**TESE DE DOUTORADO**

SEQUENCIAMENTO E BALANCEAMENTO DE TAREFAS EM LINHAS SERIAIS MULTIMODELOS COM TEMPOS ESTOCÁSTICOS

Igor Carlos Pulini

Orientador: Michel José Anzanello

Colatina, Janeiro 2016

Laboratório de Otimização de produtos e processos – LOPP

Programa de Pós-Graduação em Engenharia de Produção – PPGEP

Universidade Federal do Rio Grande do Sul - UFRGS

**SUMÁRIO**

[1 Introdução 1](#_Toc448474844)

[1.1 Considerações Iniciais 1](#_Toc448474845)

[1.2 Tema e Objetivos 4](#_Toc448474846)

[1.2.1 Tema 4](#_Toc448474847)

[1.2.2 Objetivos 4](#_Toc448474848)

[1.3 Justificativa do tema 5](#_Toc448474849)

[1.3.1 Científica 5](#_Toc448474850)

[1.3.2 Social 6](#_Toc448474851)

[1.3.3 Pessoal 6](#_Toc448474852)

[1.4 Metodologia do trabalho 6](#_Toc448474853)

[1.5 Estrutura do trabalho 7](#_Toc448474854)

[1.6 Delimitações do trabalho 7](#_Toc448474855)

[2 Revisão Bibliográfica 8](#_Toc448474856)

[2.1 Otimização 8](#_Toc448474857)

[2.2 Otimização com multiplos objetivos 8](#_Toc448474858)

[2.3 Algoritmos Evolucionários Multiobjetivos 15](#_Toc448474859)

[2.3.1 NSGA-II 16](#_Toc448474860)

[2.3.2 Tomada de Decisão 19](#_Toc448474861)

[2.3.3 Método de Borda 20](#_Toc448474862)

[2.4 Sequenciamento 20](#_Toc448474863)

[2.5 Regras ATC 29](#_Toc448474864)

[2.6 Balanceamento 30](#_Toc448474865)

[2.6.1 Arranjo físico 30](#_Toc448474866)

[2.6.2 Linhas de Montagem 32](#_Toc448474867)

[2.6.3 Balanceamento de Linha de Montagem 41](#_Toc448474868)

[2.6.4 Técnicas de Solução ALBP 46](#_Toc448474869)

[3 Bibliografia 47](#_Toc448474870)

# Introdução

## Considerações Iniciais

O setor de confecções brasileiro representa 5,7% do valor total da produção da indústria de transformação e conta com um universo de mais de 100 mil empresas que empregam diretamente mais de 1,6 milhão de brasileiros o equivalente a 16,9% do total de trabalhadores alocados na produção industrial, no qual 75% dos postos de trabalho são mulheres. Nos últimos anos o setor vem sofrendo com a falta de planejamento do governo para tornar o pais mais competitivo o que se reflete no aumento das importações de produtos do vestuário que cresceu cerca de 25 vezes nos últimos 10 anos, e como consequência resultou em várias demissões como pode ser analisado no Figura 1. Em 2015 a produção física de produtos do vestuário teve uma queda de 10% em relação a 2014 com uma perda de cerca de 100 mil postos de trabalho (ABIT, 2016).

Figura - Empregos diretos (em milhares) (ABIT, 2016)

O mapa da produção mundial do vestuário começou a mudar na década de 80, saindo dos EUA, Europa e Japão para países emergentes da Ásia e, mais recentemente, leste Europeu, norte da África e Caribe. Atualmente, a Ásia é responsável por 73% do volume total produzido no mundo, com destaque, por ordem, para: China, Índia, Paquistão, Coreia do Sul, Taiwan, Indonésia, Malásia, Tailândia e Bangladesh (ABIT, 2016). A concorrência de produtos industrializados em países asiáticos vem afetando diretamente a indústria brasileira, em especial a China que representa 49,7 % da produção mundial, onde o governo chinês subsidia parte da produção industrial de forma a garantir empregos em seu país, possibilitando um baixo custo de produção para sua indústria e, por consequência, destaque no mercado internacional.

Para o setor de confecções esta concorrência influencia diretamente na sua organização industrial, pois produtos com características de produção em massa deixam de ser o foco e as indústrias passam a investir em produtos cada vez mais personalizados, esta estratégia exige uma completa reestruturação dos processos produtivos, desde a pesquisa e desenvolvimento da coleção até a produção e expedição do produto acabado. Espera-se assim fornecer ao mercado diversidade de produtos com elevado grau de qualidade, baixo custo e ciclos de vida menores, ampliando a atratividade dos produtos nacionais frente à concorrência internacional. Estas características de produção conhecidas como customização em massa, segundo (Silveira G.;Borenstein D.;Fogliatto F. S., 2000), refere-se à capacidade de fornecer produtos e serviços customizados através de processos flexíveis a custos baixos, e vem surgindo como alternativa para diferenciar empresas neste mercado altamente competitivo e segmentado.

A maioria das linhas de produção no setor de confecções tem características semelhantes com um conjunto de estações de trabalho sequenciais ligadas de forma semiautomática ou totalmente manual. Desta forma a equalização da carga de trabalho de cada operador na linha de montagem é um trabalho desafiador para os gerentes industriais (Nassirnia & Md. Tap, 2010).

No mercado globalizado onde a demanda é impulsionada pelos clientes, o maior desafio dos fabricantes é determinar um melhor balanceamento e sequenciamento das tarefas com o objetivo de atender a grande variedade de produtos com baixo volume de produção em uma mesma linha de montagem (Uddin M. K.;Soto M. C.;Lastra J. L. M., 2010).

Diante destes desafios, impostos pelo sistema econômico vigente, intensificam-se nas empresas, bem como na academia, buscas por melhorias, que possibilitem à criação de diferenciais competitivos de mercado à indústria do vestuário. As atuais características de mercado exigem que as empresas trabalhem com lotes de produção cada vez menores, dificultando a formulação de layouts especializados de melhor desempenho e exigindo a formulação layouts de produção flexíveis para uma rápida e constante adaptação às exigências do mercado da moda, forçando as industrias a trabalharem com o conceito de just-in-time no qual as ordens de produção são criadas por demanda de acordo com a venda do produto e a posterior compra de matéria prima. Neste tipo de produção com uso intensivo de tarefas manuais, e consequente variação estocástica do tempo, existe uma grande dificuldade de fazer uso eficiente dos recursos de produção, assegurando uma rápida execução das tarefas e o cumprimento dos prazos de entrega.

A contribuição desta tese está na definição de um modelo real de balanceamento e sequenciamento das tarefas aplicado especificamente ao setor de confecções em linhas de produções seriais, múltiplos modelos com tempos estocásticos. O trabalho inova ao propor um algoritmo genético multiobjectivo que trabalha com tempos estocásticos para resolução do problema; ao propor uma modificação na regra de sequenciamento ATC para priorizar o volume de produção.

## Tema e Objetivos

### Tema

O tema desta tese é o sequenciamento e balanceamento de tarefas em linhas seriais multi-modelos com tempos estocásticos.

### Objetivos

Melhorar o balanceamento e o sequenciamento das tarefas em linhas seriais multi-modelos com tempos estocásticos.

São objetivos específicos:

1. Identificar e selecionar as melhores práticas de sequenciamento e balanceamento utilizadas no processo industrial e tecnológico na indústria de vestuário;
   1. Aplicação de entrevistas e questionários junto a empresários, gerentes de produção, engenheiros de produção e operadores das indústrias de vestuário para identificação das técnicas de sequenciamento e balanceamento atualmente utilizadas, mesmo que de forma empírica;
   2. Levantamento e análise de dados produtivos das indústrias relacionados ao sequenciamento e balanceamento das ordens de produção;
   3. Cruzamento dos dados de produção com as técnicas utilizadas no sequenciamento e balanceamento para determinar aquelas que conduzem ao melhor desempenho produtivo.
2. Propor um modelo multiobjectivo para sequenciamento e balanceamento apoiado nas melhores práticas da indústria de vestuário;
3. Revisão literária das técnicas de sequenciamento e balanceamento de produção;
4. Revisão literária sobre métodos heurísticos e exatos de otimização de sequenciamento e balanceamento de produção;
5. Executar a modelagem computacional de forma a adequar às técnicas de sequenciamento e balanceamento descritas na literatura, métodos heurísticos e exatos de otimização e as melhores práticas de sequenciamento adotadas na produção das indústrias em um software;
6. Comparar e analisar o desempenho produtivo obtido com a utilização do Modelo;
7. Comparar dados produtivos obtidos antes e depois da implantação do software

## Justificativa do tema

### Científica

Mesmo considerando a grande variedade de artigos científicos envolvendo o tema de balanceamento de linhas de produção, ainda há uma grande lacuna entre a academia e as aplicações práticas até agora. Uma das razoes para este défice está no fato de que os trabalhos de pesquisa consideram apenas um único ou apenas algumas extensões de balanceamento, na prática existe a necessidade de se trabalhar com inúmeras extensões combinadas (Boysen, et al., 2008).

Apesar dos grandes avanços na automação de processos, ainda há muitos sistemas de montagem que dependem principalmente ou completamente de trabalho manual (Zhuming & Zhang, 2001) (Tempelmeier, 2003).

A importância do setor de vestuário no estado do ES é caracterizada pelas 1.237 empresas que geram em torno de 17.042 empregos diretos (MTE/RAIS, 2009), com uma receita anual em torno de R$ 500 milhões (IBGE/PIA, 2014). Segundo (IDEIES, 2011), a abertura da economia internacional intensificou a concorrência mundial, apresentando novas formas de organização sócio espaciais, como o deslocamento de atividades produtivas mais intensivas de trabalho para os países de menor custo de mão de obra. A Ásia se destaca como principal concorrente para o setor têxtil e de vestuário.

Neste cenário competitivo, as empresas do setor têxtil e de vestuário capixaba encontram na inovação tecnológica seu principal desafio para manterem-se competitivas, tendo que priorizar investimentos em setores críticos como: Desenvolvimento de produtos e design; Aperfeiçoamento da manufatura; Flexibilidade e desenvolvimento da marca; Comercialização e distribuição.

Esta concorrência força as empresas de confecções a se tornarem mais competitivas, buscando meios que proporcionem um maior desempenho de suas linhas de produção.

### Social

Grande número de empregos que estão sendo perdidos nas indústrias de confecções do estado do espírito santo devido à concorrência com as empresas de vestuário asiáticas.

### Pessoal

Experiência com indústria de confecções desde 1996; Sócio proprietário da Matriz Sistemas (até 2009), empresa desenvolvedora de software ERP para o setor de confecções, no qual fui responsável pelo desenvolvimento de módulos voltados para o planejamento e controle de produção, explosão de materiais, expedição, compras e formação de preço de produtos; Mestrado em inteligência computacional voltado para o otimização de produção em ambientes dinâmicos com algoritmos genéticos multiobjetivo e análise multicritério; Desenvolvimento e implantação de software de balanceamento de layout produtivo para Mona Indústria do vestuário (Vila Velha); Desenvolvimento e implantação de software de pagamento de produtividade e controle de produção para P W Brasil Export SA (Colatina); Desenvolvimento de aplicativo utilizando lógica Fuzzy para classificação de produtos de vestuário para a P W Brasil Export SA; Abertura junto aos empresários do setor de vestuário; Professor efetivo IFES-Campus Colatina.

## Metodologia do trabalho

A estratégia metodológica nesta investigação é a pesquisa de **pesquisa-Ação**. A natureza desta pesquisa é **aplicada**. A abordagem é **mista** com enfoque quantitativo. O método de pesquisa é **dedutivo**, utilizando-se da **hipótese**: Se validarmos o modelo de sequenciamento e balanceamento de tarefas com tempos estocásticos junto às empresas de vestuário as empresas terão um melhor desempenho produtivo. Segundo (POPPER, 1953), a hipótese deve ser testada para ver se ela é verdadeira.

## Estrutura do trabalho

Além deste capítulo introdutório a tese está estruturada em mais cinco capítulos que abordam assuntos relacionados ao objetivo da pesquisa.

Capítulo dois é uma revisão literária dos assuntos relacionados à pesquisa: Sequenciamento e balanceamento de produção; Arranjo físico; Algoritmos genéticos multiobjectivos e Análise multicritério.

Capítulo três aborda os procedimentos metodológicos utilizados no desenvolvimento da tese, técnicas de coleta e tratamento de dados.

Capítulo quatro apresenta a formulação do problema, modelagem matemática.

Capítulo cinco apresenta solução proposta.

Capítulo seis analisa os resultados obtidos com a aplicação da proposta, comparando o resultado com as técnicas atualmente utilizadas, simulação e com a literatura.

## Delimitações do trabalho

O modelo proposto foi projetado com características específicas da industrias do vestuário, para layouts de produção que utilizam linhas de produção em série, multi-modelo, com emprego de tarefas manuais (tempos estocásticos) com equipamentos fixos.

# revisão bibliográfica

## Otimização

Otimizar significa tornar ótimo ou ideal, é extrair o melhor rendimento possível empregando técnicas para selecionar as melhores alternativas para se chegar aos objetivos, refere-se a encontrar um ou mais soluções viáveis que corresponda a valores extremos de um ou mais objetivos. Quando o problema possui apenas uma única função objetivo o processo de otimização está centrado em encontrar a melhor configuração das variáveis que respeite as restrições de forma a obter o melhor valor para a função objetivo. No entanto, na prática, problemas de otimização no mundo real, naturalmente envolvem múltiplos objetivos, ou seja, possuem mais de uma função objetivo, na maioria das vezes conflitantes, no qual a melhora de um degrada os demais (Srinivas & Deb, 1994); (Zitzler & Thiele, 1999).

O ambiente do sequenciamento é geralmente complexo e o programador de produção muitas vezes tem de lidar com uma combinação ponderada de diversos objetivos e os pesos podem ser o tempo ou dependente situação. Normalmente, um programador pode não saber os pesos exatos e pode querer realizar uma análise paramétrica e analisar os conflitos entre os objetivos. Quando um programador cria uma programação que é melhor com relação a um objetivo, ele pode querer saber como os outros objetivos são afetados, quando existem múltiplos objetivos o conceito de Pareto desempenha está função (Pinedo , 2012).

## Otimização com multiplos objetivos

Em problemas de otimização com múltiplos objetivos não existe uma solução que obtenha o melhor resultado para todos os objetivos, não permitindo a determinação uma única solução ótima como na otimização com um objetivo. Neste processo se obtém um conjunto de soluções no qual nenhuma solução é melhor que a outra para todos os objetivos, necessitando de informações adicionais para o processo de tomada de decisão. Para melhor entendimento do processo de otimização com múltiplos objetivos alguns conceitos e definições importantes e amplamente utilizados na literatura são descritos a seguir (Zitzler & Thiele, 1999); (Coello, et al., 2007); (Deb, 2009).

Maximiza / Minimizar

*(a) m= 1, 2, 3, ..., M*

*Sujeito a:*

*(b) j= 1, 2, 3, ..., J*

*(c) k= 1, 2, 3, ..., K*

*(d) i= 1, 2, 3, ..., N*

1. Existem *M* funções objetivos e cada função pode ser de maximizar ou minimizar. Cada função objetivo serve de guia, avaliando os resultados obtidos durante o processo de otimização, no qual as variáveis de decisão: = (), tem seus valores alterados.
2. e (c) São restrições de comportamento representadas por funções de igualdade ou desigualdade que delimitam situações inadequadas a solução do problema.
3. São as restrições laterais que delimitam diretamente as variáveis de decisão.

Qualquer solução que satisfaça a todas as restrições é conhecida como **solução viável** e se encontra dentro de uma região chamada **região viável**. O melhor valor, ou **valor ótimo** da função objetivo se encontra no **ponto ótimo**, que é um vetor de pontos formado pelas variáveis de decisão que juntos formam o extremo da função objetivo e devem respeitar as restrições. Uma diferença entre problemas de otimização com um objetivo e com múltiplos objetivos é que a função objetivo constitui um conjunto multidimensional de soluções, no qual cada conjunto de variáveis de decisão que forma um ponto viável é mapeado em um ponto no espaço o objetivo conforme pode ser visualizado na Figura 2. Desta maneira na possível ter diversos pontos mapeados no espaço de decisão apontando para diferentes pontos no espaço objetivo.



Figura - Espaço de decisão x espaço objetivo adaptado (Deb, 2009)

Erroneamente pode-se pensar que na otimização com múltiplos objetivos a tarefa seja buscar uma ótima solução para cada função objetivo, porém, como mais de uma função objetivo deve ser avaliada simultaneamente a otimização não pode ser aplicada a somente um objetivo enquanto os outros objetivos também são importantes. Segundo (Deb, 2009) existe 2 abordagens para solucionar problemas de otimização com múltiplos objetivos exibidas na Figura 3:

1. Buscar múltiplas soluções com diferentes valores para os objetivos e com este conjunto de ótimas soluções utilizar informações adicionais para escolher uma dentre as soluções.
2. Utilizar um peso para cada objetivo durante o processo de busca transformando o processo de busca com múltiplos objetivos em um processo de busca com um simples objetivo.



Figura - Solução com múltiplos objetivos (Deb, 2009)

A abordagem “a” é mais metódica, prática e menos subjetiva, porém se há um conhecimento confiável o bastante para estabelecer os pesos a segunda abordagem pode ser mais adequada. Utilizar pesos para combinar vários objetivos em uma única função objetivo durante o processo de otimização, tem a vantagem de produzir uma única solução, porém definir os pesos para cada objetivo requer um conhecimento profundo, muitas vezes, não disponível (Zitzler & Thiele, 1999). Outro problema é que os métodos de otimização com múltiplos são geralmente muito sensíveis a pequenas alterações nos coeficientes da função de penalidade ou nos pesos da função objetivo (Horn, et al., 1994).

É ilusório conceder uma solução ótima para um problema que trate simultaneamente objetivos conflitantes como ocorre na otimização com um único objetivo, pois na otimização com múltiplos objetivos não existe uma única solução ótima e sim um conjunto de soluções, onde todos os objetivos são simultaneamente considerados. Dessa forma, é necessário estabelecer a ordenação das soluções obtidas considerando cada um dos objetivos utilizado o conceito de dominância (Coello, et al., 2007).

Uma solução *x(1)* domina outra solução *x(2)*, se ambas as condições 1 e 2 são satisfeitas (Deb, 2009) (Lu, et al., 2013):

1. A solução *x(1*) é melhor ou igual a *x(2*) em todos os objetivos;
2. A solução *x(1*) é estritamente melhor que *x(2*) em pelo menos um objetivo.

Caso qualquer uma das condições acima seja violada, a solução *x(1*) não domina a solução *x(*2). Se *x(1)* domina a solução *x(2)* (ou matematicamente *x(1)* *x(2)*), este fato também pode ser definido como:

* *x(2)* é dominado por *x(1)*;
* *x(1)* não é dominado por *x(2)*;
* *x(1)* é melhor ou igual a *x(2)*.

Considerando objetivos de minimização e aplicando os conceitos de dominância apresentados anteriormente na Figura 4, é possível constatar que a solução “*g*” é dominada (melhor) pela a solução “*a*” em ambos objetivos, que a soluções “*e*” e “*f*” domina a solução “*a*” em ambos objetivos e que as soluções “*b*”, “*c*” e “*d”* não dominam nem são dominados pela solução “*a*”, pois a solução “a”, apresenta melhor resultado em pelo menos um dos objetivos.



Figura - Relações de dominância adaptado (Coello, et al., 2007)

Considerando 2 objetivos, no qual o objetivo 1 é de minimizar e o objetivo 2 é de maximizar conforme pode ser analisado na Figura 5. Ao analisar a solução 1 e 2 aplicando os conceitos de dominância é possível verificar que a solução 1 é melhor que a solução 2 para o objetivo 1 e para o objetivo 2. Dessa forma é possível afirmar que a solução 1 domina a solução 2. Comparando as soluções 1 e 5 é possível verificar que a solução 5 é melhor que a solução 1 para o objetivo 2 e igual para o objetivo 1. Sendo possível afirmar de acordo com os conceitos de dominância apresentados que a solução 5 domina a solução 1 (Deb, 2009).



Figura - Dominância (Deb, 2009)

O conjunto de soluções não dominadas do espaço objetivo mostrado na Figura 6, dá-se o nome de frente de Pareto. Neste conjunto de soluções não dominadas há sempre uma melhora em pelo menos um dos objetivos e um sacrifício em pelo menos um dos outros, ou seja, para cada solução deste conjunto não existe outra solução que seja melhor em todos os objetivos.



Figura - Frente de Pareto (Deb, 2009)

A existência de um grande número de alternativas pertencentes à fronteira Pareto torna complexa a escolha. Entretanto, apesar da importância e da dificuldade dessa decisão, os esforços nessa área têm se concentrado principalmente na busca pela fronteira ótima de Pareto e menos tem sido feito no sentido de se acoplar informações quanto à preferência do decisor aos algoritmos de busca. Este acoplamento pode ser implementado de 3 formas:

1. Decisão a priori - O decisor é consultado uma única vez, antes do início do processo de otimização e a informação obtida quanto aos seus interesses é usada para guiar a busca pela solução favorita pertencente à fronteira Pareto;
2. Interativa ou progressiva - O decisor é consultado várias vezes durante o processo de otimização. Assim, ele pode definir suas preferências tendo em mãos a informação disponível na iteração corrente quanto às possíveis soluções. Sua preferência guia a busca nas próximas iterações, até que nova consulta seja realizada. O processo de decisão progressiva só é interrompido quando o decisor estiver totalmente satisfeito com sua escolha;
3. Decisão a posteriori - Inicia-se com a execução de uma busca multiobjectivo. O decisor é consultado apenas depois que uma aproximação discreta satisfatória da fronteira de Pareto for encontrada.

A abordagem a posteriori possui a vantagens do decisor pode analisar todas as soluções ótimas disponíveis antes de fazer sua escolha; se seus interesses se alterarem, não é necessário executar o algoritmo de otimização novamente. Além disso, geralmente é mais fácil incluir novos critérios na análise realizada a posteriori do que a priori ou progressivamente. Por outro lado, essa abordagem geralmente exige maior tempo de processamento do que as outras, pois a busca por uma aproximação da fronteira de Pareto tende a exigir maior esforço computacional do que a busca por um único ponto ótimo (Deb, 2009).

A dificuldade associada a resolução de problemas de otimização de forma exata, naturalmente combinatória, exigiu o desenvolvimento de técnicas **heurísticas** que procuram obter boas soluções a um custo computacional razoável, sem garantir que as soluções sejam as melhores. As heurísticas surgiram para resolver certos tipos específicos de problemas, não sendo flexíveis o bastante para aplicação em um conjunto de diferentes problemas. As meta-heurísticas surgiram com uma proposta de heurísticas que podem ser aplicadas a solução de diferentes problemas de otimização. Dentre as meta-heurísticas os algoritmos evolucionários são naturalmente adaptados para trabalhar com múltiplos objetivos por encontrar múltiplas soluções em uma única evolução através da evolução de uma população de soluções, por não precisar de informações adicionais e por possibilitar escapar dos ótimos locais durante o processo de otimização (Coello, et al., 2007).

## Algoritmos Evolucionários Multiobjetivos

Algoritmos evolucionários são técnicas meta-heurísticas de busca e otimização, largamente aplicáveis, inspiradas nos mecanismos da evolução natural. A ideia é a evolução de uma população de estruturas computacionais, de modo tal que melhore a adequação média dos indivíduos que formam esta população em relação ao ambiente em que ela está submetida. Os modelos de algoritmos evolucionários com múltiplos objetivos são usualmente classificados em 2 grupos (Deb, 2009):

* Não elitistas: não utiliza nenhuma forma de elitismo nas suas interações;
* Elitistas: modelos que empregam alguma forma de elitismo: SPEA; PESA; NSGA II.

O algoritmo básico da computação evolucionária pode ser descrito da seguinte forma:

1 Algoritmo Evolucionário Conceitual

2 | Inicialização da população

3 | Avaliação dos indivíduos na população

4 | Repetição (evolução)

5 | | Seleção dos indivíduos da população

6 | | Alteração destes indivíduos para criar uma nova população

7 | | Avaliação da nova população

8 | Até que um critério de parada seja alcançado

9 Fim

Dentre as meta-heurísticas destaca-se o uso de algoritmos evolucionários para otimização com múltiplos objetivos, que devido suas características populacionais, são mais adequados do que outras buscas cegas (Fonseca & Fleming, 1995). Existem uma vasta variedade de algoritmos evolucionários que trabalham com múltiplos objetivos na literatura, por exemplo: Nondominated Sorting Genetic Algorithm (NSGA), Niched Pareto Genetic Algorithm (NPGA), Strength Pareto Evolutionary Algorithm (SPEA), Strength Pareto Evolutionary Algorithm 2 (SPEA2), Pareto Archived Evolution Strategy (PAES), Nondominated Sorting Genetic Algorithm II (NSGA-II) e MicroGenetic Algorithm for Multiobjective Optimization (µGA). Para obter mais informações sobre estes algoritmos consulte informações específicas em (Coello, et al., 2007) (Deb, 2009).

Alguns trabalhos fizeram uma comparação entre os algoritmos evolucionários multiobjectivos e o que obteve o melhor resultado nos testes foi algoritmo de classificação genética não dominada (NSGA) (Zitzler & Thiele, 1999); (Srinivas & Deb, 1994); (Dias & Vasconcelos, 2002); (Gen & Lin, 2014).

### NSGA-II

O algoritmo NSGA-II é uma heurística construtiva de busca elitista, que faz a preservação de soluções dominantes através de várias gerações. O processo se inicia a partir de uma população (*P*) com *i* indivíduos (pais) obtidos aleatoriamente, ou também, através de uma heurística construtiva. As seguintes gerações seguintes são obtidas a partir de operadores de modificação genética de mutação, seleção e recombinação. O algoritmo tem como finalidade encontrar um conjunto de pontos próximos à frente de Pareto, e que ela se mantenha a mais diversificada possível. A distância de aglomeração (*crowding distance*) permite qualificar o espaço ao redor de uma solução, pois é fundamental que os indivíduos da população estejam relativamente espaçados dentro da fronteira de Pareto, permitindo uma maior variedade do conjunto de soluções e consequentemente conduzindo com mais rapidez ao encontro de indivíduos com maior qualidade de solução (Deb, et al., 2002). O cálculo da distância aglomeração da solução *i* é dada por:

 (23)

Na expressão (23), *f1 max, f1 min, f2 max e f2 min* representam os valores máximos e mínimos sobre o espaço da solução para as funções objetivo *f1* e *f2*. Esta Expressão representa metade do perímetro do retângulo da Figura 7, sendo assim, pode-se afirmar que, para um conjunto de soluções, a solução que introduz o maior nível de diversidade é aquela com maior distância de aglomeração.



Figura 7 - Representação da distância de aglomeração da solução *i* (Deb, et al., 2002)

O esquema do NSGA-II pode ser descrito conforme a Figura 8, onde uma população inicial é criada de forma aleatória. A partir desta população cria-se, através de operadores genéticos, uma segunda população de filhos, que é combinada com a população de pais inicial e em seguida classificada de acordo com o grau de dominância da frente de Pareto, ou seja, a primeira frente teria o grau de dominância 0, a segunda frente grau 1 e assim por diante. O próximo passo é limitar o tamanho da população, criando assim um corte em determinado ponto em uma das frentes de Pareto; para que esse procedimento ocorra, a ordenação dos indivíduos dentro das frentes deve ser feita de acordo com a maior distância de aglomeração, forçando a sobrevivência de indivíduos mais bem distribuídos pela frente de Pareto. O procedimento se repete até que os critérios estabelecidos de parada sejam satisfeitos. Desse modo os descendentes finais são resultado de uma busca elitista pela frente de Pareto com menor grau de dominância e da ordenação dos indivíduos da última frente pela maior distância de aglomeração entre os indivíduos. A seguir é descrito o pseudocódigo do NSGA-II (Deb, et al., 2002).



Figura 8 - Método NSGA-II (Coello, et al., 2007)

1. Gerar uma população *P* de tamanho *N*;
2. Identificar as frentes de dominância e avaliar as distâncias de aglomeração (*crowding distance*) em cada frente;
3. Usando a seleção, recombinação e mutação, gera-se uma população de descendentes do mesmo tamanho de *P*;
4. Reunir pais e filhos em um conjunto de tamanho *2N* e classificá-los em frentes de dominância;
5. Determinar os descendentes finais selecionando as frentes com melhor grau de dominância. Se o limite de tamanho *N* é superado, eliminar as soluções com menor distância de aglomeração na última frente selecionada;
6. Se o critério de convergência é atingido, fim do processo. Caso contrário, retornar ao passo 3.

Um conjunto de soluções pertencentes à primeira frente de Pareto é obtido como resultado do algoritmo NSGA-II. A partir deste conjunto há a necessidade de se optar por uma dentre as possíveis soluções que será aplicada na distribuição das tarefas como resultado final do algoritmo, normalmente esta decisão fica a cargo do especialista. A análise multicritério pode ser utilizada para tratar convicções de diferentes partes interessadas e direcionar a tomada de decisões na presença de incertezas e conflito de interesses. Dessa maneira é possível aplicar as técnicas embasadas em análise multicritério para escolher a melhor entre as soluções obtidas pelo algoritmo NSGA-II avaliando a opinião de diferentes partes interessadas no processo de sequenciamento produtivo.

### Tomada de Decisão

Uma decisão precisa ser tomada sempre que estamos diante de um problema que possui mais de uma alternativa para sua solução. Mesmo quando, para solucionar um problema possuímos uma única ação a tomar, temos as alternativas de tomar ou não a ação (Gomes, et al., 2009).

Os métodos de análise multicritério (AMD - Apoio Multicritério a Decisão) não procuram, apenas, encontrar uma solução aleatória, mas buscar no universo de possibilidades um método para a tomada de decisão. Uma decisão deve atender a todas as necessidades, considerando os limites e restrições impostas pelo contexto do problema e auxiliar na articulação de preferências em presença de ambiguidade, bifurcações e incertezas (Gomes, et al., 2003). Os principais métodos de análise multicritério são: Método de Borda; Método Condorcert; Método de Análise Hierarquica (*Analytic Hierarchic Process* - AHP); Os Métodos da Família Electre; O Método Promethe; O Método Macbeth.

O método de borda usa uma estrutura de candidatos e eleitores. Esta estrutura permite que o método se adeque perfeitamente a função de decisor, considerando cada candidato como uma possível solução retornada pelo NSGA-II, e cada eleitor como um objetivo da otimização. Desse modo é necessário apenas classificar a posição de cada solução obtida para cada objetivo e com um simples cálculo determinar o indivíduo selecionado.

### Método de Borda

Método de decisão proposto por Jean Charles de Borda à Academia de Ciências Francesa em 1781, para ser aplicado em comitês compostos por mais de um indivíduo. O método requer que as alternativas sejam ordenadas de acordo com as preferências do decisor onde a alternativa preferida recebe 1 ponto, a segunda, 2 pontos e assim sucessivamente conforme Figura 9. Os pontos de cada alternativa são então somados, e a melhor alternativa será aquela para qual a soma dos pontos seja menor (Dias, 2000).



Figura 9- Método de borda.

## Sequenciamento

O objetivo do sequenciamento é fazer uso eficiente dos recursos de produção e assegurar a rápida execução das tarefas de forma a executar a entrega nos prazos acordados (Pinedo , 2012). É possível categorizar os principais modelos de programação, especificando as configurações dos recursos e da natureza das tarefas. Desta maneira um problema de sequenciamento pode conter uma ou várias máquinas máquina. Se o problema conter somente uma máquina, as tarefas são relacionadas a uma única fase, caso o problema contenha vários modelos de máquinas geralmente envolvem tarefas com múltiplas fases. Em ambos os casos, as máquinas podem estar distribuídas em quantidades unitárias ou em paralelo. Além disso, se o conjunto de tarefas disponíveis para a programação não se altera ao longo do tempo, o sistema é chamado de estático, em contraste com os casos em que aparecem novos trabalhos ao longo do tempo, em que o sistema é chamado de dinâmico. Tradicionalmente, os modelos estáticos são mais facilmente resolvidos que os modelos dinâmicos e têm sido estudados mais extensivamente. Embora os modelos dinâmicos estarem mais próximo da aplicação prática, modelos estáticos, frequentemente capturam a essência dos sistemas dinâmicos, e a análise de problemas estáticos frequentemente revela informações valiosas e bons princípios heurísticos que são úteis em situações dinâmicas. Finalmente, quando se presume condições de certeza o modelo é chamado de determinista. Por outro lado, quando reconhecemos a incerteza utilizando distribuições de probabilidade explícitas, o modelo é chamado estocástico (Baker & Trietsch, 2009).

Todo o sequenciamento de tarefas considera o número de postos de trabalho e o número de máquinas finitos. O número de postos de trabalho é designado por *n* e o número de máquinas por *m*. Geralmente, o índice *j* refere-se a um posto de trabalho, enquanto *i* refere-se a uma máquina. Se um trabalho exige uma série de etapas de processamento ou operações, o par (*i*, *j*) refere-se à etapa de processamento ou operação da tarefa *j* na máquina *i*. As seguintes partes de dados estão associados a tarefa *j* (Pinedo , 2012).

Tempo de processamento (). Representa o tempo de processamento da tarefa *j* na máquina *i*. O *i* pode ser omitido se o tempo de processamento da tarefa *j* não depende do equipamento. Se o trabalho for processado em apenas uma máquina j pode ser omitido.

Data de liberação (). Representa a data em que a tarefa j estará disponível para o sistema.

**Data de entrega ().** Representa a data de entrega do trabalho *j* (isto é, a data em que o trabalho está prometido ao cliente). O término do trabalho após a sua data de entrega é permitido, mas com uma grande penalidade.

**Peso ().** O peso da tarefa *j* representa sua prioridade ou importância da tarefa *j* em relação as outras tarefas.

Um problema de sequenciamento é descrito por um trio *α|β|γ* (Graham, et al., 1979)*;* (Pinedo , 2012)*;* (Brucker, 2007). No qual *α* descreve o ambiente das máquinas e contém apenas uma entrada. *β* fornece detalhes das características de processamento e restrições e pode conter nenhuma, uma ou múltiplas entradas. Já *γ* descreve o objetivo que muitas vezes contém uma única entrada.

Os ambientes de máquina possíveis especificados no por *α* são:

* **Uma máquina (1):** O caso de uma única máquina é o mais simples de todos os ambientes de máquina possíveis.
* **Máquinas idênticas em paralelo ().** São *m* máquinas idênticas em paralelo. A tarefa *j* exige uma única operação que pode ser processada em qualquer uma das *m* máquinas. Se a tarefa *j* não pode ser processada em qualquer máquina, mas em um subconjunto específico , então a entrada aparece no campo β.
* **Máquinas em paralelo com diferentes velocidades ().** Existem *m* máquinas em paralelo com velocidades diferentes. As Máquinas realizam o trabalho da mesma forma variando somente a velocidade, que é denotada porpara a máquina *i.* O tempo da tarefa *j* na máquina *i* é igual a **,** assumindo que ela possa ser processada pela máquina *i*.
* **Máquinas não-relacionadas ().** Esse ambiente é uma generalização do anterior. Existem m máquinas diferentes em paralelo. A máquina i pode processar a tarefa j com uma velocidade . O tempo que a tarefa j gasta na máquina i é igual a . Se as velocidades das máquinas são independentes da tarefa, isto é, , então o ambiente se torna idêntico ao anterior.
* **Flow shop ().** Existem *m* máquinas em série. Cada tarefa tem que ser processada por cada uma das *m* máquinas. Todas as tarefas têm o mesmo roteiro, isto é, devem ser processados primeiro pela máquina 1, depois pela máquina 2, e assim por diante. Após o término em uma máquina uma tarefa entra na fila para a próxima máquina. Normalmente, todas as filas operam com *first in first out* (FIFO), isto é, uma tarefa não pode passar outra na fila. Se o FIFO é adotado, o flow shop é referido como flow shop com permutação, e o campo β deve incluir a entrada *prmu*, que é apresentada abaixo.
* **Flow shop flexível ().** É uma generalização dos ambientes flow shop e máquinas paralelas. Em vez de *m* máquinas em série, existem *s* estágios em série com um número de máquinas em paralelo em cada estágio. Cada tarefa deve ser processada primeiro no estágio 1, depois no estágio 2, e assim por diante. Um estágio funciona como um banco de máquinas paralelas: em cada estágio, uma tarefa *j* requer apenas uma máquina e, normalmente, qualquer máquina pode processar qualquer tarefa. As filas entre os vários estágios também operam, em geral, com a estratégia FIFO.
* **Open shop ().** Existem *m* máquinas. Cada tarefa tem que ser processada em cada uma das *m* máquinas. No entanto, alguns dos tempos de processamento podem ser zero. Não há restrições quanto ao roteiro de cada tarefa pelo ambiente de máquinas. É possível determinar um roteiro para cada tarefa, e tarefas diferentes podem ter roteiros diferentes.
* **Job shop ().** Num job shop com *m* máquinas, cada tarefa tem seu próprio roteiro. Uma distinção é feita entre job shops onde cada tarefa visita qualquer máquina no máximo uma vez e job shops onde uma tarefa pode visitar uma máquina mais de uma vez. No último, *β* contém a entrada *rcerc* para recirculação, que é apresentada abaixo.

As restrições especificadas no campo β podem incluir múltiplas entradas. São algumas das entradas possíveis:

* **Data de disponibilidade ().** Se este símbolo estiver presente em β, a tarefa *j* não pode iniciar seu processamento antes da data de disponibilidade Se não aparece em β, o processamento da tarefa *j* pode começar a qualquer momento. Em contraste com datas de disponibilidade, datas de entrega não são especificadas neste campo. O tipo de função objetivo dá indicação suficiente se existem datas de entrega ou não.
* **Preempções (*prmp*).** Preempções sugerem que não é necessário manter uma tarefa em uma máquina até o término. É possível interromper o processamento de uma tarefa a qualquer hora, e colocar uma tarefa diferente na máquina. A quantidade de tempo de processamento que uma tarefa interrompida já recebeu não é perdida. Quando uma tarefa interrompida volta à máquina (ou a outra máquina no caso de máquinas paralelas), ela só precisa da máquina para o tempo de processamento restante. Quando preempções são permitidas, *prmp* é incluído a β.
* **Tempos de preparação dependentes da sequência ().** Representa o tempo de preparação dependente da sequência entre as tarefas *j* e *k*, denota o tempo de preparação para a tarefa *k* se a tarefa *k* é a primeira na sequência e denota o tempo de limpeza após a tarefa *j* se for a última na sequência. Se o tempo de preparação entre as tarefas *j* e *k* depende da máquina, o sub-escrito *i* é incluído, isto é, . Se não aparece em β, todos os tempos de preparação são assumidos zero ou independentes da sequência, podendo ser simplesmente adicionados aos tempos de processamento.
* **Famílias de tarefas (*fmls*).** As *n* tarefas pertencem, neste caso, *F* tarefas de diferentes famílias. Tarefas da mesma família podem ter tempos de processamento diferentes, mas eles podem ser processados em uma máquina um após o outro sem necessidade de qualquer tempo de *setup*. No entanto, se a máquina muda de uma família para outra, ou seja, da família *g* para a família *h*, é necessário um tempo de setup. Se este tempo de preparação depende de ambas as famílias *g* e *h e* é dependente da sequência, em seguida, é denotada por *.* Se este tempo de preparação depende apenas da família que iniciou, isto é, família *h*, significa que Se não depender de outra família é simplesmente dado por *s*.
* **Processamento em lote (*batch (b)*).** Máquina pode ser capaz de processar um número de tarefas *b* simultaneamente; isto é, ele pode processar um lote de até *b* tarefas ao mesmo tempo. Se *b* = 1, então o problema reduz-se a um ambiente de programação convencional. Outro caso especial que é de interesse é *b* = ∞, ou seja, não há limite para o número de postos de trabalho a máquina pode manipular a qualquer momento.
* **Quebras (*brkdwn*).** Quebras de máquinas sugerem que máquinas não estão sempre disponíveis. Os períodos em que uma máquina não está disponível são assumidos fixos (por exemplo, devido a turnos ou manutenções). Se há máquinas idênticas em paralelo, o número de máquinas disponíveis é uma função do tempo *m(t)*.
* **Restrições de elegibilidade ().** Uma restrição que pode aparecer em β quando o ambiente de máquinas contiver *m* máquinas em paralelo (). Quando está presente, nem todas as *m* máquinas são capazes de processar a tarefa *j*. O conjunto denota o conjunto de máquinas capazes de processar a tarefa *j*. Se β não contém , a tarefa *j* pode ser processado por qualquer uma das *m* máquinas.
* **Permutação (*prmu*).** Uma restrição que pode ser aplicada no ambiente de flow shop, é que as filas na entrada de ada máquina operam de acordo com a política FIFO. Isso quer dizer que a ordem (ou permutação) em que as tarefas passam pela primeira máquina é mantida através do sistema.
* **Bloqueio (*block*).** Bloqueio é um fenômeno que ocorre em flow shop. Se um flow shop tem um buffer limitado entre duas máquinas sucessivas, pode ser que, quando um buffer ficar cheio, a máquina anterior não possa liberar a tarefa terminada. Esse fenômeno se chama bloqueio: a tarefa terminada tem que permanecer na máquina anterior evitando, ou bloqueando, a máquina de processar outras tarefas.
* **Sem espera (*nwt*).** A restrição sem-espera é outro fenômeno que pode o correr em flow shop. Tarefas não podem esperar entre duas máquinas sucessivas. Isso quer dizer que o tempo de início de uma tarefa pode ser atrasado para garantir que a tarefa possa atravessar o flow shop sem ter que esperar por nenhuma máquina. Um exemplo de uma operação é uma linha de fabricação, onde uma determinada peça de metal não pode esperar pela próxima etapa porque ela esfriaria.
* **Recirculação (*recrc*).** Recirculação pode ocorrer em job shop, quando uma tarefa pode visitar uma máquina mais de uma vez.

Outras entradas de β são autoexplicativas. Por exemplo, = indica que todos os tempos de processamento são iguais, e = indica que todas as datas de entrega são iguais. Como dito antes, datas de entrega, ao contrário das datas de disponibilidade, não são normalmente explicitadas em β; o tipo da função objetivo nos dá indicação suficiente se as tarefas têm data de entrega ou não.

O objetivo a ser minimizado é sempre uma função dos tempos de término das tarefas, que, é claro, dependem do sequenciamento, ou também pode ser uma função das datas de entrega. O tempo de término da operação da tarefa j na máquina *i* é dado por . O tempo em que a tarefa *j* sai do sistema (o tempo de término na última máquina onde ele requer processamento) é denotado por . O atraso de uma tarefa j é definido como:

Que é um valor positivo quando a tarefa *j* é terminada atrasada e negativo quando é completada adiantada. O atraso da tarefa *j* é definida como:

A diferença entre o atraso e o adiantado reside no facto de que o atraso nunca será negativo. A pena unitária da tarefa *j* é definido como:

A seguir, temos alguns exemplos de possíveis funções objetivo a serem minimizadas.

**Makespan ().** O makespan, definido como é equivalente ao término da última tarefa a sair do sistema. Um makespan mínimo geralmente indica uma alta utilização das máquinas.

**Lateness ().** É definido como . Mede a pior violação das datas de entrega.

**Tempo de término total ponderado ().** A soma dos tempos ponderados de término das *n* tarefas, utilizando a prioridade como fator de ponderação, dá uma indicação do custo total do sequenciamento. A soma dos tempos de término é frequentemente referida na literatura como tempo de fluxo (*flow time*). O tempo de término total ponderado é chamado de tempo de fluxo ponderado.

**Tempo de término total ponderado descontado ().** Essa é uma função de custo mais geral que a anterior, onde os custos são descontados a uma taxa de *r*, 0 < *r* < 1, por unidade de tempo. Isso é, se a tarefa *j* não termina até o tempo *t*, um custo adicional é adicionado durante o período *[t, d + dt]*. Se a tarefa *j* termina no tempo *t*, o custo total do período *[0, t]* é . O valor de *r* geralmente é perto de 0, como 0,1 (ou 10%).

**Atraso total ponderado ().** Também é uma função de custo mais geral que o tempo de término total ponderado

**Número ponderado de tarefas atrasadas ().** O número ponderado de tarefas atrasadas, não somente é uma métrica de interesse acadêmico, mas também é um objetivo frequente na vida real.

É possível classificar o sequenciamento em 3 classes distintas (Pinedo , 2012):

1. **Sem espera** - Quando as operações são distribuídas entre as maquinas de forma que nenhuma máquina fica em momento algum esperando sem trabalho o termino de outra operação conforme pode ser visualizado na Figura 10a.
2. **Ativo** – Quando não existe nenhuma operação que poça ser deslocada para a esquerda sem que haja atraso em qualquer outra operação conforme Figura 10b.
3. **Semiativo** – Quando existe pelo menos uma operação que pode ser movido para uma posição a esquerda de sua localização sem atrasar nenhuma outra operação conforme pode ser visualizado na Figura 10c a operação O2.



Figura - Classificação do sequenciamento

Uma quantidade considerável de pesquisadores tem se dedicado a encontrar algoritmos com tempo polinomial para resolução de problemas de sequenciamento, porém muitos problemas não possuem um algoritmo polinomial; estes problemas são conhecidos como NP-Hard exigindo uma prova matemática para comprovar sua dificuldade (Pinedo , 2012).

Soluções utilizando técnicas inteiras ou heurísticas podem ser desenvolvidas, mas sua aplicação se torna difícil devido à natureza combinatória do problema que cresce à medida que novos produtos e rotas são criadas e a dinâmica de atualização dos dados da produção. Uma alternativa para estes casos á a aplicação de uma heurística baseada em construção que tem complexidade linear, que são frequentemente referidos como regras de sequenciamento. Estas regras atribuem uma prioridade aos trabalhos em espera na fila de acordo com uma combinação de características e parâmetros, para em seguida, selecionar o trabalho com a maior prioridade. Há muitas regras que priorizem os trabalhos de acordo com critérios diferentes, embora várias regras específicas foram desenvolvidas, não existe uma regra única que domina todos os outros (El-Bouri, 2012)

A principal vantagem na utilização de regras de despacho em ambientes dinâmicos, no entanto, é a sua capacidade para tomar suas decisões de sequenciamento com base em dados locais. Isso significa que novos postos de trabalho podem ser sequenciados em tempo real, localmente e sem a necessidade de rever ou estabelecer um novo calendário global cada vez que um novo trabalho chega. No entanto, a maioria das regras de sequenciamento comuns são míopes, ou seja, as suas decisões de sequenciamento são baseadas em informações locais, como resultado seu desempenho na maioria das situações está longe de ser ideal (El-Bouri, 2012).

As regras de sequenciamento não garantem a obtenção soluções ótimas e sua disseminação é atribuída a sua simplicidade e eficiência à luz da intratabilidade computacional de problemas de sequenciamento, no entanto, são muitas vezes a causa do baixo desempenho chão de fábrica devido à sua natureza míope (Nguyen, et al., 2013); (Sun & S. Noble, 1999). Outro aspecto que torna a regras de sequenciamento atraente para pesquisadores e profissionais é que elas não possuem problemas de escala que é o grande problema de quase todos os métodos de otimização (Nguyen, et al., 2013).

As regras de sequenciamento mais comumente aplicadas segundo (Nguyen, et al., 2013); (Su & Romanowski, 2013) são:

1. **FIFO (First in first out)** - A primeira tarefa que chegou na fila é a primeira a ser processada;
2. **SPT (Shortest processing time)** - A tarefa selecionada é a que tiver o menor tempo de processamento entre as tarefas da fila;
3. **LPT (Longest processing time)** - A tarefa selecionada é a que tiver o maior tempo de processamento entre as tarefas da fila;
4. **LSO (Longest subsequent operation)** - A tarefa selecionada é a que pertence ao trabalho que tem a mais longa tarefa subsequente;
5. **LRM (longest remaining processing time)** - A tarefa selecionada é a que pertence ao trabalho que tem o tempo restante de processamento mais longo;
6. **MWKR (most work remaining)** - Selecionar a tarefa que tem a maioria do trabalho restante;
7. **SWKR (smallest work remaining)** - Selecionar a tarefa que tem o menor trabalho restante;
8. **MOPR (largest number of operations remaining)** - Selecionar a tarefa que que tem maior número de trabalhos restantes;
9. **EDD (earliest due date)** - Selecionar a tarefa que tem menor data de entrega;
10. **MS (minimum slack)** - Selecionar a operação que tem folga mínima;
11. **WSPT (Weighted shortest processing time)** - A tarefa é selecionada a partir da ordem crescente ponderada da razão entre os tempos de processamento e a sua prioridade de atendimento.

Regras de sequenciamento mais elaboradas permitem a utilização de diferentes parâmetros, podendo formular funções objetivos mais complexas e como resultado melhorar o desempenho prático da aplicação. Tais regras surgiram com a concatenação de regras elementares ponderadas por parâmetros de escalas (Pinedo , 2012).

## Regras ATC

A heurística a*pparent tardiness cost* (ATC - custo aparente atraso) é uma regra composta que se aplica a problemas com 1 ou várias máquinas que tem como objetivo minimizar o atraso ponderado. A ATC combina a regra ponderada tempo de processamento mais curto (WSPT) e a folga mínima regra (MS), calculando um índice de prioridade para cada trabalho restante a cada vez que uma máquina se torna disponível o trabalho com a maior prioridade é selecionado para ser processado. O índice de prioridade da tarefa *j* no tempo *t* é definido como:

(1)

Onde é o peso que define a priorização da tarefa *j*, *p* é o tempo de processamento, *d* é prazo de execução e é um parâmetro de escala determinado empiricamente. Se *K* é muito grande a regra ATC reduz à regra WSPT, se *K* é muito pequena a regra reduz a regra MS (Pinedo , 2012).

Uma extensão da regra ATC para trabalhar com tempos de setup é denominada *Apparent Tardiness Cost with Setups* (ATCS), na qual foi incluso um segundo termo exponencial que considera tempos de setup entre a tarefa que foi concluída e a próxima tarefa.

(2)

No qual *slj* é o tempo médio de setup das tarefas sequenciadas, e são os parâmetros de escala empíricos relacionados com o tempo de entrega e setup.

## Balanceamento

O problema de balanceamento de linha de montagem (ALB) consiste em atribuir uma série de atividades em um conjunto de estações de trabalho de modo a otimizar uma medida de eficiência, respeitando as restrições de precedência (Becker C.;Scholl A., 2006). O principal objetivo do ALB é distribuir as tarefas uniformemente sobre cada estação de trabalho de forma que o tempo de ociosidade do homem e das máquinas sejam minimizados (Kumar & Mahto, 2013). O processo de balanceamento de produção depende de alguns do correto entendimento de alguns conceitos que serão introduzidos no decorrer do capítulo.

### Arranjo físico

O arranjo físico é uma das principais decisões que determinam a eficiência a longo prazo das operações, podendo ajudar a organização a obter uma vantagem estratégica que proporcione diferenciação, baixo custo ou resposta rápida (Heizer & Render, 2001). Segundo (Slack, et al., 2009) o arranjo físico é como seus recursos transformadores são posicionados um em relação ao outro e como as várias tarefas estão alocadas a estes recursos transformadores. A maioria dos arranjos físicos derivam apenas de quatro tipos básicos de arranjos físicos:

1. **Posicional** no qual quem sofre a transformação fica estacionário enquanto os equipamentos, maquinários, instalações e pessoas se movem a medida do necessário.
2. **Funcional** no qual os recursos ou processos similares são localizados juntos uns dos outros, desta forma, diferentes produtos terão diferentes necessidades, portanto percorrerão diferentes roteiros, fazendo com que o padrão de fluxo seja bastante complexo.
3. **Por produto** no qual os recursos transformadores são arranjados fisicamente na mesma sequência das atividades requeridas pelo produto.
4. **Celular** que mescla características do arranjo funcional e por produto no qual os recursos transformadores necessários para atender as necessidades se encontram agrupados.

Segundo (Simaria, 2001), é possível associar o perfil da procura dos produtos acabados aos tipos de arranjos físicos conforma pode ser visualizado na Figura 11, nos quais o ponto A e B correspondem a situações extremas em relação a variabilidade dos produtos e a quantidade de produtos fabricados. O ponto A produz grande número de produtos diferentes em lotes de uma unidade, já ponto B um único produto é fabricado em grandes quantidades para satisfazer a grande procura.

Em ambos os casos surge a necessidade de distribuir de forma uniforme as tarefas com a finalidade de obter altas taxas de utilização e fluxo contínuo de produção. Este problema é usualmente conhecido como balanceamento de linhas de produção.



Figura - Perfil de procura x sistema de produção (Simaria, 2001)

Conforme (Heizer & Render, 2001), os problemas das linhas de produção e das linhas de montagem são semelhantes, dessa forma é possível concentrar os esforços na resolução de problemas de linha de montagem.

### Linhas de Montagem

Uma linha de montagem pode ser definida como um conjunto de estações de trabalho composta por um ou vários operadores, dispostas de forma sequencial e ligados através de um sistema de transporte de material, no qual cada estação é responsável por um conjunto de tarefas pré-definidas no processo de montagem dos produtos. Originalmente linhas de montagem foram desenvolvidas e aplicadas na produção em massa de produtos padronizados explorando a especialização de trabalho e os efeitos da aprendizagem, porém atualmente vem ganhando importância na produção em baixos volumes (Boysen, et al., 2006).

O processo de construção de um produto é dividido em operações elementares chamadas de tarefas V = {1,...n}, e o tempo de execução de uma tarefa *j* é dado pelo tempo *tj,* onde cada tarefa necessita de um equipamento e capacitação do operador para ser executada. Devido as condições tecnológicas e organizacionais é necessário respeitar as restrições de precedência das operações que podem ser representadas por meio de um grafo direcionado composto por nós numerados que representam as tarefas e por arcos que definem o tempo e definem as relações de precedência entre as tarefas. As relações de precedência podem ser resumidas e presentadas por um grafo direcionado conforme Figura 12 no qual *n* = 10 tarefas todas com seus respectivos tempos e suas precedências. O grafo descreve que a tarefa 5 requer que as tarefas 1 e 4 (predecessores diretos) e a tarefa 3 (predecessor indireto) sejam completadas antes da sua execução (Scholl & Becker, 2006).



Figura - Grafo de precedência (Scholl & Becker, 2006)

As linhas de produção podem ser classificadas quanto ao número de produtos conforme Figura 13, onde cada figura geométrica representa um diferente modelo produzido (Becker C.;Scholl A., 2006):

1. Linha de produção simples: Produção em massa de um único produto.
2. Linha de produção mista: Utilizada na produção de vários modelos de um produto, geralmente com tempo de setup insignificante.
3. Linha multi-modelo: Utilizada quando existem diferenças significativas entre cada modelo de produto, geralmente utilizado em lotes.



Figura - Linhas de montagem (Becker C.;Scholl A., 2006)

Uma característica importante da linha de montagem diz respeito a variabilidade do tempo da tarefa. Tarefas com pouca variação no tempo, geralmente tarefas simples ou estações automatizadas os tempos são considerados deterministas, já variações consideráveis principalmente devido à instabilidade de trabalhos manuais relacionados a habilidade e motivação dos funcionários exigem que o tempo seja considerado estocástico (Becker C.;Scholl A., 2006).

Ao projetar uma linha de montagem algumas restrições devem ser impostas ao agrupamento de elementos de trabalho tais como ( Ponnambalam, et al., 2000):

* Gráfico de precedência das operações
* O número de estações de trabalho deve ser menor que o número de estações de trabalho disponíveis.
* O tempo de ciclo deve ser maior ou igual ao máximo tempo de qualquer estação de trabalho.

As linhas de montagem foram classificadas com base em 3 elementos representados pela notação [α│β│γ] proposta por (Boysen, et al., 2006), onde:

* α – Representa as características do grafo de precedência que é representado por 6 atributos α1 a α6;
* β – Representa as características da estação e da linha de produção que é representado por 6 atributos β1 a β6;
* γ – Representa os objetivos, no qual γ ϵ {*m*, *c*, *E*, *Co*, *Pr, SSLk, score,* o}.

Para cada um dos elementos descreve uma série de possibilidades e restrições que são encontrados em problemas reais e que são retratados pela literatura. As características do grafo de precedência são descritas na Tabela 1.

|  |  |
| --- | --- |
| **Características do grafo de precedência** | |
| **Grafo de precedência do produto: α1 ϵ {mix, mult, o}** | |
| α1 = mix | Modelo de produção misto |
| α1 = mult | Modelo de produção multi-modelo |
| α1 = o | Modelo de produção de um único modelo |
| **Estrutura do grafo de precedência: α2 ϵ {spec, o}** | |
| α2 = spec | Precedência de acordo com grafo |
| α2 = o | A precedência pode ter qualquer estrutura acíclica |
| **Tempo de processamento: α3 ϵ {tsto, tby, o}** | |
| α3 = tsto | Tempo de processamento estocástico |
| α3 = tby | Tempo de processamento dinâmico (aprendizagem) |
| α3 = o | Tempo estático e determinístico |
| **Incremento no tempo dependendo da sequência α4 ϵ {Δtdir, Δtind, o}** | |
| α4 = Δtdir | Se 2 tarefas são executadas na mesma estação um tempo adicional pode ser necessário para o setup ou alteração das ferramentas |
| α4 = Δtind | O tempo é incrementado se o estado de conclusão de uma tarefa tem efeito sobre o tempo de processamento de outras tarefas que são executadas mais tarde na mesma ou em outra estação |
| α4 = o | Não considera incremento de tempo de acordo com a sequência |
| **Restrições de atribuição α5 ϵ {link, inc, cum, fix, excl, type, min, max, o}** | |
| α5 = link | Subconjunto de tarefas são ligadas de tal forma que devem ser atribuídos a mesma estação |
| α5 = inc | Subconjunto de tarefas são incompatíveis e não devem ser atribuídos a mesma estação |
| α5 = cum | A atribuição da tarefa está sujeita a restrições sobre o valor acumulado de determinado atributo (Ex. espaço restrito de material) |
| α5 = fix | Algumas tarefas só podem ser atribuídas a estações particulares |
| α5 = excl | Algumas tarefas não podem ser atribuídas a estações particulares |
| α5 = type | Algumas tarefas devem ser atribuídas a uma estação a partir de um determinado conjunto |
| α5 = min | Ao atribuir uma tarefa distâncias mínimas em relação a outras tarefas medida em tempo, espaço ou sequência tem que ser observadas |
| α5 = max | Ao atribuir uma tarefa distâncias máximas em relação a outras tarefas devem ser observadas |
| α5 = o | Não há restrições de atribuição |
| **Alternativas de processamento α6 ϵ {, o}** | |
| α6 = | Alternativas de processamento podem ser separados seguindo alternativas do grafo de precedência definido por ϵ {o, prec, subgraph} |
| = o | Alternativas de processamento só se desviam em relação a tempo e custo |
| = prec | Alternativas de processamento não só afeta o tempo e o custo, mas também a relação de precedência entre as tarefas |
| = subgraph | Alternativas de processamento podem alterar o processo de produção de modo que subgrafos podem ser substituídos. Ex. Quando um conjunto de ações podem ser instalados separadamente ou completamente substituído por uma parte comprada |
| α6 = o | Não são consideradas alternativas de processamento |

Tabela - Características do grafo de precedência (Boysen, et al., 2006)

As características da estação de trabalho e da linha de produção são descritas na Tabela 2.

|  |  |
| --- | --- |
| **Características da estação de trabalho e da linha de produção** | |
| **Movimentação dos produtos: β1 ϵ {o, }** | |
| β1 = o | Em linhas de produção ritmadas o tempo de ciclo restringe o tempo de cada estação de trabalho com ϵ {o, each, prob} e ϵ {o, div} |
| = o | A cada ciclo o trabalho em uma estação é limitado pela média do tempo de ciclo. Esta restrição é rigorosa na produção de um modelo único, é cumprida em média no caso da produção de modelo misto, algum modelo de paralelização ou tempos estocásticos. |
| = each | A cada modelo o tempo de ciclo deve ser respeitado seja para modelo misto ou multi-modelo |
| = prob | A restrição de tempo de ciclo é obedecida seguindo determinada probabilidade ou proporção |
| = o | Todas as estações e modelos devem considerar o mesmo tempo de ciclo global |
| = div | O tempo de ciclo diverge entre as estações de trabalho e modelos dos produtos |
| β1 = | Linhas unpaced não são restringidas por um tempo de ciclo. Em vez disso ela avança quando as estações tiverem completado suas tarefas, com ϵ {o, cync} |
| = o | Em linhas assíncronas, logo que uma estação de trabalho termina uma tarefa, a peça é movida para a estação seguinte, ou para um buffer a frente, a menos que ocorra um bloqueio. Neste caso o problema de ALB é acompanhado de uma decisão adicional de posicionamento e dimensionamento de buffers. |
| = sync | Em linhas de trabalho síncronas o movimento dos produtos é coordenado entre as estações de trabalho, ou seja, um produto é movido para a estação seguinte quando todas as estações tiverem completado a tarefa |
| **Layout da linha: β2 ϵ {o, }** | |
| β2 = o | As estações são dispostas em série ao longo do fluxo da linha de produção |
| β2 = | Utilizada linha de produção em formato de U com estações se cruzando, onde ϵ {o, n} |
| = o | Linha em formato simples de U |
| = n | A linha e composta por *n* seguimentos em formado de U |
| **Paralelização: β3 ϵ {, }** | |
| β3 = | Mais de uma linha paralela deve ser balanceada ou o número de linhas instaladas faz parte do problema de decisão |
| β3 = | Quando as estações estão em paralelo, os seus recursos e conteúdos de trabalho são duplicados para permitir o processamento de todas as tarefas alternadamente |
| β3 = | Uma tarefa paralelizada é relacionada a mais de uma estação. Dessa forma um trabalho adicional é inserido ao processar uma tarefa paralela alternadamente |
| β3 = | Várias estações trabalham simultaneamente na mesma parte em locais diferentes de montagem de modo que um não obstrua o trabalho do outro. O problema de equilíbrio está ligado a programação detalhada de forma a evitar que operadores diferentes interferir uns com os outros. |
| β3 = o | Nenhum tipo de paralelização é considerado |
| = o | Seja um nível fixo ou máximo de paralelização que constitui o problema o número de linhas em paralelo é dado por = {1, 2, 3...} |
| Atribuição de recursos: β4 = {*equip*, , o} | |
| β3 = *equip* | Para cada estação exatamente um equipamento deve ser escolhido entre um conjunto de alternativas especificados. O problema de equilíbrio está ligado a um problema de seleção de equipamento. |
| β3 = | Em vez de selecionar um único equipamento a partir de um conjunto de alternativas de equipamentos, o equipamento de uma estação está configurado juntamente com a atribuição da tarefa. Se várias tarefas requerem o mesmo recurso, sinergias podem combinar estas tarefas na mesma estação de trabalho, pois os recursos são necessários apenas uma vez reduzindo os custos. Existem 2 tipos básicos de sinergia dados por = {o, 01, *max*} |
| = o | Os recursos são modelados explicitamente para a conta ou outro tipo de sinergia ou dependência |
| = 01 | Se mais de uma tarefa pode ser realizada com o mesmo recurso (ferramenta ou máquina), o custo de investimento é reduzido se as tarefas são atribuídas a mesma estação, pois o recurso tem que ser instalado apenas uma vez. Isto é uma decisão de investimento 0-1 tem de ser feita para cada combinação de estação e recurso |
| = *max* | Se as tarefas diferem em relação a qualidade dos recursos (velocidade, capacidade ou qualificação), é exigido a seleção dos recursos de forma a cumprir o nível de demanda máxima das tarefas. Por exemplo, se os operadores precisam de uma qualificação especifica para cumprir uma tarefa difícil |
| Incremento de tempo dependente da estação: β5 = { *Δtun*p, o} | |
| β5 = *Δtu*np | Quando alguma parte do tempo da estação é consumida por atividades improdutivas como transporte da peça ou o retorno dos trabalhadores para o início da estação no final do ciclo |
| β5 = *o* | Incremento de tempo dependente da estação não é considerado |
| Aspectos adicionais da configuração da linha: β6 = {*buffer, feeder, mat, change, o*} | |
| β6 = *buffer* | Locais de armazenagem intermediários são necessários e devem ser alocados e dimensionados |
| β6 = *feeder* | Um ou mais linhas de alimentação fluem de uma linha principal e exigem uma coordenação simultânea de atribuições de tarefa e tempo de ciclo |
| β6 = *mat* | Caixas contendo a necessidade de material deve ser posicionada e dimensionada |
| β6 = *change* | Se determinadas tarefas exigir que a peça de trabalho esteja em determinada posição (levantada, inclinada, etc.), uma decisão tem que ser feita, se a posição é fixa dentro da estação de trabalho ou se maquinas especiais são necessárias para a mudança de posição |
| β6 = *o* | Não são considerados aspectos adicionais na configuração da linha |

Tabela - Características da estação de trabalho e da linha de produção

O processo de balanceamento é guiado por um objetivo que se busca ao executar o processo de balanceamento da linha de produção. As características destes objetivos estão descritas na Tabela 3.

|  |  |
| --- | --- |
| **Objetivos** | |
| **Objetivos: ϵ {*m, c, E, Co, Pr, , score, o}*** | |
| = *m* | Minimizar o número de estações *m* sujeita a um determinado alvo de saída para um determinado horizonte de planejamento (especificado o tempo de ciclo ou a taxa de produção) |
| = *c* | Minimizar o tempo de ciclo para um dado número de estações |
| = *E* | Maximizar a eficiência E da linha respeitando as restrições de taxa de produção e número de estações |
| = *Co* | Minimizar os custos para uma dada meta de produção |
| = *Pr* | Resultado definido como a diferença entre receita e custo é maximizada |
| = | Os tempos das estações devem ser suavizados. Há 2 maneiras de suavizar os dados para = {*stat*, *line*} |
| = *stat* | Na produção de modelos misto onde os tempos da estação são variáveis por causa dos diferentes modelos |
| = *line* | Os tempos das estações são suavizados sobre todas as estações da linha |
| = *score* | O objetivo é minimizar ou maximizar alguma pontuação composta relacionada com um ou mais atributos que descrevem aspectos de gargalo ou outras medidas de eficiência |
| = *o* | A função objetivo não é necessária, somente soluções viáveis são procuradas |

Tabela – Objetivos

### Balanceamento de Linha de Montagem

O problema de balanceamento da linha surge quando se decide distribuir o trabalho entre as estações de forma que cada estação fique com uma mesma carga produtiva (Becker C.;Scholl A., 2006),

O balanceamento de uma linha de montagem é processo de distribuir de forma equilibrada o trabalho de montagem entre as estações guiado por um objetivo conhecido (ALBP- Assembly line balancing problem). Qualquer ALBP consiste em encontrar uma alternativa viável e equilibrada para a linha de produção na qual cada tarefa esteja relacionada a uma estação de trabalho e as restrições de precedência sejam respeitadas. Algumas definições dadas por (Kumar & Mahto, 2013) são necessárias para uma melhor compreensão dos cálculos de balanceamento de uma linha de montagem:

* Tempo total necessário para a produção de um produto é igual ao somatório dos tempos de todas as operações conforme Equação 1, no qual *ti* é tempo necessário para completar a tarefa i da linha de produção e *k* é úmero total de operações a serem executadas na linha de produção.

(1)

* Tempo de ciclo é o tempo entre as saídas consecutivas de um produto de uma linha de montagem, ou seja, corresponde ao maior tempo gasto em uma estação de trabalho na linha de montagem e

(2)

* O cálculo do número mínimo de estações necessárias está descrito na Equação 3, na qual o resultado deve ser sempre arredondado para cima.

(3)

* Eficiência da linha de montagem.

(4)

* Folga na linha de montagem.

(5)

A classificação do ALBP proposta por (Baybars, 1986) divide os problemas em 2 classes:

1. Problema de Balanceamento da Linha de Montagem Simples (SALBP, em inglês Simply Assembly Line Balancing Problem)
2. Problema de Balanceamento da Linha de Montagem Geral (GALBP, em inglês General Assembly Line Balancing Problem).

A SALBP é o problema mais estudado do campo de balanceamento de linha de montagem e segundo (Scholl & Becker, 2006) possui as seguintes propriedades:

* Produção em massa de um produto homogêneo;
* Todas as tarefas são processadas de um modo pré-determinado;
* Linha compassada com tempo de ciclo *c* fixo de acordo com uma desejada quantidade de resultado;
* Tempos de operação (de tarefa) determinísticos e inteiros;
* Nenhuma restrição de atribuição além das restrições de precedência;
* Layout da linha em série com *m* estações somente de um lado e sem linha de alimentação;
* A sequência do processamento das tarefas está sujeita a restrições de precedência;
* Uma tarefa não pode ser dividida entre duas ou mais estações;
* Todas as estações são igualmente equipadas com respeito a máquinas e trabalhadores;
* Maximização da eficiência da linha: , no qual *m* é o número de estações de trabalho, *c* é o tempo de ciclo e *tsum* = é a soma dos tempos de todas as tarefas.

Os SALBP foram classificados em 4 tipos diferentes em relação a função objetivo conforme pode ser visualizado na Tabela 4 (Scholl, 1999):

1. SALBP-1: Dado um tempo de ciclo c, minimizar o tempo ocioso das estações.
2. SALBP-2: Dado o número de estações, minimizar o tempo de ciclo.
3. SALBP-E: Se nem o tempo de ciclo nem o número de estações é informado, o objetivo é encontrar qual a melhor configuração da linha para alcançar uma melhor eficiência.
4. SALBP-F: Se o tempo de ciclo e o número de estações são informados o objetivo passa a ser encontrar uma configuração equilibrada e viável para a linha.



Tabela - Versões SALBP (Scholl, 1999)

É possível perceber que as características do SALBP são restritas o que não permite representar adequadamente problemas do mundo real. Para estes problemas do mundo real com um número maior de considerações é utilizado os chamados problemas de balanceamento de linha de montagem gerais (GALBP), que atualmente vem ganhando mais interesse por parte dos pesquisadores (Martino & Pastor, 2010).

Mesmo com várias descrições do ALBP na literatura há 5 características presentes em qualquer tipo de ALBP (Boysen, et al., 2006). São elas:

1. Os produtos são conhecidos
2. Um conjunto de processamentos alternativos é dado.
3. A linha deve ser configurada de modo que as quantidades alvo de produção sejam satisfeitas considerando certo horizonte de planejamento.
4. O fluxo da linha é unidirecional.
5. A sequência de processamento das tarefas está sujeita a restrições de precedência.

É possível visualizar a classificação dos ALB na Figura 14, na qual os problemas são subdivididos em linha de modelo simples ou multi-modelo/misto e com relação a variabilidade dos tempos determinístico ou estocástico e no ultimo nível em problemas de SALBP e GALBP (Ghosh, 1989); (Kumar & Mahto, 2013). As linhas multi-modelos são utilizadas para montagem de dois ou mais modelos de um mesmo produto simultaneamente. A quantidade de cada modelo produzido pela linha é normalmente baixa, devido a necessidade do cliente por modelos exclusivos. Estas características podem ser encontradas em diversas áreas da indústria; como exemplo, indústrias de vestuário; automobilística e aparelhos eletrônicos (SARKER & PAN, 1998).



Figura Classificação ALBP (Ghosh, 1989)

Uma classificação similar pode ser analisada na Figura 15, no qual o autor substituiu o ultimo nível pelas características da linha de montagem entre em série e formato de U (Sivasankaran & Shahabudeen, 2014).



Figura - Classificação ALBP (Sivasankaran & Shahabudeen, 2014)

Os problemas de balanceamento também podem ser classificados conforme seus objetivos ( Ponnambalam, et al., 2000):

**Tipo 1:** Dado o tempo de ciclo e a precedência, o objetivo é minimizar o número de estações de trabalho. Este modelo surge quando há a concepção de novas linhas de montagem.

**Tipo 2:** Dado as estações de trabalho o objetivo é minimizar o tempo de ciclo. Neste modelo é necessário tratar a precedência das operações e o posicionamento dos postos de trabalho.

### Técnicas de Solução ALBP

A literatura inclui um grande número de procedimentos para solução dos ALBP como pode ser analisado na pesquisa (Boysen, et al., 2006). As diversas técnicas de solução podem ser classificadas como (Bautista & Pereira, 2009):

* **Modelos Heurísticos:** Que utilizam regra de prioridade para atribuir tarefas a diferentes estações de trabalho, produzindo boas soluções em um tempo computacional reduzido. Dentre os procedimentos é possível destacar: Método dos pesos posicionais (RPW-Rank Positional Weight); Método de Kilbridge e Wester (KWM); Este Regra do maior candidato (LCR – Largest Candidate Rule).
* **Modelos Matemáticos**: Normalmente baseadas árvores de busca, programação dinâmica ou algoritmo de Branch and Bound.
* **Modelos metaheurísticas:** Não produzem resultados tão bons como procedimentos de enumeração, porém, possuem uma maior aplicabilidade para procedimentos para os problemas gerais. Dentre as principais é possível destacar: Busca local; colônia de formigas; Simulated anneling e Algoritmos genéticos.

Não há um modelo desenvolvido versátil o suficiente para atender juntamente todas as condições especiais que envolvem os ALBP (Boysen, et al., 2008). Uma união de todos os aspectos relacionados ao processo de balanceamento em um único modelo é uma ambição distante, por este motivo diversos modelos são pesquisados e propostos para resolver problemas específicos.

# Bibliografia

Ponnambalam, S. G. P., Aravindan, P. A. & Naidu, G. M. N., 2000. A Multi-Objective Genetic Algorithm for Solving Assembly Line Balancing Problem. *The International Journal of Advanced Manufacturing Tecnology,* 16(1), p. 341–352.

ABIT, 2016. *Relatório de Atividades ABIT 2015,* São Paulo: Associação Brasileira da Indústria Têxtil e de Confecção.

Baker, K. R. B. & Trietsch, D. T., 2009. *PRINCIPLES OF SEQUENCING AND SCHEDULING.* 1 ed. New Jersey: JohnWiley & Sons.

Bautista, J. B. & Pereira, J. P., 2009. A dynamic programming based heuristic for the assembly line balancing problem. *European Journal of Operational Research,* Volume 194, p. 787–794.

Baybars, I. B., 1986. A survey of exact algorithms for the simple assembly line balancing problem. *Management Science,* 32(1), pp. 909-932.

Becker C.;Scholl A., 2006. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research,* p. 694–715.

Boysen, N. B., Fliedner , M. F. & Scholl, A. S., 2006. A classification of assembly line balancing problems. *European Journal of Operational Research*, 14 Dezembro, p. 674–693.

Boysen, N. B., Fliedner, M. F. & Scholl, A. S., 2008. Assembly line balancing: Which model to use when?. *Int. J. Production Economics*, 7 Março, p. 509–528.

Boysen, N. B., Fliedner, M. F. & Scholl, A. S., 2009. Sequencing mixed-model assembly lines: Survey, classification and model critique. *European Journal of Operational Research,* 192(1), p. 349–373.

Brucker, P. B., 2007. *Scheduling Algorithms.* 5 ed. s.l.:Springer.

Chen J. C.;Chen C. C.;Su L. H.;Wub H. B.;Sun C. J., 2012. Assembly line balancing in garment industry. *Expert Systems with Applications,* p. 10073–10081.

Coello, C. A. C., Lamont, G. B. L. & Veldhuizen, D. A. V. V., 2007. *Evolutionary Algorithms for Solving Mult-Objective Problems.* 2 ed. s.l.:Springer.

Deb, K. D., 2009. *Multi-Objective Optimization using Evolutionary Algorithms.* 1 ed. s.l.:John Wiley & Sons, LTD.

Deb, K. D., Pratab, S. P., Agarwal, S. A. & Meyarivan, T. M., 2002. A Fast and Elitist Multiobjective Genetic Algorithm: NGSA-II. *IEEE Transactions on Evolutionary Computing,* 6(2), p. 182–197.

Dias, A. H. F. D. & Vasconcelos, J. A. V., 2002. Multiobjective Genetic Algorithms Applied to Solve Optimization Problems. *IEEE TRANSACTIONS ON MAGNETICS,* 38(2), pp. 1133 - 1136.

Dias, L. M. C., 2000. *A Informação Imprecisa e os Modelos Multicritério de Apoio a Decisão,* Coimbra: s.n.

El-Bouri, A. E.-B., 2012. A cooperative dispatching approach for minimizing mean tardiness in a dynamic flowsho. *Computers & Operations Research,* 39(7), p. 1305–1314.

Fonseca, C. M. F. F. & Fleming, P. J. F., 1995. An overview of evolutionary algorithms in multiobjective optimization. *Journal Evolutionary Computation,* 3(1), pp. 1-16.

Gen, M. G. & Lin, L. L., 2014. Multiobjective evolutionary algorithm for manufacturing scheduling problems: State-of-the-art survey. *Journal of Intelligent Manufacturing,* 25(5), p. 849–866.

Ghosh, S. G., 1989. A Comprehensive Literature Review and Analysis of the Design, Balancing and Scheduling of Assembly Systems. *International Journal of Production Research*, pp. 637-670.

Gomes, L. F. A. M. G., Araya, M. C. G. A. & Carignano, C. C., 2003. *Tomada de Decisões em Cenários Complexos.* 1 ed. São Paulo: Cengage Learning.

Gomes, L. F. A. M. G., Gomes, C. F. S. G. & Almeida, A. T. A., 2009. *Tomada de Decisão Gerencial.* 3 ed. São Paulo: ATLAS.

Graeml A. R.;Peinado J., 2007. *Administração da Produção (Operações Industriais e de Serviços).* Curitiba: UnicenP.

Graham, R. G., Lawler, E. L., Lenstra, J. L. & Kan, A. R. K., 1979. Optimization and approximation in deterministic sequencing and scheduling: A survey. *Annals of Discrete Mathematics,* Volume 5, pp. 287-326.

Heizer, J. H. & Render, B. R., 2001. *Administração de Operações.* 5 ed. s.l.:LTC.

Horn, J. H., Nafpliotis, N. N. & Goldberg, D. E. G. G., 1994. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. *IEEE World Congress on Computational Intelligence,* 1(1), pp. 82-87.

IBGE/PIA, 2014. *Relatório de acompanhamento setorial: Têxtil e confecção.* [Online]   
Available at: http://www.abdi.com.br/Estudo/textil%20e%20confeccao%20junho%2008.pdf  
[Acesso em 10 6 2014].

IDEIES, 2011. *Textil e Confecções do Espírito Santo, Caderno Inteligência Competitiva.* Vitória-ES: IDEIES.

Kumar, N. K. & Mahto, D. M., 2013. Assembly Line Balancing: A Review of Developments and Trends in Approach to Industrial Application. *Global Journal of Researches in Engineering*, pp. 28-49.

Kumar, N. K. & Mahto, D. M., 2013. Assembly Line Balancing: A Review of Developments and Trends in Approach to Industrial Application. *Global Journal of Researches in Engineering,* 13(2), pp. 29-50.

LEE, Y. H. L., BHASKARAN, K. B. & PINEDO, M. P., 1997. A heuristic to minimize the total weighted tardiness with sequence-dependent setups. *IIE Transactions,* Volume 29, pp. 45-52.

Lu, H. L., Niu, R. N., Liu, J. L. & Zhu, Z. Z., 2013. A chaotic non-dominated sorting genetic algorithm for the multi-objective automatic test task scheduling problem. *Applied Soft Computing,* 13(1), pp. 2790-2802.

MARCONI, M. D. A. & LAKATOS, E. M., 2010. *Fundamentos da Metodologia Científica 7ª. ed..* São Paulo: Atlas.

Martino, L. M. & Pastor, R. P., 2010. Heuristic procedures for solving the general assembly line balancing problem with setups. *International Journal of Production Research,* 48(1), pp. 1787-1804.

MTE/RAIS, 2009. *Ministério do trabalho e emprego.* [Online]   
Available at: http://www3.mte.gov.br/rais/  
[Acesso em 10 6 2014].

Nassirnia, P. N. & Md. Tap, M. M. T., 2010. Strategies to Achieve Labor Flexibility in the Garment Industry. *IJCSI International Journal of Computer Science Issues,* 7(4).

Nguyen, S. N., Zhang, M. Z., Johnston, M. J. & Chen Tan, K. C. T., 2013. A computational study of representations in genetic programming to evolve dispatching rules for the job shop scheduling problem. *IEEE Transactions on Evolutionary Computation,* 17(5), pp. 621- 639.

Pinedo , M. L. P., 2012. *Scheduling: Theory, Algorithms, and Systems.* 4 ed. s.l.:Springer.

POPPER, K., 1953. *Science: Conjectures and refutations. philosophyfaculty.ucsd.edu/.* [Online]   
Available at: http://philosophyfaculty.ucsd.edu/faculty/rarneson/Courses/popperphil1.pdf  
[Acesso em 24 Março 2014].

SARKER, B. R. S. & PAN, H. P., 1998. Designing a mixed-model assembly line to minimize the costs of idle and utility times. *Computers & industrial engineering,* 34(3), pp. 609 -628.

Scholl, A., 1999. *Balancing and sequencing assembly lines.* s.l.:Darmstadt Technical.

Scholl, A. S. & Becker, C. B., 2006. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*, 11 Setembro, p. 694–715.

Scholl, A. S. & Becker, C. B., 2006. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research*, 11 Setember, p. 666–693.

Silveira G.;Borenstein D.;Fogliatto F. S., 2000. Mass customization: Literature review and research directions. *Production Economics,* pp. 1-13.

Simaria, A. S. d. A. S., 2001. *Uma Metodologia para o Balanceamento de Linhas de Montagem.* Porto: EGP.

Sivasankaran, P. S. & Shahabudeen, P. S., 2014. Literature review of assembly line balancing problems. *The International Journal of Advanced Manufacturing Technology*, p. 1665–1694.

Sivasankaran, P. S. & Shahabudeen, P. S., 2014. Literature review of assembly line balancing problems. *The International Journal of Advanced Manufacturing Technology,* 73(9), pp. 1665-1694.

Slack, N. S., Chambers, S. C. & Johnston, R. J., 2009. *Administração da Produção.* 3 ed. São Paulo: Atlas S.A..

Srinivas, N. S. & Deb, k. D., 1994. Multiobjective Optimization Using Nodominated Sorting in Genetic Algorithms. *Journal of Evolutionary Computation ,* 2(3), pp. 221-248.

Su, M. S. & Romanowski, R. R., 2013. Multicontextual dispatching rules for job shops with dynamic job arrival. *International Journal of Advanced Manufacturing Technology,* 67(1-4), pp. 19-33.

Sun, X. S. & S. Noble, J. S. N., 1999. An Approach to Job Shop Scheduling with Sequence-Dependent Setups Xiaoq. *Journal of Manufacturing System,* 18(6).

Tempelmeier, H. T., 2003. Practical considerations in the optimization of flow production systems. *International Journal of,* 41(1), p. 149–170.

Uddin M. K.;Soto M. C.;Lastra J. L. M., 2010. An integrated approach to mixed-model assembly line balancing and sequencing. *Assembly Automation,* pp. 164 - 172.

Valente, J. M. V., 2007. Improving the performance of the ATC dispatch rule by using workload data to determine the lookahead parameter value. *International Journal of Production Economics,* 106(1), p. 563–573.

Zhuming , B. Z. & Zhang, W. Z., 2001. Flexible fixture design and automation: review, issues and future directions. *International Journal of Production Research,* 39(13), pp. 2867- 2894.

Zitzler, E. Z. & Thiele, L. T., 1999. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION,* 3(4), pp. 257 - 271.