

Estacionamento Autônomo Urbano de Automóveis Através de Caminhos Otimizados por Algoritmos Genéticos^{*}

Renan P. Vieira^{*} Téo C. Revoredo, IEEE Member^{**}

^{*} Programa de Pós-Graduação em Engenharia Eletrônica,
Departamento de Engenharia Eletrônica e de Telecomunicações,
Universidade do Estado do Rio de Janeiro, RJ (e-mail:
renan.porto.vieira@gmail.com).

^{**} Departamento de Engenharia Eletrônica e de Telecomunicações,
Universidade do Estado do Rio de Janeiro, RJ (e-mail:
teorevredo@uerj.br).

Abstract: Vehicle parking maneuvers in large cities have a great influence on traffic flow, measures that minimize time and space spent on these maneuvers are of great value. As a starting point, one must plan the best way to carry out parking, a non-trivial task with regard to non-holonomic vehicles prevalent in large metropolises. Based on this scenario, this work presents a path planning algorithm to park a car based on polynomial parameterization optimized by genetic algorithm. The objective is to define a law of motion to steer the vehicle from an initial pose near a parking space to a final pose within the latter smoothly, without interruption, avoiding any obstacles on the way. Path tracking is performed via pure pursuit controller. Results are presented in a 3D physical simulation environment and compared with another approach present in the literature, demonstrating its usefulness and providing the basis for broader studies.

Resumo: As manobras de estacionamento de veículos nas grandes cidades têm grande influência na fluidez do trânsito, medidas que minimizem o tempo e o espaço gasto nessas manobras são de grande valia. Como ponto de partida, deve-se planejar a melhor forma de realizar o estacionamento, tarefa não trivial em relação aos veículos não holonômicos prevalentes nas grandes metrópoles. Com base neste cenário, este trabalho apresenta um algoritmo de planejamento de caminhos para estacionar um carro baseado em parametrização polinomial otimizada por algoritmo genético. O objetivo é definir uma lei de movimento para conduzir o veículo de uma posição inicial perto de uma vaga de estacionamento para uma posição final na vaga de forma suave, sem interrupção, evitando quaisquer obstáculos no caminho. O rastreamento de caminho é realizado através do controlador de perseguição puro. Os resultados são apresentados em um ambiente de simulação física 3D e comparados com outra abordagem presente na literatura, demonstrando sua utilidade e fornecendo base para estudos mais amplos.

Keywords: Path planning; Autonomous parking; polynomial Parametrization; Optimization; Genetic algorithm; Path tracking; Pybullet.

Palavras-chaves: Planejamento de caminho; Estacionamento autônomo; Parametrização polinomial; Otimização; Algoritmo genético; Rastreamento de caminho; Pybullet

1. INTRODUÇÃO

Avanços sem precedentes em tecnologia da informação, robótica e veículos vêm transformando os automóveis tradicionais em máquinas inteligentes capazes não apenas de se movimentarem eficientemente e autonomamente, mas de interagirem com outros veículos e com a infraestrutura das cidades. Trata-se de robôs móveis que conseguem

^{*} O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Os autores agradecem, também, à Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ) pelos seus suportes financeiros no desenvolvimento deste trabalho.

interpretar o ambiente ao seu redor, tomar decisões rápidas e oportunas, navegar sem intervenção humana na estrada, manter padrões de mobilidade seguros e executar diferentes categorias de manobras, buscando atender às necessidades cada vez mais exigentes da sociedade e das cidades, tais como segurança para os usuários, otimização de deslocamento em um ambiente com cada vez mais pessoas e veículos, expansão de infraestrutura, utilização de energias renováveis e aumento dos níveis de conforto.

Nesse contexto, o estacionamento é um dos maiores desafios nas grandes cidades modernas devido ao aumento do número de veículos e da densidade populacional, além da necessidade de manutenção de distanciamento suficiente

entre veículos estacionados que permita abertura de portas, embarque e desembarque adequados. A realização de manobras de estacionamento contribui significativamente para os engarrafamentos nas grandes metrópoles atualmente (Rashid et al., 2019). Dotar carros com a capacidade de estacionar autonomamente tende a aliviar os problemas atuais. Por exemplo, após deixar passageiros, um carro autônomo poderia estacionar sozinho mesmo em vagas de estacionamento estreitas, uma abordagem que economizaria espantosos 6,2 bilhões de metros quadrados apenas nos estacionamentos dos Estados Unidos (Hussain and Zeadally, 2019). Não por acaso, sistemas de assistência ativa ao estacionamento vêm sendo implementados em veículos comerciais por diversos fabricantes, os quais são capazes de identificar vagas e manobrar o veículo até o seu interior sem colidir, com a mínima participação do motorista (United States. Department of Transportation. National Highway Traffic Safety Administration, 2021).

Muitas técnicas vêm sendo investigadas para o planejamento de caminhos para estacionar carros autônomos. D. Wang et al. (Wang et al., 2011) usa um método de dois arcos para modelar o caminho de estacionamento paralelo e otimizar seu comprimento total, entretanto, o caminho é limitado a dois seguimentos e dificultam a exatidão de rastreamento. E. Argento et al. (Argento et al., 2020) combina uma técnica de interpolação para gerar caminhos suaves com um algoritmo de busca por força bruta. Todavia, a abordagem oferece apenas soluções sub ótimas determinadas pelo intervalo de busca, embora abordagens deste tipo garantam alta exatidão de rastreamento enquanto evitam modos naturais da estrutura mecânica ou do sistema de servo-controle (Erkorkmaz and Altintas, 2001).

Na busca por otimizar a solução de problemas de movimentação autônoma, dá-se destaque à utilização de inteligência artificial (IA) (Ni et al., 2018). Entre as técnicas utilizadas, os algoritmos genéticos (AG) estão entre as mais promissoras, passíveis de aplicação direta no processo de planejamento (Tuncer and Yildirim, 2012; Tu and Yang, 2003). As aplicações deste algoritmo cresceram muito nos últimos anos, principalmente devido às vantagens que este apresenta quando comparado a outros métodos de otimização. Como principais vantagens pode-se destacar (Godberg, 1989):

- Não requer informações sobre o gradiente da superfície de resposta;
- Eventuais descontinuidades da superfície de resposta não afetam o desempenho da otimização;
- A presença de mínimos locais não reduzem a eficiência do algoritmo;
- A atuação do algoritmo tem apresentado excelentes resultados para problemas de otimização de grande escala

Este trabalho pretende contribuir para a melhoria de planejamento de caminhos para automóveis autônomos, propondo um algoritmo cujo propósito é conduzir o veículo de maneira suave e contínua de uma configuração inicial a uma final, desviar de eventuais obstáculos ao longo do trajeto e deslocar-se da menor distância possível, respeitando os limites físicos impostos pelo veículo. A proposta combina as vantagens da interpolação polinomial com a capacidade de otimização dos algoritmos genéticos para

gerar caminhos ininterruptos, exequíveis e otimizados com relação à distância percorrida, contribuindo especialmente em implementação na associação das diferentes técnicas. Como foco de aplicação, aborda-se o estacionamento autônomo. Uma vez definido o caminho, ele é efetuado através de um controlador de perseguição pura (Coulter, 1992). Os resultados apresentados são emulados em plataforma de simulação física que confere maior realismo e permite avaliar de maneira efetiva o desempenho da implementação proposta, provendo as bases para melhorias futuras.

2. MODELO CINEMÁTICO DO VEÍCULO

Sendo as manobras de estacionamento executadas com baixas velocidades, a influência da dinâmica pode ser desprezada (Zhu and Rajamani, 2006) e o planejamento de caminhos é fundamentado na cinemática do veículo. Considera-se o mesmo como um corpo rígido contendo quatro rodas em contato com o solo, sendo a sua orientação determinada pelas rodas dianteiras, tal como apresentado na Figura 1.

Para obtenção das equações cinemáticas que regem o movimento do veículo, utiliza-se o modelo de bicicleta (Pineiro, 2009). O quadro de referência do veículo, que descreve o seu caminho plano, tem origem no centro do eixo traseiro sendo denotado por P na Figura 1. A escolha deste ponto se dá por serem suas coordenadas cartesianas uma saída diferencialmente plana para o sistema, ou seja, uma vez especificadas estas variáveis e suas derivadas de segunda ordem, todas as outras variáveis de estado e a entrada do sistema são também definidas (Revoredo et al., 2016).

O raio de curvatura (R) do caminho é dado pela Equação (1), em que L é a distância entre os eixos do veículo e ϕ o seu ângulo de esterçamento. As especificações destes e dos outros parâmetros do modelo são resumidos na Tabela 1. Onde o ângulo de esterçamento (ϕ) é limitado.

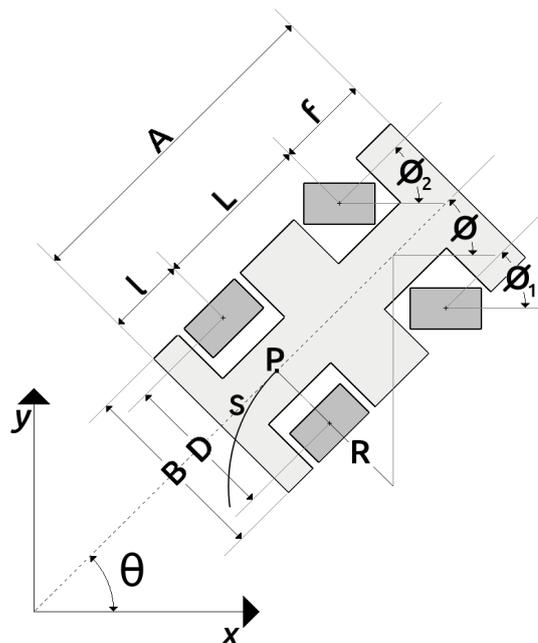


Figura 1. Modelo cinemático do veículo.

Tabela 1. Parâmetros da Cinemática do Veículo.

Parâmetro	Descrição
A	Comprimento do veículo
B	Largura do veículo
L	Distância entre os eixos do veículo
D	Distância entre as rodas do eixo dianteiro
f	Distância entre o eixo traseiro e a traseira do veículo
l	Distância entre o eixo dianteiro e a dianteira do veículo
θ	Ângulo do veículo em relação ao eixo x
R	Raio de curvatura instantâneo
P	Ponto de referência entre eixo traseiro
V	Velocidade linear
ϕ	Ângulo de esterçamento do veículo
ϕ_i	Ângulo de esterçamento da roda i ($i = 1, 2$)
$\dot{\phi}$	Velocidade angular de esterçamento

$$R = \frac{L}{\tan(\phi)} \quad (1)$$

$$\phi_{\min} \leq \phi \leq \phi_{\max} \quad (2)$$

A distância (S) percorrida pelo ponto de guiamento e a sua derivada (V) são definidas por:

$$S = R \cdot \theta \quad (3)$$

$$\dot{\theta} = \frac{V}{R} \quad (4)$$

Sendo a decomposição da velocidade, no plano x e y :

$$\dot{x}(t) = V \cdot \cos(\theta(t)) \quad (5)$$

$$\dot{y}(t) = V \cdot \sin(\theta(t)) \quad (6)$$

A inclinação do veículo (θ) e a sua derivada podem ser expressas em função de x e y , da seguinte maneira:

$$\theta(t) = \arctan\left(\frac{\dot{y}(t)}{\dot{x}(t)}\right) \quad (7)$$

$$\dot{\theta}(t) = \frac{\dot{x} \cdot \ddot{y} - \ddot{x} \cdot \dot{y}}{\dot{x}^2 + \dot{y}^2} \quad (8)$$

Assim, a velocidade linear do automóvel e o seu ângulo de esterçamento são definidos:

$$V = \pm \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2} \quad (9)$$

$$\phi = \arctan\left(\frac{L\dot{\theta}}{V}\right) \quad (10)$$

A distância entre as rodas de um mesmo eixo implica em raios de curvatura diferentes para um dado ângulo de esterçamento. Para adaptar o modelo cinemático de maneira a representar adequadamente a movimentação do veículo, utiliza-se a geometria de Ackermann para o sistema de esterçamento (Gillespie, 1992), a qual especifica ângulos distintos para as rodas dianteiras, ϕ_1 e ϕ_2 , dados pela Equação (11).

$$\phi_1, \phi_2 = \arctan\left(\frac{L}{R \pm \frac{D}{2}}\right) \quad (11)$$

Além disso, as rodas devem assumir velocidades diferentes entre si (Hartani et al., 2009). As velocidades das rodas

traseiras direita e esquerda, V_{td} e V_{te} , respectivamente, e das rodas dianteira direita e esquerda, V_{fd} e V_{fe} , são determinadas pelas Equações (12), (13) e (14).

$$V_{td}, V_{te} = V \cdot \left(1 \pm \frac{D}{2 \cdot R}\right) \quad (12)$$

$$V_{fd} = \frac{V_{td}}{\cos(\phi_1)} \quad (13)$$

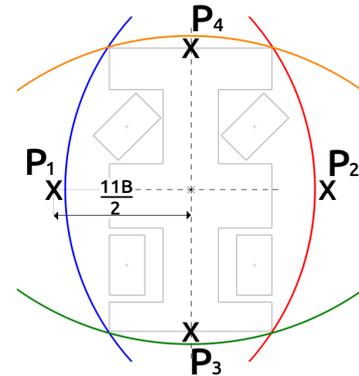
$$V_{fe} = \frac{V_{te}}{\cos(\phi_2)} \quad (14)$$

3. MODELAGEM DO AMBIENTE

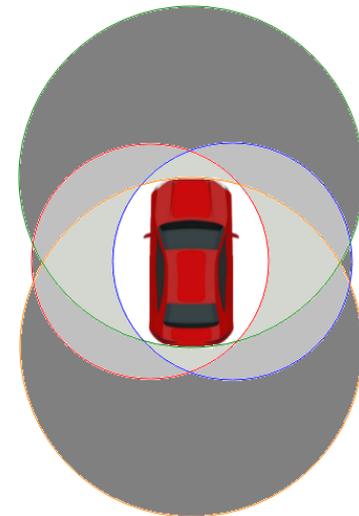
O ambiente onde o veículo se movimenta é representado por um mapa geométrico (Thrun, 2003), compondo uma abstração de alto nível em uma abordagem de fácil interpretação, implementação computacional simples e representação consistente.

3.1 Presença de Obstáculos Fixos

Para detectar obstáculos no ambiente que o veículo transita, as regiões de detecção são definidas com base em quatro pontos de referência (Argento et al., 2020), conforme exemplificado na Figura 2. As zonas de detecção de



(a) Quatro pontos de referência de detecção.



(b) Zonas de detecção.

Figura 2. Zonas de detecção de obstáculos.

obstáculos são círculos, centrados em pontos de referência (P_1, P_2, P_3 e P_4), cujos raios são definidos por:

$$r_1 = r_2 = \sqrt{\left(\frac{A}{2}\right)^2 + (6,5 \cdot B)^2} \quad (15)$$

$$r_3 = r_4 = \sqrt{(A)^2 + \left(\frac{B}{2}\right)^2} \quad (16)$$

onde A e B denotam o comprimento e a largura do veículo, respectivamente.

Conhecendo os obstáculos ao redor, se ao longo de todo o caminho previsto, nenhum obstáculo pertencer às quatro zonas de detecção simultaneamente, então o caminho está livre de obstáculos.

4. PLANEJAMENTO DE CAMINHOS

O planejamento de caminhos proposto é realizado utilizando polinômios de quinto grau (Demirli and Khoshnejad, 2009) para duas variáveis de uma função espacial, garantindo alta exatidão de rastreamento enquanto evitam modos naturais da estrutura mecânica ou do sistema de servo-controle (Erkorkmaz and Altintas, 2001), conforme a seguinte equação:

$$\begin{cases} x(s) = \sum_{i=0}^5 C_{xi}s^i \\ y(s) = \sum_{i=0}^5 C_{yi}s^i \end{cases} \quad (17)$$

em que s denota a posição do veículo no espaço bidimensional. O caminho $P(s)$ é definido por:

$$P(s) = (x(s), y(s)) \quad (18)$$

A lei do movimento é imposta da seguinte forma:

$$S = S(t) \quad (19)$$

Assim, a velocidade e a aceleração do veículo são dadas por:

$$\dot{P} = \frac{dP}{ds} \dot{S}(t) \quad (20)$$

$$\ddot{P} = \frac{dP}{ds} \ddot{S}(t) + \frac{d^2P}{ds^2} \dot{S}(t) \quad (21)$$

A parametrização leva a:

$$\mathbf{x} = [x_0 \ x_f \ V_s \cos(\theta_0) \ V_s \cos(\theta_f) \ x_1]^\top \quad (22)$$

$$\mathbf{y} = [y_0 \ y_f \ V_s \sin(\theta_0) \ V_s \sin(\theta_f) \ y_1]^\top \quad (23)$$

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ k_f^5 & k_f^4 & k_f^3 & k_f^2 & k_f & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5k_f^4 & 4k_f^3 & 3k_f^2 & 2k_f & 1 & 0 \\ k_1^5 & k_1^4 & k_1^3 & k_1^2 & k_1 & 1 \end{bmatrix} \quad (24)$$

$$\mathbf{c}_x = [C_{x_5} \ C_{x_4} \ C_{x_3} \ C_{x_2} \ C_{x_1} \ C_{x_0}]^\top \quad (25)$$

$$\mathbf{c}_y = [C_{y_5} \ C_{y_4} \ C_{y_3} \ C_{y_2} \ C_{y_1} \ C_{y_0}]^\top \quad (26)$$

Os parâmetros de entrada são as posições inicial e final (x_0, y_0 e x_f, y_f) e orientações (θ_0 e θ_f).

Para divergir os caminhos possíveis e encontrar o melhor, alguns parâmetros são variados. A constante k_f é definida conforme a Equação 27. A constante k_1 deve ser maior que zero e diferente de k_f . O ponto auxiliar (x_1, y_1) é definido no intervalo $x_0 < x_1 < x_f$ e $y_0 < y_1 < y_f$ e pode ocorrer entre os pontos inicial e final ($k_1 < k_f$) ou após este último ($k_1 > k_f$), mas a curva é descrita apenas até o ponto final (x_f e y_f). A velocidade do veículo pode ter duas direções, movimento para frente ($V_s = 1$) ou para trás ($V_s = -1$).

$$k_f \geq \sqrt{(x_f - x_0)^2 + (y_f - y_0)^2} \quad (27)$$

$$\mathbf{c}_x = \mathbf{T}^{-1} \times \mathbf{x} \quad (28)$$

$$\mathbf{c}_y = \mathbf{T}^{-1} \times \mathbf{y} \quad (29)$$

4.1 Implementação Computacional

A determinação dos diferentes caminhos, dentre as possibilidades que a parametrização temporal proporciona, é realizada computacionalmente. Tendo como informações as configurações espaciais iniciais e finais desejadas e as restrições impostas pela cinemática do veículo e pelo ambiente. Um algoritmo genético (AG) é associado à interpolação polinomial, para escolher o caminho nos critérios pré-definidos.

Algoritmos genéticos são métodos de otimização e busca inspirados nos mecanismos de evolução de populações de seres vivos (Holland et al., 1992). Caracterizam-se pelo uso do princípio da seleção natural e sobrevivência do mais apto, declarado em 1859 pelo naturalista e fisiologista inglês Charles Darwin (Darwin, 2020). Essa classe de algoritmo permite a utilização de diferentes representações das variáveis de entrada, como real e binária, bem como diferentes níveis de precisão ou mesmo restrições conforme o problema, formando os chamados “cromossomos” que fundamentam um indivíduo. Um grupo de indivíduos, para você, é conhecido como população.

Seguindo a inspiração mencionada acima, o algoritmo busca soluções variando suas entradas de acordo com dois parâmetros de objetivos diferentes. Para potencializar a exploração do espaço de busca, são aplicadas mutações individuais nos cromossomos, buscando unir as melhores características e obter avaliações aprimoradas através do cruzamento das mesmas. Por fim, a seleção natural conhecida pela teoria darwiniana é representada por uma função responsável por selecionar os indivíduos que compõem a próxima população.

Dentre as várias implementações possíveis para a escolha da mais adequada, a abordagem aqui implementada utiliza a proposta de otimização de funções feita por R. Houck et al. (Houck et al., 1995). É um algoritmo que avalia um grupo (população) de informações de entrada (indivíduos) e, com base nas restrições impostas, determina o caminho mais adequado, arbitrado em gerações. No presente trabalho, as entradas são os parâmetros da parametrização que variam os caminhos possíveis (k_f, x_1, y_1, k_1 e V_s).

Cada caminho gerado é submetido à função de custo (f_c) definida pela Equação 30, e o AG deve minimizá-lo.

Tabela 2. Parâmetros do AG.

Parâmetro	Especificações
Representação	Binária
Domínio x_1	Entre x_0 e x_f
Domínio y_1	Entre y_0 e y_f
Domínio k_1	Entre 0,01 e $3k_{f_{\min}}$
Domínio k_f	Entre $k_{f_{\min}}$ e $3k_{f_{\min}}$
Domínio V_s	-1 ou 1
Precisão	1μ
Aptidão	Avaliação
População	80
Gerações	100
Taxa de <i>crossover</i>	60%
Taxa de mutação	4%
Função de seleção	Seleção por Roleta

$$f_c = \sqrt{|S|^2 + |\phi|_{\max}^2} \quad (30)$$

onde $|S|$ é o comprimento do trajeto.

Para lidar com outras restrições, tais como evitar obstáculos, são utilizadas técnicas de penalização (Reeves, 1997), ou seja, se alguma outra restrição não for respeitada, o algoritmo suprime o resultado natural e o substitui por um valor inadequado. Neste caso, o comprimento do caminho ($|S|$) assume um valor cem vezes maior. Ao longo de todas as gerações, se o caminho escolhido atender aos requisitos, ele é classificado como válido.

A Tabela 2 aponta as especificações do AG aplicado.

5. CONTROLADOR DE PERSEGUIÇÃO PURA

O seguimento de caminho visa projetar técnicas de controle para posicionar o veículo em um caminho predeterminado (Wang et al., 2019), tarefa para a qual muitos autores apresentam modelos baseados em cinemática (Gonzalez et al., 2016). Neste trabalho, visando fornecer um ambiente de simulação física 3D de um estacionamento autônomo e auxiliar na validação do algoritmo de planejamento de caminhos, um controlador é utilizado para manipular as ações do veículo de modo a rastrear o caminho de referência. A estratégia de controle é chamada perseguição pura. Trata-se de um algoritmo de rastreamento, baseado na geometria da cinemática do veículo (Coulter, 1992). Tomando como referência o modelo cinemático tipo bicicleta e dado um ponto alvo, l_d e α são definidos respectivamente como a distância e a inclinação entre ele e o veículo, conforme mostrado na Figura 3(a).

Definindo o raio de curvatura (R), chega-se ao triângulo isósceles mostrado na Figura 3(b) e tem-se a seguinte relação:

$$R = \frac{l_d}{2\text{sen}(\alpha)} \quad (31)$$

Correlacionando com o modelo cinemático, o ângulo de direção (ϕ) pode ser calculado como:

$$\phi = \arctan\left(\frac{2L\text{sen}(\alpha)}{l_d}\right) \quad (32)$$

onde L é a distância entre os eixos do veículo.

O objetivo é controlar o veículo em todos os pontos do caminho sequencialmente. No entanto, como essa sequência

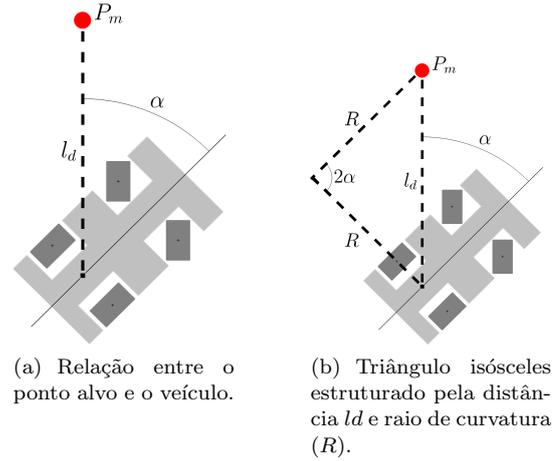


Figura 3. Geometria do controlador de perseguição pura.

pré-definida é geralmente descrita por pontos muito próximos uns dos outros, é comum ter dificuldade em alcançar o próximo ponto em casos de curvas mais acentuadas ou na recuperação de uma perturbação. Portanto, busca-se adaptar o algoritmo para torná-lo mais robusto (Coulter, 1992). É implementada uma distância de antecipação (L_{ad}), definida como a distância mínima necessária para que o algoritmo consiga antecipar o próximo ponto alvo e, assim, suavizar a curva necessária e estabilizar a rota.

6. RESULTADOS

Para os resultados apresentados, as características consideradas para o veículo de pequena escala estão presentes na Tabela 3. As vagas têm suas dimensões definidas em função das características do veículo (Vorobieva et al., 2015) e seus dados são apresentados na Tabela 4.

A abordagem é baseada no ambiente de simulação Py-Bullet (Coumans and Bai, 2016). Trata-se de um módulo Python para simulação física para robótica, jogos, efeitos visuais e aprendizagem de máquinas, com foco na transferência da simulação para o robô real (Michalik and Janota, 2020), aproximando a emulação computacional de resultados experimentais. O ambiente permite uma variedade de simulações, incluindo, dinâmica, cinemáticas e detecção de colisão.

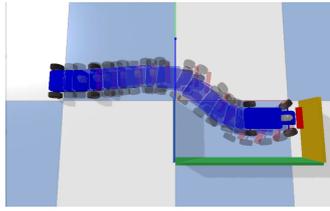
Tabela 3. Parâmetros do veículo.

Parâmetro	Valor
L	325 mm
B	290 mm
f	100 mm
l	50 mm
A	475 mm
ϕ	$-45^\circ/45^\circ$
$ \phi _{\max}$	1 rad/s
$ V _{\max}$	0,1 m/s

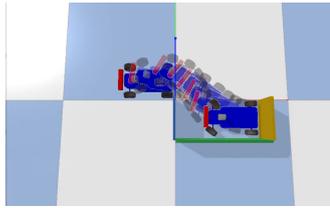
Tabela 4. Dados das Vagas.

Tipo de vaga	Comprimento	Largura
Paralelo de frente	1091,8 mm	495,6 mm
Paralelo de ré	729,9 mm	335,1 mm
Perpendicular de frente/ré	631,7 mm	519,2 mm
Inclinado de frente/ré	631,7 mm	519,2 mm

As Figuras 4, 5 e 6 apresentam os resultados para o estacionamento paralelo, perpendicular e inclinado respectivamente.

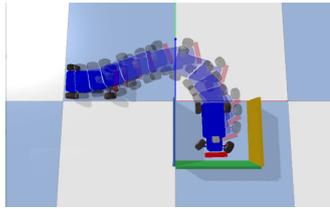


(a) Frente.

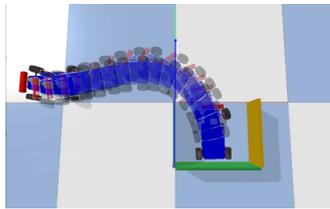


(b) Ré.

Figura 4. Estacionamento paralelo.

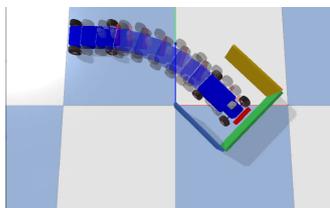


(a) Frente.

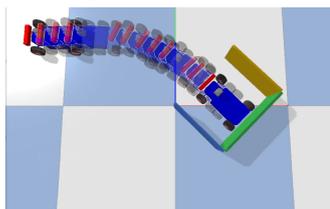


(b) Ré.

Figura 5. Estacionamento perpendicular.



(a) Frente.



(b) Ré.

Figura 6. Estacionamento inclinado.

Os resultados mostram a capacidade de o veículo de realizar movimentos contínuos e suaves enquanto rastreia eficientemente os caminhos de referência sem interrupções de movimento, efetuando as diferentes categorias de estacionamento e desviando dos obstáculos definidos pelas vagas.

6.1 Comparação dos Resultados

O algoritmo (AG) aqui proposto é comparado com outra abordagem. E. Argento et al. (Argento et al., 2020) também utiliza uma técnica de interpolação de quinto grau para gerar caminhos, porém combinado com um algoritmo de busca por força bruta (FB).

Inspirado no método de parametrização espacial, a abordagem que utiliza busca por força bruta leva a:

$$\mathbf{x} = \begin{bmatrix} x_0 \\ x_f \\ V_s \cos(\theta_0) \\ V_s \cos(\theta_f) \\ -V_s \sin(\theta_0) \dot{\theta}_0 \\ -V_s \sin(\theta_f) \dot{\theta}_f \end{bmatrix} \quad (33)$$

$$\mathbf{y} = \begin{bmatrix} y_0 \\ y_f \\ V_s \sin(\theta_0) \\ V_s \sin(\theta_f) \\ V_s \cos(\theta_0) \dot{\theta}_0 \\ V_s \cos(\theta_f) \dot{\theta}_f \end{bmatrix} \quad (34)$$

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ k_f^5 & k_f^4 & k_f^3 & k_f^2 & k_f & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5k_f^4 & 4k_f^3 & 3k_f^2 & 2k_f & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 20k_f^3 & 12k_f^2 & 6k_f & 2 & 0 & 0 \end{bmatrix} \quad (35)$$

$$\mathbf{c}_x = [C_{x_5} \ C_{x_4} \ C_{x_3} \ C_{x_2} \ C_{x_1} \ C_{x_0}]^\top = \mathbf{T}^{-1} \times \mathbf{x} \quad (36)$$

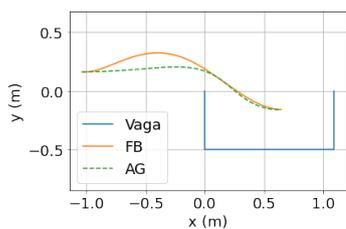
$$\mathbf{c}_y = [C_{y_5} \ C_{y_4} \ C_{y_3} \ C_{y_2} \ C_{y_1} \ C_{y_0}]^\top = \mathbf{T}^{-1} \times \mathbf{y} \quad (37)$$

Os parâmetros de entrada são as posições e orientações inicial (x_0, y_0, θ_0) e final (x_f, y_f, θ_f) . Diferentes caminhos são criados variando k_f como descreve a Equação (27). Por sua vez, $\dot{\theta}_0$ e $\dot{\theta}_f$ são definidos por:

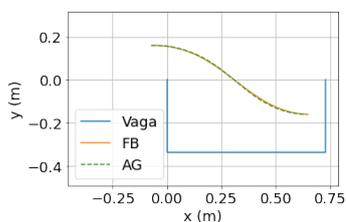
$$\frac{\tan(\phi_{\min})}{L} \leq \dot{\theta}_0, \dot{\theta}_f \leq \frac{\tan(\phi_{\max})}{L} \quad (38)$$

Para comparar os resultados, ambos os caminhos gerados tiveram seu rastreamento realizado utilizando o controlador de perseguição pura. Diante dos resultados, foram avaliados o tempo de viagem e a distância percorrida.

As Figuras 7, 8 e 9 apresentam a comparação dos algoritmos para o estacionamento paralelo, perpendicular e inclinado respectivamente. A Tabela 5 ostenta o tempo de viagem e a distância percorrida para cada algoritmo.

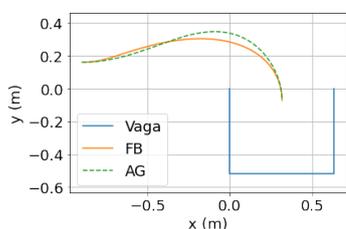


(a) Frente.

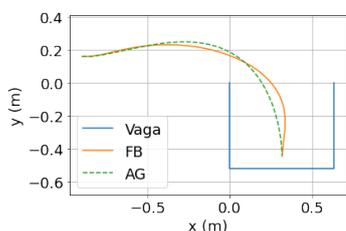


(b) Ré.

Figura 7. Resultados de estacionamento paralelo.

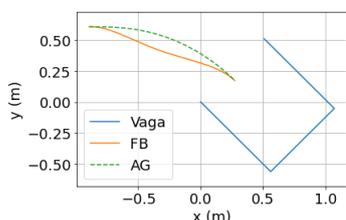


(a) Frente.

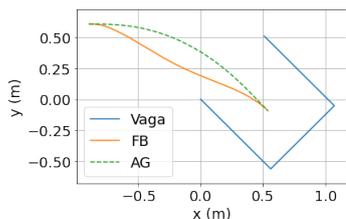


(b) Ré.

Figura 8. Resultados de estacionamento perpendicular.



(a) Frente.



(b) Ré.

Figura 9. Resultados de estacionamento inclinado.

Tabela 5. Resultados.

Tipo de vaga	AG	FB
Paralelo de frente	1,768 m - 17,59 s	1,830 m - 18,20 s
Paralelo de ré	0,802 m - 8,07 s	0,805 m - 8,09 s
Perpendicular de frente	1,467 m - 14,71 s	1,426 m - 14,28 s
Perpendicular de ré	1,645 m - 16,39 s	1,679 m - 16,75 s
Inclinado de frente	1,283 m - 12,74 s	1,255 m - 12,48 s
Inclinado de ré	1,655 m - 16,37 s	1,607 m - 15,95 s

Os resultados mostram que conforme o tempo e a distância percorrida nas manobras de estacionamento, o AG obteve o melhor desempenho para o estacionamento paralelo (frente e ré) e perpendicular de ré.

7. CONCLUSÃO

Este artigo apresenta um algoritmo de planejamento de caminhos baseado na combinação entre interpolação polinomial e um algoritmo genético de otimização, capaz de realizar manobras de estacionamento autônomo de automóveis de maneira contínua e suave, evitando-se obstáculos pela rota e respeitando os limites físicos impostos pelo veículo. A modelagem do automóvel é baseada na geometria de Ackermann para evitar derrapagem e o ambiente é representado por um mapa geométrico.

Como exemplo de aplicação, são simulados diferentes categorias de estacionamento. A proposta prova-se viável e útil para desenvolvimentos futuros para o planejamento de caminhos. A comparação de desempenho também é realizada com base em um algoritmo encontrado na literatura. Além disso, são apresentados resultados de simulações em ambiente 3D para ajudar a validar a viabilidade do algoritmo proposto, que estabelece as bases para estudos mais amplos.

Cabe ressaltar que o processo de aplicação do algoritmo genético é quase exclusivamente empírico, demandando experiência para definir seus parâmetros. Além disso, sua execução pode demandar um tempo vultoso.

Mesmo com os resultados obtidos em ambiente de simulação física, deve-se considerar a dificuldade de transferir a experiência simulada para o mundo real. Recomenda-se que investigações futuras incluam o detalhamento na utilização do modelo dinâmico e a comparação de outras técnicas de otimização com o algoritmo aqui proposto, bem como com outras soluções propostas na literatura, além de validação experimental.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Os autores agradecem, também, à Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ) pelos seus suportes financeiros no desenvolvimento deste trabalho.

REFERÊNCIAS

Argento, E.V., Vieira, R.P., and Revoredo, T.C. (2020). Estacionamento paralelo autônomo de veículos leves de passeio. *Anais da Sociedade Brasileira de Automática*, 2(1).

- Coulter, R.C. (1992). Implementation of the pure pursuit path tracking algorithm. Technical report, Carnegie-Mellon UNIV Pittsburgh PA Robotics INST.
- Coumans, E. and Bai, Y. (2016). Pybullet, a python module for physics simulation for games, robotics and machine learning. URL <http://pybullet.org>.
- Darwin, C. (2020). *A origem das espécies*. EDITORA BIBLIOMUNDI SERVIÇOS DIGITAIS LTDA.
- Demirli, K. and Khoshnejad, M. (2009). Autonomous parallel parking of a car-like mobile robot by a neuro-fuzzy sensor-based controller. *Fuzzy Sets and Systems*, 160(19), 2876–2891. doi:10.1016/j.fss.2009.01.019. URL <https://doi.org/10.1016/j.fss.2009.01.019>.
- Erkorkmaz, K. and Altintas, Y. (2001). High speed CNC system design. part i: jerk limited trajectory generation and quintic spline interpolation. *International Journal of Machine Tools and Manufacture*, 41(9), 1323–1345. doi:10.1016/s0890-6955(01)00002-5. URL [https://doi.org/10.1016/s0890-6955\(01\)00002-5](https://doi.org/10.1016/s0890-6955(01)00002-5).
- Gillespie, T.D. (1992). Fundamentals of vehicle dynamics. *SAE International*.
- Godberg, D.E. (1989). Genetic algorithms in search. *Optimization, and Machine Learning*.
- Gonzalez, D., Perez, J., Milanes, V., and Nashashibi, F. (2016). A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(4), 1135–1145. doi:10.1109/tits.2015.2498841.
- Hartani, K., Bourahla, M., Miloud, Y., and Sekour, M. (2009). Electronic differential with direct torque fuzzy control for vehicle propulsion system. *Turk J Elec Eng end Comp Sci*, 17(1).
- Holland, J.H. et al. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.
- Houck, C.R., Joines, J., and Kay, M.G. (1995). A genetic algorithm for function optimization: a matlab implementation. *Ncsu-ie tr*, 95(09), 1–10.
- Hussain, R. and Zeadally, S. (2019). Autonomous cars: Research results, issues, and future challenges. *IEEE Communications Surveys & Tutorials*, 21(2), 1275–1313. doi:10.1109/comst.2018.2869360. URL <https://doi.org/10.1109/comst.2018.2869360>.
- Michalik, R. and Janota, A. (2020). The PyBullet module-based approach to control the collaborative YuMi robot. In *2020 ELEKTRO*. IEEE. doi:10.1109/elektro49696.2020.9130233. URL <https://doi.org/10.1109/elektro49696.2020.9130233>.
- Ni, S., Liu, Z., Cai, Y., and Wang, X. (2018). Modelling of ship's trajectory planning in collision situations by hybrid genetic algorithm. *Polish Maritime Research*, 25(3), 14–25. doi:10.2478/pomr-2018-0092. URL <https://doi.org/10.2478/pomr-2018-0092>.
- Pinheiro, B.C. (2009). Sistema de controle tempo real embarcado para automação de manobra de estacionamento. *Dissertação de Mestrado em Automação e Sistemas, Universidade Federal de Santa Catarina*.
- Rashid, M.M., Rahman, M.M., Islam, M.R., Alwahedy, O.N., and Abdullahi, A. (2019). Autonomous 4wd smart car parallel self-parking system by using fuzzy logic controller. *American International Journal of Sciences and Engineering Research*, 2(2), 1–31.
- Reeves, C.R. (1997). Genetic algorithms for the operations researcher. *INFORMS journal on computing*, 9(3), 231–250.
- Revoredo, T.C., Mora-Camino, F., and Slama, J. (2016). A two-step approach for the prediction of dynamic aircraft noise impact. *Aerospace Science and Technology*, 59, 122–131. doi:10.1016/j.ast.2016.10.017. URL <https://doi.org/10.1016/j.ast.2016.10.017>.
- Thrun, S. (2003). Robotic mapping: a survey. *Exploring Artificial Intelligence in the NewMillenium, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.*, 1–35.
- Tu, J. and Yang, S.X. (2003). Genetic algorithm based path planning for a mobile robot. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 1, 1221–1226. IEEE.
- Tuncer, A. and Yildirim, M. (2012). Dynamic path planning of mobile robots with improved genetic algorithm. *Computers & Electrical Engineering*, 38(6), 1564–1572.
- United States. Department of Transportation. National Highway Traffic Safety Administration (2021). Active park assist draft test procedure validation. *U.S. Department of Transportation*. doi:10.21949/1526030. URL <https://rosap.ntl.bts.gov/view/dot/58216>.
- Vorobieva, H., Glaser, S., Minoiu-Enache, N., and Mammam, S. (2015). Automatic parallel parking in tiny spots: Path planning and control. *IEEE Transactions on Intelligent Transportation Systems*, 16(1), 396–410.
- Wang, D., Liang, H., Mei, T., and Zhu, H. (2011). Research on self-parking path planning algorithms. In *Proceedings of 2011 IEEE International Conference on Vehicular Electronics and Safety*. IEEE. doi:10.1109/icves.2011.5983825. URL <https://doi.org/10.1109/icves.2011.5983825>.
- Wang, H., Liu, B., Ping, X., and An, Q. (2019). Path tracking control for autonomous vehicles based on an improved MPC. *IEEE Access*, 7, 161064–161073. doi:10.1109/access.2019.2944894.
- Zhu, C. and Rajamani, R. (2006). Global positioning system-based vehicle control for automated parking. *I MECH E Part D Journal of Automobile Engineering*, 220(1), 37–52.