

Sistema de decisão multicritério e multiobjetivo para programação de produção e dimensionamento de lote em uma indústria de beneficiamento de malha

Maxsuel Vieira Orti
Laboratório de Inteligência Aplicada - LIA
Universidade do Vale do Itajaí
Itajaí/SC/BR
maxsuel@edu.univali.br

Raimundo C. G. Teive
Laboratório de Inteligência Aplicada – LIA
Universidade do Vale do Itajaí Itajaí/SC/BR
rteive@univali.br

ABSTRACT

As atividades de programação de produção e dimensionamento de lotes são essenciais no processo de decisão operacional de uma indústria, sendo a otimização destas fundamentais para a lucratividade de uma empresa. Essa otimização tem uma complexidade computacional alta, sendo classificado como um problema NP-Difícil. Neste trabalho foi aplicado a este problema de otimização um Algoritmo Genético Multiobjetivo NSGA-II, combinado com um modelo multicritério, o *Rank Order Centroid* (ROC), com o objetivo de maximizar a quantidade produzida, minimizar o custo médio e minimizar o desvio entre o que é produzido e o que está em estoque. Os resultados obtidos com os experimentos realizados, em uma indústria de beneficiamento de malhas, apontam que o algoritmo desenvolvido diminuiu em 12% o custo médio, aumentou em 10% o peso total programado e minimizou a distância para o estoque regulador em 8%, indicando que a solução proposta é viável, apresentando um desempenho superior, em todas as funções objetivos, ao modelo implementado pela empresa.

KEYWORDS

Programação da produção, Dimensionamento de lote, Algoritmo genético multiobjetivos, Análise multicritério.

1 INTRODUÇÃO

A atividade de gerenciamento de estoque e programação de produção são fundamentais para o crescimento sustentável de uma empresa. O desenvolvimento de estratégias para inventário, planejamento de produção e programação de produção são indispensáveis para a sobrevivência de uma empresa no longo prazo e para obter vantagens competitivas [1]. As empresas precisam garantir a quantidade ideal de produto em estoque conforme a demanda e ao mesmo tempo otimizar a alocação de recursos e as tarefas do ambiente de produção, balanceando a demanda com a capacidade de fornecer um produto [2].

O dimensionamento de lote (DL) é uma operação de gerenciamento de estoque que tem por objetivo determinar a quantidade ideal de produção de um produto em um período, considerando a disponibilidade de estoque, a demanda e as necessidades dos clientes [1], [2]. Por natureza, o DL busca manter uma quantidade certa de produto em estoque, evitando a falta ou

excesso de itens, assim como minimizando o custo de armazenamento [3], sendo uma ponte entre o estoque e a produção.

A programação de produção (PP), por sua vez, é um processo industrial que tem como propósito decidir quando um produto será feito, quais são os recursos humanos necessários e quais equipamentos serão utilizados [4], [5]. Uma PP otimizada é uma que minimiza o tempo de produção e os custos, tentando maximizar a eficiência da operação [6]. Enquanto a programação de produção busca atribuir recursos, o dimensionamento de lotes tenta definir a quantidade ótima de produção. Desta forma, observa-se uma relação de proximidade entre os dois processos, sendo importante que as duas decisões sejam feitas de forma conjunta para usar a capacidade industrial de forma eficiente [1].

A literatura científica considera tanto o problema de PP quanto o de DL como de complexidade NP-Difícil [7], sendo usualmente tratados de forma mono-objetivos, devido aos aspectos conflitantes envolvidos, no qual busca-se maximizar a quantidade da produção e minimizar o custo e o tempo de entrega. Porém, esta situação é conflitante, pois a maior produção não significa menor custo ou atendimento de demanda. Portanto, uma abordagem multiobjetivo e multicritério é necessária para tratar o problema. Entretanto, observa-se na literatura que os poucos trabalhos que apresentam abordagens multiobjetivos para estes problemas, em geral não fazem o ranqueamento das soluções ótimas de Pareto.

O presente trabalho propõe a aplicação de um Algoritmo Genético Multiobjetivo (AGMO) para resolver os problemas de PP e DL de uma indústria têxtil, especializada em beneficiamento de malha, de forma integrada. O objetivo é maximizar a quantidade de itens produzidos, minimizar o custo e minimizar o desvio entre os itens produzidos e os que estão em estoque. As soluções de Pareto encontradas pelo AGMO serão ranqueadas seguindo um modelo de decisão multicritério.

Este artigo está organizado da seguinte maneira. Na seção 2 o problema de pesquisa é descrito, focando na programação da produção e no dimensionamento de lote, discutindo como estes problemas têm sido resolvidos na literatura. Nas seções 3 e 4 as

técnicas adotadas para solução destes problemas, otimização multiobjetivos e análise de decisão multicritérios, são exploradas. Na sequência, os aspectos técnicos do sistema computacional desenvolvido são apresentados, detalhando o algoritmo NSGA-II implementado, bem como o modelo de decisão utilizado. Alguns resultados são apresentados na seção 6, e comentários finais na seção 7.

2 PROBLEMA DE PESQUISA

Os problemas de Programação da Produção (PP) e Dimensionamento de Lotes (DL) de uma indústria são objetos deste estudo. Nesta seção serão apresentados os conceitos básicos referentes a estes problemas, bem como o estado da arte sobre como estes problemas têm sido abordados na literatura técnica.

2.1 Problemas de PP e DL

De forma prática, o objetivo do DL é determinar o que produzir e quando [8]. Determinar o tamanho de um lote representa uma das atividades mais importante para o processo de programação de produção [9]. Dentre os processos contidos em um gerenciamento de estoque, o dimensionamento de lote (DL) é a ponte entre o estoque e a produção.

Desta forma, o objetivo principal de um DL é manter a quantidade ideal de produtos em estoque, com o propósito de satisfazer demanda, evitando-se a falta ou o excesso de itens [3]. Adicionalmente, DL pode se referir à quantidade de produto que deve ser produzido em uma data específica, ou em uma execução de produção [2].

A programação de produção (PP), que faz parte do planejamento e controle de produção, é fundamental para o desempenho de uma indústria [10]. O propósito da PP é minimizar o tempo de produção e o seu custo ao decidir quando produzir um item, quais recursos humanos utilizar e quais equipamentos serão necessários [4]; [11]; [12]. De forma geral, os problemas de PP são classificados principalmente como: *flow shop* e *job shop* [13].

O *Flow Shop* (FS) é uma PP que considera que as operações para a produção de um item são feitas de forma sequencial, sempre seguindo a mesma ordem, sendo que as máquinas da linha de produção, geralmente, estão posicionadas de forma a gerar um fluxo unidirecional [13]. Já para o *Job Shop* (JS), segundo [13], as operações para a produção de um item têm rota fixa, mas não é necessariamente a mesma para todos os itens produzidos. Dentro de um ambiente industrial, dependendo da perspectiva analisada, é possível perceber que tanto o JS quanto o FS podem coexistir. Adicionalmente, é perceptível que a operação de PP e DL acabam se complementando.

2.2 Estado da Arte

Os problemas de Dimensionamento de Lotes (DL) e Programação da Produção (PP) foram tratados inicialmente na literatura técnica de forma isolada. No caso do DL, essa operação tem sido pesquisada por mais de cem anos, sempre focada na quantidade de produção que é considerada a mais econômica. O início do estudo de DL, denominado ordem econômica (*Economic Order Quantity* – EOQ), foi proposto por [14]. É considerado o primeiro modelo para auxiliar na tomada de decisão sobre o que produzir e quando.

O modelo de [14] foi estendido por [15]; que juntos são a base da literatura científica nesse assunto [8]; [9]. De forma geral, todos esses modelos partem de uma premissa básica de minimizar o custo de armazenamento do estoque. Neste contexto, definir o tamanho ideal de um lote pode ser considerado um dos problemas mais importantes para a programação de produção [9].

A pesquisa de PP por sua vez começou com o estudo de derivações em busca de um agendamento otimizado, que evoluiu para o estudo da complexidade e classificação dos problemas de PP, como NP-Difícil, e que nos últimos anos tem apresentado um foco na utilização de metaheurísticas para a solução do problema [4]. De forma geral, os problemas de PP são classificados principalmente como: *flow shop* e *job shop* [13].

Os problemas de DL e PP foram tratados de forma conjunta, inicialmente de forma mono-objetivo, buscando a minimização dos custos globais de produção, incluindo o armazenamento, utilizando-se Algoritmo Genético, em [16]. No mesmo ano, foi proposta em [17] a primeira abordagem multiobjetivos para estes problemas, utilizando-se o algoritmo NSGA-II. Em [18] foi demonstrado a viabilidade da aplicação do algoritmo NSGA-II para resolver o problema de DL com uma PP do tipo *Job Shop* flexível, em uma atividade que recicla fibra carbono, buscando demonstrar como a PP afeta o consumo de energia.

Tanto [17] e [18] buscavam minimizar o tempo de produção e o custo total da linha de produção, considerando um ambiente com apenas uma máquina e vários produtos diferentes. Porém, reduzir o custo total pode significar reduzir a produção de forma geral, não sendo a modelagem mais indicada para este problema. O mais correto seria reduzir o preço médio por quilo de malha e não o custo total da produção. Além disto, ambos trabalhos consideram apenas um máximo de duas máquinas em paralelo.

A questão de utilização da análise multicritério para ranqueamento das soluções ótimas de Pareto, obtidas pela otimização multiobjetivos, somente apareceu em 2016, em [19]. Neste trabalho foram aplicados diversos algoritmos multi-objetivos (NSGA-II, SPEA2 e VEGA), juntamente com a técnica multicritério TOPSIS. Nesses casos, o objetivo era minimizar os custos de um sistema, a carga de trabalho das máquinas, e o tempo de produção ao mesmo tempo que atendia as demandas dos clientes. Os algoritmos foram testados com um sistema acadêmico com apenas cinco máquinas em paralelo.

Mais recentemente, [20] e [21] apresentaram uma abordagem multiobjetivos para resolver os problemas de DL e PP. Enquanto [20] propuseram uma solução híbrida utilizando NSGA-II e SPEA2, para resolver o problema de uma fábrica de lâminas para turbinas de aviões; [21] apresentaram uma modificação do algoritmo de colônia de abelhas aplicado a uma fábrica de ônibus. Em ambos os trabalhos não são aplicados métodos multicritérios para ranqueamento das soluções. Além disto, o modelo é testado em um sistema de produção com apenas dez máquinas em paralelo.

Neste trabalho busca-se resolver os problemas de PP e DL em uma indústria de beneficiamento de malhas, utilizando Algoritmos Genéticos Multiobjetivos, considerando-se três objetivos a serem otimizados: peso total programado, custo médio por quilograma de malha produzida e distância para o estoque regulador. São

consideradas 24 máquinas em paralelo e o ranqueamento das soluções ótimas de Pareto é feito utilizando-se os pesos ROC.

3 OTIMIZAÇÃO MULTIOBJETIVOS

No caso estudado por esse trabalho, a empresa tem demonstrado dificuldades de sinergia entre o departamento industrial e comercial. Os dados de produção e vendas indicam que os produtos produzidos podem não ser necessários para atender a demanda do comercial que requer produtos que não estão sendo produzidos. Dessa forma, ao formalizar os processos de PP e DL de forma conjunta, busca-se minimizar esses problemas. O processo de decisão de uma empresa pode se tornar complexo, quando os problemas são conflitantes, o bom desempenho em um critério implica, na maioria dos casos, uma avaliação negativa em outro critério [22].

No caso de um problema de otimização multiobjetivo, procura-se encontrar uma solução ótima, de compromisso entre o conjunto de soluções e que envolvem todos os objetivos de um problema, buscando uma solução ótima no sentido de Pareto [23]; [24]. O resultado de um algoritmo multiobjetivo, será a apresentação de várias soluções ótimas, sendo necessário ao decisor escolher uma solução de compromisso que é mais alinhada aos objetivos da empresa [25]. Nesses casos, pode-se utilizar um modelo de decisão multicritério, que considera a opinião dos gestores da empresa.

Constatada a necessidade de aplicação de uma abordagem multiobjetivos ao problema e de uma metaheurística para otimização, pois existem objetivos conflitantes, partiu-se para a escolha do algoritmo a ser utilizado. No levantamento do estado da arte do problema, identificou-se que o algoritmo NSGA-II tem sido aplicado em problemas correlatos com grande sucesso, como em [17]; [18]; [19] e [20].

Um AGMO consegue tratar problemas que são conflitantes por natureza e um dos algoritmos que faz isso é o NSGA-II. O NSGA-II é uma modificação do modelo tradicional do AG, onde é utilizado um procedimento de ordenação de soluções não dominadas (*fast nondominated sorting*), que busca manter o elitismo de uma solução, e utiliza um operador para calcular a concentração da população (*crowding distance*), o qual foca na diversidade das soluções [23]. O objetivo é encontrar um grupo de soluções que esteja na fronteira Pareto-ótima e mais diversificado possível [29]. Segundo [24], NSGA-II tem sido amplamente utilizado para tratar diferentes problemas com múltiplos objetivos a serem otimizados, apresentando grande eficiência.

4 MODELO DE DECISÃO MULTICRITÉRIO

O processo de tomada de decisão de uma empresa pode se tornar complexo quando se tem vários critérios ou objetivos a serem considerados, e eventualmente a serem otimizados. Um modelo de decisão multicritério lida com problemas que, geralmente, possuem critérios conflitantes e são complexos quando considerado a escalabilidade de um problema [25]. Um desempenho bom em um critério, na maioria dos casos, implica em uma baixa avaliação em outro critério, sendo necessário uma forma de lidar com múltiplas variáveis e buscando soluções viáveis [22]. Nestes casos, deve-se buscar uma solução de compromisso que seja a melhor para a

empresa, considerando a importância de cada critério para a empresa.

Desta forma, um dos modelos multicritério que consideram a opinião dos administradores é o *rank order centroid weights* (ROC). Esse modelo, proposto por [26], solicita a um gestor para fazer um ranque de prioridade entre os critérios e com base nesse ranque calcula o peso que cada critério vai ter na função de avaliação de uma solução.

Há várias formas de agregação dos desempenhos locais de uma alternativa em uma medida de desempenho global, porém, a mais utilizada é a função de agregação aditiva ponderada, demonstrada na equação (1) [27].

$$V(a) = \sum_{i=1}^n w_i \cdot v_i(a) \quad (1)$$

Onde $V(a)$ é o valor do desempenho global da alternativa a , $v_i(a)$ é o valor local (parcial) de a no i -ésimo critério, w_i é o peso do i -ésimo critério ($w_i \geq 0$), n é o número total de critérios e $\sum_{i=1}^n w_i = 1$. Os valores de $v_i(\cdot)$ devem ser normalizados, de forma que o valor local de uma alternativa no melhor nível seja unitário (ou 100%), e no pior nível seja nulo (ou 0%), tanto para critérios de custo como os de benefícios. Uma vez obtidos os desempenhos globais, pode-se obter um ranque das pontuações onde a melhor alternativa é aquela que maximiza o valor da função de agregação.

Os pesos ROC são calculados a partir do vértices do simplex \mathcal{S}_n , definido por $w_1 \geq w_2 \geq \dots \geq w_n \geq 0$ e restrito por $\sum_{i=1}^n w_i = 1$, cujos vértices são $\mathbf{e}_1 = (1, 0, \dots, 0)$, $\mathbf{e}_2 = (1/2, 1/2, 0, \dots, 0)$, ..., $\mathbf{e}_n = (1/n, 1/n, 1/n, \dots, 1/n)$. Os pesos são determinados pelas médias das coordenadas correspondentes aos vértices que definem o centroide. De forma geral, para o i -ésimo atributo mais importante, o peso centroide é calculado pela equação (2).

$$w_i(ROC) = \frac{1}{n} \sum_{j=1}^n \frac{1}{j}, \quad i = 1, \dots, n \quad (2)$$

O ranqueamento com pesos baseados no centroide foram comparados com outros modelos por [28], e foi demonstrado, com base nas simulações, que esse modelo converge para a melhor alternativa em 75% a 87% dos casos.

5 MATERIAIS E MÉTODO

5.1 Modelo de Negócio

O experimento foi realizado em uma empresa que possui 12 processos industriais definidos, mas o foco principal de PP e DL é no momento de tingimento de malhas. A operação de tingimento de malhas possui um PP e DL mais complexo se comparado as outras atividades, uma vez que a definição das cores e a quantidade que deve ser produzida durante um período acontece nesse processo. Nesse caso, quando se considera somente as máquinas de tingir, é possível definir que a classificação de PP que se assemelha ao modelo é a do *Job Shop*.

A indústria possui 24 máquinas de tingir, categorizadas conforme capacidade de produção, sendo assim divididas: 14

equipamentos de 200kg, 6 equipamentos de 400kg e 4 equipamentos de 1000kg.

Qualquer ordem de produção (OP), passa por somente uma máquina e a escolha é feita considerando o tamanho do lote. Ao adicionar a malha na máquina de tingir, é informado qual a cor que deve ser tingida, conforme a PP. Após a seleção da cor, o armazém de corantes busca automaticamente o corante para tingimento, a cozinha industrial seleciona os produtos químicos e as quantidades adequadas para aquela receita. A máquina começa a trabalhar para garantir que os produtos químicos serão aplicados por toda a malha. No fim do processo industrial, os lotes produzidos são divididos em rolos de malha com peso médio de 20kg, essas peças são pesadas e embaladas.

Um problema recorrente que acontece na empresa é o conflito de interesse entre o departamento industrial (DI) e o departamento comercial (DC). O DI tem como objetivo obter a maior quantidade em kg de produção, enquanto o DC quer os produtos estejam sempre disponíveis para atender demanda futura.

No modelo comercial atual, um pedido é recebido pelo DC, e então é conferido se o produto existe em estoque. Se o produto existe em estoque, o produto é enviado ao cliente e o pedido é fechado. Se o produto não existe em estoque, uma OP é gerada, encaminhado ao DI para avaliação. O DI define em que momento essa OP será agendada para produção, e após a produção, o pedido é enviado ao cliente. Este procedimento está representado na Figura 1.

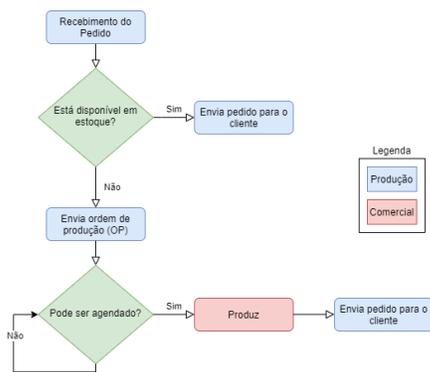


Figura 1 - Modelo de negócio atual

Desta forma, esse estudo propõe a utilização de um AGMO para balancear os objetivos dos dois departamentos e otimizar a relação entre DI e DC. Adicionalmente, propõe-se a criação de um estoque mínimo para os produtos.

5.2 Dimensionamento de Lote

No modelo atual de negócio, não existe informação suficiente para calcular o custo de manter um item em estoque. Da mesma forma, o cálculo de juros e depreciação não se aplica ao produto de malha de viscoso. Adicionalmente, o foco desse trabalho não é, necessariamente, buscar o menor custo de armazenamento, mas sim buscar a quantidade de produto que deve estar disponível em estoque para atender as demandas comerciais.

A quantidade ideal de itens a serem produzidos por variante (Q_{var}^*), depende de dados que devem ser extraídos do banco de dados. O primeiro dado é o estoque atual E_a , que busca fornecer o

somatório do peso de todas as peças em estoque, por variante. O segundo passo é recuperar informações referentes aos pedidos em aberto, P_a , isso acontece ao somar a quantidade de peso de cada pedido por cor.

A terceira variável é referente a demanda, e é o estoque mínimo E_m . O estoque mínimo foi feito avaliando-se todas as vendas, por variante, do ano de 2019, calculando-se uma média mensal de vendas, por variante e por mês, e então definido que esse valor seria o estoque mínimo de uma cor.

$$Q_{var}^* = \sum E_a - \sum P_a - E_m \quad (3)$$

Para realizar o cálculo de quantidade ideal de itens a serem produzidos, se faz a subtração do E_a pelo P_a e se tem o que seria a posição do estoque no futuro, ou seja, quando os pedidos forem faturados. O resultado desse cálculo é subtraído do E_m , e nesse momento é obtido a quantidade ideal de itens a serem produzidos.

5.3 AGMO

5.3.1 Design do Indivíduo

O indivíduo é a base do AGMO, e neste trabalho ele possui uma identificação (id), uma programação de produção (*scheduling*), a quantidade total de peso produzido na programação (*total_weight*), o custo médio por quilograma apresentado pela solução (*average_cost_per_kg*), e a distância que a solução proposta está do estoque regulador ideal (*total_distance_to_minimum_stock*). Adicionalmente, as variáveis auxiliares do NSGA-II também estão inseridas nessa classe.

A PP é a variável mais importante do algoritmo e contém todas as informações necessárias para gerar um agendamento para o DI. O DI necessita apenas da identificação da cor e qual máquina utilizar para gerar uma PP de um produto. A Figura 2 apresenta uma visualização das variáveis e seus tipos.

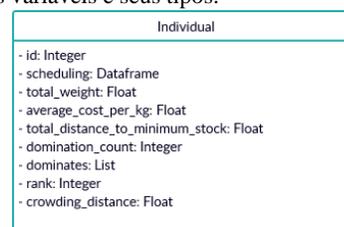


Figura 2 - Classes do indivíduo

A criação de uma PP é uma tarefa complexa, por manipular diferentes conjuntos de dados. Para a PP, é necessária uma lista de máquinas ativas, lista de cores liberadas para a produção, e uma análise de quanto tempo de produção é necessário para atender a demanda comercial. Todos esses dados foram extraídos dos bancos de dados da empresa pesquisada.

Para a PP, é definido um tempo de produção igual ao utilizado pela indústria pesquisada, desconsiderando eventuais paradas para manutenção. A produção começa a trabalhar no domingo às 22 horas e continua, ininterruptamente, até sábado às 13 horas. Então esse estudo parte do princípio que todas as máquinas ativas trabalham 7800 minutos por semana.

A PP é criada ao iterar sobre todas as máquinas ativas e adicionando cores, aleatoriamente, até que o tempo de produção da máquina seja atingida. Para cada máquina, em uma lista de máquinas ativas, e enquanto a variável de tempo de produção não for maior que o tempo produtivo da máquina; o algoritmo seleciona uma cor aleatória em uma lista de cores disponíveis. Posteriormente, uma busca sobre dados de produção histórica, envolvendo a máquina e a cor é feita. Se não existe informação sobre a cor e a máquina, uma nova pesquisa é feita buscando por dados da cor e de máquinas do mesmo grupo, nesse caso, assume-se que máquinas de um mesmo grupo vão ter dados de peso, custo e tempo parecidos. Se não existe informação sobre a cor e máquina, uma nova cor aleatória é selecionada e o procedimento é repetido.

O resultado desse algoritmo é a criação de uma programação de produção em uma estrutura de *dataframe*, conforme apresentado na Figura 3, que permite a manipulação de dados de forma semelhante a utilizada pelo departamento industrial. Após a criação, as variáveis de avaliação do indivíduo são calculadas, demonstrando um ponto comparativo para saber se a solução proposta é boa ou não.

```

for machine in list_machines:
    while time <= production_time:
        random_color = get_random_color()
        production_info = search(machine, random_color)
        if not production_info.empty:
            time += production_info['time_in_minutes']
            weight = production_info['weight']
            cost_per_kg = production_info['cost_per_kg']
            time_in_minutes = production_info['time_in_minutes']
            schedule = [(machine, color_id, weight, cost_per_kg,
            time_in_minutes)]
            scheduling.append(schedule)
        calculate()
    return scheduling

```

Figura 3 - Criação de uma programação de produção

A função *calculate()* somará todo o peso produzido do PP gerado, vai fazer a média do custo por quilograma e realizar o cálculo da distância da solução proposta para o estoque regulador ideal. Utilizando o *dataframe* criado para o PP, é possível realizar o somatório do peso ao chamar a função *pandas.DataFrame.sum()*, essa função simplifica e automatiza esse processo. O mesmo pode ser feito com relação a média de custo, nesse caso, se utiliza a função *pandas.DataFrame.mean()*. diferente, e é nesse momento que o dimensionamento de lote atua.

5.3.2 Funções Objetivos

Com o projeto do indivíduo realizado, se vê a necessidade de explicitar quais são as variáveis que ditaram a evolução da solução e como serão usadas na avaliação do algoritmo. De forma geral, busca-se maximizar a quantidade da produção, minimizar o custo de produção e atender a demanda. Nesse sentido, três variáveis são utilizadas para avaliar a solução: peso total programado, custo médio por quilograma de malha produzida, e distância para o estoque regulador (DER).

- Peso total programado (PTP)

O PTP é um objetivo que calcula o total, em peso, de malha que será produzida se a programação de produção de um indivíduo for executada. Esse objetivo é simples por natureza, e está diretamente relacionado aos desejos do Departamento Industrial, onde a maior quantidade de peso é a melhor solução.

Com base na variável programação de produção (*total_weight*), é feito a soma de todos os pesos e de todas as variantes de uma determinada PP. Essa soma é feita utilizando uma função da biblioteca *Pandas*, chamada *pandas.DataFrame.sum()*, que faz a iteração com todas as linhas da PP e soma todos os pesos.

- Custo médio por quilograma (CMQ)

O custo médio por quilograma busca controlar os custos industriais de produção da empresa pesquisada, e tem como base a PP gerada em uma solução. Nesse estudo, busca-se minimizar o custo de produção. Esse objetivo pode ou não ser oposto ao PTP, uma vez que a maior quantidade produtiva pode não significar um custo mínimo.

A variável utilizada para esse cálculo é o custo médio por quilograma (*average_cost_per_kg*), onde a média de todas as variantes de uma determinada PP é feita. A média é calculada utilizando a função *pandas.DataFrame.mean()*, da biblioteca *Pandas*. Com esse cálculo, é possível determinar o custo médio da PP aleatória gerada e avaliar as futuras gerações.

- Distância para estoque regulador (DER)

A distância para o estoque regulador (DER) é uma função que considera a quantidade ideal de itens a serem produzidos, conforme a equação (4), em relação a PP gerada aleatoriamente pelo AGMO. A DER é uma função que pega a variável peso total (P) da programação de produção recém-criada e subtrai da quantidade ideal Q_{var}^* de todas as variantes a ser produzida que o dimensionamento de lote forneceu, sendo representada pela equação (4).

$$DER = P - \sum Q_{var}^* \quad (4)$$

Quanto mais próximo de 0 for o resultado dessa equação, melhor é a solução quando o assunto é essa função objetivo, ou seja, neste problema de otimização, buscar-se-á a minimização do desvio DER; se o resultado for negativo, significa que a PP aleatória deixou de suprir a demanda de alguma cor; se o resultado for positivo, significa que a PP produziu uma quantidade superior ao que a demanda necessitava.

5.3.3 Restrições

As restrições que são implementadas nesse trabalho referem-se principalmente ao processo de tingimento de uma malha. O processo químico, de aplicação de uma cor em uma malha, gera sujeira dentro do equipamento, que dependendo da cor subsequente força uma limpeza completa no equipamento. Portanto, o tempo/custo de setup de um equipamento pode ser consideravelmente reduzido se o AGMO considerar e evoluir conforme essa restrição.

Um exemplo dessa restrição seria se, a máquina M22 produzir a cor PRETO e a próxima cor a ser produzida na mesma máquina é a cor BRANCO. Com base na experiência industrial dos especialistas, existe uma alta probabilidade do produto da cor BRANCO ser manchado pela sujeira que a cor PRETO deixou na

máquina. Logo, para a cor BRANCO ser produzida após a cor PRETO, é necessário uma limpeza da máquina que pode levar até 4 horas, envolve 3 funcionários da manutenção e produtos químicos de limpeza.

Entretanto, é possível ir da cor PRETO até a cor BRANCO se existir um degradê de cores produzidas, da mais escura para a mais clara, até chegar no BRANCO. Nesse caso, dado um tempo de produção, vai ser possível produzir PRETO e BRANCO na mesma máquina, sem qualquer custo de setup, desde que outras cores sejam produzidas entre elas.

A tarefa de determinar a sequência das cores a serem produzidas requer o auxílio integral do gerente de produção, que com base em sua experiência e conhecimento químico, orienta sobre quais cores podem ser produzidas em sequência e quais não podem. Supõem-se que se as restrições forem aplicadas ao AGMO e ele conseguir evoluir seguindo-a, o resultado da programação de produção vai ser significativamente diferente da forma como funciona atualmente.

5.3.4 População

A população, no algoritmo, é representada por uma lista de indivíduos (*individuals*), uma variável que define a quantidade de indivíduos que a população vai ter (*n_individuals*), e uma que apresenta os indivíduos como um dataframe (*individuals_df*). Métricas da população são gerados posteriormente, para que possa comparar se a evolução das espécies, está realmente acontecendo.

A classe população ainda contém funções auxiliares, que criam uma população (*new*), que transforma uma lista de indivíduos em dataframe (*individuals_asdf*), que busca um indivíduo na população (*search*), e que verifica se existem indivíduos duplicados na população (*isDuplicated*). Com a utilização de dataframes e estrutura de classes para um indivíduo, é possível acessar qualquer indivíduo em qualquer momento durante a execução do código. Neste estudo foram realizados testes com populações iniciais de 10 a 100 indivíduos.

5.3.5 Particularidades do NSGA-II

O NSGA-II é o algoritmo multiobjetivo adotado por esse estudo, e o desenvolvimento dele se dá ao criar estratégias para determinar a não dominância de uma solução e para calcular a concentração da população, garantindo o elitismo e a diversidade.

- Ordenação por não dominância

O fator de não dominância é feito ao comparar todas as soluções, avaliando as três funções objetivos. O estudo busca minimizar os custos de produção, maximizar a quantidade produzida, e deixar o DER o mais próximo possível de 0. Esses objetivos são conflitantes por natureza de forma que a maior quantidade de produção não significa o melhor custo, da mesma forma o melhor custo não significa o melhor DER.

De forma geral, sabendo que um indivíduo I_i vai ser comparado com todos os outros indivíduos da população I_j , a dominância é dada pela equação (5). Em um primeiro momento, é comparado se o indivíduo é melhor ou igual para todas as funções objetivos. Posteriormente, é avaliado se o indivíduo é melhor em pelos menos uma função objetivo. Se a resposta for sim para os dois critérios, considera-se que o indivíduo I_i não é dominado pelo indivíduo I_j .

$$D = (CMQ_i \leq CMQ_j \wedge DER_i \leq DER_j \wedge PTP_i \geq PTP_j) \quad (5)$$

A implementação da não dominância nesse trabalho, utiliza uma classe que tem como objetivo criar as fronteiras Pareto-ótima, zerar essas fronteiras e as variáveis de dominância quando necessário, e calcular o fator de dominância. Essa classe é utilizada durante todo o processo do AGMO, e é usada em todas as gerações quando é necessário aproveitar do elitismo e zerar as informações de dominância de um indivíduo que foi selecionado para estar presente na próxima geração.

- Operador de comparação de concentração

O operador de comparação de concentração é implementado de forma a utilizar as capacidades de um *dataframe*, de ordenar o conjunto de dados por coluna e reordenar conforme necessidade. Uma classe chamada *CrowdingDistance* foi criada, e o primeiro passo foi implementar uma função que normaliza as colunas que se referem aos objetivos do trabalho. A informação sobre a fronteira em que o indivíduo está contido está presente na variável *ranque*, dentro da classe *indivíduo*, e então uma lista contendo os membros de uma fronteira é verificada para serem atribuídos os valores. O *dataframe* é ordenado em forma ascendente pela variável que se refere ao PTP, calculando assim a densidade da população para todas as fronteiras. Após isso, o mesmo processo é repetido para CMQ e DER. O resultado é um dataframe que contém todas as variáveis, *crowding_distance*, diferente de nulo.

- Operador seleção

Este trabalho utiliza o operador de seleção por torneio, junto com o operador de comparação de concentração para selecionar os indivíduos para reprodução. Dessa forma, considerando que o elitismo é controlado pela ordenação de não dominância, dois indivíduos são selecionados aleatoriamente, comparados usando o comparador de concentração, e retornando o melhor indivíduo, entre os dois, para a reprodução.

O operador de seleção está implementado na classe chamado *GeneticAlgorithm*, que é instanciada com a população da geração atual. A função de seleção é chamada de *tourney_selection()*, e recebe somente a quantidade de indivíduos que vão participar do torneio, sendo que por padrão 2 indivíduos participam do torneio.

- Operador Crossover

A estratégia de crossover utilizada nessa implementação, foi a de selecionar um ponto no material genético dos pais e criar um indivíduo com a quantidade proporcional de material de cada pai. A implementação do operador de *crossover* é semelhante a implementação de um novo indivíduo, principalmente no que se refere a criação da programação de produção. A principal diferença é que ao invés de selecionar cores aleatórias, o crossover pega a parte da programação existente do pai1 e a outra parte do pai2. A Figura 4 representa a operação de *crossover* entre dois indivíduos.

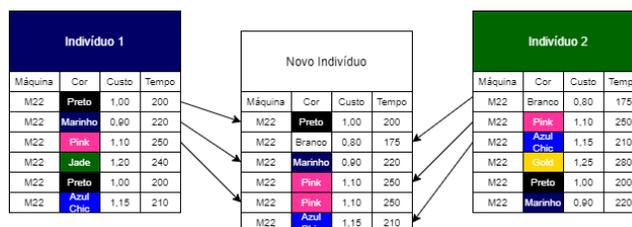


Figura 4 - Operador crossover

Além de todas as variáveis contidas na classe *GeneticAlgorithm*, a função de *crossover* recebe dois indivíduos, o *parent1* e o *parent2*. Para cada máquina em uma lista de máquinas ativas, o operador de *crossover* vai pegar a programação de produção dos pais e armazenar nas variáveis *p1* e *p2*. Em seguida, são selecionadas, para aquela máquina, uma programação do *p1* e uma programação do *p2*. Os tempos de produção vão ser computados e, se o tempo não for maior que o tempo de produção da máquina, o procedimento será repetido. No fim desse *loop*, uma nova PP vai ter sido gerada com base nos dois pais. Um novo indivíduo é criado e é atribuído a nova PP para ele, e a função de *calculate* é invocada para atribuir os valores das funções objetivos.

- Operador mutação

O operador de mutação tem como objetivo uma mudança aleatória em um indivíduo, e nesse estudo isso é feito ao alterar a PP. Inicialmente é necessário aplicar a seleção por torneio para buscar um indivíduo na população atual que será a base para o indivíduo novo. O algoritmo então gera um número aleatório *n* de 5 a 50, indicando que a quantidade mínima de alteração em um indivíduo é de 5 PP, e o máximo de 50. Em seguida, são selecionadas aleatoriamente *n* programações que vão ser substituídas por novas. O procedimento é demonstrado na Figura 5.

Indivíduo 1				Novo Indivíduo			
Máquina	Cor	Custo	Tempo	Máquina	Cor	Custo	Tempo
M22	Preto	1,00	200	M22	Gold	1,25	280
M22	Marinho	0,90	220	M22	Marinho	0,90	220
M22	Pink	1,10	250	M22	Pink	1,10	250
M22	Jade	1,20	240	M22	Pink	1,10	250
M22	Preto	1,00	200	M22	Preto	1,00	200
M22	Azul Chic	1,15	210	M22	Azul Chic	1,15	210

Figura 5 - Operador mutação

A seleção aleatória do que será modificado é realizada pela própria biblioteca Pandas, através da função *pandas.DataFrame.sample(n)*. Apesar do exemplo na Figura 4 demonstrar a seleção de apenas uma máquina, a mutação não seleciona por máquinas, pegando qualquer programação de produção no conjunto de dados.

- Gerações

As gerações de um NSGA-II funcionam de forma diferente de um AG normal. Uma população de indivíduos aleatórios P_0 de tamanho N é criada e ranqueada conforme descrito anteriormente. Uma população “filha” Q_0 é criada utilizando os operadores de seleção, crossover e mutação descritos acima. As populações P_0 e Q_0 são combinadas, formando uma população de tamanho $2N$ denominada R_0 . Essa população R_0 , é então ranqueada seguindo a ordenação por não dominância, e o operador de comparação de concentração. Considerando que R_0 está devidamente ranqueada, é selecionado os primeiros N indivíduos para fazer parte da população P_1 , que marca a próxima geração. Esse procedimento é repetido até que o algoritmo alcance uma condição de parada, que neste estudo é o número de gerações.

5.3.6 Modelo Multicritério

A tomada de decisão e a conciliação entre o departamento industrial e comercial é algo que precisa ser definido pela diretoria da empresa, dessa forma, tanto os objetivos quanto o ranqueamento deles foram decisões tomadas pelo diretor administrativo da indústria, que em suas atribuições, é responsável pela definição da estratégia adotada pela empresa.

Nesse sentido, o diretor estabeleceu que a prioridade é: (i) minimizar o custo médio por quilograma (CMQ), (ii) minimizar o desvio da distância para o estoque regulador (DER) e (iii) maximizar o peso total programado (PTP).

Sabendo que a equação (6) descreve como o peso baseado no centroide é calculado, e o ranque estabelecido pelo diretor da empresa, foi possível estabelecer qual a prioridade entre os critérios, que foram calculados nas equações abaixo.

$$CMQ = \frac{1 + \frac{1}{2} + \frac{1}{3}}{3} = \frac{11}{18} = 0,611$$

$$DER = \frac{\frac{1}{2} + \frac{1}{3}}{3} = \frac{5}{18} = 0,278$$

$$PTP = \frac{\frac{1}{3}}{3} = \frac{1}{9} = 0,111$$

6 RESULTADOS

Foram realizados diversos testes envolvendo diferentes valores de população inicial, número de gerações e taxa de mutação. Neste artigo serão apresentados apenas os resultados relativos ao AGMO com 100 indivíduos, 100 gerações e taxa de mutação de 5%, os quais produziram um melhor desempenho.

As restrições aplicadas ao algoritmo impactam as funções objetivos avaliadas por esse trabalho. Nesse sentido, quanto maior o número de restrições, maior vai ser o custo e menor a quantidade produzida. Foi definido, na modelagem do AGMO, que para cada vez que uma restrição fosse identificada, ela sofresse uma punição de 10 vezes o custo e que adicionasse um tempo de produção igual a 250 minutos, sendo estes valores definidos pelos especialistas da empresa. Ao aplicar essa solução foi possível perceber que a otimização das funções objetivo acabam gerando um resultado positivo nas restrições. Este cenário é demonstrado na Figura 6, sendo possível observar que o custo das restrições tende a cair conforme o algoritmo evolui.

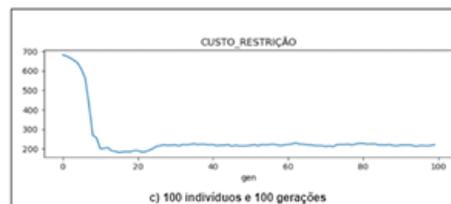


Figura 6 - Restrição por indivíduo e gerações

Outro experimento realizado com relação à mudança da ordem de preferência dos critérios, o qual impacta nos pesos ROC e consequentemente no ranking das soluções. Na tabela 1, são apresentados os resultados obtidos nas soluções ranqueadas pelo multicritério, com destaque para o teste 2, o qual manteve a ordem de preferência indicada pelos especialistas da empresa. Cabe

destacar que todas as soluções apresentadas na tabela 1 são consideradas ótimas, no sentido de Pareto, sendo geradas pelo algoritmo NSGA-II.

Tabela 1 - Ranqueamento das soluções ótimas

Teste	1º Obj	2º Obj	3º Obj	CMQ	PTP	DER
1	CMQ	PTP	DER	0,75	618.172,32	167.489,56
2	CMQ	DER	PTP	0,71	447.625,03	3057,73
3	PTP	CMQ	DER	0,75	618.172,32	167.489,56
4	PTP	DER	CMQ	0,75	618.172,32	167.489,56
5	DER	PTP	CMQ	0,71	447.625,03	3057,73
6	DER	CMQ	PTP	0,71	447.625,03	3057,73

O resultado do teste 1 retrata o foco em custo e produção, sendo percebível que o estoque regulador é deixado de lado, em favor de uma produção em massa. Já no teste 2, que foi a solicitação do gestor da empresa, o foco é custo e DER, foi possível obter uma solução com um custo melhor e um DER mínimo, porém neste caso existe um sacrifício sobre o total de malha produzida.

7 COMENTÁRIOS FINAIS

A solução proposta pelo AGMO, considerando o período avaliado, melhorou todas as funções objetivos, apresentando uma solução melhor do que a utilizada pela fábrica. Essa solução apresentou um custo médio por quilograma 12% menor (R\$ 0,71 / kg), um peso total programado 10% superior (447.625,03 kg) e uma redução de 8% na distância para o estoque regulador (3057,73 kg). Esta solução fornecida pelo algoritmo NSGA-II foi a primeira do ranking gerado pelos pesos ROC, dado a ordem de prioridade definida pelos gestores: 1º) CMQ, 2º) PTP e 3º) DER. Esta é uma solução viável, a qual minimiza o custo médio por quilograma e a distância para o estoque regulador, maximizando a quantidade produzida de viscoso.

A empresa pesquisada tem um histórico de falta de sinergia entre o departamento comercial e industrial, fazendo com que o setor produtivo realize suas atividades sem considerar o setor comercial, e vice-versa. Com o modelo integrado de otimização multiobjetivos e modelo de decisão multicritério para ranqueamento das soluções, foi possível resolver os problemas complexos de Programação de Produção e Dimensionamento de Lote, os quais são classificados pela literatura como problemas NP-difícil, em uma indústria de beneficiamento de malha.

Como continuação desta pesquisa, pretende-se ampliar o conjunto de dados para testes, considerando uma base de dados mensal, pois no presente trabalho utilizou-se uma base semanal. Além disto, outros algoritmos AGMO serão aplicados, tais como NSGA-III e SPEA-II, buscando-se fazer uma comparação de desempenho com relação ao NSGA-II.

REFERÊNCIAS

[1] CLARK, A.; ALMADA-LOBO, B.; ALMEDER, C. Lot sizing and scheduling: industrial extensions and research opportunities. *International Journal of Production Research*, v. 49, p. 2457-2461, February 2011.
 [2] KORKULU, S.; BÓNA, K. Ergonomics as social component of sustainable lot-sizing: a review. *Periodica Polytechnica Social and Management Sciences*, v. 17, p. 1-8, 2019.

[3] ALFARES, H. K.; GHAITHAN, A. M. EOQ and EPQ Production-Inventory Models with Variable Holding Cost: State-of-the-Art Review. *Arabian Journal for Science and Engineering*, v. 44, p. 1737-1755, October 2018.
 [4] FAN, K.; ZHAI, Y.; WANG, M. Review and classification of hybrid shop scheduling. *Production Engineering*, v. 12, p. 597-609, June 2018.
 [5] BAKER, Kenneth R.; TRIETSCH, Dan.; *Principles of sequencing and scheduling*. 2. ed. Hoboken: John Wiley & Sons, 2019.
 [6] ONAR, S. C.; OZTAYSI, B.; KAHRAMAN, C.; YANIK, S.; SENVAR, O. In: TALBI, E-G.; YALAOUI, AMOEDO, L. (Editors). *Metaheuristics for Production Systems*. New York: Springer, 2016. p. 1-22.
 [7] JANS, R.; DEGRAEVE, Z. Meta-heuristics for dynamic lot sizing: A review and comparison of solution approaches. *European Journal of Operational Research*, v. 177, p. 1855-1875, March 2007.
 [8] BUSHUEV, M. A.; GUIFRIDA, A.; JABER, M. Y.; KHAN, M. A review of inventory lot sizing review papers. *Management Research Review*, vol. 68, p. 283-298, 2015.
 [9] ANDRIOLO, A.; BATTINI, D.; GRUBBSTROM, R. B.; PERSONA, A. A century of Evolution from Harri's basic lot size model: Survey and research agenda. *International Journal of Production Economics*, v. 155, p. 16-38, January 2014.
 [10] STOOP, P. P. M.; WIERS, V. C. S. The complexity of scheduling in practice. *International Journal of Operations & Production Management*, v. 15, n. 10, p. 37-53, 1996.
 [11] KHATAB, A.; DAHANE, D.; AGHEZZAF, E-H, AIT-KADI, D. Joint maintenance and production planning optimization model for production systems with operation-dependent failure. In: *Proceeding of 44th International Conference on Computers & IE*, 2014, p. 1-13. Turkey, 2014.
 [12] EMMONS, H.; VAIRAKTARAKIS, G. *Flow Shop Scheduling: theoretical results, algorithms, and applications*. Boston, USA: Springer Boston, 2013.
 [13] PINEDO, M. L. *Scheduling: theory, algorithms, and systems*. 5. ed. Cham, Switzerland: Springer Nature Switzerland, 2016.
 [14] HARRIS, F. W. How many parts to make at once. *Factory, The Magazine of Management*, v. 10, n. 2, p. 135-152, 1913.
 [15] TAFT, E. W. The most economical production lot. *The Iron Age*, v. 101, p. 1410-1412, 1918.
 [16] TOLEDO, C. F. M.; DE OLIVEIRA, L.; OLIVEIRA, R. R. R.; PEREIRA, M. R. Parallel genetic algorithm approaches applied to solve a synchronized and integrated lot sizing and scheduling problem. In: *Proceedings of the 2010 ACM Symposium on Applied Computing*, 2010. p. 1148-1152.
 [17] CAPA, C.; AKGUNDUZ, A.; DEMIRLI, K. A bi-objective genetic algorithm for economic lot scheduling problem under uncertainty. In: *Proceedings of the 2014 Industrial and Systems Engineering Research Conference*, 2014.
 [18] LIU, Y.; TIWARI, A. An investigation into minimizing total energy consumption and total completion time in a flexible job shop for recycling carbon fiber reinforced polymer. *Procedia CIRP*, v. 29, p. 722-728, May 2015.
 [19] ROHANINEJAD, M.; SAHRAEIAN, R.; NOURI, B. V. Multi-objective optimization of integrated lot-sizing and scheduling problem in flexible job shops. *RAIRO Operations Research*, v. 50, n. 3, p. 587-609, 2016.
 [20] ZHOU, Y.; YANG, J-J.; ZHENG, L-Y. Multi-agent based hyper-heuristics for multi-objective flexible job shop scheduling: a case study in an aero-engine blade manufacturing plant. *IEEE Access*, v. 7, p. 21147-21176, 2019.
 [21] YUE, L.; GUAN, G.; ZHANG, L.; ULLAH, S.; GUI, Y. Multi objective lotsizing and scheduling with material constraint in flexible parallel lines using a Pareto based guided artificial bee colony algorithm. *Computers & Industrial Engineering*, v. 128, p. 659-680, 2019.
 [22] GAL, T.; STEWART, T. J.; HANNE, T. *Multicriteria decision making: advances in MCDM models, algorithms, theory, and applications*. New York, USA, Springer New York, 1999.
 [23] DEB, K.; PRATAP, A.; AGARWAL, S. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEE Transactions on Evolutionary Computation*, v. 6, p. 182-196, April 2002.
 [24] MUSSOI, F. L. R. Modelo de decisão integrado para a priorização multiestágio de projetos de distribuição considerando a qualidade da energia elétrica. 2013. 387 f. Tese (Doutorado em Engenharia Elétrica) – UFSC, Florianópolis, 2013.
 [25] GHOSH, A.; MAL, P.; ABHIJIT, M. *Advanced Optimization and Decision-Making Techniques in Textile Manufacturing*. USA: CRC Press, 2019.
 [26] BARRON, F. H. Selecting a Best multi-attribute alternative with partial information about attribute weights. *Acta Psychologica*, n. 80, p. 91-103, 1992.
 [27] KEENEY, R. L.; RAIFFA, H. (1993). *Decisions with multiple objectives: preferences and value tradeoffs*. Cambridge: University Press, 1993.
 [28] TZENG, G. H.; HUANG, J. J. (2011). *Multiple attribute decision making: methods and applications*. United States: CRC Press, 349 p., 2011.
 [29] GHOSH, A.; MAL, P.; ABHIJIT, M. *Advanced Optimization and Decision-Making Techniques in Textile Manufacturing*. Boca Raton, USA: CRC Press, 2019.