Intro al MidTerm

CORSO DI INTELLIGENZA ARTIFICIALE, A.A. 2021-22

R. BASILI

Outline

Programma del MidTerm

Esempi di domande a risp. multiple

- Introduzione all'AI
- Agenti
  - Scopi e metodologie
  - Ambiente
  - Conoscenza e Modelli
- Problem Solving: Ricerca Semplice
- Problem Solving: Ricerca Informata
- Tecniche avanzate di ricerca
- Modelli e Logica

Esempi domande aperte

Altri Esercizi libro

Progettazione di Agenti: Vacuum Agent World

Programma

Il primo test MidTerm insistera' sulle seguenti tematiche (vedi AIMA book)

- Introduzione all'AI (Chapt. 1)
- Agenti (Chapt 2)
  - Scopi e metodologie
  - Ambiente
  - Conoscenza e Modelli
- Problem Solving: Ricerca Semplice (Chapt. 3, 3.1-3.4)
- Problem Solving: Ricerca Informata (Capt. 3, 3.5-3.6)
- Tecniche avanzate di ricerca (Capt. 4, 4.1, 4.4; ~~Chapt. 5, 5.1-5.3~~)
- Agenti basati su conoscenza e calcolo delle proposizioni (Capt. 7, 7.1-7-6)

Domane a Risposte Chiuse

Esame di Intelligenza Artificiale
Prima Prova MidTerm (a.a. 2019-2020)

14 Novembre 2019

Docente: R. Basili

Rispondente alle seguenti domande marcando le risposte che ritenete corrette. Tempo a disposizione: 30 minuti. In sede di valutazione, ogni risposta sbagliata abbassa il punteggio.

1. Quale tra queste definizioni di IA e' *Human-centered* e *caratteristica dell'agire intelligente*:
(A) Lo studio delle facolta' algoritmiche attraverso l'uso di modelli cognitivi [−0]
(B) L'arte di creare algoritmi che svolgono funzioni su macchine che richiedono intelligenza quando svolte da esseri umani [−0]
(C) L'arte di creare macchine che svolgono funzioni che richiedono intelligenza quando svolte da esseri umani [−0]
(D) Lo studio delle facolta' mentali attraverso l'uso di modelli computazionali [−0]

Domande a Risposte Chiuse (2)

2. Un agente e' razionale:

1. perche' prende decisioni in analogia con le decisioni dell'uomo (esperto)
2. perche' conosce con completezza l'ambiente circostante in cui agisce
3. perche' agisce con i suoi attuatori in un ambiente
4. perche' agisce massimizzando un vantaggio in analogia con l'esperto umano
5. perche' persegue un obbiettivo usando le azioni ad esso disponibili in modo da minimizzare costi e rischi

(A) La 1, la 2 e la 4. [−0]
(B) La 1, la 4 e la 5. [−0]
(C) La 1, la 2 e la 3. [−0]
(D) La 2, la 3 e la 4. [−0]

Domande a Risposte Chiuse (3)

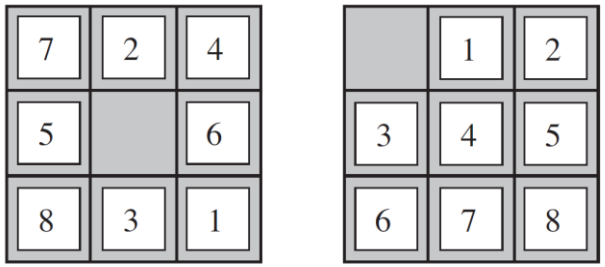
3. Un simulatore di ambienti e' uno strumento software che consente:

1. La generazione di stimoli per gli agenti
2. La valutazione delle prestazioni degli agenti
3. La acquisizione delle azioni di risposta dagli agenti
4. La attuazione di azioni che modificano gli agenti

(A) Falso. [−0]
(B) Vero solo se escludiamo la 4. [−0]
(C) Vero sempre. [−0]
(D) Falso se escludiamo la 4. [−0]
(E) Nessuna delle altre risposte. [−0]

Domande a Risposte chiuse (4)

4. Date le due configurazione del gioco dell'8 puzzle qui sotto rappresentate



si determini la affermazione corretta:

(A) La distanza euclidea tra le due configurazione conta il quadrato del numero di mosse che porta la prima nella seconda [−0]

(B) Una euristica possibile considera il numero di caselle che sono ordinate nello stato corrente [−0]

(C) Una euristica possibile (Manhattan distance) conta il numero di spostamenti (orizzonatli e verticali) da applicare alle diverse caselle dello Start State per condurle tutte nella posizione ad esse assegnata nel Goal State finale. [−0]

(D) La combinazione lineare di euristiche diverse non Ã¨ una euristica valida [−0]

Domande a Risposte Chiuse (5)

5. L'algoritmo di A* non e' ottimale:
(A) Sempre vero. [ ]
(B) Dipende dal problema [ ]
(C) Nessuna delle alternative [ ]
(D) In genere e' falso poiche' la funzione euristica e' un criterio approssimato. [ ]
(E) In genere e' vero per euristiche ammissibili. [ ]

Domande a Risposte Chiuse Soluzioni

tba

Domande Aperte

Discutere la relazione tra la nozione di agente e di ambiente in AI. Si esplicitino esempi di applicazioni di AI in cui distinguere le due nozioni. Si facciano esempi di applicazioni che caratterizzano i diversi tipi di ambiente e di agente.

Data un puzzle da n celle (ad es. 9), definire su tale gioco la nozione di agente e di ambiente; di percezione e di azione; si determini lo spazio degli stati (dell'ambiente). Si determinino le azioni dell'agente i caso di agente semplice dotati di riflessi o di agente a comportamento randomizzato. Si definisca un insieme di euristiche possibili. Si discuta infine facoltativamente un algoritmo di ricerca euristica per la soluzione del gioco.

Domande Aperte (2)

Si discuta la nozione di algoritmi di Ricerca non Informati per il Problem Solving ed in particolare si discuta le proprietà di completezza, e ottimalità per tali algoritmi. Si confrontino tra loro almeno due algoritmi.

Si discuta l'algoritmo di A* e se ne esemplifichi la applicazione ad almeno due problemi diversi.

Si confrontino tra loro gli algoritmi di Breadth-first, Depth-first e di Best First: si utilizzi un esempio di applicazione ad un problema (ad es. le 8 regine)

Problemi ed Agenti

Riempire e giustificare la seguente tabella

Agente	Prestazione	Ambiente	Attuatori	Sensori
TAXI Driver	Arrivare alla destinazione, sicuro, veloce, ligio alla legge, viaggio confortevole, minimo consumo di benzina, profitti massimi	Strada, altri veicoli, pedoni, clienti	Sterzo, acceleratore, freni, frecce, clacson, schermo di interfaccia o sintesi vocale	Telecamere, sensori a infrarossi e sonar, tachimetro, GPS, contachilometri, acelerometro, sensori sullo stato del motore, tastiera o microfono
Giocatore Sudoku				
Robot cameriere in una sala ristorante				
Motore di ricerca su Facebook (ricerca post e altri utenti)				
Speech recognizer (e.g. SIRI)				

Agenti: formalizzazione del PEAS model

Riempire e giustificare la seguente tabella

Agente	Prestazione	Ambiente	Attuatori	Sensori
8-puzzle	Ordinare il puzzle con il minimo tempo t (o numero di mosse, n)	Il puzzle da 81 celle	Spostamento celle	Osservazione dell'intero stato del puzzle
8-puzzle	Given Start state *S*, apply the sequence P=(A_1, A_2, …, A_n) of actions A_i to reach the Goal *G* such that: n = $\min_P$ such that (A_n °(A_{n-1} … A_2°(A_1(*S*))…)=*G*	A 9x9 matrix *M* with coefficients in m_{ij} ∈{1, …, 8, *blank*} where ∀(i,j)≠(k,l) $m_{ij} \neq m_{kl}$	Due opzioni: • Muovi un qualsiasi m_{ij} in *blank* • Blank-left, …, Blank-down	Full Matrix *M*
Vacuum World Cleaner with two rooms				
Ricerca di un percorso Start-Goal in un a mappa connessa di N città				

Caratterizzazione ambienti di agenti razionali

2.4 For each of the following activities, give a PEAS description of the task environment and characterize it in terms of the properties listed in Section 2.3.2.

A. • Playing soccer.
B. • Exploring the subsurface oceans of Titan.
C. • Shopping for used AI books on the Internet.
D. • Playing a tennis match.
E. • Practicing tennis against a wall.
F. • Performing a high jump.
G. • Knitting a sweater.
H. • Bidding on an item at an auction.

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete

2.6 This exercise explores the differences between agent functions and agent programs.

a. Can there be more than one agent program that implements a given agent function? Give an example, or show why one is not possible.

b. Are there agent functions that cannot be implemented by any agent program?

c. Given a fixed machine architecture, does each agent program implement exactly one agent function?

d. Given an architecture with n bits of storage, how many different possible agent programs are there?

e. Suppose we keep the agent program fixed but speed up the machine by a factor of two. Does that change the agent function?

Agenti e complessità

2.11 Consider a modified version of the vacuum environment in Exercise 2.8, in which the geography of the environment—its extent, boundaries, and obstacles—is unknown, as is the initial dirt configuration. (The agent can go *Up* and *Down* as well as *Left* and *Right*.)

a. Can a simple reflex agent be perfectly rational for this environment? Explain.

b. Can a simple reflex agent with a *randomized* agent function outperform a simple reflex agent? Design such an agent and measure its performance on several environments.

c. Can you design an environment in which your randomized agent will perform poorly? Show your results.

d. Can a reflex agent with state outperform a simple reflex agent? Design such an agent and measure its performance on several environments. Can you design a rational agent of this type?

Vacuum Cleaner

function GOAL-BASED-AGENT(*percept*) **returns** an action
  persistent: *state*, the agent's current conception of the world state
    model, a description of how the next state depends on current state and action
    goal, a description of the desired goal state
    plan, a sequence of actions to take, initially empty
    action, the most recent action, initially none

  state ← UPDATE-STATE(*state*, *action*, *percept*, *model*)
  if GOAL-ACHIEVED(*state*,*goal*) **then return** a null action
  if *plan* is empty **then**
plan ← PLAN(*state*,*goal*,*model*)
  action ← FIRST(*plan*)
  plan ← REST(*plan*)
  return *action*

Figure S2.1 A goal-based agent.

Vacuum Cleaner

2.11 Consider a modified version of the vacuum environment in Exercise 2.8, in which the geography of the environment—its extent, boundaries, and obstacles—is unknown, as is the initial dirt configuration. (The agent can go *Up* and *Down* as well as *Left* and *Right*.)

a. Can a simple reflex agent be perfectly rational for this environment? Explain.

b. Can a simple reflex agent with a *randomized* agent function outperform a simple reflex agent? Design such an agent and measure its performance on several environments.

c. Can you design an environment in which your randomized agent will perform poorly? Show your results.

d. Can a reflex agent with state outperform a simple reflex agent? Design such an agent and measure its performance on several environments. Can you design a rational agent of this type?

(a) and (b)

Vacuum Cleaner

(c) A complex environment

2.11 Consider a modified version of the vacuum environment in Exercise 2.8, in which the geography of the environment—its extent, boundaries, and obstacles—is unknown, as is the initial dirt configuration. (The agent can go *Up* and *Down* as well as *Left* and *Right*.)

a. Can a simple reflex agent be perfectly rational for this environment? Explain.

b. Can a simple reflex agent with a *randomized* agent function outperform a simple reflex agent? Design such an agent and measure its performance on several environments.

c. Can you design an environment in which your randomized agent will perform poorly? Show your results.

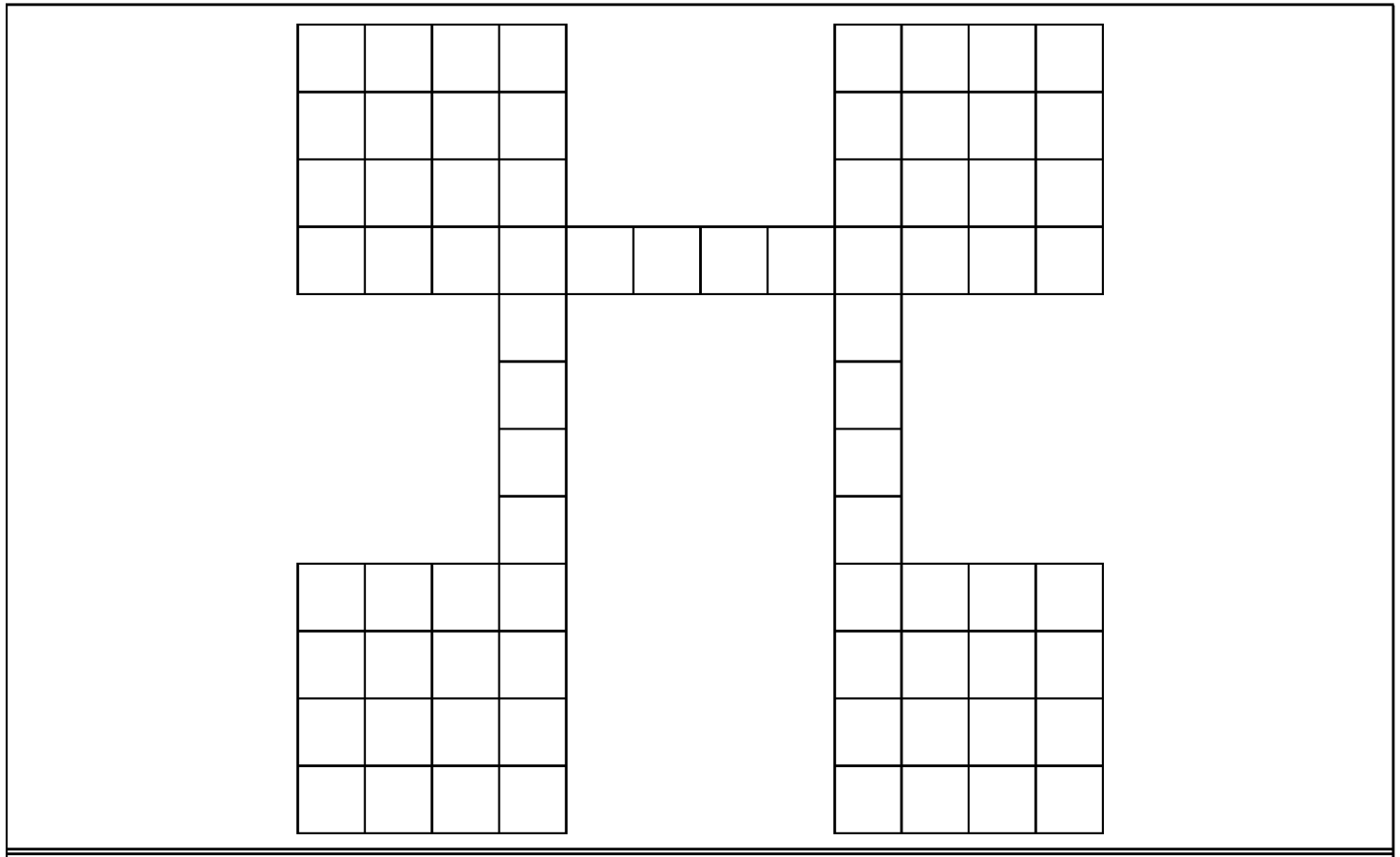


Figure S2.3 An environment in which random motion will take a long time to cover all the squares.

Vacuum Cleaner

(d)

2.11 Consider a modified version of the vacuum environment in Exercise 2.8, in which the geography of the environment—its extent, boundaries, and obstacles—is unknown, as is the initial dirt configuration. (The agent can go *Up* and *Down* as well as *Left* and *Right*.)

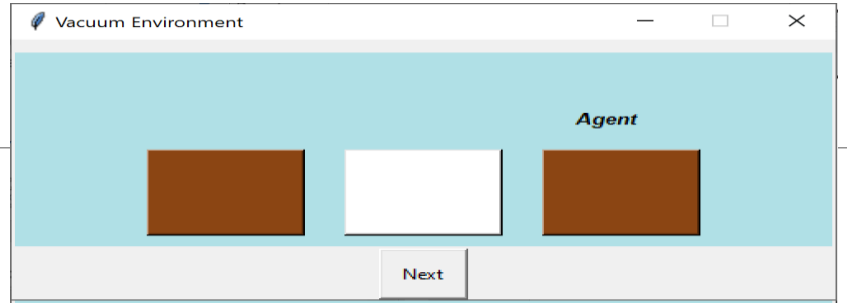
a. Can a simple reflex agent be perfectly rational for this environment? Explain.

b. Can a simple reflex agent with a *randomized* agent function outperform a simple reflex agent? Design such an agent and measure its performance on several environments.

c. Can you design an environment in which your randomized agent will perform poorly? Show your results.

d. Can a reflex agent with state outperform a simple reflex agent? Design such an agent and measure its performance on several environments. Can you design a rational agent of this type?

Vacuum Cleaner: esercizio




```
function GOAL-BASED-AGENT(percept) returns an action
  persistent: state, the agent's current conception of the world state
              model, a description of how the next state depends on current state and action
              goal, a description of the desired goal state
              plan, a sequence of actions to take, initially empty
              action, the most recent action, initially none

  state ← UPDATE-STATE(state, action, percept, model)
  if GOAL-ACHIEVED(state,goal) then return a null action
  if plan is empty then
plan ← PLAN(state,goal,model)
  action ← FIRST(plan)
  plan ← REST(plan)
  return action
```

Figure S2.1 A goal-based agent.

Adattare il programma tabellare

- Esistono 3 stanze (non due sole)
  - Monodirezionale ma lungo tre e non più due stanze
  - Inizializzazione randomica dello stato delle stanze
- L'agente ha una percezione «completa» dell'ambiente, cioè l'intero insieme di stanze
- L'agente è GoalBased e quindi può
  - Elaborare un piano rispetto alla situazione iniziale
  - Eseguire il piano ad ogni passo
- Si confrontino il comportamento di
  - Breadth-first Search
  - A* (basati su una euristica, ad es. numero di stanze sporche)

Exercise 3.25

3.25 The **heuristic path algorithm** (Pohl, 1977) is a best-first search in which the evaluation function is $f(n) = (2 - w)g(n) + wh(n)$. For what values of w is this complete? For what values is it optimal, assuming that h is admissible? What kind of search does this perform for $w = 0$, $w = 1$, and $w = 2$?

Exercise 4.1

4.1 Give the name of the algorithm that results from each of the following special cases:

a. Local beam search with $k = 1$.

b. Local beam search with one initial state and no limit on the number of states retained.

c. Simulated annealing with $T = 0$ at all times (and omitting the termination test).

d. Simulated annealing with $T = \infty$ at all times.

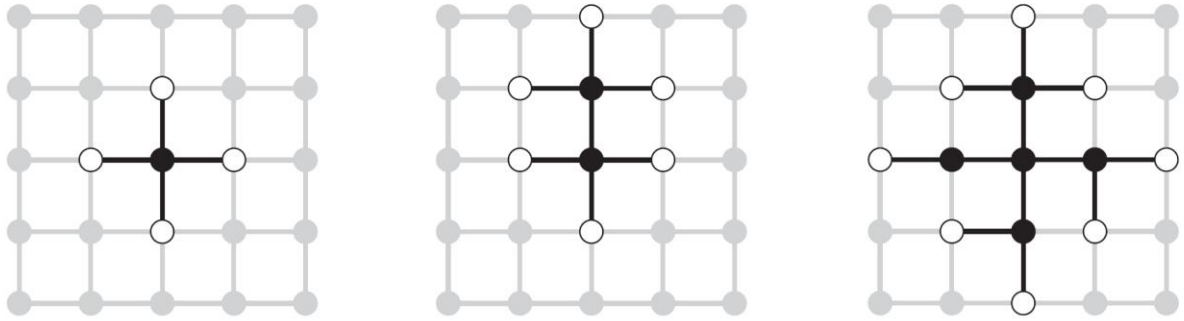


Figure 3.9 The separation property of GRAPH-SEARCH, illustrated on a rectangular-grid problem. The frontier (white nodes) always separates the explored region of the state space (black nodes) from the unexplored region (gray nodes). In (a), just the root has been expanded. In (b), one leaf node has been expanded. In (c), the remaining successors of the root have been expanded in clockwise order.

3.26 Consider the unbounded version of the regular 2D grid shown in Figure 3.9. The start state is at the origin, (0,0), and the goal state is at (x, y).

a. What is the branching factor b in this state space?

b. How many distinct states are there at depth k (for $k > 0$)?

c. What is the maximum number of nodes expanded by breadth-first tree search?

d. What is the maximum number of nodes expanded by breadth-first graph search?

e. Is $h = |u - x| + |v - y|$ an admissible heuristic for a state at (u, v)? Explain.

f. How many nodes are expanded by A* graph search using h?

g. Does h remain admissible if some links are removed?

h. Does h remain admissible if some links are added between nonadjacent states?

Domande

3.10 Define in your own words the following terms: state, state space, search tree, search node, goal, action, transition model, and branching factor.

3.9 The **missionaries and cannibals** problem is usually stated as follows. Three missionaries and three cannibals are on one side of a river, along with a boat that can hold one or two people. Find a way to get everyone to the other side without ever leaving a group of missionaries in one place outnumbered by the cannibals in that place. This problem is famous in AI because it was the subject of the first paper that approached problem formulation from an analytical viewpoint (Amarel, 1968).

a. Formulate the problem precisely, making only those distinctions necessary to ensure a valid solution. Draw a diagram of the complete state space.

b. Implement and solve the problem optimally using an appropriate search algorithm. Is it a good idea to check for repeated states?

c. Why do you think people have a hard time solving this puzzle, given that the state space is so simple?

3.14 Which of the following are true and which are false? Explain your answers.

a. Depth-first search always expands at least as many nodes as A* search with an admissible heuristic.

b. $h(n) = 0$ is an admissible heuristic for the 8-puzzle.

c. A* is of no use in robotics because percepts, states, and actions are continuous.

d. Breadth-first search is complete even if zero step costs are allowed.

e. Assume that a rook can move on a chessboard any number of squares in a straight line, vertically or horizontally, but cannot jump over other pieces. Manhattan distance is an admissible heuristic for the problem of moving the rook from square A to square B in the smallest number of moves.

3.15 Consider a state space where the start state is number 1 and each state k has two successors: numbers $2k$ and $2k + 1$.

a. Draw the portion of the state space for states 1 to 15.

b. Suppose the goal state is 11. List the order in which nodes will be visited for breadth-first search, depth-limited search with limit 3, and iterative deepening search.

c. How well would bidirectional search work on this problem? What is the branching factor in each direction of the bidirectional search?

d. Does the answer to (c) suggest a reformulation of the problem that would allow you to solve the problem of getting from state 1 to a given goal state with almost no search?

e. Call the action going from k to $2k$ Left, and the action going to $2k + 1$ Right. Can you find an algorithm that outputs the solution to this problem without any search at all?

Exercise 3.25

3.25 The **heuristic path algorithm** (Pohl, 1977) is a best-first search in which the evaluation function is $f(n) = (2 - w)g(n) + wh(n)$. For what values of w is this complete? For what values is it optimal, assuming that h is admissible? What kind of search does this perform for $w = 0$, $w = 1$, and $w = 2$?

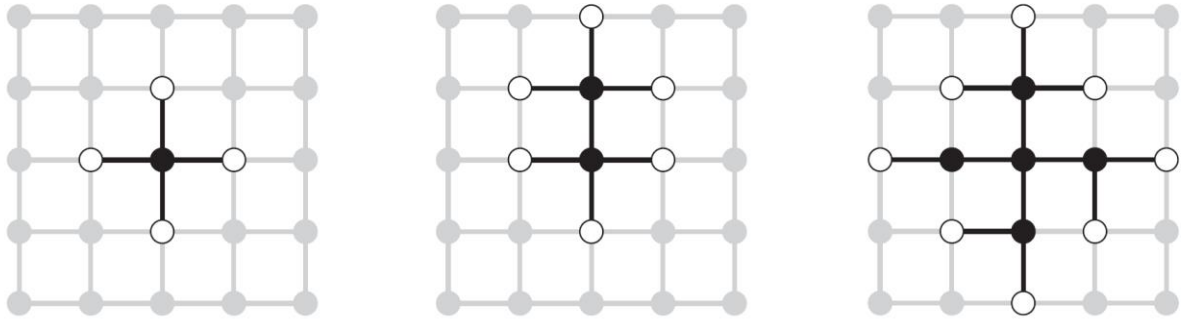


Figure 3.9 The separation property of GRAPH-SEARCH, illustrated on a rectangular-grid problem. The frontier (white nodes) always separates the explored region of the state space (black nodes) from the unexplored region (gray nodes). In (a), just the root has been expanded. In (b), one leaf node has been expanded. In (c), the remaining successors of the root have been expanded in clockwise order.

3.26 Consider the unbounded version of the regular 2D grid shown in Figure 3.9. The start state is at the origin, (0,0), and the goal state is at (x, y).

a. What is the branching factor b in this state space?

b. How many distinct states are there at depth k (for $k > 0$)?

c. What is the maximum number of nodes expanded by breadth-first tree search?

d. What is the maximum number of nodes expanded by breadth-first graph search?

e. Is $h = |u - x| + |v - y|$ an admissible heuristic for a state at (u, v)? Explain.

f. How many nodes are expanded by A* graph search using h?

g. Does h remain admissible if some links are removed?

h. Does h remain admissible if some links are added between nonadjacent states?

Exercise 4.1

4.1 Give the name of the algorithm that results from each of the following special cases:

a. Local beam search with $k = 1$.

b. Local beam search with one initial state and no limit on the number of states retained.

c. Simulated annealing with $T = 0$ at all times (and omitting the termination test).

d. Simulated annealing with $T = \infty$ at all times.

4.13 In this exercise, we examine hill climbing in the context of robot navigation, using the environment in Figure 3.31 as an example.

a. Repeat Exercise 4.11 using hill climbing. Does your agent ever get stuck in a local minimum? Is it *possible* for it to get stuck with convex obstacles?

b. Construct a nonconvex polygonal environment in which the agent gets stuck.

c. Modify the hill-climbing algorithm so that, instead of doing a depth-1 search to decide where to go next, it does a depth-k search. It should find the best k-step path and do one step along it, and then repeat the process.

d. Is there some k for which the new algorithm is guaranteed to escape from local minima?

e. Explain how LRTA* enables the agent to escape from local minima in this case.

Risposte