

Demanda de Desenvolvimento

Sistema de Distribuicao Automatica de PAX

Bacia de Sergipe

1. Contexto Operacional

A operacao offshore na Bacia de Sergipe utiliza embarcacoes do tipo Surfer (capacidade: 24 passageiros) e Aqua Helix (capacidade: 100 passageiros) para transportar passageiros do Terminal Maritimo Inacio Barbosa (TMIB) ate plataformas offshore. Diariamente, um operador monta manualmente os itinerarios de cada embarcacao, decidindo quais plataformas cada barco visita, quantos passageiros embarcam e desembarcam em cada parada, e em que ordem as paradas ocorrem.

Esse processo manual e demorado, sujeito a erros, e depende fortemente da experiencia individual do operador. O objetivo desta demanda e automatizar a geracao desses itinerarios por meio de um solver algoritmico em Python.

2. Descricao do Problema

2.1 Entrada

O sistema recebe:

- Demanda de passageiros: tabela com plataformas de destino e quantidade de passageiros, discriminados por origem (TMIB ou PCM-09/M9).

2.2 Saida Esperada

Uma lista de itinerarios codificados, um por embarcacao, no formato:

```
SURFER 1905 06:30 TMIB +24/M6 -2/M9 -6 +2/M5 -3/PD01 (-2) -13
SURFER 1870 07:20 TMIB +24/M3 -10/M7 -9/M9 +6/M4 (-4) -5/M3 (-2)
SURFER 1930 07:30 TMIB +23/M2 -15/M9 -1 +1/B1 -3/B4 (-1) -4
```

Notacao:

Notacao	Significado
+N	Embarca N passageiros
-N	Desembarca N passageiros de origem TMIB
(-N)	Desembarca N passageiros de origem M9
/	Separador de paradas

2.3 Objetivo de Otimizacao

Minimizar a distancia total percorrida pela frota, respeitando todas as restricoes operacionais, e garantindo que toda a demanda seja atendida.

3. Regras de Negocio e Restricoes

3.1 Estrutura da Rota

Cada rota segue o padrao:

TMIB --> [paradas pre-M9] --> M9 --> [paradas pos-M9]

- TMIB e sempre a origem. Todos os passageiros embarcam no TMIB.

3.2 Visita Dupla (Loop)

Uma plataforma pode ser visitada duas vezes na mesma rota: uma vez pre-M9 (para desembarcar passageiros TMIB) e uma vez pos-M9 (para desembarcar passageiros M9). Exemplo: M3 recebe 10 pax TMIB na ida e 2 pax M9 na volta.

3.3 Capacidade

Embarcacao	Capacidade maxima
Surfer (1905, 1906, 1930, 1931, 1870)	24 pax
Aqua Helix	100 pax

A lotacao a bordo nunca pode exceder a capacidade em nenhum ponto da viagem.

3.4 Seguranca

Nunca desembarcar apenas 1 passageiro em uma plataforma. O minimo por desembarque e 2 passageiros.

3.5 Agrupamento Geografico (Clusters)

Plataformas proximas devem ser agrupadas na mesma rota para minimizar distancias:

Cluster	Plataformas	Dist. Interna
M6/B	M6, M8, B1, B2, B3, B4	0,42-1,48 NM
M2/M3	M2, M3	1,04 NM
M1/M7	M1, M7	1,24 NM
PDO/PGA	PDO1-3, PGA1-8	0,8-6,8 NM
PRB	PRB-01 (isolado)	20,15 NM do TMIB

Combinacoes proibidas (nunca na mesma rota):

- Cluster B com PDO/PGA (>10 NM entre eles)

Pares obrigatorios (devem ir juntos quando ambos tem demanda e cabem na capacidade):

- M2 + M3 (1,04 NM entre si)

3.6 Embarque M9 Proporcional

Cada barco embarca no M9 apenas os passageiros M9 que ele mesmo vai entregar nas paradas seguintes. Nao ha

embarque de M9 em excesso.

3.7 Rotas Diretas (sem M9)

Quando todas as plataformas de uma rota tem demanda exclusivamente TMIB (sem passageiros de origem M9), a rota pode ir direto do TMIB as plataformas, sem parar em M9, economizando o desvio de 8,76 NM.

3.8 Troca de Turma (cenario especial)

Em dias de troca de turma:

- Barcos precisam recolher rendidos concentrados em M9 e trazê-los de volta ao TMIB.

3.9 Aqua Helix - Restricoes de Gangway

O Aqua Helix opera via gangway (aproximacao lateral) e so pode atracar nas seguintes plataformas: M9, M6, B1, M7, M5, M3, PGA3. Cada parada do Aqua Helix incorre uma penalidade de 25 minutos para aproximacao.

4. Exemplo Completo

4.1 Demanda de entrada

Plataforma	Origem M9	Origem TMIB	Total
M2	0	15	15
M3	2	10	12
M4	4	5	9
M5	0	3	3
M6	0	2	2
M7	0	9	9
M9	0	7	7
B1	0	3	3
B4	1	4	5
PDO1	2	13	15
Total	9	71	80

Barcos disponiveis: SURFER 1905 (06:30), SURFER 1870 (07:20), SURFER 1930 (07:30)

4.2 Distribuicao esperada (solucao manual do operador)

Rota 1 - SURFER 1905 (06:30):

TMIB +24 / M6 -2 / M9 -6 +2 / M5 -3 / PDO1 (-2) -13

Logica: Sai do TMIB com 24 pax. Passa por M6 (proxima, 0,95 NM de M9) e desembarca 2 pax TMIB. Segue para M9, desembarca 6 TMIB e embarca 2 M9 (destinados a PDO1). Segue para M5 (1,58 NM de M9), desembarca 3 TMIB. Termina em PDO1 (6,68 NM de M9), desembarca 13 TMIB e 2 M9.

Rota 2 - SURFER 1870 (07:20):

TMIB +24 / M3 -10 / M7 -9 / M9 +6 / M4 (-4) -5 / M3 (-2)

Logica: Sai com 24 pax. Passa por M3 (pre-M9) e desembarca 10 TMIB. Passa por M7 (pre-M9) e desembarca 9 TMIB. Chega ao M9 com 5 pax, embarca 6 M9. Segue para M4 e desembarca 5 TMIB + 4 M9. Volta a M3 (visita dupla) e

desembarca 2 M9.

Rota 3 - SURFER 1930 (07:30):

TMIB +23 / M2 -15 / M9 -1 +1 / B1 -3 / B4 (-1) -4

Logica: Sai com 23 pax. Passa por M2 (pre-M9, 2,00 NM de M9), desembarca 15 TMIB. Segue para M9, desembarca 1 TMIB e embarca 1 M9. Segue para B1 (1,56 NM de M9), desembarca 3 TMIB. Termina em B4 (0,41 NM de B1), desembarca 4 TMIB + 1 M9.

Resultado: 80/80 pax entregues, 3 barcos utilizados, clusters respeitados.

5. Algoritmo Proposto (Visao Geral)

1. Ler entrada (demanda, barcos, distancias)
2. Subtrair rotas fixas (se houver) da demanda
3. Formar pacotes de demanda respeitando pares obrigatorios
4. Para cada atribuicao possivel de pacotes --> barcos:
 - a. Decidir quais paradas ficam pre-M9 e pos-M9
 - b. Ordenar paradas pre-M9 por caminho otimo a partir de TMIB
 - c. Ordenar paradas pos-M9 por caminho otimo a partir de M9
 - d. Calcular distancia total
5. Selecionar a atribuicao de menor distancia total
6. Gerar strings de rota codificadas
7. Validar: demanda atendida, capacidade respeitada

A decisao pre vs pos-M9 e o diferencial critico do algoritmo. Para cada plataforma atribuida a um barco:

- Somente M9 --> obrigatoriamente pos-M9

6. Stack Tecnologica

Componente	Detalhe
Linguagem	Python 3.12
Dependencias	openpyxl (Excel), bibliotecas padrao
Entrada	solver_input.xlsx (planilha com config, barcos, demanda)
Saida	distribuicao.txt (itinerarios codificados)
Dados	distplat.json, velocidades.txt, gangway.json

7. Arquivos do Repositorio

Arquivo	Descricao
solver.py	Solver v4 atual (resultados insatisfatorios)
solverGPT.py	Solver v4.1 com melhorias parciais
criarInputSolver.py	Gera template solver_input.xlsx
criarTabela6.py	Simula viagens e gera planilha de programacao
distplat.json	Matriz de distancias entre plataformas (NM)
velocidades.txt	Velocidades das embarcacoes (nos)
gangway.json	Plataformas com operacao de gangway
Demandas/demanda.xlsx	3 exemplos historicos demanda + distribuicao

8. Entregaveis Esperados

- Solver funcional que, dada uma demanda e embarcacoes disponiveis, gere itinerarios otimizados no formato codificado.

9. Roadmap Futuro (informativo)

- Fase 2: Coleta de dados - registrar cada (entrada, saida, correcao do operador) para acumular historico.

Documentacao de referencia completa disponivel no arquivo CLAUDE.md do repositorio.