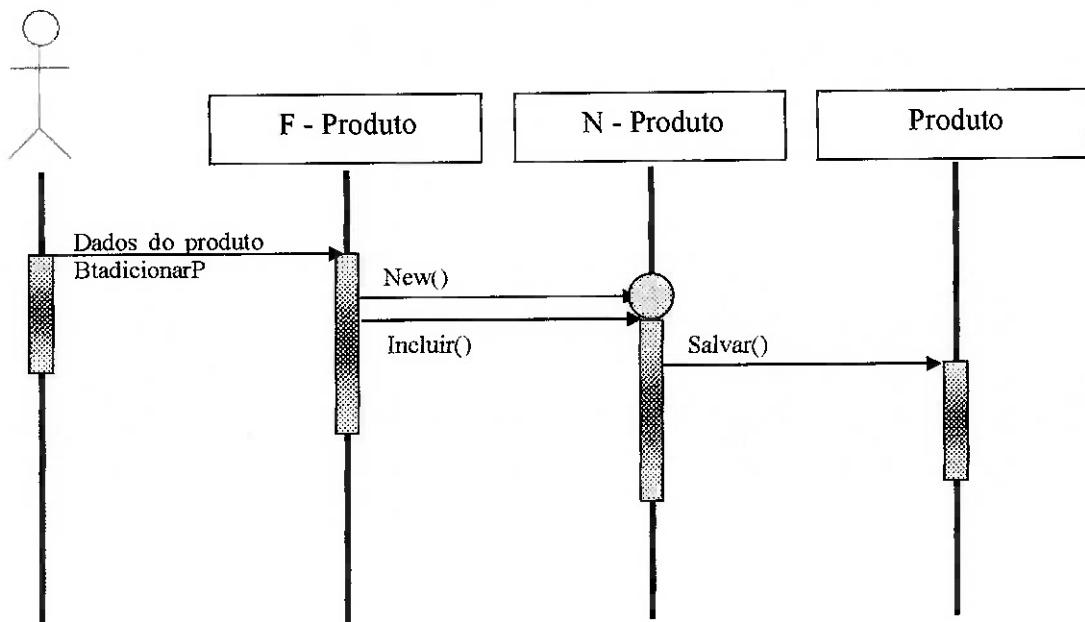


Figura 6 – Diagrama de seqüência

Dessa forma, chegamos às especificações que serão necessárias para o funcionamento do sistema.

#### 8.3.1. Telas

- Adicionar produto
- Alterar produto
- Excluir produto
- Pesquisar produto
- Adicionar Linha de produção
- Alterar Linha de produção
- Excluir Linha de produção
- Pesquisar Linha de produção
- Adicionar Roteiro de montagem
- Alterar Roteiro de montagem

- Excluir Roteiro de montagem
- Pesquisar Roteiro de montagem
- Ordenação da produção
- Relatórios
- Login

### 8.3.2. Funções / Transições

- Adicionar produto
- Alterar produto
- Excluir produto
- Pesquisar produto
- Adicionar Linha de produção
- Alterar Linha de produção
- Excluir Linha de produção
- Pesquisar Linha de produção
- Adicionar Roteiro de montagem
- Alterar Roteiro de montagem
- Excluir Roteiro de montagem
- Pesquisar Roteiro de montagem
- Ordenar
- Lista produtos
- Roteiro produto
- Salvar
- Reordenar roteiro

### 8.3.3. Tabelas

- Produtos (IDProduto : integer, NomeProduto : String)
- Linha de produção (IDLP : integer, NomeLP : String, TempoFabricacao : Hora, QdeMax : integer)

- Roteiro de montagem (IDProduto : integer, IDLP: integer, Ordem : integer)
- Ordenação da produção (IDProduto : integer, Quantidade : integer)
- Ordem Final (OrdemFinal : integer, IDProduto : integer)

#### 8.3.4. Relatórios

- Produtos
- Linha de produção
- Roteiro de montagem
- Ordem final

#### 8.3.5. Código

O algoritmo do programa que foi desenvolvido para o projeto tratado neste trabalho segue no Anexo A.

## 9. RESULTADOS

Inicialmente o sistema foi testado com uma seqüência de 5 produtos, como o exemplo da seção anterior. A seqüência dos produtos foi determinada e chegamos ao resultado previsto no fim da montagem bruta, teste suficiente para provar que o sistema suportava as bifurcações da linha e os vários tipos de produtos. Depois modificamos o sistema para calcular o tempo de uma seqüência de 20 produtos, tarefa que foi computacionalmente mais lenta, mas mostrou resultados condizentes com as expectativas.

Segue abaixo alguns resultados experimentais obtidos:

ID Produto	Nome Produto
1	LN
2	L98
3	FPN
4	HPN
5	HSK
6	SKL
7	Canavieiro
8	FSK
9	LTC

Tabela 2 – Lista de produtos

P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	Tempo
1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	2	6	3835
1	2	3	5	4	6	7	8	9	1	2	3	4	5	6	7	8	9	2	6	3634
1	2	3	5	4	6	7	8	9	1	2	3	5	4	6	7	8	9	2	6	3433
1	2	3	5	4	6	7	8	9	1	2	5	3	4	6	7	8	9	2	6	3396
1	2	3	5	4	6	7	8	9	1	2	5	3	6	4	7	8	9	2	6	3285

Tabela 3 – Seqüência de produtos e tempo de produção a cada iteração

## 10. CONCLUSÃO

Com o projeto do sistema em mãos, partiu-se para um maior detalhamento do sistema a ser utilizado e do método de escolha da melhor solução. Assim, foi realizado um estudo mais profundo sobre o Tabu Search e foram levantados dados mais detalhados sobre a linha de produção que serviram para nossas simulações. Também foram construídos as tabelas e formulários básicos para o sistema. Logo após foi codificado o algoritmo que calcula o tempo de uma seqüência de produtos na linha e que faz a escolha do melhor resultado baseado no Tabu Search.

Levantados estes dados e feita a análise mais minuciosa sobre o método de sequenciamento, iniciamos a fase de construção do algoritmo. Após esta fase, partimos para uma fase de testes e simulações mais intensivas.

Com os testes pudemos perceber que o código estava funcionando assim como projetado e o tempo sendo calculado assim como o previsto.

O sistema construído revelou a possibilidade de uma melhoria do sequenciamento da linha estudada. Também lança base para uma futura ampliação, englobando por sua vez toda a linha de caminhões.

## 11. BIBLIOGRAFIA

- Morton, Thomas E. & Pentico, David W., “*Heuristic scheduling systems : with applications to production systems and project management*”, Wiley, c1993
- UML Tutorial  
<http://uml.tutorials.trireme.com/>  
Acesso em: 20/03/04
- DGA - UNICAMP  
[http://www.dga.unicamp.br/treinamento/apostilas/apostila\\_uml.pdf](http://www.dga.unicamp.br/treinamento/apostilas/apostila_uml.pdf)  
Acesso em: 20/03/04
- Barreto, Marcos Ribeiro Pereira; *Notas de aula da disciplina PMR2490 - Sistemas de Informação*; São Paulo, 2003
- Hertz, Alain; Taillard, Eric; de Werra, Dominique, *A tutorial on Tabu Search*, Proc. of Giornate di Lavoro AIRO'95, 1995

**ANEXO A**

Nas páginas seguintes estão descritos os casos de uso do sistema tratado nesse trabalho.

---

## **ESPECIFICAÇÃO DE CASOS DE USO PARA SEQÜENCIAMENTO DE PRODUÇÃO**

---

Versão 1.0

## Histórico das Revisões

Data	Versão	Descrição	Autor

---

# Índice

---

Especificação de Casos de Uso para <i>Sequenciamento de produção</i> .....	I
1 Apresentação .....	I
1.1 Objetivo.....	I
2 Conceitos Gerais.....	II
2.1 Dicionário de conceitos.....	II
3 Inclusão de produto .....	III
3.1 Breve Descrição.....	III
3.2 Atores.....	III
3.3 Pré-Condições.....	III
3.4 Fluxo de Eventos .....	III
3.4.1 Fluxo Básico .....	III
3.4.2 Fluxos Alternativos .....	III
3.4.3 Requerimentos Especiais.....	III
3.4.4 Pós-Condições .....	IV
3.4.5 Pontos de Extensão .....	IV
4 Alteração de produto .....	V
4.1 Breve Descrição.....	V
4.2 Atores.....	V
4.3 Pré-Condições.....	V
4.4 Fluxo de Eventos .....	V
4.4.1 Fluxo Básico .....	V
4.4.2 Fluxos Alternativos .....	V
4.4.3 Requerimentos Especiais.....	VI
4.4.4 Pós-Condições .....	VI
4.4.5 Pontos de Extensão .....	VI
5 Exclusão de produto .....	VII
5.1 Breve Descrição.....	VII
5.2 Atores.....	VII
5.3 Pré-Condições.....	VII
5.4 Fluxo de Eventos .....	VII
5.4.1 Fluxo Básico .....	VII
5.4.2 Fluxos Alternativos .....	VII
5.4.3 Requerimentos Especiais.....	VIII
5.4.4 Pós-Condições .....	VIII
5.4.5 Pontos de Extensão .....	VIII
6 Pesquisa de produto .....	IX
6.1 Breve Descrição.....	IX
6.2 Atores.....	IX
6.3 Pré-Condições.....	IX
6.4 Fluxo de Eventos .....	IX
6.4.1 Fluxo Básico .....	IX
6.4.2 Fluxos Alternativos .....	IX
6.4.3 Requerimentos Especiais.....	IX

6.4.4	Pós-Condições .....	X
6.4.5	Pontos de Extensão .....	X
7	Inclusão de linha de produção .....	XI
7.1	Breve Descrição .....	XI
7.2	Atores .....	XI
7.3	Pré-Condições .....	XI
7.4	Fluxo de Eventos .....	XI
7.4.1	Fluxo Básico .....	XI
7.4.2	Fluxos Alternativos .....	XI
7.4.3	Requerimentos Especiais .....	XII
7.4.4	Pós-Condições .....	XII
7.4.5	Pontos de Extensão .....	XII
8	Alteração de Linha de produção .....	XIII
8.1	Breve Descrição .....	XIII
8.2	Atores .....	XIII
8.3	Pré-Condições .....	XIII
8.4	Fluxo de Eventos .....	XIII
8.4.1	Fluxo Básico .....	XIII
8.4.2	Fluxos Alternativos .....	XIV
8.4.3	Requerimentos Especiais .....	XIV
8.4.4	Pós-Condições .....	XIV
8.4.5	Pontos de Extensão .....	XIV
9	Exclusão de Linha de produção .....	XV
9.1	Breve Descrição .....	XV
9.2	Atores .....	XV
9.3	Pré-Condições .....	XV
9.4	Fluxo de Eventos .....	XV
9.4.1	Fluxo Básico .....	XV
9.4.2	Fluxos Alternativos .....	XV
9.4.3	Requerimentos Especiais .....	XVI
9.4.4	Pós-Condições .....	XVI
9.4.5	Pontos de Extensão .....	XVI
10	Pesquisa de Linha de produção .....	XVII
10.1	Breve Descrição .....	XVII
10.2	Atores .....	XVII
10.3	Pré-Condições .....	XVII
10.4	Fluxo de Eventos .....	XVII
10.4.1	Fluxo Básico .....	XVII
10.4.2	Fluxos Alternativos .....	XVII
10.4.3	Requerimentos Especiais .....	XVII
10.4.4	Pós-Condições .....	XVIII
10.4.5	Pontos de Extensão .....	XVIII
11	Inclusão de Roteiro de montagem .....	XIX
11.1	Breve Descrição .....	XIX
11.2	Atores .....	XIX
11.3	Pré-Condições .....	XIX
11.4	Fluxo de Eventos .....	XIX
11.4.1	Fluxo Básico .....	XIX

11.4.2	Fluxos Alternativos .....	XIX
11.4.3	Requerimentos Especiais.....	XIX
11.4.4	Pós-Condições .....	XX
11.4.5	Pontos de Extensão .....	XX
12	<i>Alteração</i> dos Roteiros de montagem.....	XXI
12.1	Breve Descrição.....	XXI
12.2	Atores.....	XXI
12.3	Pré-Condições.....	XXI
12.4	Fluxo de Eventos .....	XXI
12.4.1	Fluxo Básico .....	XXI
12.4.2	Fluxos Alternativos .....	XXI
12.4.3	Requerimentos Especiais.....	XXII
12.4.4	Pós-Condições .....	XXII
12.4.5	Pontos de Extensão .....	XXII
13	Exclusão de Roteiro de montagem.....	XXIII
13.1	Breve Descrição.....	XXIII
13.2	Atores.....	XXIII
13.3	Pré-Condições.....	XXIII
13.4	Fluxo de Eventos .....	XXIII
13.4.1	Fluxo Básico .....	XXIII
13.4.2	Fluxos Alternativos .....	XXIII
13.4.3	Requerimentos Especiais.....	XXIV
13.4.4	Pós-Condições .....	XXIV
13.4.5	Pontos de Extensão .....	XXIV
14	<i>Pesquisa</i> de Roteiro de montagem.....	XXV
14.1	Breve Descrição.....	XXV
14.2	Atores.....	XXV
14.3	Pré-Condições.....	XXV
14.4	Fluxo de Eventos .....	XXV
14.4.1	Fluxo Básico .....	XXV
14.4.2	Fluxos Alternativos .....	XXV
14.4.3	Requerimentos Especiais.....	XXV
14.4.4	Pós-Condições .....	XXVI
14.4.5	Pontos de Extensão .....	XXVI
15	<i>Ordenação</i> da produção .....	XXVII
15.1	Breve Descrição.....	XXVII
15.2	Atores.....	XXVII
15.3	Pré-Condições.....	XXVII
15.4	Fluxo de Eventos .....	XXVII
15.4.1	Fluxo Básico .....	XXVII
15.4.2	Fluxos Alternativos .....	XXVII
15.4.3	Requerimentos Especiais.....	XXVII
15.4.4	Pós-Condições .....	XXVIII
15.4.5	Pontos de Extensão .....	XXVIII

---

## ***1. Apresentação***

---

### ***1.1 Objetivo***

O objetivo deste documento é a especificação de requisitos para o conjunto de funcionalidades referido como Sequenciamento de produção.

---

## **2. Conceitos Gerais**

---

### ***2.1. Dicionário de conceitos***

---

- Produto: Modelos que serão fabricados
- Linha de produção: Conjunto de equipamentos e máquinas dispostas em seqüência de forma a realizar uma seqüência de fabricação.
- Roteiro de montagem: Seqüência de Linha de produção que o Produto segue para a sua fabricação total.
- Melhor Ordenação da produção: Melhor ordem de montagem para atingir os objetivos da linha de produção

---

### ***3. Inclusão de produto***

---

#### ***3.1. Breve Descrição***

---

Destina-se ao cadastramento de novos produtos.

#### ***3.2. Atores***

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

#### ***3.3. Pré-Condições***

---

- os atores devem estar logados no sistema e com acesso à função.

#### ***3.4. Fluxo de Eventos***

---

##### ***3.4.1. Fluxo Básico***

---

1.O ator informa, em tela fornecida pelo sistema, o seguinte dado:

- nome descritivo do produto, de acordo com o conceito Produto, string com até 200 caracteres, obrigatório

2.O sistema gera um código de produto, que é um número sequencial inteiro

3.O sistema armazena os dados relativos ao produto

4.O sistema informa ao operador o código de produto que foi gerado, re-enviando o formulário de cadastramento de produto para que o ator possa cadastrar um novo produto, se desejar.

##### ***3.4.2. Fluxos Alternativos***

---

a.Desistência

1.Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

##### ***3.4.3. Requerimentos Especiais***

---

Não aplicável.

#### **3.4.4. Pós-Condições**

---

- produto incluído, se as regras de validação forem verificadas.

#### **3.4.5. Pontos de Extensão**

---

Nenhum.

---

## **4. Alteração de produto**

---

### ***4.1. Breve Descrição***

---

Destina-se à alteração de produtos cadastrados.

### ***4.2. Atores***

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### ***4.3. Pré-Condições***

---

- os atores devem estar logados no sistema e com acesso à função.

### ***4.4. Fluxo de Eventos***

---

#### ***4.4.1. Fluxo Básico***

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código do produto, E/OU
- parte ou a totalidade do nome descritivo do produto.

2. O sistema verifica a validade do código ou nome fornecido

3. O sistema exibe através de uma tela, uma lista com os produtos que condizem com os dados fornecidos.

4. O ator seleciona o produto desejado na tela fornecida.

5. O ator altera, em tela fornecida pelo sistema alterando um ou mais dos seguintes dados:

- nome descritivo do produto, string com até 200 caracteres, obrigatório

6. O sistema atualiza os dados relativos ao produto

7. O sistema re-envia o formulário de alteração de produto para que o ator possa alterar outro produto, se desejar.

#### ***4.4.2. Fluxos Alternativos***

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

b. Código de produto inexistente

1. O sistema informa a inexistência do código, e apresenta novamente o formulário de alteração de produto.

c. Nome de produto inexistente

1. O sistema informa a inexistência do nome, e apresenta novamente o formulário de alteração de produto.

---

#### *4.4.3. Requerimentos Especiais*

---

Não aplicável.

---

#### *4.4.4. Pós-Condições*

---

- produto alterado, se as regras de validação forem verificadas.

---

#### *4.4.5. Pontos de Extensão*

---

Nenhum.

---

## 5. Exclusão de produto

---

### 5.1. Breve Descrição

---

Destina-se à exclusão de produtos cadastrados.

### 5.2. Atores

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### 5.3. Pré-Condições

---

- os atores devem estar logados no sistema e com acesso à função.

### 5.4. Fluxo de Eventos

---

#### 5.4.1. Fluxo Básico

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código do produto, E/OU
- parte ou a totalidade do nome descritivo do produto.

2. O sistema verifica a validade do código ou nome fornecido

3. O sistema exibe através de uma tela, uma lista com os produtos que condizem com os dados fornecidos.

4. O ator seleciona o produto desejado na tela fornecida.

5. O ator altera, em tela fornecida pelo sistema apresentando os seguintes dados:

- nome descritivo do produto, string com até 200 caracteres, obrigatório

6. O sistema exclui o produto

7. O sistema re-envia o formulário de exclusão de produto para que o ator possa excluir outro produto, se desejar.

#### 5.4.2. Fluxos Alternativos

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

b. Código de produto inexistente

1. O sistema informa a inexistência do código, e apresenta novamente o formulário de alteração de produto.

c. Nome de produto inexistente

1. O sistema informa a inexistência do nome, e apresenta novamente o formulário de alteração de produto.

#### ***5.4.3. Requerimentos Especiais***

---

*Não aplicável.*

#### ***5.4.4. Pós-Condições***

---

- produto excluído, se as regras de validação forem verificadas.
- Verificação e abertura de tela mostrando os roteiros de montagem que eram vinculados ao produto excluído. Só sairá desta tela se o usuário alterar o(s) roteiro(s) mostrados.

#### ***5.4.5. Pontos de Extensão***

---

*Nenhum.*

---

## 6. Pesquisa de produto

---

### 6.1. Breve Descrição

---

Destina-se à listagem de produtos cadastrados.

### 6.2. Atores

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### 6.3. Pré-Condições

---

- os atores devem estar logados no sistema e com acesso à função.

### 6.4. Fluxo de Eventos

---

#### 6.4.1. Fluxo Básico

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código do produto, E/OU
- parte ou a totalidade do nome descritivo do produto.

2. O sistema verifica a validade do código ou nome fornecido

3. O sistema lista os produtos que atendam ao critério de pesquisa, por semelhança. Apresentando novamente o formulário de pesquisa com o campos contendo o critério da pesquisa anterior.

#### 6.4.2. Fluxos Alternativos

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

#### 6.4.3. Requerimentos Especiais

---

Não aplicável.

#### ***6.4.4. Pós-Condições***

---

- Listagem de produtos, ou Nenhum produto encontrado caso a listagem seja nula.

#### ***6.4.5. Pontos de Extensão***

---

*Nenhum.*

---

## **7. Inclusão de linha de produção**

---

### **7.1. Breve Descrição**

---

Destina-se ao cadastramento das linhas de produção de acordo com o conceito de Linha de produção.

### **7.2. Atores**

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### **7.3. Pré-Condições**

---

- os atores devem estar logados no sistema e com acesso à função.

### **7.4. Fluxo de Eventos**

---

#### **7.4.1. Fluxo Básico**

---

1.O ator informa, em tela fornecida pelo sistema, os seguintes dados

- Nome da Linha de produção, de acordo com o conceito Linha de produção, string com até 50 caracteres, obrigatório
- Tempo de fabricação, Hora ,obrigatório
- Quantidade máxima, inteiro maior que zero, obrigatório

2.O sistema gera um código da Linha de produção, que é um número seqüencial inteiro

3.O sistema armazena os dados relativos à linha

4.O sistema informa ao operador a descrição do produto que incluído, re-enviando o formulário de cadastramento de item de embarcação para que o ator possa cadastrar um novo produto, se desejar.

#### **7.4.2. Fluxos Alternativos**

---

a.Desistência

1.Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

#### 7.4.3. Requerimentos Especiais

Não aplicável.

#### 7.4.4. Pós-Condições

- Linha de produção incluída, se as regras de validação forem verificadas.

#### 7.4.5. Pontos de Extensão

Nenhum.

---

## **8. Alteração de Linha de produção**

---

### ***8.1. Breve Descrição***

---

Destina-se à alteração de Linhas de produção cadastradas.

### ***8.2. Atores***

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### ***8.3. Pré-Condições***

---

- os atores devem estar logados no sistema e com acesso à função.

### ***8.4. Fluxo de Eventos***

---

#### ***8.4.1. Fluxo Básico***

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código da Linha de produção, E/OU
- parte ou a totalidade do nome descritivo da Linha de produção.

2. O sistema verifica a validade do código ou nome fornecido

3. O sistema exibe através de uma tela, uma lista com as Linhas de produção que condizem com os dados fornecidos.

4. O ator seleciona a Linha de produção desejada na tela fornecida.

5. O ator altera, em tela fornecida pelo sistema alterando um ou mais dos seguintes dados:

- Nome da Linha de produção, de acordo com o conceito Linha de produção, string com até 50 caracteres, obrigatório
- Tempo de fabricação, Hora, obrigatório
- Quantidade máxima, inteiro maior que zero, obrigatório

6. O sistema atualiza os dados relativos ao produto

7. O sistema re-envia o formulário de alteração de item de embarcação para que o ator possa alterar outro item de embarcação, se desejar.

#### ***8.4.2. Fluxos Alternativos***

---

**a. Desistência**

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

**b. Código da Linha de produção inexistente**

1. O sistema informa a inexistência do código, e apresenta novamente o formulário de alteração da Linha de produção.

**c. Nome da Linha de produção inexistente**

1. O sistema informa a inexistência do nome, e apresenta novamente o formulário de alteração da Linha de produção.

#### ***8.4.3. Requerimentos Especiais***

---

Não aplicável.

#### ***8.4.4. Pós-Condições***

---

- Linha de produção alterada, se as regras de validação forem verificadas.

#### ***8.4.5. Pontos de Extensão***

---

Nenhum.

---

## **9. Exclusão de Linha de produção**

---

### **9.1. Breve Descrição**

---

Destina-se à exclusão de Linhas de produção cadastradas.

### **9.2. Atores**

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### **9.3. Pré-Condições**

---

- os atores devem estar logados no sistema e com acesso à função.

### **9.4. Fluxo de Eventos**

---

#### **9.4.1. Fluxo Básico**

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código da Linha de produção, E/OU
- parte ou a totalidade do nome descritivo da Linha de produção.

2. O sistema verifica a validade do código ou nome fornecido

3. O sistema exibe através de uma tela, uma lista com as Linhas de produção que condizem com os dados fornecidos.

4. O ator seleciona a Linha de produção desejada na tela fornecida.

5. O ator confirma a exclusão do item de embarcação em tela fornecida pelo sistema contendo os seguintes dados:

- Nome da Linha de produção
- Tempo de fabricação
- Quantidade máxima

6. O sistema exclui o produto

7. O sistema re-envia o formulário de exclusão de produto para que o ator possa excluir outra Linha de produção, se desejar.

#### **9.4.2. Fluxos Alternativos**

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

b. Código da Linha de produção inexistente

1. O sistema informa a inexistência do código, e apresenta novamente o formulário de exclusão da Linha de produção.

c. Nome da Linha de produção inexistente

1. O sistema informa a inexistência do nome, e apresenta novamente o formulário de exclusão da Linha de produção.

---

#### *9.4.3. Requerimentos Especiais*

---

Não aplicável.

---

#### *9.4.4. Pós-Condições*

---

- Linha de produção excluída, se as regras de validação forem verificadas.
- Verificação e abertura de tela mostrando os roteiros de montagem que eram vinculados ao produto excluído. Só sairá desta tela se o usuário alterar o(s) roteiro(s) mostrados.

---

#### *9.4.5. Pontos de Extensão*

---

Nenhum.

---

## **10. Pesquisa de Linha de produção**

---

### ***10.1. Breve Descrição***

---

Destina-se à listagem de Linhas de produção cadastradas.

### ***10.2. Atores***

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### ***10.3. Pré-Condições***

---

- os atores devem estar logados no sistema e com acesso à função.

### ***10.4. Fluxo de Eventos***

---

#### ***10.4.1. Fluxo Básico***

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código da Linha de produção, E/OU
- parte ou a totalidade do nome descritivo da Linha de produção, E/Ou
- Tempo de fabricação, E/Ou
- Quantidade máxima, E/Ou

2. O sistema verifica a validade do código, nome, tempo e/ou quantidade fornecidos

3. O sistema lista as Linhas de produção que atendam ao critério de pesquisa, por semelhança. Apresentando novamente o formulário de pesquisa com os campos contendo o critério da pesquisa anterior.

#### ***10.4.2. Fluxos Alternativos***

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

#### ***10.4.3. Requerimentos Especiais***

---

Não aplicável.

#### **10.4.4. Pós-Condições**

---

- Listagem das Linhas de produção, ou Nenhuma Linha de produção encontrada caso a listagem seja nula.

#### **10.4.5. Pontos de Extensão**

---

*Nenhum.*

---

## **11. Inclusão de Roteiro de montagem**

---

### **11.1. Breve Descrição**

---

Destina-se ao cadastramento dos Roteiros de montagem de acordo com o conceito de Roteiro de montagem.

### **11.2. Atores**

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### **11.3. Pré-Condições**

---

- os atores devem estar logados no sistema e com acesso à função.

### **11.4. Fluxo de Eventos**

---

#### **11.4.1. Fluxo Básico**

---

1. O ator informa, em tela fornecida pelo sistema, os seguintes dados

- Código do Produto, inteiro, obrigatório
- Ordem de montagem, inteiro, obrigatório
- Código da Linha de produção, inteiro, obrigatório

2. O sistema verifica a consistência dos dados

3. O sistema armazena os dados relativos ao Roteiro de montagem

4. O sistema informa ao operador a descrição do Roteiro de montagem que incluído, re-enviando o formulário de cadastramento de Roteiro de montagem para que o ator possa cadastrar um novo produto, se desejar.

#### **11.4.2. Fluxos Alternativos**

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

#### **11.4.3. Requerimentos Especiais**

---

Não aplicável.

#### ***11.4.4. Pós-Condições***

---

- Roteiro de montagem incluído, se as regras de validação forem verificadas.

#### ***11.4.5. Pontos de Extensão***

---

Nenhum.

---

## **12. Alteração dos Roteiros de montagem**

---

### **12.1. Breve Descrição**

Destina-se à alteração dos Roteiros de montagem cadastrados.

### **12.2. Atores**

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### **12.3. Pré-Condições**

- os atores devem estar logados no sistema e com acesso à função.

### **12.4. Fluxo de Eventos**

#### **12.4.1. Fluxo Básico**

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código do Produto

2. O sistema verifica a validade do código fornecido

3. O sistema exibe através de uma tela, uma lista com o Roteiro de montagem do Produto fornecido

5. O ator altera, em tela fornecida pelo sistema alterando um ou mais dos seguintes dados:

- Ordem de montagem, inteiro, obrigatório
- Código da Linha de produção, inteiro, obrigatório

6. O sistema atualiza os dados relativos ao produto

7. O sistema re-envia o formulário de alteração dos Roteiros de montagem para que o ator possa alterar outro Roteiro de montagem, se desejar.

#### **12.4.2. Fluxos Alternativos**

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

b. Código do Produto inexistente

1. O sistema informa a inexistência do código, e apresenta novamente o formulário de alteração do Roteiro de montagem.

#### **12.4.3. Requerimentos Especiais**

---

Não aplicável.

#### **12.4.4. Pós-Condições**

---

- Roteiro de montagem alterado, se as regras de validação forem verificadas.

#### **12.4.5. Pontos de Extensão**

---

Nenhum.

---

## **13. Exclusão de Roteiro de montagem**

---

### **13.1. Breve Descrição**

---

Destina-se à exclusão de Roteiro de montagem cadastrados.

### **13.2. Atores**

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### **13.3. Pré-Condições**

---

- os atores devem estar logados no sistema e com acesso à função.

### **13.4. Fluxo de Eventos**

---

#### **13.4.1. Fluxo Básico**

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código do Produto

2. O sistema verifica a validade do código fornecido

3. O sistema exibe através de uma tela, uma lista com o Roteiro de montagem do Produto fornecido

4. O ator seleciona o Roteiro de montagem desejado na tela fornecida.

5. O ator confirma a exclusão do item de embarcação em tela fornecida pelo sistema contendo os seguintes dados:

- Código do Produto, inteiro, obrigatório
- Ordem de montagem, inteiro, obrigatório
- Código da Linha de produção, inteiro, obrigatório

6. O sistema exclui o produto

7. O sistema re-envia o formulário de exclusão de produto para que o ator possa excluir outro Roteiro de montagem, se desejar.

#### **13.4.2. Fluxos Alternativos**

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

b. Código do Produto inexistente

1. O sistema informa a inexistência do código, e apresenta novamente o formulário de alteração do Roteiro de montagem.

#### **13.4.3. Requerimentos Especiais**

---

Não aplicável.

#### **13.4.4. Pós-Condições**

---

- Roteiro de montagem excluído, se as regras de validação forem verificadas.

#### **13.4.5. Pontos de Extensão**

---

Nenhum.

---

## **14. Pesquisa de Roteiro de montagem**

---

### **14.1. Breve Descrição**

---

Destina-se à listagem dos Roteiros de montagem cadastrados.

### **14.2. Atores**

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção.

### **14.3. Pré-Condições**

---

- os atores devem estar logados no sistema e com acesso à função.

### **14.4. Fluxo de Eventos**

---

#### **14.4.1. Fluxo Básico**

---

1. O ator informa, em tela fornecida pelo sistema:

- parte ou a totalidade do código do Produto, E/OU
- parte ou a totalidade do Código da Linha de produção, E/OU

2. O sistema verifica a validade do(s) código(s) fornecido(s)

3. O sistema lista os Roteiros de montagem que atendam ao critério de pesquisa, por semelhança. Apresentando novamente o formulário de pesquisa com os campos contendo o critério da pesquisa anterior.

#### **14.4.2. Fluxos Alternativos**

---

a. Desistência

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

#### **14.4.3. Requerimentos Especiais**

---

Não aplicável.

#### ***14.4.4. Pós-Condições***

---

- Listagem das Linhas de produção, ou Nenhuma Linha de produção encontrada caso a listagem seja nula.

#### ***14.4.5. Pontos de Extensão***

---

*Nenhum.*

---

## **15. Ordenação da produção**

---

### ***15.1. Breve Descrição***

---

Destina-se à Ordenação da produção.

### ***15.2. Atores***

---

Este caso de uso é de uso exclusivo de um engenheiro de produção ou do gerente de produção e do programador.

### ***15.3. Pré-Condições***

---

- os atores devem estar logados no sistema e com acesso à função.

### ***15.4. Fluxo de Eventos***

---

#### ***15.4.1. Fluxo Básico***

---

1. O Sistema gera uma lista com os produtos cadastrados
2. O ator informa, em tela fornecida pelo sistema:
  - Quantidade a ser fabricada de cada produto
2. O sistema verifica a validade dos dados fornecidos
3. O sistema gera a melhor Ordenação da produção, de acordo com o conceito de melhor Ordenação da produção.

#### ***15.4.2. Fluxos Alternativos***

---

##### **a. Desistência**

1. Se o ator selecionar a opção "Cancelar", o caso de uso se encerra, voltando o sistema à tela principal.

#### ***15.4.3. Requerimentos Especiais***

---

## Implementação do algoritmo de sequenciamento

### 15.4.4. Pós-Condições

- Listagem da Ordenação da produção.

### 15.4.5. Pontos de Extensão

Nenhum.

**ANEXO B**

Nas páginas seguintes está o manual de utilização do programa desenvolvido para este trabalho.

# **Seqüenciamento de Produção**

v1.0

Novembro de 2004

# Índice

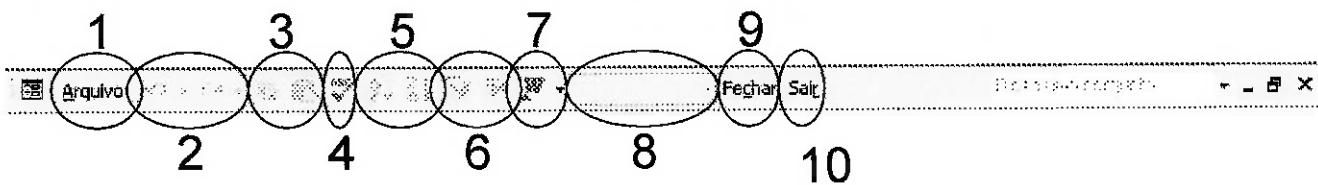
- Introdução
- Barra de Ferramentas
- Login
- Alterar Senha
- Níveis de Acesso
- Menu Inicial
  - Gerenciamento de Níveis de Acesso
  - Gerenciamento de Usuários
  - Gerenciamento de Produtos
  - Gerenciamento de Maquinas
  - Gerenciamento de Roteiros de Montagem
  - Seqüencial

# Introdução

Este manual tem como objetivo a familiarização e o aprendizado do software de seqüenciamento de produção.

# Barra de Ferramentas

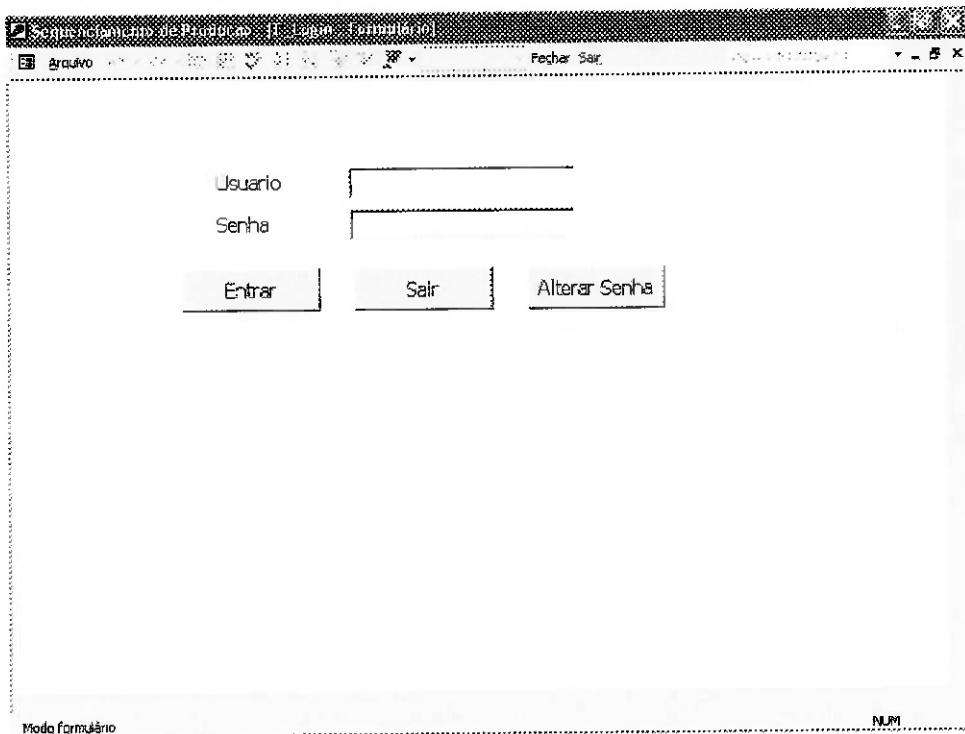
## Funcionalidades da barra de ferramentas



- 1 – Menu com as principais opções o sistema
- 2 – Desfazer e Refazer, respectivamente
- 3 – Copiar e Colar, respectivamente
- 4 – Verificação ortográfica
- 5 – Ordenação crescente ou decrescente, respectivamente.  
Valido quando selecionado um campo em um formulário
- 6 – Executar filtro e desfazer filtro, respectivamente. Também valido quando se seleciona algum informação em um formulário
- 7 – Vínculos do Office, permite analisar os dados do formulário nos aplicativos do Office, como Word e Excel.
- 8 – Zoom, aplicável principalmente em relatórios
- 9 – Fechar. Fecha a tela atualmente aberta, voltando para a tela anterior. Salva automaticamente todas as alterações efetuadas.
- 10 – Sair. Sai do aplicativo. Salva automaticamente todas as alterações efetuadas.

# Login

Responsável por permitir acesso ao sistema somente para as pessoas autorizadas.



- Entre com o usuário e a senha
  - Clique em “Entrar” para entrar no sistema
  - Clique em “Alterar Senha” caso deseje mudar sua senha.
- Clique em “Sair” para sair do sistema

# Alterar Senha

- Caso deseje alterar sua senha, digite a nova senha duas vezes nos campos “Nova Senha” e “Nova Senha (Confirmar)”.
  - Clique em “Entrar” para alterar a senha e entrar no sistema
  - Clique em “Voltar” para desistir de alterar a senha.

Obs: A senha deve ter no maximo 10 caracteres.

Alterar Senha

Arquivo Fechar Sair

Usuario: 1

Senha: \*

Nova Senha: 1

Nova Senha (Confirmar): 1

Entrar Voltar

Modo Formulário NJM

# Níveis de Acesso

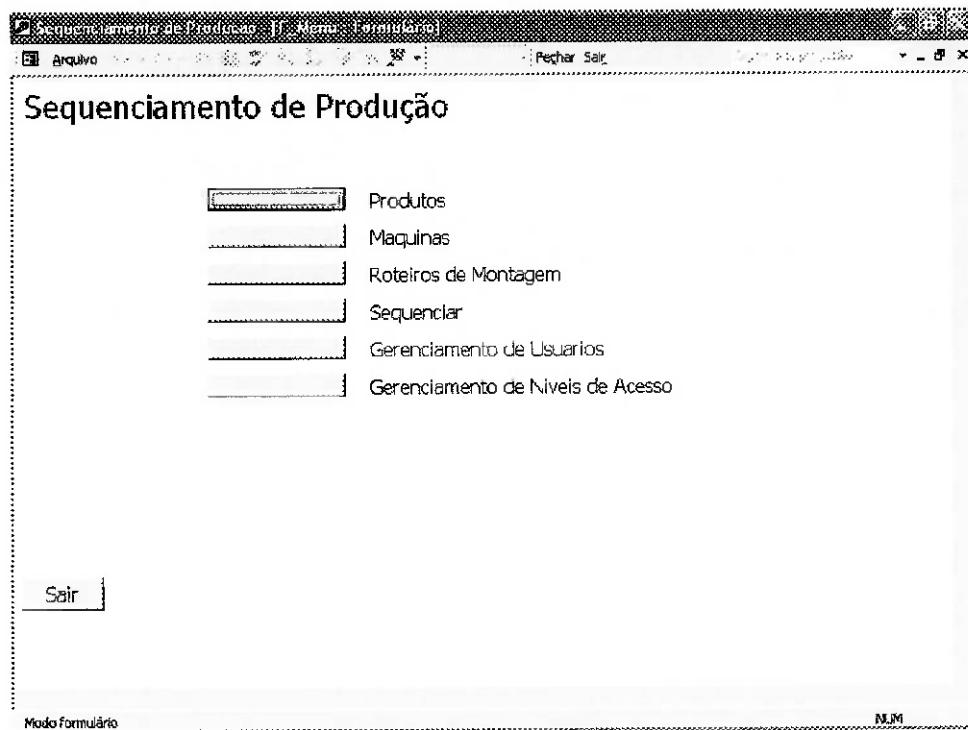
O sistema possui três níveis diferentes de acesso. São eles:

- Administrador
  - Responsável pela manutenção do sistema
  - Acesso total ao sistema, incluindo a parte de programação.
- Engenheiro ou Gerente de Produção
  - Responsável pelo cadastro dos produtos, máquinas e roteiros
  - Acesso quase total ao sistema, excluindo a parte de programação.
- Programador
  - Acesso ao menu de seqüenciamento

# Menu Inicial

Primeira tela do programa. Menu principal para todas as funcionalidades do sistema.

Obs: O acesso às funções e a quantidade de opções pode variar de acordo com o seu nível de acesso.



Gerenciamento de:

- Produtos (Botão “Produtos”)
- Maquinas (Botão “Maquinas”)
- Roteiros (Botão “Roteiros de Montagem”)
- Seqüenciamento (Botão “Seqüencial”)
- Usuários (Botão “Gerenciamento de usuários”)
- Níveis de Acesso (Botão “Gerenciamento de Níveis de Acesso”)

Botão de saída do sistema “Sair”

# Gerenciamento de Níveis de Acesso

Gerenciamento (Inclusão, Alteração e exclusão) dos níveis de acesso existentes.

Obs1: No caso de adição de níveis, será necessária a atribuição de acessos em nível de programação, com acesso de administrador.

Obs2: Os números em Nível de Acesso precisam ser diferentes.

Seqüenciamento de Produção

Nível de Acesso

Nível de Acesso	Descrição do Nível de Acesso	Excluir
1	Administrador	X
2	Engenheiro ou Gerente de Produção	X
3	Programador	X
0		X

Voltar

Botão para a exclusão do registro

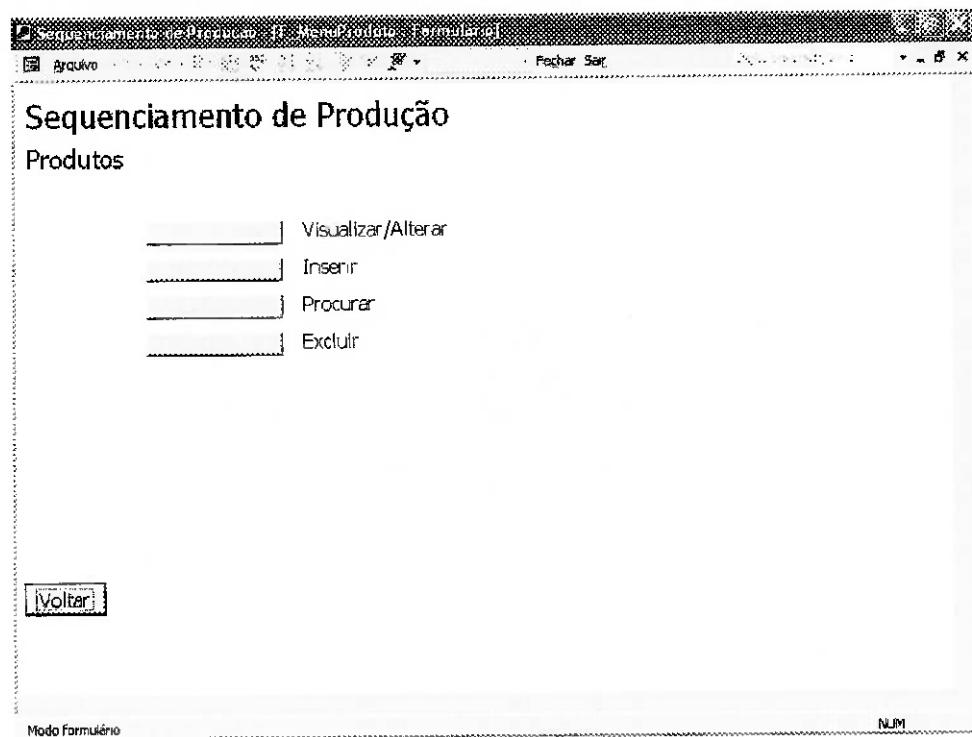
# Gerenciamento de Usuários

Responsável pelo gerenciamento dos usuários e da atribuição dos seus respectivos níveis de acesso.

Usuario	Senha	NivelAcesso
1	1	Engenheiro ou Gerente de Produção
3	3	Programador
ithis	ithis	Administrador
Usuario	Usuario	Engenheiro ou Gerente de Produção

# Gerenciamento dos Produtos

Menu com as opções para o gerenciamento dos produtos.



# Gerenciamento dos Produtos -Visualizar

Visualização e alteração dos produtos já cadastrados. Os dados são salvos de maneira automática.

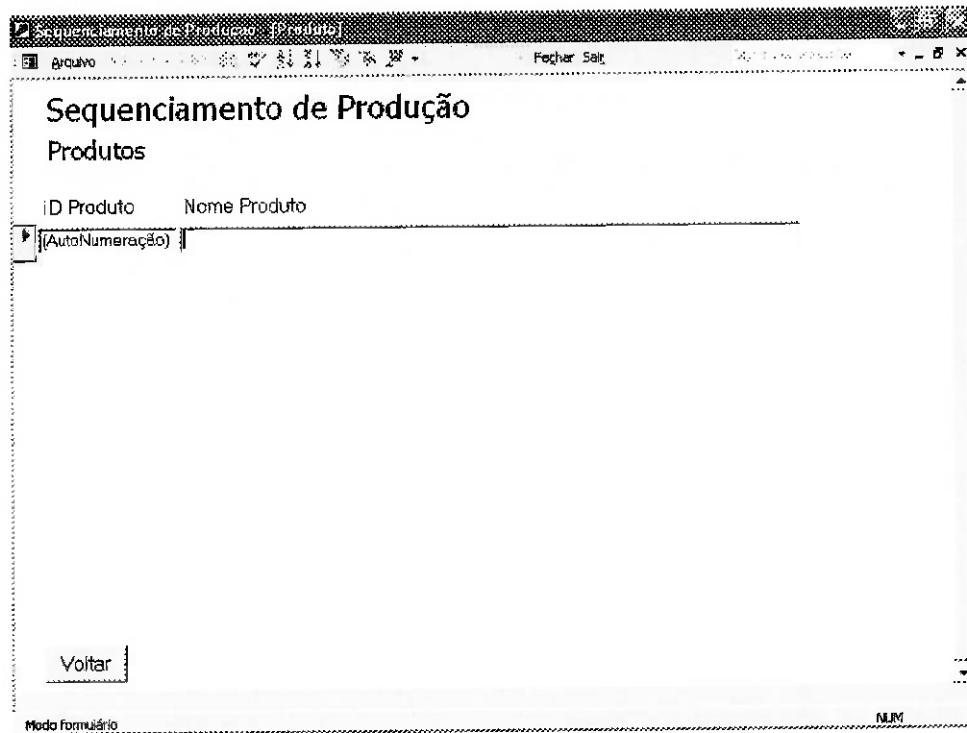
ID Produto	Nome Produto
1	LN
2	L98
3	FPN
4	HPN
5	HSK
6	SKL
7	Canavieiro
8	FSK
9	LTC

# Gerenciamento dos Produtos - Inserir

Inserção de novos produtos.

- Inserir o Nome do novo produto
- Clicar em “Voltar”

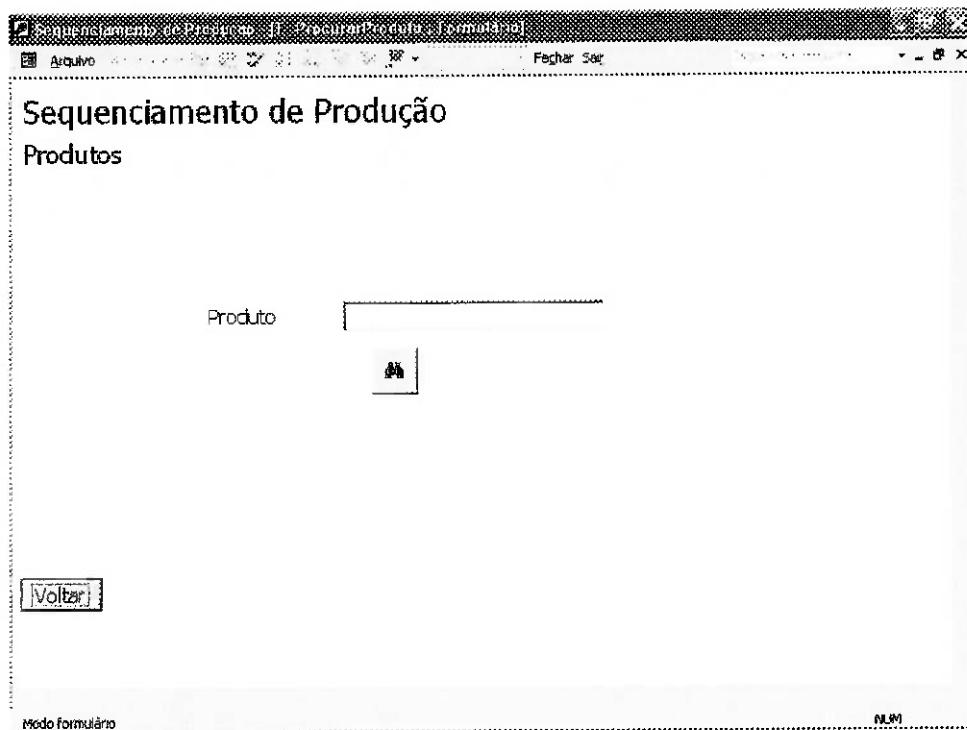
Obs: Os dados são salvos de maneira automática, quando apertado o botão “Voltar”.



# Gerenciamento dos Produtos - Procurar

Procura de produtos já cadastrados.

- Inserir o Nome do produto desejado.
- Clicar no botão com o binóculos. O sistema exibira a lista de produtos com o nome semelhante ao desejado.

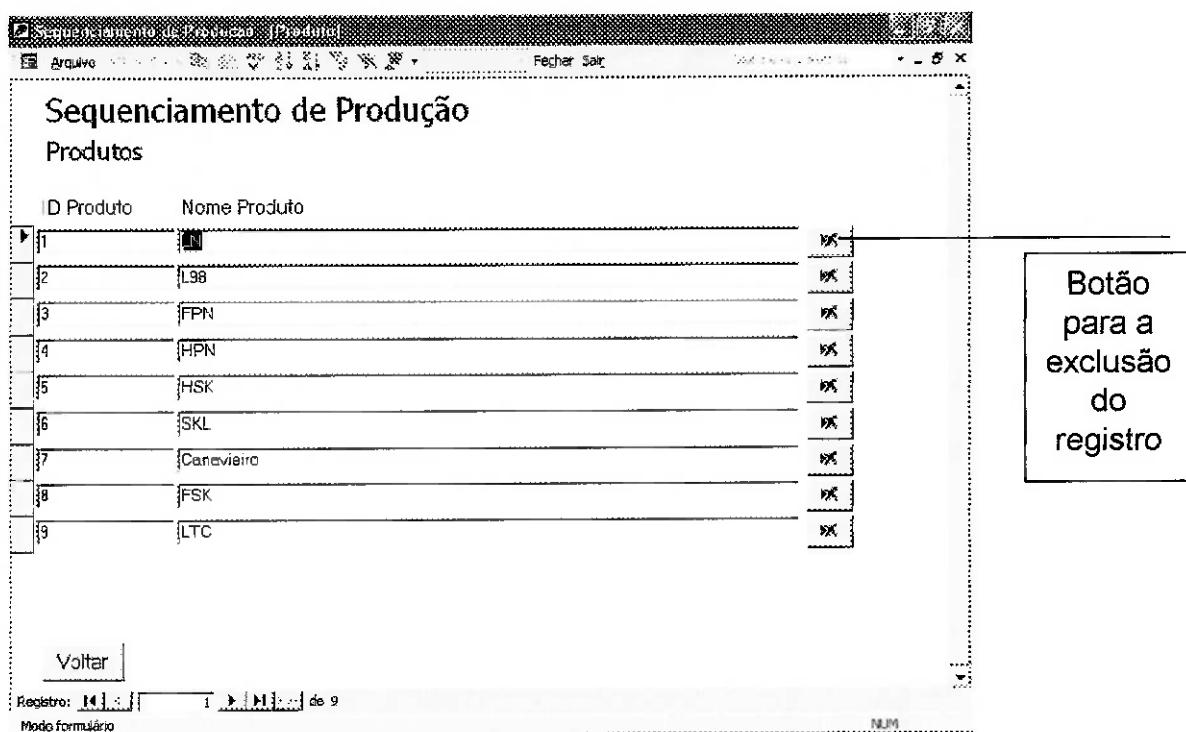


# Gerenciamento dos Produtos - Excluir

Exclusão de produtos.

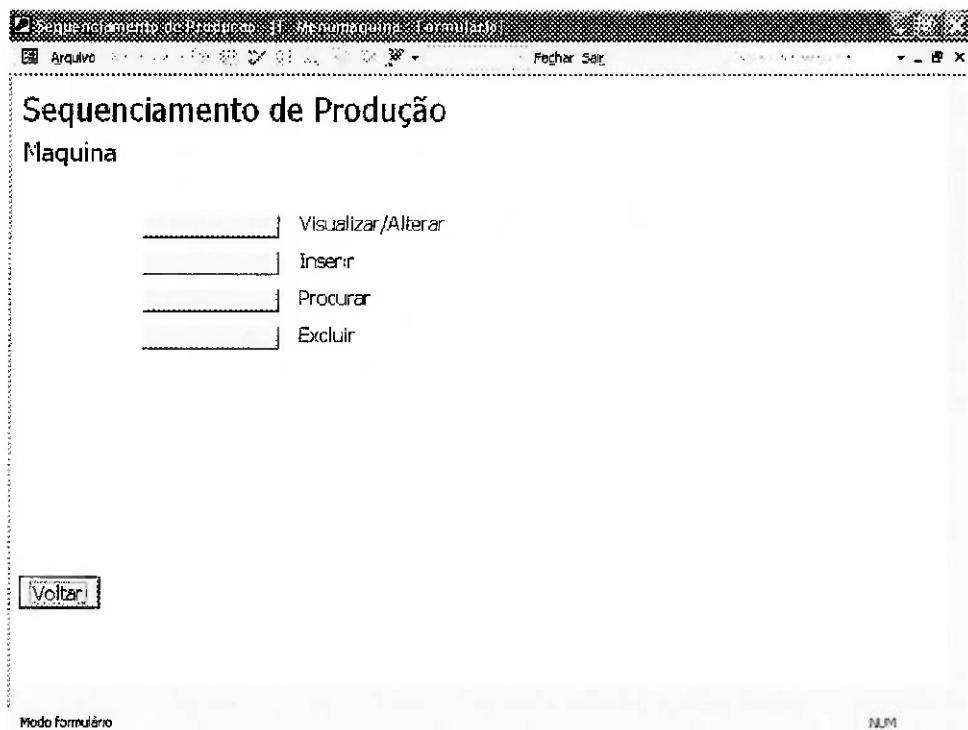
- Verificar o produto que se deseja excluir.
- Clicar no botão com X vermelho no canto direito da tela do produto que se deseja excluir.

Obs: Os dados são salvos de maneira automática, quando apertado o botão “Voltar”.



# Gerenciamento de Maquinas

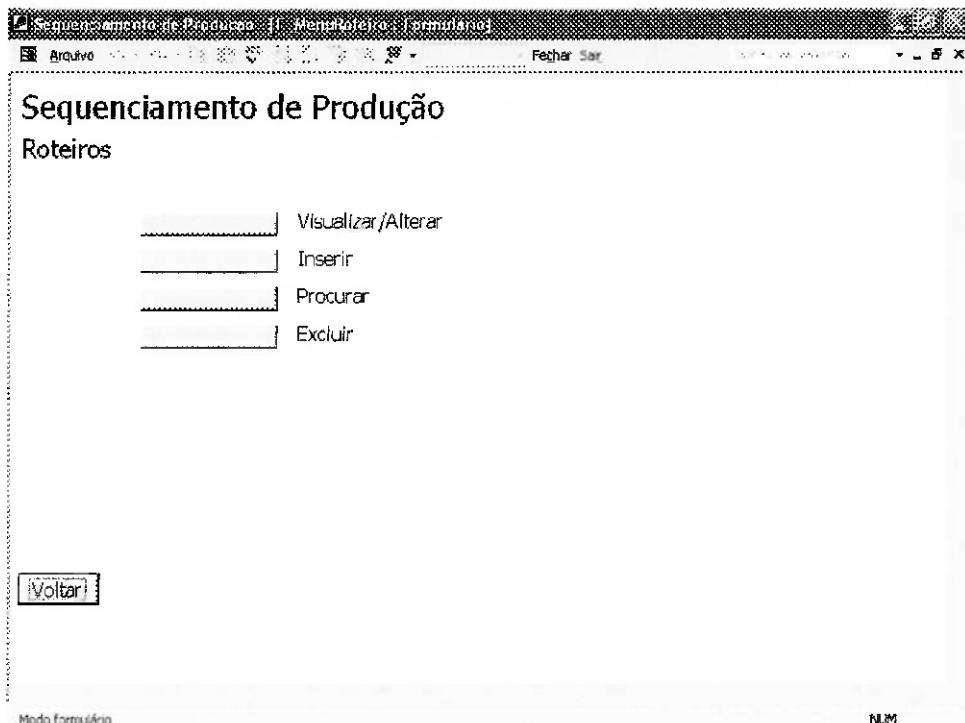
Os procedimentos para o gerenciamento das maquinas são semelhantes ao gerenciamento dos produtos. Basta trocar os nomes produto por maquina nas paginas 11 a 15 deste manual.



# Gerenciamento de Roteiros de Montagem

Menu para o gerenciamento dos roteiros de montagem.

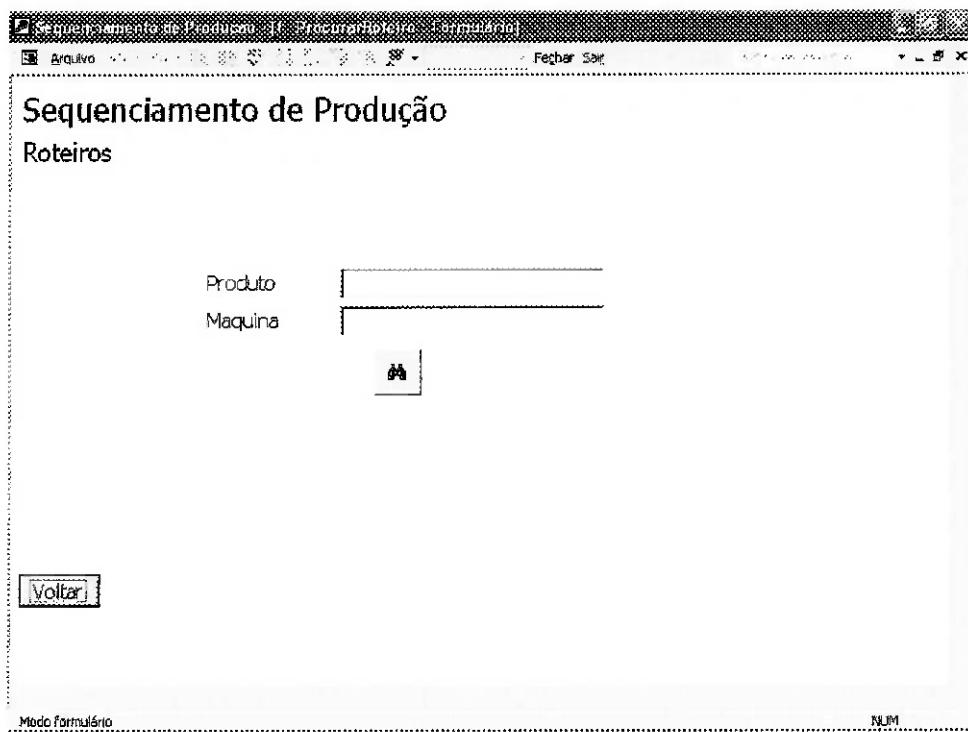
Para facilitar o gerenciamento dos roteiros, todos os botões abrem a tela de procura do roteiro. Isto ocorre pois o gerenciamento dos roteiros de montagem possui uma complexidade muito maior do que os gerenciamentos de produtos e maquinas.



# Gerenciamento de Roteiros de Montagem - Procura

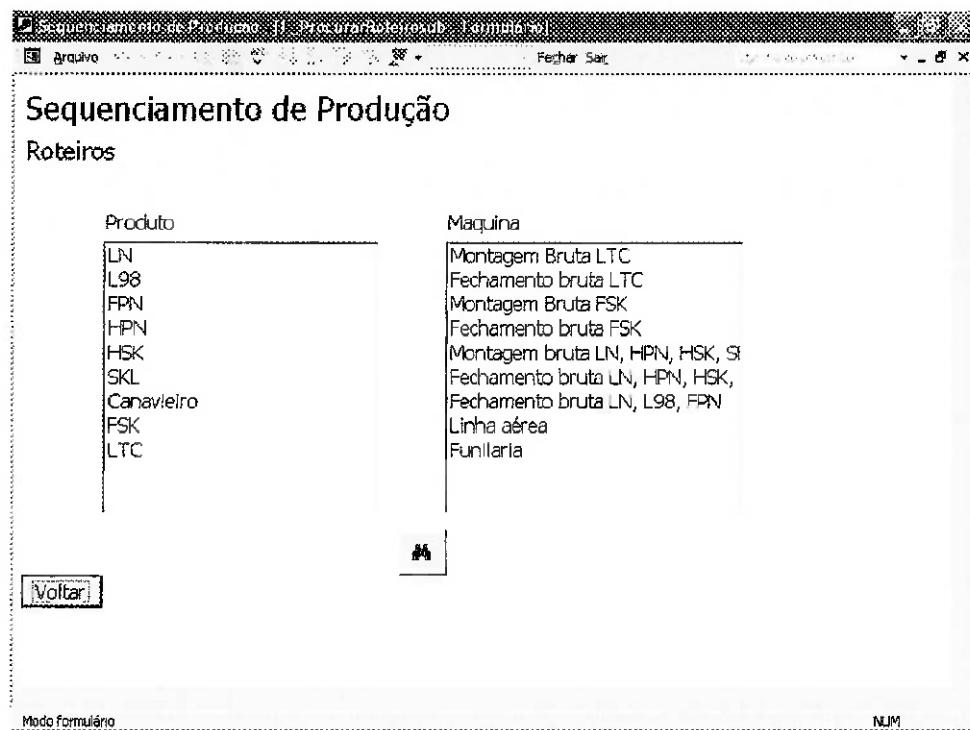
Menu para a procura dos roteiros de montagem.

- Entrar com os nomes parciais ou inteiros do produto e/ou da maquina ou deixar ambos em branco para obter a lista completa.
- Clicar no botão com o desenho de um binóculos.



# Gerenciamento de Roteiros de Montagem - Procura

- Abre-se a tela com as opções de produto e máquina disponíveis de acordo com o que foi digitado na tela anterior.
- Selecionar um produto e/ou uma máquina desejada.
- Clicar no botão com a figura de um binóculos.



# Gerenciamento de Roteiros de Montagem - Procura

- O sistema gera a lista de acordo com o que foi procurado na tela anterior.

Sequenciamento de Produção - Roteiro de montagem

Arquivo Editar Busca Formularios Ajuda Fechar Sair

Sequenciamento de Produção

Roteiro

Produto	Maquina	Posicao	Tempo (min)
LN	Montagem bruta LN, HPN, HSK, SKL, Can, L98, FPN	1	49
LN	Fechamento bruta LN, HPN, HSK, SKL, Can	2	66
LN	Fechamento bruta LN, L98, FPN	2	66
LN	Linha aérea	4	15
LN	Funilaria	3	7

Voltar

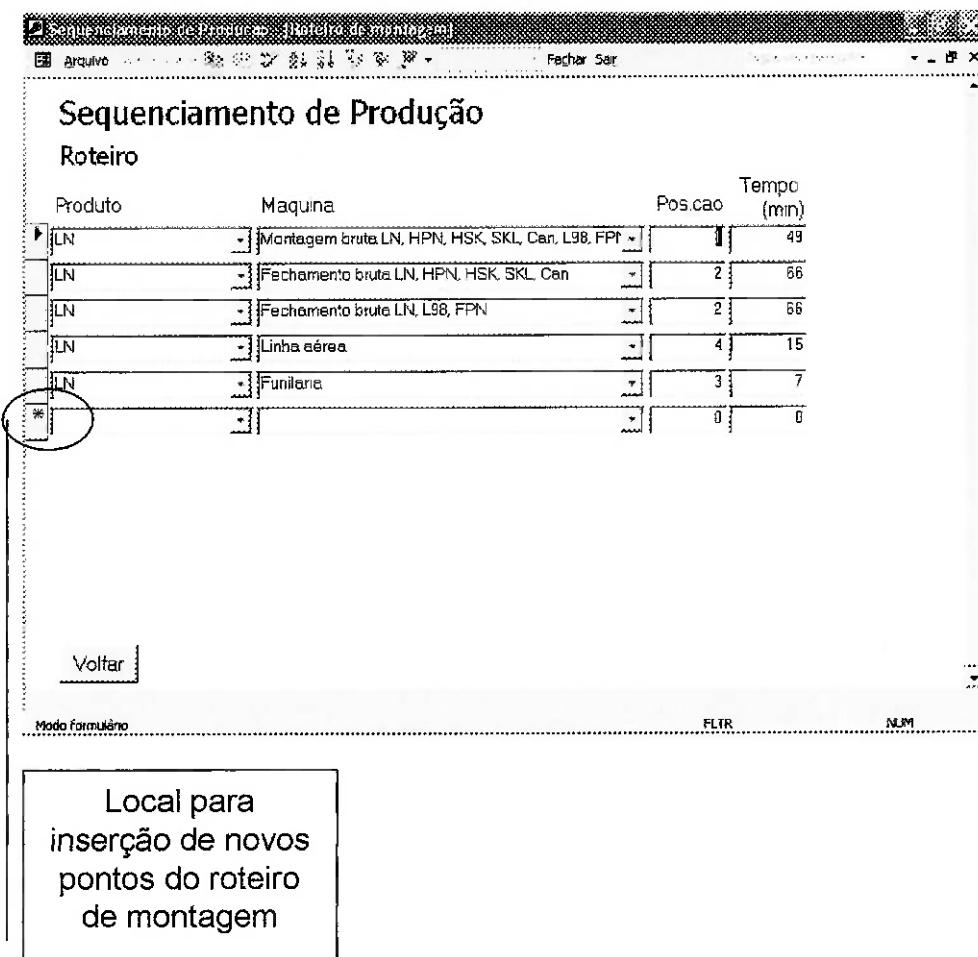
Modo formulário

FLTR

NUM

# Gerenciamento de Roteiros - Inclusão

Para a inserção de um novo ponto do roteiro. Ir com a barra de rolagem ate a extremidade inferior. O ultimo campo que aparecer será o campo para a inclusão de um novo registro.



# Gerenciamento de Roteiros - Exclusão

- Localizar o registro que se deseja excluir.
- Clicar no botão na parte direita do registro que se deseja excluir.

Sequenciamento de Produção - Roteiro de montagem

Arquivo Editar Visualizar Roteiro de montagem Fechar Sair

Sequenciamento de Produção

Roteiro

Produto	Maquina	Pos.cao	Tempo (min)	
LN	Montagem bruta LN, HPN, HSK, SKL, Can, L98, FPN	1	49	<input checked="" type="checkbox"/>
LN	Fechamento bruta LN, HPN, HSK, SKL, Can	2	66	<input checked="" type="checkbox"/>
LN	Fechamento bruta LN, L98, FPN	2	66	<input checked="" type="checkbox"/>
LN	Linha aérea	4	15	<input checked="" type="checkbox"/>
LN	Funiaria	3	7	<input checked="" type="checkbox"/>

Voltar

Modo formulário

FLTR

NLM

Botão para a exclusão do registro

# Seqüencial

Menu para seqüenciamento de produção.

Seqüenciamento de Produção

Produtos

Seqüencia	Produto
1	FPN
2	FPN
3	FPN
4	FPN
5	FPN
6	L98
7	L98
8	L98
9	L98
10	L98
11	LN
12	LN

Importar Seqüencia 1

2

Voltar 3

Seqüenciar

Modo Formulário

NUM

Executar as operações na seguinte ordem:

- 1 – Importação da seqüência a ser seqüenciada.
- 2 – Visualização e possíveis alterações, se desejar.
- 3 – Botão que iniciara o seqüenciamento.

# Seqüencial - Resultado

Exibição do resultado do seqüenciamento, com o tempo previsto para a produção.

The screenshot shows a software window titled "Seqüenciamento de Produção" with the sub-section "Sequencia Final". The sequence of products is as follows:

PROD1	PROD2	PROD3	PROD4	PROD5	PROD6	PROD7	PROD8	PROD9	PROD10
3	3	3	3	3	2	2	2	2	2
PROD11	PROD12	PROD13	PROD14	PROD15	PROD16	PROD17	PROD18	PROD19	PROD20
101	201	101	201	101	3	3	3	3	3

Below this, there is a table titled "Produtos" with the following data:

IDProduto	NomeProduto
1	LN
2	L96
3	FPN
4	HPN
5	H5K
6	SKL
7	Canavieira
8	FSK
9	LTC

At the bottom of the window, there are buttons for "Voltar" (Back), "Finalizar" (Finish), "Modo formulário" (Form mode), and "NUM". A callout box points to the "Finalizar" button with the text "Botão para exportação dos resultados" (Button for exporting results).