

- **Trabalho:** “Otimização de Pontos de Corte em CNC utilizando Computação Evolutiva”
- **Nome Completo:** Lucas Araújo de Castro
- **Assunto:** Computação Evolutiva
- **Algoritmo Escolhido:** GA
- **Youtube:** https://youtu.be/xm_42k2w4FU
- **GitHub:** <https://github.com/LucasCastroo/otimizador-pontos-de-recorte-cnc>

Documentação

1. Introdução

• **Contextualização:** Apresentar brevemente o problema de otimização de corte em CNC, destacando a importância da redução de matéria-prima, energia e tempo de máquina.

O corte de chapas metálicas utilizando máquinas CNC (Controle Numérico Computadorizado) é amplamente utilizado na indústria devido à sua precisão e repetibilidade. No entanto, a eficiência do posicionamento dos recortes impacta diretamente três fatores essenciais para a produtividade e competitividade das empresas:

- Redução de matéria-prima: Melhor aproveitamento das chapas reduz o desperdício de material.
- Otimização do consumo de energia: Um layout bem planejado diminui o tempo da máquina em operação, reduzindo custos de energia elétrica.

- Menor tempo de processamento: Um planejamento eficiente minimiza a quantidade de movimentos e cortes desnecessários.

Para solucionar esse problema, implementei um Algoritmo Genético (GA) que organiza a disposição dos recortes em uma chapa metálica, utilizando a abordagem de Retângulos Livres (Free Rectangles).

2. Escolha do Algoritmo Evolutivo

- **Justificativa:** Por que esse algoritmo foi escolhido em detrimento dos demais?

Particle Swarm Optimization (PSO)

- Vantagem - Simples de implementar, funciona bem para otimização contínua.
- Desvantagem - Pouco eficiente para problemas combinatórios como corte de chapas.

Ant Colony Optimization (ACO)

- Vantagem - Bom para problemas de roteamento e caminhos mínimos.
- Desvantagem - Exige um tempo elevado para convergir para boas soluções.

Differential Evolution (DE)

- Vantagem - Eficiente para problemas de otimização contínua.
- Desvantagem - Não adequado para otimização combinatória.

Genetic Algorithm (GA)

- Vantagem - Flexível, funciona bem com problemas combinatórios e permite múltiplos operadores.
- Desvantagem - Pode exigir mais ajustes para convergência rápida.

Motivos que escolhi o Algoritmo Genético:

- Capacidade de lidar com problemas combinatórios: Como o problema envolve a disposição de recortes fixos em uma área delimitada, o GA se destaca.
- Versatilidade: O GA permite ajustes finos em seus operadores (crossover, mutação, seleção) para melhorar a performance.
- Exploração e Exploração Balanceadas: Através da mutação e recombinação de indivíduos, o GA mantém uma boa diversidade na população, evitando soluções locais ruins.

3. Descrição da Solução Implementada

- **Arquitetura do Código:** Explicar como o código está organizado (classes, métodos principais, etc.).
- **Modificações Realizadas:** Descrever detalhadamente cada modificação feita nos métodos de otimização (avaliação, seleção, mutação, cruzamento, parâmetros, etc.).
- **Ferramentas Utilizadas:** Mencionar se foi usado ChatGPT, bibliotecas específicas de Python, etc.

Arquitetura do Código

O código foi estruturado da seguinte forma:

Arquivo `app.py`

- Responsável por iniciar o processo de otimização.
- Define as dimensões da chapa e os recortes disponíveis.
- Instancia o Algoritmo Genético e executa a otimização.
- Exibe os layouts inicial e final.



UNITINS
UNIVERSIDADE ESTADUAL DO TOCANTINS

TOCANTINS
GOVERNO DO ESTADO



Arquivo genetic_algorithm.py

- Contém a implementação do Algoritmo Genético (GA).
- Utiliza a técnica de Retângulos Livres (Free Rectangles) para alocar os recortes na chapa.
- Inclui funções de avaliação, seleção, cruzamento e mutação.

Modificações Realizadas

1. Melhorias na Avaliação (evaluate_individual)
 - cálculo do fitness foi ajustado para penalizar fortemente layouts que descartam peças.
 - Agora, o bounding box final da solução é considerado para otimizar a ocupação da chapa.
2. Implementação de Seleção por Roleta (roulette_selection)
 - Foi usada uma roleta viciada, onde soluções com melhor fitness têm maior chance de serem escolhidas.
 - Evita a perda de diversidade na população, garantindo que as melhores soluções sejam favorecidas.
3. Crossover de Dois Pontos (crossover_two_point)
 - operador de crossover foi otimizado para permitir uma mistura eficiente das características dos pais.
 - Garante diversidade genética, evitando estagnação do algoritmo.

4. Operador de Mutação (mutate)

- Implementado um mecanismo que troca aleatoriamente a posição de duas peças com probabilidade de 10%.
- A mutação impede convergência prematura, aumentando a chance de explorar outras soluções.

Ferramentas Utilizadas

- Python: Linguagem principal do projeto.
- Bibliotecas:
 - random: Utilizada para embaralhamento e seleção aleatória.
 - copy: Para cópias seguras das soluções durante a otimização.
 - math: Para cálculos geométricos.
 - matplotlib (no `common.layout_display`): Para exibir os layouts.
- ChatGPT: Usado como suporte na otimização dos operadores genéticos e na escrita da documentação.

4. Resultados

- **Disposição Inicial vs. Disposição Otimizada:** Descrever como o algoritmo se comportou (podem ser imagens ou capturas de tela).
- **Impacto Econômico:** Comentar o potencial de economia de matéria-prima e energia.
- **Tempo de Processamento:** Avaliar se o tempo de execução do algoritmo é adequado para uso prático.



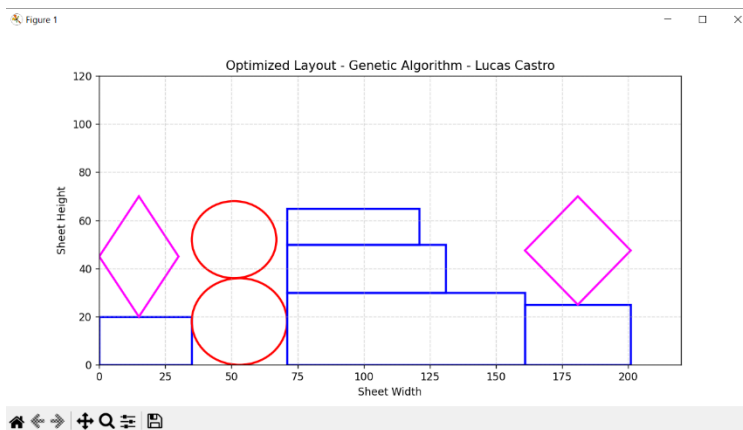
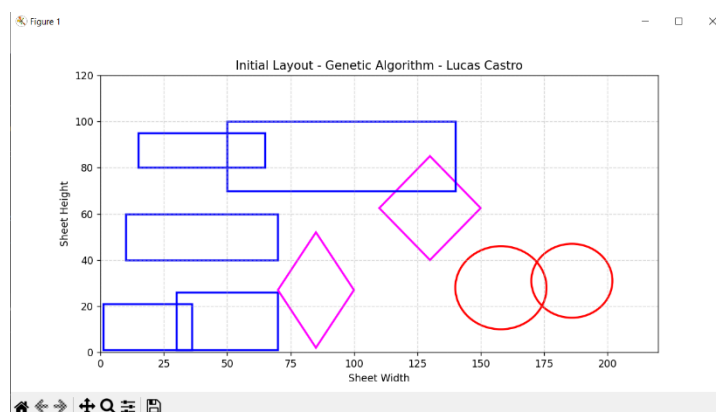
UNITINS
UNIVERSIDADE ESTADUAL DO TOCANTINS

TOCANTINS
GOVERNO DO ESTADO



Disposição Inicial vs. Disposição Otimizada

- A disposição inicial das peças é desorganizada, muitas delas ficam sobrepostas ou saem dos limites da chapa.
- Após a otimização, o algoritmo geneticamente evolui soluções que reduzem o espaço desperdiçado, realocando melhor os recortes.



Impacto Econômico

A otimização realizada pelo GA pode resultar em grandes economias para a indústria, pois:

- Menos material desperdiçado → Redução de custos na compra de chapas metálicas.
- Menos tempo de corte → Economia de energia elétrica e maior produtividade.

- Eficiência no aproveitamento do espaço → Pode permitir o uso de chapas menores, reduzindo desperdício.

Tempo de Processamento

- O algoritmo foi configurado para rodar 100 gerações.
- O tempo médio de execução foi de menos de 5 segundos em cada geração, tornando-o viável para uso em tempo real na indústria.

5. Análise Crítica

- **Vantagens e Desafios:** Pontuar os principais benefícios e possíveis limitações da abordagem escolhida.
- **Sugestões de Melhoria:** Indicar eventuais aprimoramentos futuros.

Vantagens

- GA encontra boas soluções em poucos segundos, tornando-o prático para uso real.
- A abordagem de Retângulos Livres evita sobreposição e melhora o aproveitamento do espaço.
- A seleção por roleta e mutação garantem diversidade na população, evitando soluções ruins.

Desafios

- A convergência pode ser lenta para casos com muitas peças pequenas.
- algoritmo não permite rotações além de 0° e 90° , limitando algumas soluções.

Sugestões de Melhoria

- Permitir rotação arbitrária das peças, aumentando as possibilidades de encaixe.
- Implementar hibridização com heurísticas gulosa, reduzindo a carga do GA.
- Melhorar a avaliação do fitness, considerando espaços vazios internos.

6. Conclusão

- **Resumo do Trabalho:** Reforçar os principais resultados obtidos.
- **Próximos Passos:** Possíveis extensões ou aplicações futuras.

Resumo do Trabalho

- Algoritmo Genético foi implementado com sucesso para otimizar o corte de chapas CNC, reduzindo o desperdício de material e otimizando o tempo de corte.
- Os resultados mostraram que o método reduz significativamente o espaço desperdiçado, e o tempo de execução é rápido o suficiente para aplicações industriais.



UNITINS
UNIVERSIDADE ESTADUAL DO TOCANTINS

TOCANTINS
GOVERNO DO ESTADO



Próximos Passos

- Introdução de novos operadores genéticos para melhorar a eficiência.
- Testes com diferentes tipos de materiais e formatos de recortes.
- Comparação do GA com algoritmos híbridos, combinando heurísticas de encaixe.

7. Referências (Opcional)

- Listar artigos, livros ou outros materiais utilizados.