v7.n10.2026

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

SISTEMA INTELIGENTE DE SEQUENCIAMENTO DE PRODUÇÃO BASEADO EM ALGORITMOS GENÉTICOS E *MACHINE LEARNING* PARA A INDÚSTRIA AERONÁUTICA

INTELLIGENT PRODUCTION SCHEDULING SYSTEM BASED ON GENETIC ALGORITHMS AND MACHINE LEARNING FOR THE AEROSPACE INDUSTRY

SISTEMA INTELIGENTE DE SECUENCIACIÓN DE LA PRODUCCIÓN BASADO EN ALGORITMOS GENÉTICOS Y MACHINE LEARNING PARA LA INDUSTRIA AERONÁUTICA

Diogo Marconato Garcia[1], Lucas Diniz Dallilo[2], Fabiana Florian[3]

e7119527
https://doi.org/10.47820/recima21.v7i11.9527

PUBLICADO: 11/2026

RESUMO

Este trabalho aborda o problema de definir a ordem de entrada das aeronaves na linha de montagem, decisão que afeta diretamente o tempo total de produção. A regra usualmente adotada, a ordem de chegada (FIFO), considera apenas datas e ignora pendências conhecidas, como falta de material e bloqueios de qualidade. O objetivo foi desenvolver e avaliar, em ambiente experimental, um sistema que combina um algoritmo genético, responsável por buscar a ordem de menor tempo total, e um modelo de aprendizado de máquina, que estima o risco de atraso de cada aeronave e é incorporado à função de avaliação do algoritmo. O sistema foi avaliado sobre 351 ordens de produção reais, distribuídas em 11 aeronaves. Por limitações da base disponível, o tempo de cada etapa foi aproximado pela quantidade de ordens abertas, e o modelo preditivo foi treinado com respostas de atraso geradas por regras do processo, e não com atrasos efetivamente observados. Nessas condições, a ordem sugerida reduziu em cerca de 38% o tempo total estimado em relação à ordem de chegada, e o componente preditivo reduziu em aproximadamente 14% a exposição da fila ao risco, sem aumento significativo do tempo total. Os resultados demonstram o potencial da combinação das duas técnicas, mas têm caráter experimental e não comprovam o desempenho preditivo nem os ganhos em condições reais de produção, o que exige validação com tempos-padrão e atrasos históricos.

PALAVRAS-CHAVE: Algoritmo genético. Aprendizado de máquina. Indústria aeronáutica. Inteligência artificial. Otimização de produção. Sequenciamento de produção.

ABSTRACT

This study addresses the problem of defining the order in which aircraft enter the assembly line, a decision that directly affects total production time. The commonly adopted rule, arrival order (FIFO), considers only dates and ignores known issues such as material shortages and quality holds. The objective was to develop and evaluate, in an experimental setting, a system that combines a genetic algorithm, which searches for the order with the shortest total time, and a machine learning model, which estimates the delay risk of each aircraft and is incorporated into the algorithm's fitness function. The system was evaluated on 351 real production orders covering 11 aircraft. Owing to limitations of the available data, the time of each stage was approximated by the number of open orders, and the predictive model was trained on delay labels generated by process rules rather than on actually observed delays.

[1] Graduando em Engenharia da Computação pela Universidade de Araraquara (UNIARA), com atuação em desenvolvimento de sistemas e inteligência artificial.
[2] Mestre em Ciência da Computação pela Universidade Federal de São Carlos (UFSCar), doutorando pela Universidade de São Paulo (USP) e docente da Universidade de Araraquara (UNIARA), na área de Engenharia de Software.
[3] Doutora em Alimentos e Nutrição pela Universidade Estadual Paulista (UNESP), docente da Universidade de Araraquara (UNIARA), com atuação em Economia e Metodologia Científica.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Under these conditions, the suggested order reduced the estimated total time by about 38% compared with the arrival order, and the predictive component reduced the queue's risk exposure by approximately 14%, with no significant increase in total time. The results demonstrate the potential of combining the two techniques, but they are experimental and do not prove predictive performance or gains under real production conditions, which requires validation with standard times and historical delays.

KEYWORDS: *Aircraft manufacturing. Artificial intelligence. Genetic algorithm. Machine learning. Production optimization. Production scheduling.*

RESUMEN
Este trabajo aborda el problema de definir el orden de ingreso de las aeronaves en la línea de montaje, decisión que afecta directamente el tiempo total de producción. La regla habitualmente adoptada, el orden de llegada (FIFO), considera solo fechas e ignora problemas conocidos, como la falta de material y los bloqueos de calidad. El objetivo fue desarrollar y evaluar, en un entorno experimental, un sistema que combina un algoritmo genético, encargado de buscar el orden de menor tiempo total, y un modelo de aprendizaje automático, que estima el riesgo de retraso de cada aeronave y se incorpora a la función de evaluación del algoritmo. El sistema se evaluó sobre 351 órdenes de producción reales, distribuidas en 11 aeronaves. Debido a limitaciones de la base disponible, el tiempo de cada etapa se aproximó por la cantidad de órdenes abiertas, y el modelo predictivo se entrenó con respuestas de retraso generadas por reglas del proceso, y no con retrasos efectivamente observados. En estas condiciones, el orden sugerido redujo en cerca del 38% el tiempo total estimado en relación con el orden de llegada, y el componente predictivo redujo en aproximadamente un 14% la exposición de la cola al riesgo, sin aumento significativo del tiempo total. Los resultados demuestran el potencial de la combinación de las dos técnicas, pero tienen carácter experimental y no comprueban el desempeño predictivo ni las ganancias en condiciones reales de producción, lo que exige validación con tiempos estándar y retrasos históricos.

PALABRAS CLAVE: *Algoritmo genético. Aprendizaje automático. Industria aeronáutica. Inteligencia artificial. Optimización de la producción. Secuenciación de la producción.*

INTRODUÇÃO

Montar uma aeronave envolve milhares de componentes e dezenas de etapas que dependem umas das outras. Quando uma etapa atrasa, o efeito se propaga para as etapas seguintes, deixa equipes e equipamentos parados e pode gerar multas contratuais (SLACK; BRANDON-JONES; JOHNSTON, 2018).

Decidir a ordem em que as aeronaves entram na linha é, portanto, uma decisão de alto impacto e também um problema de difícil solução, pois o número de ordens possíveis cresce rapidamente com a quantidade de aeronaves. Na Pesquisa Operacional, problemas com essa característica são classificados como NP-difíceis, isto é, não se conhece método capaz de encontrar a solução exata em tempo razoável quando o tamanho cresce (PINEDO, 2016).

Na prática, é comum que essa decisão siga a regra mais simples possível: atender na ordem de chegada, conhecida como FIFO (do inglês *First-In, First-Out*, ou primeiro a entrar,

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

primeiro a sair). Essa regra é fácil de aplicar, mas tem uma limitação séria: ela considera apenas datas e desconsidera se a aeronave tem material em falta, ordem travada ou pendência de qualidade.

A proposta deste trabalho é unir duas técnicas para superar essa limitação. A primeira é o algoritmo genético, que procura boas ordens de produção imitando o processo de evolução natural. A segunda é um modelo de aprendizado de máquina, que analisa a situação atual de cada aeronave e estima a probabilidade de ela atrasar. A união acontece no critério que o algoritmo usa para julgar se uma ordem é boa: além do tempo total, ele passa a considerar também o risco estimado, de modo que aeronaves problemáticas tendam a ser posicionadas mais cedo na fila.

Diante disso, o problema que orienta este trabalho é: a incorporação de uma estimativa de risco de atraso ao algoritmo genético permite definir ordens de produção que reduzam o tempo total e, ao mesmo tempo, antecipem as aeronaves mais problemáticas? A pesquisa justifica-se porque as informações sobre pendências já estão registradas no sistema corporativo, mas não são aproveitadas pela regra FIFO. O objetivo geral foi desenvolver e avaliar esse sistema. Como objetivos específicos, definiram-se: (i) tratar os dados de produção extraídos do sistema corporativo; (ii) implementar o cálculo do tempo total de produção; (iii) construir o modelo de risco e medir sua qualidade de previsão; (iv) verificar, por experimento, se o modelo de risco contribui para o resultado; (v) testar a sensibilidade do sistema aos seus parâmetros; e (vi) entregar o resultado em um painel utilizável pelo gestor de produção. A avaliação foi conduzida em ambiente experimental, com dados reais de ordens de produção, porém com tempos de etapa aproximados e respostas de atraso geradas por regras, condições detalhadas nas Seções 2.3 e 2.5.2 e discutidas na Seção 3.9.

1. REFERENCIAL TEÓRICO

1.1. Algoritmos genéticos aplicados ao sequenciamento

O algoritmo genético é um método de busca inspirado na evolução biológica, proposto por Holland (1975) e consolidado por Goldberg (1989). Parte-se de um conjunto de soluções candidatas (no caso, ordens de produção), selecionam-se as melhores, combinam-se partes delas e introduzem-se pequenas alterações aleatórias; com a repetição desse ciclo, as soluções tendem a melhorar.

Quando a solução é uma ordem, é preciso um cuidado extra: ao combinar duas ordens, a nova ordem não pode repetir nem omitir aeronaves. O operador PMX (*Partially Matched*

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Crossover) foi criado para garantir isso e é amplamente usado nesse tipo de problema (GOLDBERG; LINGLE JR., 1985). Trabalhos recentes confirmam o bom desempenho de algoritmos evolutivos em ambientes de manufatura complexos (CHANG; JIA; REN, 2025; LI *et al.*, 2025; WANG *et al.*, 2022).

Destaca-se também a heurística NEH, proposta por Nawaz, Enscore Jr. e Ham (1983), referência clássica para esse tipo de linha, que ordena as tarefas pelo tempo total de processamento e insere cada uma na posição de menor tempo parcial. Neste trabalho, ela inspira a ideia de fornecer ao algoritmo um ponto de partida de boa qualidade (Seção 2.4). A Tabela 1 compara essas estratégias com a proposta deste trabalho.

Tabela 1. Comparação entre estratégias de definição da ordem de produção

Estratégia	Como decide a ordem	Considera problemas da aeronave?	Tempo total obtido
FIFO (ordem de chegada)	Segue a data de entrada, sem nenhum cálculo.	Não.	Alto: gera esperas entre as etapas.
Regras simples de despacho	Escolhe pelo menor tempo local ou pela data mais próxima.	Não.	Intermediário: melhora pouco e estagna.
Heurística construtiva (NEH)	Insere cada aeronave na posição que minimiza o tempo parcial.	Não.	Baixo: referência clássica para tempo.
Algoritmo genético com risco (proposta deste trabalho)	Evolui ordens completas, avaliando tempo e risco juntos.	Sim.	Baixo, e ainda reorganiza a fila por risco.

Fonte: Elaborada pelos autores (2026).

1.2. Aprendizado de máquina aplicado à manufatura

O *Random Forest*, proposto por Breiman (2001), é um modelo de previsão formado por várias árvores de decisão, isto é, sequências de perguntas simples sobre os dados (por exemplo, “esta aeronave tem mais de três ordens travadas?”). A combinação dos resultados de todas as árvores torna o modelo mais estável do que uma árvore isolada e, além da classificação, fornece uma probabilidade, que é a informação utilizada neste trabalho.

Uma revisão recente mostra que técnicas de aprendizado de máquina vêm sendo aplicadas em diversas frentes da engenharia aeronáutica, como aerodinâmica, projeto de aeronaves e monitoramento da condição de motores (WANG; MA, 2024), o que indica maturidade crescente dessas ferramentas no setor.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

1.3. Cuidados na avaliação de um modelo de previsão

Dois cuidados são essenciais na avaliação de um modelo. O primeiro é evitar o vazamento de dados (*data leakage*), que ocorre quando o modelo tem acesso, no treinamento, a informações indisponíveis no momento real da previsão, o que produz desempenho excelente nos testes e insatisfatório na prática (KAUFMAN *et al.*, 2012). Um caso particular ocorre quando a resposta a ser prevista é calculada por uma fórmula a partir das próprias variáveis de entrada, situação em que o modelo apenas redescobre a fórmula, caso tratado na Seção 2.5.2.

O segundo é a forma de medir o acerto. Avaliar o modelo com os mesmos dados do treinamento produz resultado otimista, pois o modelo já conhece aquelas respostas. Recomenda-se, por isso, a validação cruzada, na qual os dados são divididos em partes e o modelo é treinado em algumas e testado na restante, até que todas tenham servido de teste (KOHAVI, 1995), procedimento adotado neste trabalho (Seção 2.5.3).

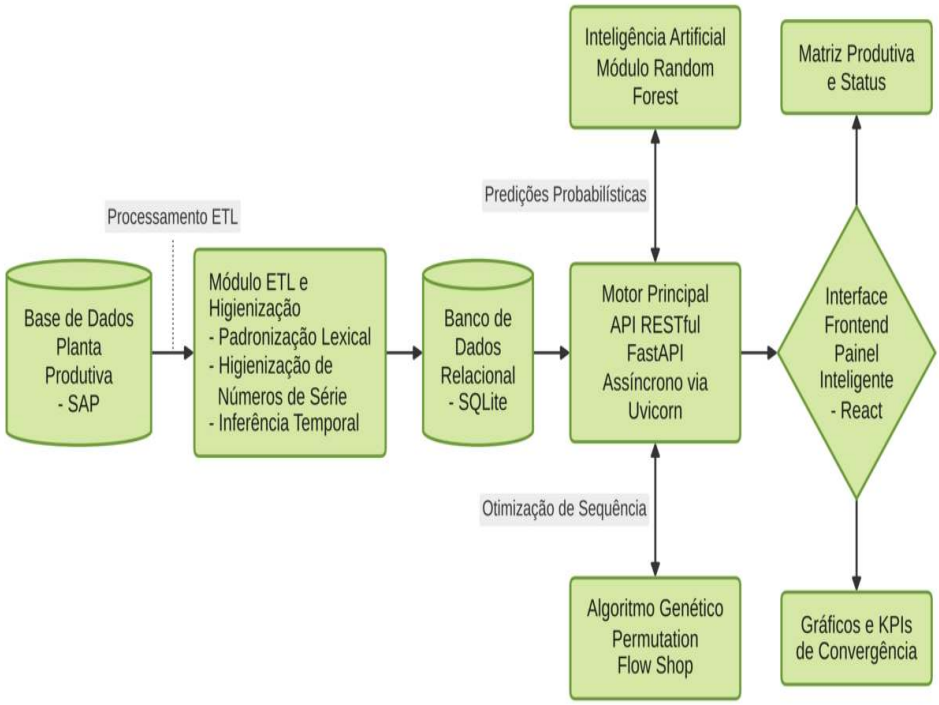
2. METODOLOGIA

2.1. Arquitetura do sistema

O sistema foi construído em Python e organizado em quatro partes: uma rotina que lê e trata os dados de produção; um banco de dados local (SQLite) que armazena o resultado; um servidor que executa os cálculos e responde às solicitações, feito com FastAPI (FASTAPI, 2023); e um painel visual construído em React, por meio do qual o gestor acessa os resultados. O modelo de previsão foi implementado com a biblioteca scikit-learn (PEDREGOSA *et al.*, 2011). A Figura 1 resume essa arquitetura.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 1. Arquitetura geral do sistema



Fonte: Elaborada pelos autores (2026).

2.2. Origem e tratamento dos dados

Os dados foram obtidos de uma extração bruta de 351 ordens do sistema de gestão SAP da fábrica, entregue em planilha. Após a higienização e o descarte de 9 registros sem identificação de aeronave (resultando em 342 ordens válidas vinculadas), o conjunto de modelagem abrangeu 11 aeronaves distribuídas em 8 etapas de montagem com ordens ativas. Informações que identificam a empresa ou o produto foram mascaradas por exigência de confidencialidade.

Antes de qualquer cálculo, uma rotina de tratamento padroniza os nomes das colunas, preserva os códigos de identificação como texto (evitando que 0123 seja lido como 123), traduz as siglas internas de planejamento para nomes de etapas, padroniza o número de série das aeronaves e converte as datas. A Tabela 2 apresenta as verificações de qualidade realizadas e o tratamento adotado em cada caso; a extração não apresentou registros repetidos nem falhas no identificador principal.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Tabela 2. Verificação da qualidade dos dados e tratamentos aplicados

O que foi verificado	O que se encontrou	O que foi feito
Volume do arquivo bruto	351 registros e 22 colunas	Todos os registros foram lidos na etapa inicial de ingestão.
Número da ordem (identificador) ausente ou repetido	Nenhum caso	A rotina de descarte existe, mas não precisou ser acionada.
Número de série da aeronave ausente	9 registros	Marcados como “sem número de série” e retirados do cálculo de sequenciamento.
Campo de situação da ordem em branco	34 registros	Preenchidos como vazio e não contados como problema.
Data de entrega inválida	1 registro	Convertida para nulo e excluída do cálculo de atraso.
Códigos de erro do sistema (#N/D e #N/A)	Presentes na coluna de número de série	Padronizados antes da carga no banco.
Colunas com muitos campos em branco	Seis colunas de apoio	Não utilizadas na modelagem; mantidas apenas para consulta.

Fonte: Elaborada pelos autores (2026).

2.3. Como o tempo total de produção é calculado

A linha de montagem estudada segue um padrão em que todas as aeronaves passam pelas mesmas etapas, na mesma sequência. Na literatura, esse arranjo é chamado de *Flow Shop* de permutação, e o tempo total do início ao fim de toda a produção é chamado de *makespan* (PINEDO, 2016).

O cálculo segue uma regra simples: uma aeronave só inicia uma etapa depois de concluir a anterior e de a estação ser liberada pela aeronave que a precede, prevalecendo a condição que ocorrer mais tarde (PINEDO, 2016):

$$C_{i,k} = \max(C_{i-1,k},\ C_{i,k-1}) + p_{i,k} \qquad (1)$$

Nessa equação, $C_{i,k}$ é o instante em que a aeronave da posição i conclui a etapa k; $C_{i-1,k}$ é quando a aeronave anterior liberou essa mesma etapa; $C_{i,k-1}$ é quando a própria aeronave terminou a etapa anterior; e $p_{i,k}$ é o tempo que ela consome nessa etapa. O tempo total da produção (C_{max}) é o valor obtido para a última aeronave na última etapa.

O tempo de cada etapa merece uma observação. A extração recebida não continha o tempo-padrão de cada operação. Por isso, adotou-se como medida de carga de trabalho a quantidade de ordens abertas da aeronave em cada etapa. Trata-se de uma medida aproximada (*proxy*), e não de um tempo efetivamente medido: os tempos totais apresentados neste trabalho são expressos em unidades de carga, e não em horas ou dias de produção. Os efeitos dessa aproximação são discutidos na Seção 3.9.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

2.4. O algoritmo genético

Cada solução candidata é uma ordem completa das 11 aeronaves. Há 11! = 39.916.800 ordens possíveis, quantidade que torna inviável a enumeração exaustiva em cenários de maior escala e justifica o uso de um método de busca. O algoritmo mantém 50 ordens, avalia todas, seleciona as melhores, combina-as, aplica pequenas alterações aleatórias e repete o processo, com a configuração apresentada na Tabela 3.

Tabela 3. Configuração do algoritmo genético

Parâmetro	Valor adotado	Justificativa
Tamanho da população	50 ordens candidatas	Equilíbrio entre qualidade e tempo de cálculo; testado na Seção 3.7.
Número máximo de rodadas	100	Limite de segurança; na prática, a execução é encerrada antes.
Parada antecipada	20 rodadas sem melhora	Evita continuar calculando depois que o resultado estabiliza.
Taxa de combinação (*crossover*)	0,85	Faixa usual da literatura; alternativas testadas na Seção 3.7.
Forma de combinação	PMX	Garante que nenhuma aeronave seja repetida ou esquecida.
Taxa de alteração (mutação)	0,10	Mantém variedade nas soluções; alternativas testadas na Seção 3.7.
Forma de alteração	Troca de duas posições	Alteração simples e adequada a uma ordem.
Forma de seleção	Torneio entre 3 candidatas	Favorece as melhores sem eliminar a variedade.
Preservação das melhores	2 soluções por rodada	Impede que a melhor solução encontrada se perca.
Peso do risco (α)	0,3	Define o quanto o risco influencia a decisão; justificado na Seção 3.7.
Ponto de partida	Ordens aleatórias mais a ordem FIFO	Dá ao algoritmo uma solução conhecida de partida, ideia inspirada em heurísticas construtivas (NAWAZ; ENSCORE JR.; HAM, 1983).

Fonte: Elaborada pelos autores (2026).

2.5. O modelo de previsão de risco

2.5.1 Variáveis utilizadas

O modelo observa 12 características de cada aeronave, todas calculadas a partir das ordens de produção reais tratadas na Seção 2.2. A Tabela 4 apresenta cada uma delas.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Tabela 4. Variáveis utilizadas pelo modelo de previsão

Variável	O que ela representa
Ordens travadas	Quantas ordens da aeronave estão bloqueadas no sistema.
Ordens com falta de material	Quantas ordens estão paradas esperando peça.
Ordens com restrição de qualidade	Quantas ordens estão retidas por inspeção ou engenharia.
Ordens em execução	Quantas ordens estão sendo trabalhadas no momento.
Ordens a encerrar	Quantas ordens estão marcadas para fechamento.
Total de ordens	Quantas ordens abertas a aeronave possui ao todo.
Proporção com pendência	Que fatia das ordens da aeronave tem alguma pendência registrada.
Proporção de ordens críticas	Que fatia das ordens está travada ou com falta de material.
Diversidade de centros de trabalho	Em quantos setores diferentes a aeronave tem ordens abertas.
Defasagem da data de entrega	Quantos dias separam a data atual da data prevista de entrega.
Etapa de montagem	Em que fase da montagem a aeronave se encontra.
Mês de referência	Mês a que se refere a extração dos dados.

Fonte: Elaborada pelos autores (2026).

2.5.2. De onde vêm as respostas e como se evitou a circularidade

Esta etapa do trabalho apresenta uma limitação metodológica importante. As variáveis da Tabela 4 são reais, extraídas do sistema da fábrica. Já a resposta que o modelo aprende a prever, isto é, se a aeronave atrasou ou não, não pôde ser obtida do histórico, porque a base disponibilizada mostra apenas a situação atual das ordens, sem registro de quais aeronaves efetivamente atrasaram no passado.

As respostas foram, então, geradas a partir de regras de conhecimento do processo, que consideram falta de material, ordens travadas, proporção de pendências e defasagem da data de entrega. Como alertado na Seção 1.3, se essa resposta fosse calculada por uma fórmula fixa, o modelo apenas a redescobriria, e o desempenho medido não teria significado (KAUFMAN *et al.*, 2012).

Para evitar isso, adotaram-se três medidas. A primeira foi aplicar uma variação aleatória nos valores das variáveis, de modo que não existam dois casos idênticos. A segunda foi variar aleatoriamente o peso de cada regra a cada caso gerado, para que não haja um conjunto fixo de pesos a ser memorizado. A terceira, e mais importante, foi transformar o valor calculado em uma probabilidade e sortear a resposta final a partir dela: assim, duas aeronaves em situação parecida podem receber respostas diferentes, como pode ocorrer na fábrica. Esse sorteio impede que qualquer modelo alcance 100% de acerto e o obriga a estimar probabilidades em vez de memorizar uma regra.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Esse procedimento gerou 360 casos, dos quais 165 (45,8%) foram marcados como atraso. Consequentemente, o desempenho relatado na Seção 3.4 mede a capacidade do modelo de recuperar um processo parcialmente aleatório, construído a partir de regras, e não seu acerto sobre atrasos efetivamente observados na fábrica. Esses indicadores, portanto, não comprovam desempenho preditivo em condições reais de produção, limitação retomada na Seção 3.9.

2.5.3. Configuração do modelo e forma de avaliação

A configuração do modelo foi definida por busca entre combinações de valores, resultando em 100 árvores de decisão, profundidade máxima de 5 níveis e mínimo de 10 casos para dividir um nó e para formar um grupo final. A profundidade reduzida foi intencional, pois árvores rasas produzem probabilidades mais suaves, evitando valores extremos de 0% ou 100% que prejudicariam o cálculo de risco do algoritmo genético. Aplicou-se também o balanceamento entre as classes, para que o modelo não privilegiasse a mais frequente.

Na avaliação, adotou-se a validação cruzada em 10 partes, número recomendado por Kohavi (1995): o modelo foi treinado em nove grupos e testado no décimo, repetindo-se o processo até que todos os 360 casos fossem usados como teste. Assim, toda previsão avaliada refere-se a um caso não visto no treinamento (resultados na Seção 3.4).

2.6. Como as duas técnicas se unem

O algoritmo genético atribui a cada ordem de produção uma nota (função de aptidão) para compará-la às demais. Sem o modelo de risco, essa nota depende apenas do tempo total:

$$\text{Nota}_{\text{tempo}}(\pi) = 1 / C_{\max}(\pi) \quad (2)$$

Com o modelo de risco ativado, a nota passa a ser dividida também por um fator de penalidade Ψ (psi):

$$\text{Nota}_{\text{tempo+risco}}(\pi) = 1 / [\, C_{\max}(\pi) \times \Psi(\pi) \,] \quad (3)$$

O fator Ψ mede o quanto aquela ordem expõe a produção ao risco:

$$\Psi(\pi) = 1 + \alpha \times E(\pi) \quad (4)$$

A exposição ao risco $E(\pi)$ é a soma, para todas as aeronaves, da posição que ela ocupa na fila multiplicada pelo risco estimado para ela:

$$E(\pi) = \sum (i \times r_i), \text{ para } i = 1, 2, \ldots, n \quad (5)$$

Nessa equação, i é a posição na fila (1 para a primeira, 2 para a segunda, e assim por diante) e r_i é o risco estimado da aeronave que ocupa essa posição, um valor entre 0 e 1. Como

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

a posição multiplica o risco, colocar uma aeronave arriscada no fim da fila faz $E(\pi)$ crescer bastante, o que aumenta Ψ e derruba a nota daquela ordem. Assim, o algoritmo tende a trazer as aeronaves mais problemáticas para o início.

O parâmetro α (alfa) regula a força desse efeito. Quando α vale 0, o fator Ψ vale 1 e a Equação 3 volta a ser exatamente a Equação 2, ou seja, o risco é ignorado. Essa propriedade será usada no experimento da Seção 3.6. Quanto maior o α, mais o sistema evita deixar aeronaves arriscadas para o final. Adotou-se α = 0,3, valor justificado por experimento na Seção 3.7.

2.6.1. A unidade de cada termo e o peso real do parâmetro α

A Tabela 5 apresenta a unidade e a faixa de valores de cada termo da função de aptidão, o que permite dimensionar o peso real do parâmetro α.

Como mostra a última linha da Tabela 5, apresentada a seguir, Ψ não é uma correção somada ao tempo, mas um fator que o multiplica. Nas 30 execuções com o modelo de risco ativado (Seção 3.6), a exposição média obtida foi $E(\pi) \approx 23{,}47$ (23,5 nas tabelas resumidas). Com α = 0,3:

$$\Psi = 1 + 0{,}3 \times 23{,}47 = 8{,}04 \quad (6)$$

Assim, o valor que entra no denominador da nota é o tempo total multiplicado por 8,04:

$$C_{max} \times \Psi = 119{,}8 \times 8{,}04 \approx 963 \quad (7)$$

contra apenas 119,8 sem penalidade (α = 0). Na faixa de exposição observada, esse α faz o risco pesar de 7 a 9 vezes mais que o próprio tempo na nota de cada sequência. Essa força explica o resultado da Seção 3.6: o algoritmo prioriza reduzir $E(\pi)$ sempre que há várias ordens com tempo total parecido, situação comum neste problema, e só aceita um tempo maior quando não há alternativa de baixo risco com o mesmo C_{max}.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Tabela 5. Unidade e faixa de valores de cada termo da função de aptidão

Termo	O que representa	Unidade / faixa de valores
$C_{max}(\pi)$	Tempo total de produção da ordem π.	Unidades de carga (aproximadas pela contagem de ordens em aberto por etapa).
i	Posição da aeronave na fila.	Número inteiro sem unidade, de 1 a 11.
r_i	Risco de atraso estimado pelo modelo para a aeronave da posição i.	Probabilidade sem unidade, entre 0 e 1.
$E(\pi)$	Soma de (posição × risco) de todas as aeronaves da fila.	Sem unidade; de 0 (risco nulo) a 66 (limite teórico: 11 aeronaves com risco 1,0).
α	Peso atribuído ao risco na fórmula.	Sem unidade, escolhido por quem configura o sistema; adotado 0,3.
$\Psi(\pi) = 1 + \alpha \times E(\pi)$	Fator que multiplica o tempo total.	Sem unidade; multiplicador aplicado sobre C_{max}.

Fonte: Elaborada pelos autores (2026).

3. RESULTADOS E DISCUSSÃO

Esta seção apresenta primeiro as telas do sistema (Seções 3.1 a 3.3) e depois os experimentos que sustentam os resultados (Seções 3.4 a 3.8), todos realizados sobre a base descrita na Seção 2.2. Ressalta-se que se trata de resultados experimentais, obtidos com tempos de etapa aproximados pela contagem de ordens abertas (Seção 2.3) e com um modelo de risco treinado a partir de respostas geradas por regras (Seção 2.5.2). Os ganhos relatados devem, portanto, ser interpretados como evidência do potencial da abordagem, e não como ganhos esperados na operação industrial real.

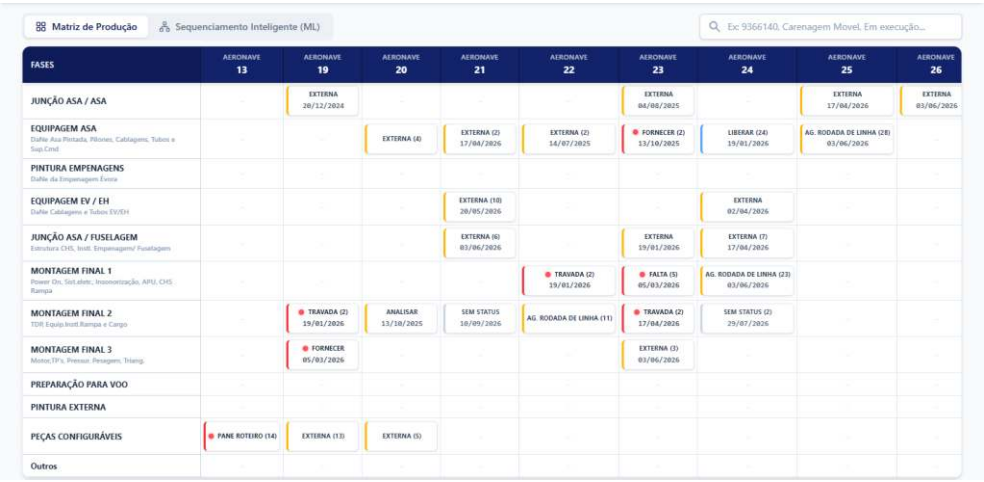
3.1. Painel de controle e matriz de produção

A tela principal apresenta uma matriz em que cada linha é uma etapa de montagem e cada coluna é uma aeronave (Figura 2). A interface lista todas as fases cadastradas no padrão da fábrica (8 delas com ordens ativas no lote analisado) e permite rolagem horizontal entre as 11 aeronaves avaliadas. As células são coloridas conforme a situação: vermelho para bloqueios críticos e amarelo para pendências. Quando uma célula reúne ordens em situações diferentes, exibe-se a mais grave, para que nenhuma ocorrência crítica fique oculta; trata-se de uma decisão de projeto deste trabalho, e não de uma técnica da literatura. Pela mesma tela, o gestor ajusta o tamanho da população, o número de rodadas e o peso do risco (α) e, ao final de cada execução,

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

consulta o tempo total obtido, o tempo da ordem de chegada, a diferença percentual entre ambos e o tempo de cálculo.

Figura 2. Matriz de produção com a situação de cada aeronave por etapa



Fonte: Elaborada pelos autores (2026).

O painel foi apresentado a quatro profissionais de planejamento e controle da produção da empresa, que confirmaram a aderência da matriz e da fila sugerida à realidade operacional da linha. Trata-se de uma validação qualitativa, que não substitui um teste formal de usabilidade.

3.2. Evolução do resultado ao longo das rodadas

A Figura 3 mostra a evolução do tempo total da melhor solução a cada rodada, comparada ao tempo do FIFO, de 193 unidades (linha tracejada). Já nas primeiras rodadas o tempo cai para a faixa de 119 a 120 unidades, o que evidencia a espera entre etapas gerada pela ordem de chegada. Em seguida, a curva se estabiliza, porque as soluções da população se tornam semelhantes e a combinação entre elas deixa de gerar novidade, comportamento esperado em algoritmos genéticos (GOLDBERG, 1989); nesse ponto, o critério de parada encerra a execução.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 3. Evolução do tempo total ao longo das rodadas, comparada ao método FIFO



Fonte: Elaborada pelos autores (2026).

3.3. Fila sugerida pelo sistema

A Figura 4 mostra o resultado final: a ordem sugerida, o risco estimado de cada aeronave e um alerta visual de prioridade. Nota-se que a aeronave de maior risco não ocupa necessariamente a primeira posição, porque o sistema precisa respeitar também as restrições de tempo descritas na Equação 1. Ainda assim, as aeronaves marcadas como críticas aparecem concentradas nas primeiras posições, efeito que é confirmado numericamente na Seção 3.6.

Figura 4. Fila sugerida pelo sistema, com o risco estimado de cada aeronave



Fonte: Elaborada pelos autores (2026).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.4. Qualidade das previsões do modelo

A Tabela 6 apresenta o resultado das previsões na validação cruzada, isto é, sempre sobre casos não vistos pelo modelo; esses quatro valores são a base dos indicadores seguintes.

Tabela 6. Resultado das previsões na validação cruzada (360 casos com respostas geradas por regras)

Resposta gerada \ Previsão	**O modelo previu: sem atraso**	**O modelo previu: com atraso**
Rotulado sem atraso	145 (acerto)	50 (erro)
Rotulado com atraso	67 (erro)	98 (acerto)

Fonte: Elaborada pelos autores (2026).

A partir desses quatro valores calculam-se os indicadores usuais, reunidos na Tabela 7: a acurácia, fração de acertos sobre o total, (145 + 98) / 360 = 0,675; a precisão, que indica, entre os casos apontados como atraso, quantos estavam de fato rotulados como atraso, 98 / (98 + 50) = 0,662; o *recall* (sensibilidade), que indica, entre os casos rotulados como atraso, quantos o modelo identificou, 98 / (98 + 67) = 0,594; e o F1, média harmônica entre precisão e *recall*, igual a 0,626.

A coluna da direita da Tabela 7 mostra o resultado que se obteria avaliando o modelo sobre os próprios dados de treino: todos os indicadores sobem entre dez e quinze pontos percentuais, efeito descrito na Seção 1.3. Por isso, apenas a coluna da esquerda deve ser considerada como desempenho esperado.

Tabela 7. Indicadores do modelo: validação cruzada frente à medição sobre o treino

Indicador	**Validação cruzada**	**Medido sobre os dados de treino**
Acurácia	67,5%	77,5%
Precisão	66,2%	76,3%
Recall (sensibilidade)	59,4%	73,9%
F1	62,6%	75,1%

Fonte: Elaborada pelos autores (2026).

Para saber se 67,5% de acerto é um bom resultado, é preciso uma referência. Um modelo que atribuísse sempre a resposta mais frequente acertaria 195 dos 360 casos, ou seja, 54,2%. O modelo, portanto, acerta cerca de 13 pontos percentuais acima desse patamar, o que indica que ele capturou a relação entre as variáveis e as respostas geradas, embora esse resultado não possa ser transposto diretamente para atrasos reais.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

O *recall* também exige leitura cuidadosa: 59,4% significa que cerca de 4 em cada 10 casos rotulados como atraso não foram sinalizados. Para o uso proposto, isso é menos grave, pois o sistema utiliza a probabilidade estimada para ordenar as aeronaves, e não a resposta “sim ou não”. Ainda assim, o modelo deve ser entendido como apoio à decisão, e não como substituto do julgamento do gestor.

3.5. Estabilidade do algoritmo genético

Como o algoritmo genético usa sorteios, cada execução pode produzir um resultado diferente. Para verificar a estabilidade, o sistema foi executado 30 vezes, com sorteios iniciais distintos, obtendo tempo total médio de 119,8 unidades. A dispersão em torno dessa média é medida pelo desvio padrão amostral (s):

$$s = \sqrt{[\ \Sigma\ (x_i - \bar{x})^2 / (N - 1)\ ]} = 2{,}36 \quad (8)$$

Em que N é o número de execuções (30), x_i é o tempo obtido em cada uma e $\bar{x}$ é a média amostral (119,8). Para facilitar a leitura, divide-se o desvio padrão pela média, obtendo o coeficiente de variação (CV):

$$CV = 2{,}36 / 119{,}8 = 0{,}020 \text{ (ou seja, 2,0\%)} \quad (9)$$

Uma variação de 2% é baixa e indica resultado estável: 18 execuções (60%) chegaram a 119,0, 11 (36,7%) a 120,0 e apenas 1 (3,3%) a 132,0, valores muito distantes dos 193,0 do FIFO. O teste t para uma amostra confirma que a diferença entre a média obtida e o valor do FIFO não pode ser atribuída ao acaso ($p < 0{,}001$).

3.6. Contribuição do modelo de risco

Superar o FIFO não indica, por si só, se o ganho decorre apenas do algoritmo genético ou se o modelo de risco também contribui. Para responder a essa questão, o algoritmo foi executado 30 vezes com o modelo de risco ligado ($\alpha = 0{,}3$) e 30 vezes desligado ($\alpha = 0$), com os mesmos 30 sorteios iniciais nas duas condições.

Para que a comparação fosse válida, as duas filas resultantes foram medidas com os mesmos valores de risco (Equação 5), o que permite quantificar a exposição ao risco da fila produzida sem o modelo, ainda que o algoritmo não o tenha considerado durante a busca.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Tabela 8. Comparação entre o sistema com e sem o modelo de risco (30 execuções em cada condição)

Indicador	Com modelo de risco (α = 0,3)	Sem modelo de risco (α = 0)
Tempo total médio	119,8	119,1
Exposição ao risco E(π)	23,5	27,4
Posição média das aeronaves de risco alto	2,9	5,0
Redução frente ao FIFO	37,9%	38,3%

Fonte: Elaborada pelos autores (2026).

A troca observada na Tabela 8 é vantajosa. No tempo total, a diferença entre 119,8 e 119,1 é pequena e, segundo o teste t, pode ser explicada pelo acaso ($p = 0,123$); ligar o modelo de risco, portanto, não torna a produção perceptivelmente mais lenta. Já a exposição ao risco cai de 27,4 para 23,5, redução de 14,3% estatisticamente significativa ($p < 0,001$).

Essa redução decorre da reorganização da fila: as aeronaves de risco alto passam da quinta para a terceira posição média, ou seja, são antecipadas em cerca de duas posições. Observa-se ainda que, sem o modelo de risco, a fila otimizada apresenta exposição de 27,4, superior à do próprio FIFO (24,1); otimizar apenas o tempo pode, portanto, deslocar aeronaves críticas para o fim da fila. As duas técnicas cumprem papéis distintos: o algoritmo genético reduz o tempo, e o modelo de risco reorganiza a fila praticamente sem custo de tempo.

3.7. Escolha dos parâmetros

Os valores da Tabela 3 foram definidos por experimentos com 10 execuções por configuração. A Tabela 9 mostra o efeito do peso do risco α.

Tabela 9. Efeito do peso do risco (10 execuções por configuração)

Peso do risco (α)	Tempo total médio	Exposição ao risco E(π)	Posição média das aeronaves de risco alto
0,0 (risco ignorado)	119,0	27,6	5,1
0,1	120,3	23,5	2,9
0,3 (valor adotado)	120,9	23,4	2,9
0,5	120,5	23,6	2,9
1,0	119,3	23,6	3,0
2,0	119,5	23,5	3,1

Fonte: Elaborada pelos autores (2026).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Praticamente todo o benefício surge na passagem de α = 0 para α = 0,1, quando a exposição cai de 27,6 para 23,5; valores maiores, até 2,0, quase não alteram o resultado. O sistema, portanto, não é sensível ao valor exato de α, desde que seja maior que zero, e o valor 0,3 foi adotado por situar-se nessa região estável.

Quanto aos operadores do algoritmo, testaram-se três taxas de combinação (0,70, 0,85 e 0,95) e quatro taxas de alteração (0,02, 0,05, 0,10 e 0,20), totalizando doze configurações. Todas terminaram com tempo total entre 119,1 e 120,9, faixa estreita que indica que o problema não é sensível a esses ajustes dentro das faixas usuais.

Tabela 10. Efeito do tamanho da população (10 execuções por configuração)

Tamanho da população	Tempo total médio	Desvio padrão	Tempo de cálculo (s)
20	122,3	5,36	0,043
30	120,5	4,06	0,068
50 (valor adotado)	120,9	3,93	0,109
80	119,1	0,32	0,124
100	119,2	0,42	0,138

Fonte: Elaborada pelos autores (2026).

O tamanho da população (Tabela 10) mostrou-se o parâmetro mais influente. Com 20 soluções, o resultado oscila bastante (desvio padrão de 5,36), enquanto com 80 ou 100 o sistema chega de forma consistente a 119 unidades, com desvio padrão abaixo de 0,5, a um custo adicional pequeno (de 0,043 para 0,138 segundos). Assim, ampliar a população é o ajuste mais promissor caso o número de aeronaves aumente; o valor 50 representa um meio-termo entre estabilidade e tempo de cálculo.

3.8. Tempo de resposta da aplicação

A consulta que carrega os registros da matriz de produção levou em média 4,2 milissegundos (20 repetições), e o ciclo completo de sequenciamento, que envolve ler os dados das aeronaves, calcular o risco e executar o algoritmo genético, levou em média 205 milissegundos, dos quais cerca de 100 milissegundos correspondem ao algoritmo genético. São tempos compatíveis com uso interativo, medidos com um único usuário e com o volume de dados descrito.

3.9. Limitações e trabalhos futuros

- As respostas usadas no treinamento foram geradas, e não observadas. Como explicado na Seção 2.5.2, as variáveis são reais, mas o indicador de atraso foi construído por

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

regras com componente aleatório. Por isso, os indicadores da Seção 3.4 não comprovam acerto sobre atrasos que de fato ocorreram, e o desempenho do modelo em condições reais de produção permanece desconhecido. O próximo passo é levantar no histórico do sistema quais ordens efetivamente ultrapassaram a data prevista e treinar e avaliar o modelo com essas respostas reais.

- O tempo de cada etapa é uma aproximação. Na falta do tempo-padrão por operação, usou-se a quantidade de ordens abertas como medida de carga (Seção 2.3). Assim, os tempos totais e a redução de cerca de 38% frente ao FIFO referem-se a essa medida aproximada, e não a horas ou dias de produção. Com os tempos reais, os valores absolutos devem mudar, e a manutenção da vantagem relativa entre as ordens precisa ser verificada.
- A escala é pequena. Foram 11 aeronaves e 8 etapas ativas, quantidade suficiente para demonstrar o funcionamento, mas modesta frente aos conjuntos de teste usados na literatura. Recomenda-se repetir a avaliação em recortes maiores da fábrica.
- Os ganhos não devem ser generalizados para o ambiente industrial real. Os experimentos foram conduzidos em laboratório, sobre uma única extração de dados, com um usuário por vez e sem testes de carga com usuários simultâneos. A confirmação dos resultados depende de um estudo piloto na linha de montagem, que compare a sequência sugerida pelo sistema com a efetivamente executada.

4. CONSIDERAÇÕES FINAIS

Este trabalho desenvolveu e avaliou, em ambiente experimental, um sistema que une um algoritmo genético, que busca a ordem de menor tempo total, e um modelo de aprendizado de máquina, que estima o risco de atraso de cada aeronave e influencia a decisão por meio de um fator de penalidade. Os experimentos, conduzidos nas condições descritas na Seção 3.9, permitem três conclusões.

A primeira é que, considerando a medida aproximada de tempo adotada, a ordem sugerida pelo sistema reduziu o tempo total estimado em cerca de 38% em relação à ordem de chegada, resultado que se repetiu ao longo de 30 execuções independentes e foi confirmado por teste estatístico.

A segunda conclusão, mais relevante do ponto de vista da proposta, é que as duas técnicas cumprem papéis distintos. A redução de tempo vem do algoritmo genético. O modelo de risco, por sua vez, reorganiza a fila: ele reduziu em 14,3% a exposição da produção ao risco

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

estimado e trouxe as aeronaves mais problemáticas em média duas posições para frente, sem aumento significativo do tempo total. Verificou-se ainda que, sem esse componente, a fila otimizada chega a ficar mais exposta ao risco do que a própria ordem de chegada.

A terceira é que o sistema se mostrou pouco sensível ao ajuste fino de seus parâmetros, sendo o tamanho da população o fator de maior influência sobre a estabilidade dos resultados.

Do ponto de vista prático, mostrou-se viável reunir em uma única aplicação a leitura dos dados corporativos, o cálculo da ordem e a apresentação do resultado em uma tela compreensível para o gestor, com tempo de resposta adequado ao uso interativo. Ressalta-se, contudo, que os ganhos relatados são experimentais e não comprovam o desempenho em condições reais de produção, pelas razões discutidas na Seção 3.9. A validação com atrasos históricos e tempos-padrão, seguida de um estudo piloto na linha de montagem, constitui a continuidade imediata do trabalho.

DECLARAÇÃO DE USO DE INTELIGÊNCIA ARTIFICIAL

Os autores utilizaram as ferramentas de inteligência artificial generativa Claude (Anthropic) e Gemini (Google) como apoio na revisão e na organização do texto, na tradução dos resumos e na adaptação do manuscrito ao modelo de formatação da revista. Todo o conteúdo produzido com o auxílio dessas ferramentas foi revisado, avaliado criticamente e validado pelos autores, que assumem integral responsabilidade pelo conteúdo, pelos dados, pelos resultados e pelas referências apresentados.

FINANCIAMENTO

Esta pesquisa não recebeu financiamento de agências de fomento, instituições ou empresas.

CONFLITO DE INTERESSES

Um dos autores possui vínculo empregatício com a empresa que disponibilizou os dados utilizados neste estudo. Os autores declaram não haver outros conflitos de interesses.

REFERÊNCIAS

BREIMAN, L. Random forests. **Machine Learning**, v. 45, n. 1, p. 5-32, 2001. DOI: 10.1023/A:1010933404324.

CHANG, X.; JIA, X.; REN, J. A reinforcement learning enhanced memetic algorithm for multi-objective flexible job shop scheduling toward Industry 5.0. **International Journal of Production Research**, v. 63, n. 1, p. 119-147, 2025. DOI: 10.1080/00207543.2024.2357740.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

FASTAPI. **FastAPI framework, high performance, easy to learn, fast to code, ready for production**. 2023. Disponível em: https://fastapi.tiangolo.com/. Acesso em: 8 mar. 2026.

GOLDBERG, D. E. **Genetic algorithms in search, optimization, and machine learning**. Reading: Addison-Wesley, 1989.

GOLDBERG, D. E.; LINGLE JR., R. Alleles, loci and the traveling salesman problem. *In:* GREFENSTETTE, J. J. (ed.). **Proceedings of the First International Conference on Genetic Algorithms and Their Applications**. Hillsdale: Lawrence Erlbaum Associates, 1985. p. 154-159.

HOLLAND, J. H. **Adaptation in natural and artificial systems**: an introductory analysis with applications to biology, control, and artificial intelligence. Ann Arbor: University of Michigan Press, 1975.

KAUFMAN, S.; ROSSET, S.; PERLICH, C.; STITELMAN, O. Leakage in data mining: formulation, detection, and avoidance. **ACM Transactions on Knowledge Discovery from Data**, v. 6, n. 4, artigo 15, 2012. DOI: 10.1145/2382577.2382579.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. *In:* **Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)**. San Francisco: Morgan Kaufmann, 1995. v. 2, p. 1137-1143. Disponível em: https://www.ijcai.org/Proceedings/95-2/Papers/016.pdf. Acesso em: 19 set. 2026.

LI, J.; LIU, Q.; LI, X.; GAO, L. An efficient problem-specific evolutionary algorithm for flexible job shop scheduling problem with specific workers in highly customised manufacturing systems. **International Journal of Production Research**, v. 63, n. 19, p. 7238-7259, 2025. DOI: 10.1080/00207543.2025.2496971.

NAWAZ, M.; ENSCORE JR., E. E.; HAM, I. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. **Omega**, v. 11, n. 1, p. 91-95, 1983. DOI: 10.1016/0305-0483(83)90088-9.

PEDREGOSA, F. *et al.* Scikit-learn: machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825-2830, 2011. Disponível em: http://jmlr.org/papers/v12/pedregosa11a.html. Acesso em: 19 set. 2026.

PINEDO, M. L. **Scheduling**: theory, algorithms, and systems. 5. ed. Cham: Springer, 2016. DOI: 10.1007/978-3-319-26580-3.

SLACK, N.; BRANDON-JONES, A.; JOHNSTON, R. **Administração da produção**. 8. ed. São Paulo: Atlas, 2018.

WANG, H. *et al.* Genetic algorithm with local search for the multi-target scheduling in flexible manufacturing system. **Journal of Circuits, Systems and Computers**, v. 31, n. 16, artigo 2250279, 2022. DOI: 10.1142/S0218126622502796.

WANG, W.; MA, J. A review: applications of machine learning and deep learning in aerospace engineering and aero-engine engineering. **Advances in Engineering Innovation**, v. 6, n. 1, p. 54-72, 2024. DOI: 10.54254/2977-3903/6/2024060.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.