Trabalho de Otimização
Relatório Técnico

Vinícius César Sena Torres (RA 22409225)

25 de Outubro de 2025

1. Introdução

Este relatório descreve a resolução do problema de levar um robô do ponto inicial $(0, 0)$ ao ponto final $(10, 10)$ no intervalo contínuo $t \in [0, 5]$, minimizando a energia acumulada ao longo do percurso. A energia integra o quadrado da velocidade instantanea somado ao termo $\sin(t)$. O problema foi formulado como um Problema de Programacao Nao Linear (NLP), modelado com Pyomo e resolvido com o solver IPOPT.

Os objetivos principais foram:

- Formular matemáticamente o problema contínuo e sua versão discretizada.
- Implementar o modelo no Pyomo.
- Resolver o problema com IPOPT e analisar os resultados.
- Registrar os achados e discutir possiveis extensoes.

2. Formulação Matemática

2.1 Problema contínuo

A energia a minimizar é dada por:

$$E = \int_0^5 \left(v(t)^2 + \sin(t)\right) \, dt,$$

onde $v(t) = \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2}$.

As restricoes de contorno sao:

$$x(0) = 0, \quad y(0) = 0, \qquad x(5) = 10, \quad y(5) = 10.$$

2.2 Discretizacao

O intervalo $[0, 5]$ foi dividido em $N = 5$ subintervalos uniformes. Define-se $t_i = i \cdot \Delta t$ com $\Delta t = 1$ e $i = 0, 1, \ldots, 5$.

2.3 Modelo discretizado

Para cada ponto t_i determinam-se variaveis x_i e y_i. A energia aproximada pela regra dos trapezios resulta em:

$$E \approx \sum_{i=0}^{N-1} \frac{1}{2} \left[\left(v_i^2 + \sin(t_i)\right) + \left(v_{i+1}^2 + \sin(t_{i+1})\right) \right] \Delta t,$$

com

$$v_i^2 = \left(\frac{x_{i+1} - x_i}{\Delta t} \right)^2 + \left(\frac{y_{i+1} - y_i}{\Delta t} \right)^2.$$

As condicoes de contorno discretas permanecem $x_0 = 0$, $y_0 = 0$, $x_N = 10$, $y_N = 10$.

3. Justificativa do Solver IPOPT

O IPOPT (Interior Point OPTimizer) foi escolhido por:

- Ser especializado em problemas nao lineares suaves com restrções de igualdade e desigualdade.
- Utilizar metodos de pontos interiores estáveis e com boa taxa de convergência.
- Integrar-se nativamente ao Pyomo, dispensando codificação manual de derivadas.
- Operar com software livre, com desempenho comparavel a solvers comerciais.

4. Implementação em Pyomo

O modelo foi implementado com:

- **ConcreteModel** para representar o problema.
- **RangeSet** para indices de tempo.
- **Var** para coordenadas x_i e y_i.
- Uma funcao objetivo Python que computa a energia via regra dos trapezios.
- **SolverFactory('ipopt')** apontando para o executavel instalado via Miniconda.

5. Resultados do Caso Base ($N = 5$)

5.1 Diagnostico do solver

O IPOPT retornou status `ok` com condicao de parada `optimal`. A energia minimizada foi:

$$E^* = 35.949741.$$

5.2 Trajetoria otima

A trajetória obtida é quase linear, conforme a Tabela 1. As coordenadas x_i e y_i sao praticamente iguais, indicando trajeto próximo da diagonal $y = x$.

Table 1: Posições ótimas no caso base

Tempo (s)	x_i	y_i
0.0	0.000000	0.000000
1.0	3.529412	3.529412
2.0	5.294118	5.294118
3.0	7.058824	7.058824
4.0	8.823529	8.823529
5.0	10.000000	10.000000

5.3 Interpretação

- O termo dominante da energia e v^2; $\sin(t)$ introduz pequenas correções.
- A trajetoria linear uniforme é energeticamente eficiente para ligar os pontos extremos sem outras restricoes.
- A convergencia ocorreu sem violacoes, confirmando a adequacao do solver e do modelo.

6. Conclusões do Caso Base

- A combinacao Pyomo + IPOPT resolve o problema de forma direta e robusta.
- O caminho quase retilineo confirma a intuicao fisica de menor distancia com custo quadratico de velocidade.
- As condicoes de contorno e relacoes de velocidade foram satisfeitas exatamente.
- O modelo serve de base para adicionar novas restricoes ou funcoes objetivo.

7. Experimentos Extras

Após validar o caso base foram conduzidas variações para analisar sensibilidades e restrições adicionais.

7.1 Sensibilidade ao refinamento temporal

O intervalo $[0, 5]$ foi subdividido com diferentes números de subintervalos N. A Tabela 2 mostra a energia obtida em cada caso. Mesmo com a energia ligeiramente crescente (devido ao termo $\sin(t)$ ser avaliado em mais pontos positivos), a trajetória permanece essencialmente linear.

Table 2: Energia em função do refinamento temporal

Subintervalos (N)	Energia
5	35.9497
10	38.2014
20	39.4223
40	40.0597

7.2 Checkpoint intermediário

Ao impor a passagem pelo ponto $(5, 6)$ no instante $t = 2.5$, a energia sobe para 36.364 unidades (acrescimo aproximado de 1,2%). A trajetória desvia temporariamente da diagonal para satisfazer a restrição, ilustrando cenários com inspeções ou paradas obrigatórias.

7.3 Penalização adicional

Incluiu-se um termo $0.01[(x_i - 10)^2 + (y_i - 10)^2]$ na função objetivo, incentivando aproximação precoce ao destino. A energia resultante foi 39.390 unidades. As coordenadas intermediárias ficam mais próximas de $(10, 10)$, o que pode ser desejável para manter o robô sob cobertura de comunicação ou sensores.

7.4 Biblioteca de pistas

Foram comparadas três configurações padronizadas:

- **Linha direta:** modelo original, energia de 35.950 unidades.
- **Checkpoint central:** obrigatoriedade em $(5, 6)$, energia de 36.364 unidades.
- **Corredor guiado:** restrições $-1.5 \leq y - x \leq 1.5$ e penalização leve, energia de 43.080 unidades.

Os resultados (energias, status do solver e trajetórias) estão registrados graficamente no notebook, evidenciando o trade-off entre liberdade geométrica e custo energético.

7.5 Visualização animada

A trajetória de menor energia entre as pistas comparadas foi animada para facilitar a apresentação. A animação destaca a progressão temporal do robô e auxilia em defesas do trabalho ou em comunicação com clientes.

Esses experimentos mostram que a formulação e flexível e suporta demandas adicionais sem perda de robustez numérica.