

Tarefa 1: Explorar e Coordenar ações

Resgate de Vítimas de Catástrofes Naturais, Desastres ou Grandes Acidentes

1 Versão

2 Estrutura do documento

A seção 3 apresenta uma descrição da primeira parte do cenário de resgate após ocorrência de um evento catastrófico. A seção 4 descreve os objetivos da tarefa que devem atender os requisitos de modelagem, as fórmulas de cálculo de desempenho e os formatos dos arquivos de entrada (seção 5). A seção 6 trata da forma de entrega da tarefa.

3 Descrição do Cenário

Em um cenário de resgate de vítimas após um acidente, dois grupos de agentes artificiais (robôs) são lançados em uma área de risco. Todos os agentes iniciam na mesma posição base. Os grupos de agentes trabalham em duas etapas sequenciais. Na primeira etapa, os agentes exploradores têm por objetivo localizar as vítimas e construir um mapa da região. Na segunda etapa, os agentes socorristas devem levar suprimentos para as vítimas localizadas pelo explorador. Os socorristas só entram em ação somente quando os agentes exploradores finalizam as buscas.

Cada agente explorador constrói um mapa da região que explorou contendo os obstáculos e as vítimas encontradas (V_e). Ao localizar uma vítima, coleta seus sinais vitais (e.g. de respiração, pulsação, pressão). Cada agente tem um tempo limitado para explorar o ambiente (T_e) em função da capacidade da bateria. Terminado o período, cada explorador deve ir até a posição base. Uma vez que todos os agentes cheguem na base, eles montam um mapa unificado a partir dos mapas individuais. Um dos agentes exploradores repassa o mapa unificado para os agentes socorristas que também partem da mesma posição base. Os socorristas então devem separar as vítimas em grupos para que possam dividir a tarefa de socorro.

A figura 1 apresenta um exemplo ilustrativo de uma região 5 x 5 na qual temos os dois grupos de agentes na posição base. Os agentes exploradores são definidos pelo conjunto $A_e = \{E_1, E_2\}$, e os socorristas pelo conjunto $A_s = \{S_1, S_2\}$, o conjunto das vítimas dispersas no ambiente, $V = \{v_1, v_2, v_3\}$, e as barreiras (obstáculos), pelos quadrados pretos. Ressalta-se que os agentes não conhecem o tamanho do grid nem a indexação das células.

	-3	-2	-1	0	1
-3		v_1			
-2				v_2	
-1					v_3
0				$E_1 E_2$ $S_1 S_2$	
1					

Figura 1: Ambiente com os agentes na posição base, vítimas ($V=\{v_1, v_2, v_3\}$) e obstáculos na cor preta. Os agentes representam a posição de cada elemento do mapa de forma relativa à posição base (0, 0)

4 Objetivo da tarefa

Localizar o maior número das $|V|$ vítimas que estão dispersas em um ambiente desconhecido levando em conta que os agentes exploradores (A_e) tem um tempo T_e dado como parâmetro de entrada para encontrá-las. Cada agente explorador deve retornar ao seu ponto de resgate antes que T_e termine, caso contrário, ele é dado por perdido assim como o mapa que construiu até o momento. É importante ressaltar que os agentes exploradores não conhecem o tamanho do grid e, tão pouco, a identificação da posição em que se encontra. Os agentes terão que construir um sistema de coordenadas relativo à posição base. Os exploradores devem estar preparados para rodar em ambientes diversos (diferentes tamanhos, posições de obstáculos e vítimas) e para iniciar em diferentes posições base.

O segundo objetivo é unificar os mapas construídos individualmente e enviar aos agentes socorristas.

O grupo As deve agrupar as vítimas por algum critério definido pela equipe de projeto.

4.1 Parâmetros e requisitos

O número de agentes exploradores e socorristas não sofrerá variação durante todo o projeto:

$A_e = \{E_1, E_2, E_3, E_4\}$

$A_s = \{S_1, S_2, S_3, S_4\}$

Os demais parâmetros podem mudar a qualquer momento: dimensões do grid, vítimas, tempo limite de exploração e de socorro, posição base, etc.

Algoritmos:

- um algoritmo de IA (visto em sala de aula) para busca exploratória
- um algoritmo de IA (visto em sala de aula) para *clustering* das vítimas
- a construção e montagem do mapa é livre

4.2 Desempenho:

Considere os conjuntos abaixo nas fórmulas de cálculo de desempenho que vocês utilizarão para comparar as soluções em sala de aula e para analisar o desempenho do agente no artigo.

- V : conjunto de vítimas dispersas no ambiente, sendo $|V|$ o número de vítimas (cardinalidade do conjunto). Este parâmetro é calculado a partir da leitura do arquivo de entrada. As vítimas estão classificadas de acordo com o risco de morte. Portanto, o conjunto V é composto pela união dos seguintes conjuntos disjuntos ($V = \bigcup_{i=1}^4 V_i$):
 - V_1 : estado **CRÍTICO** (CLASSE=1)
 - V_2 : estado **INSTÁVEL** (CLASSE=2)
 - V_3 : estado **POTENCIALMENTE INSTÁVEL** (CLASSE=3)
 - V_4 : estado **ESTÁVEL** (CLASSE=4)
- V_e : conjunto de vítimas localizadas pelo explorador tal que $|V_e| \leq |V|$. O conjunto das vítimas encontradas também é composto pela união dos conjuntos disjuntos ($V_e = \bigcup_{i=1}^4 V_{e_i}$):
 - V_{e_1} : estado **CRÍTICO** (CLASSE=1)
 - V_{e_2} : estado **INSTÁVEL** (CLASSE=2)
 - V_{e_3} : estado **POTENCIALMENTE INSTÁVEL** (CLASSE=3)
 - V_{e_4} : estado **ESTÁVEL** (CLASSE=4)
- t_e : tempo efetivamente gasto pelo explorador tal que $t_e \leq T_e$

As métricas de desempenho são as seguintes (calculadas pelo simulador):

- **pve: porcentual de vítimas encontradas:**

$$pve = |V_e|/|V|$$

- **pte: porcentual do tempo de exploração utilizado:**

$$pte = t_e/T_e$$

- **veg: pontuação de vítimas encontradas ponderada por classe de gravidade** retrata a capacidade do agente em localizar vítimas em estado mais grave, daí o uso da ponderação mais alta para as vítimas mais graves.

$$veg = \frac{6|V_{e_1}| + 3|V_{e_2}| + 2|V_{e_3}| + |V_{e_4}|}{6|V_1| + 3|V_2| + 2|V_3| + |V_4|}$$

- **peg: acumulado da gravidade das vítimas encontradas sobre o acumulado dos valores de gravidade de todas as vítimas**

$$peg = \frac{\sum_{i=1}^{|V_e|} g_i}{\sum_{j=1}^{|V|} g_j}$$

5 Modelagem

5.1 Elementos básicos

AMBIENTE: tem a forma de um quadriculado com barreiras que impedem a passagem de uma célula para outra. A célula base da região é um dos parâmetros de entrada para o simulador do ambiente. As vítimas estão distribuídas no quadriculado e há somente uma vítima por célula. Nenhuma célula contém vítima e obstáculo.

AGENTES: A_e e A_s devem criar um sistema de coordenadas a partir da base (origem) onde estão inicialmente já que eles não têm acesso ao sistema de coordenadas utilizado pelo

simulador. O agente explorador consegue ler os sinais vitais de uma vítima somente se estiver na célula onde ela se encontra.

Sensores

- **colisão:** todo agente é dotado de um sensor de colisão para detectar os obstáculos e a saída da região do acidente. Quando executam uma ação de movimentação e batem em um obstáculo, recebem um sinal (BUMPED) do simulador do ambiente. O mesmo ocorre ao tentar fazer um movimento que o leve para fora do quadriculado.
- **obstáculos:** todo agente é dotado de um sensor de detecção de obstáculos nas posições da vizinhança imediata

Ações

- A_e e A_s são capazes de ir para qualquer célula na vizinhança da atual desde que esteja dentro do quadriculado e não tenha obstáculo. Portanto, podem se mover na horizontal, vertical ou diagonal;
- A_e e A_s podem andar nas diagonais se não há um obstáculo na posição de destino. Na figura 1, um agente pode passar de (1,1) para (2,2) e vice-versa;
- A_e e A_s podem ocupar a mesma posição (x, y) de uma vítima;
- A_e são capazes de ler os sinais vitais das vítimas.

As ações que os agentes são capazes de executar com suas durações estão na Tabela 1.

	Agente(s)	Tempo de execução (minutos)	Método no VictimSim
Mover hor. ou ver. (1 step). dx e dy podem ser {-1, 0, +1}	A_e A_s	1	$r:=walk(dx, dy)$ $r=\{BUMPED, EXECUTED\}$
Mover diagonal (1 step) dx e dy podem ser {-1, 0, +1}	A_e A_s	1,5	$walk(dx, dy)$
Ler os sinais vitais de uma vítima identificad por id	A_e	2	$read_vital_signals(id)$
Verificar se há vítima na posição atual do agente. Retorna -1 se não há e um inteiro maior do que zero quando há (id).	A_e A_s	0	$id:=check_for_victim()$ $id=\{-1, <int \geq 0\}$
Deixar kit de primeiros-socorros para a vítima identificada por id (agente deve estar na posição ocupada pela vítima id)	A_s	1	$first_aid(id)$

Tabela 1: ações dos agentes e suas durações em minutos.

5.2 Arquivos de configuração e de entrada

5.2.1 Ambiente

O ambiente muda de acordo com as configurações contidas nos arquivos abaixo. Os agentes que você criar **não devem ler estes arquivos**.

env_size.txt: tamanho do ambiente que é um grid de altura x comprimento

```

BASE 0,0          ## posição inicial dos agentes
GRID_WIDTH 30     ## largura do grid em células
GRID_HEIGHT 30    ## altura do grid em células
WINDOW_WIDTH 400  ## tamanho da janela em pixels

```

```

WINDOW_HEIGHT 400    ## altura da janela em pixels
DELAY 1.0             ## delay para atrasar a atualização da GUI em segundos

```

env_victims.txt: contém o posicionamento das vítimas no grid em coordenadas (lin, col)

```

x1,y1             ## coordenada (x, y) da vítima 1
...
xn,yn             ## coordenada (x, y) da vítima n

```

env_walls: contém o posicionamento das paredes no grid em coordenadas (lin, col)

```

x1,y1             ## coordenada (x, y) da parede 1
...
xm,ym             ## coordenada (x, y) da parede m

```

Observe que a origem do sistema de coordenadas do simulador do ambiente está no **canto superior esquerdo** como ilustra a figura abaixo.

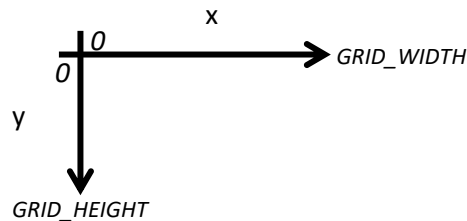


Figura 2: Sistema de coordenadas utilizado no simulador.

5.2.2 Agente

Cada agente deve responder a diferentes parametrizações de tempo de exploração e de salvamento. Estes parâmetros são lidos de arquivos de configuração cujo formato está abaixo. Cada agente possui um arquivo. **Todos os parâmetros devem estar com valores mesmo que não sejam utilizados para evitar erros.** Os custos das ações não serão mudados durante as etapas de teste durante todo o projeto. Já o tempo limite pode ter valores diferentes por cenário de treinamento e/ou teste.

```

rescuer<n>_config.txt
NAME RESCUER<n>                ## nome do agente para print de mensagens
COLOR (255, 0, 127)            ## cor para desenho do agente
TRACE_COLOR (255,153,204)      ## cor para pintar as células visitadas
TLIM 40.0                       ## tempo limite para o agente executar sua tarefa
COST_LINE 1.0                  ## custo para andar uma célula na horizontal ou vertical
COST_DIAG 1.5                  ## custo para andar uma célula na diagonal
COST_READ 2.0                  ## custo para ler os sinais vitais de uma vítima
COST_FIRST_AID 1.0             ## custo para deixar o pacote de primeiro-socorros

```

```

explorer<n>_config.txt
NAME EXPLORER<n>
COLOR (0, 0, 255)
TRACE_COLOR (153, 153, 255)
TLIM 100.0
COST_LINE 1.0
COST_DIAG 1.5
COST_READ 2.0
COST_FIRST_AID 1.0

```

5.2.3 Sinais Vitais

O arquivo **sinais_vitais.txt** contém os sinais vitais das vítimas. Cada linha representa uma vítima. Há duas versões:

- com label de gravidade: **sinais_vitais.txt** (tem y_i)

- sem label de gravidade: **sinais_vitais_sem_label.txt** (não tem y_i)

5.2.3.1 Formato

Para uma vítima i do histórico temos 5 sinais vitais (s_1 até s_5) que resultam a gravidade g_i da vítima. Todos os valores são números reais criados de modo randômico dentro dos intervalos apresentados.

$$i \ s_{i1} \ s_{i2} \ s_{i3} \ s_{i4} \ s_{i5} \ g_i \ y_i$$

i : **identificação** da vítima (número sequencial)

s_{i1} : pressão **sistólica (pSist)**: [5, 22] - não usar, é utilizada no cálculo de s_{i3}

s_{i2} : pressão **diastólica (pDiast)**: [0, 15] - não usar, é utilizada no cálculo de s_{i3}

s_{i3} : qualidade da pressão (**qPA**): [-10,10] onde 0 é a qualidade máxima -10 é a pior qualidade quando a pressão está excessivamente baixa, +10 é a pior qualidade quando a pressão está excessivamente alta

s_{i4} : **pulso**: [0,200] bpm

s_{i5} : **respiração**: [0,22] FpM (frequência de respiração)

g_i : **gravidade**: deve ser inferido pela técnica escolhida

y_i : **label do risco de morte que representa a classe de saída**:

1=CRÍTICO 2=INSTÁVEL 3=POTENCIALMENTE ESTÁVEL 4=ESTÁVEL

Exemplo

i	si1	si2	si3	si4	si5	g1	y1
	pSist	pDiast	qPA	pulso	resp	gravid	classe
1,	8.5806,	2.2791,	-8.4577,	56.8384,	9.2229,	33.5156,	2

5.2.4 Relação entre env_victims.txt e sinais_vitais.txt

Na tabela 2, observar que o arquivo de sinais vitais tem uma linha para cada uma das vítimas. Assim, a vítima 1 que está na coordenada (3, 1) tem gravidade $g_1 = 0.9$ (*classe* = 4 = *estável*). O ambiente liga as vítimas do ambiente com os sinais vitais pela posição sequencial de ocorrência nos arquivos.

env_victims.txt	sinais_vitais.txt
3,1	1 0.1 0.2 0.3 0.4 0.5 0.9 4
0,2	2 0.8 0.9 0.7 0.5 0.7 0.2 1
0,3	3 0.8 0.8 0.8 0.9 0.9 0.7 3

Tabela 2: relação entre os arquivos env_victims.txt e sinais_vitais.txt

6 APRESENTAÇÃO E ENTREGA

6.1 Forma de apresentação e entrega

O trabalho pode ser feito em equipes de até $<n>$ pessoas. A apresentação e a entrega final se darão em dois momentos distintos:

MOMENTO 1: Apresentação - durante a aula

As soluções devem estar implementadas e funcionando para que vocês possam:

- apresentar a solução e mostrar os resultados obtidos para as métricas que estão na seção 4 das duas partes da tarefa;
- rodar a solução com os arquivos de teste que serão passados pelo professor no momento 1;
- comparar com as soluções de outras equipes com base nos testes realizados em sala de aula e discutir melhorias;
- não é necessário fazer upload no Moodle.

MOMENTO 2: Entrega - fora do horário de aula (semana seguinte ao do momento 1)

- Corrigir a solução implementada
- Revisar o artigo da tarefa que engloba ambas as partes da tarefa: treinamento e testes
- Carregar a versão final do artigo e do código no Moodle.

6.2 Artefatos para entregar

- 1) Os códigos fonte na linguagem que desejar com as instruções para rodar.
- 2) Um artigo PDF de até 3 páginas, sem contar os apêndices, no [formato da SBC](#) com a estrutura abaixo e fonte tamanho 11.

Introdução: dentro do problema como um todo, quais subproblemas atacará e por quais razões: quais são as motivações e justificativas para resolvê-los.

Fundamentação Teórica: tipos de busca vistas e que são pertinentes às tarefas

Metodologia: caracterize o ambiente, o problema com seus estados e tamanho do espaço de estados, as estratégias de busca escolhidas com justificativa (por que esta e não as outras?) e a forma de modelagem. Qual estratégia utilizou em cada agente para utilizar o máximo do tempo dado e conseguir retornar à base?

Resultados e análise: mostrar os resultados numéricos comparando os resultados de treinamento e de teste das estratégias e, se possível, da solução da equipe com estratégias de *baseline* (e.g. aleatórias ou gulosas).

Conclusões: atingiu os objetivos, a solução está sobreadaptada ou generaliza bem para diferentes cenários, o que pode ser melhorado, o que poderia ser feito no futuro para completar a solução, há problemas éticos na solução – como ela afeta a vida das pessoas envolvidas? A solução é neutra? A solução é enviesada?

Referências bibliográficas

---- 3 páginas no máximo até aqui ----

Apêndices:

- 1) obrigatório: instruções claras de como executar o código (deve respeitar os formatos de arquivos de entrada e de configuração), print das telas se desejar (não colocar print das telas no corpo do artigo)
- 2) opcionais (outras informações, trechos relevantes do código, gráficos, dados, cenários de treinamento).