II esercitazione

Nell'esercitazione ci concentreremo sugli esercizi con asterisco, ma lasciamo gli altri come esercizi.

Sommario:

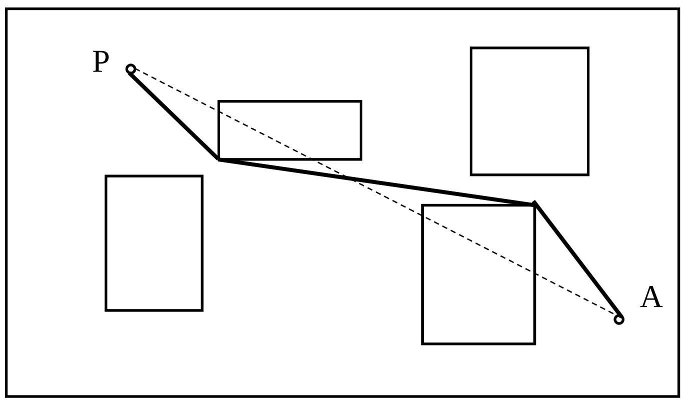
- **Formulazione di Problemi**
  - Navigazione di un robot *
  - Scala di parole *
  - Il cavallo
- **Confronto di euristiche ammissbili**
  - Confronto di euristiche ammissibili basato su proprietà matematiche *
  - Combinazione lineare di euristiche ammissibili *
  - Algoritmo del percorso euristico
  - Euristica di Gasdching

Formulazione di Problemi

Navigazione di un robot *

Si tratta di pianificare per il robot il percorso minimo tra due punti dati in una stanza popolata di ostacoli a forma di poligoni convessi del tipo di quella mostrata in figura. Il robot è dotato di un normale apparato percettivo e riesce a vedere gli ostacoli ma non gli oggetti che rispetto al suo punto di vista sono nascosti dagli ostacoli.

a. Si dia una formulazione del problema come problema di ricerca in uno spazio di stati prestando attenzione alle dimensioni dello spazio di ricerca.
b. Si definisca una euristica ammissibile da usare con un algoritmo A*.



Soluzione

Si potrebbe sovrapporre una griglia più o meno fitta sopra la stanza e fare muovere il robot per piccoli passi nel mondo a griglia.
Una *riduzione drastica dello spazio degli stati* si ha considerando solo i punti del piano che corrispondono ai vertici dei poligoni, più i due punti di partenza P e di arrivo A, e considerando spostamenti in linea retta tra questi punti.

Questa semplificazione non fa perdere l'ottimalità.

Un modo di vedere le cose è che la linea retta tratteggiata da P ad A, distanza in linea d'aria tra i due punti, è in assoluto il percorso minimo, ma tipicamente non percorribile per la presenza di ostacoli. La minima deviazione da questo percorso percorribile passa per qualche vertice di poligono. È come se la linea tratteggiata fosse un elastico che allunghiamo il minimo indispensabile per aggirare gli ostacoli (vedi figura, dove la linea solida rappresenta l'elastico allungato).

Lo spazio degli stati è dato dai vertici dei poligoni più i due punti di partenza e di arrivo. Gli operatori sono gli spostamenti in linea retta da un punto ai successivi visibili (se sono visibili possono essere raggiunti con spostamenti in linea retta con una azione).

Il problema quindi è del tutto equivalente a quello del *route finding* e può pertanto essere risolto con un algoritmo di tipo A* con funzione di valutazione $f(s) = g(s) + h(s)$ dove la $g(s)$ è la somma delle lunghezze dei tratti percorsi per arrivare nel punto del piano che corrisponde allo stato s e $h(s)$ la distanza in linea d'aria da s al punto di arrivo A.

Questa è chiaramente una sottostima per la presenza di ostacoli, che il robot deve aggirare, visto che non vola e non può penetrare gli ostacoli. Essendo una euristica *ammissibile* (e anche *consistente*, perché?) la soluzione trovata dall'algoritmo è una soluzione ottimale.

Nota: Commento a proporre una ricerca locale su spazio continuo:

Una rappresentazione su spazio continuo non soddisfa la richiesta di "porre attenzione alle dimensioni dello spazio di ricerca" formulata nel testo, che in questo caso sarebbe infinita.

Ammesso di definire una funzione costo adeguata, ad esempio per utilizzare un discesa di gradiente, va anche detto che:

- Gli algoritmi di ricerca locale non sono concepiti per questi problemi di percorso minimo come in questo problema, ma quando la sequenza di azioni non è importante;
- Anche se possiamo provare a utilizzarli (lo abbiamo fatto anche con Teseo e lo faremo nella terza parte del corso in modo sistematico), l'algoritmo può fallire nella ricerca del percorso ottimo, che è ciò che è chiesto qui: l'ottimo non è garantito a causa della possibile incoerenza in minimi locali (si provi a visualizzarlo su un blocco in cui impattiamo, ad esempio il più vicino blocco al punto A in basso a destra, in cui sia andando in su che in giù avremmo sempre un aumento della distanza da A e quindi un aumento della funzione costo, bloccandosi in un minimo locale).

La rappresentazione con spigoli mantiene invece lo spazio degli stati molto ridotto (si contino quanti spigoli abbiamo nell'esempio in figura) e come discusso si può garantire l'ottimo usando gli algoritmi di ricerca già noti.

Scala di parole *

Una scala di parole è una sequenza di parole significative (appartenenti al vocabolario italiano) tale che ogni parola differisce dalla precedente esattamente per una lettera. Sia dato il problema di trovare una scala di parole, più breve possibile, che porta da una parola data ad un'altra, anch'essa data. Ad esempio, una scala da **auto** a **vita** potrebbe essere:

auto > muto > mito > dito > dita > vita

a. Si formuli il problema come un problema di ricerca in uno spazio di stati.
b. Quale è la dimensione massima dello spazio di ricerca per parole di lunghezza *k?* Dare una stima, anche approssimata, del fattore di diramazione relativamente all'esempio.
c. Quale algoritmo di ricerca non informata usereste per ricercare una soluzione nello spazio degli stati? Discutere completezza, ottimalità, complessità delle possibili soluzioni.
   Inoltre, la ricerca bidirezionale sarebbe appropriata per questo problema?
d. Si proponga un'euristica da usare con un algoritmo di ricerca informata. Si dica se l'euristica è ammissibile e se ne discuta la completezza e l'ottimalità (con ricerca "ad albero") .
e. Considerare il caso della ricerca locale.

Soluzione

a)
Stati: parole
Stato iniziale: parola da cui si parte (es. auto)
Stato finale: parola che si vuole ottenere (es. vita)
Operatori: i successori di una parola p di lunghezza *k* sono tutte le parole che appartengono a Diz(*k*)—le parole del dizionario di lunghezza *k*--- e che hanno una sola lettera diversa da p.
Costo azione: 1

b)
Numero delle parole in Diz(*k*) è un limite superiore. Una stima al fattore di diramazione è *k* x *n*, dove *n* è il numero di parole significative ottenute sostituendo a una delle *k* lettere un'alternativa, tipicamente molto minore delle 25 possibili (come sarebbe considerando un alfabeto di 26 lettere).

c)
Ricerca in ampiezza: Il fattore di diramazione *b* è comunque piuttosto alto e quindi l'occupazione di memoria per tenere in memoria tutta la frontiera fino ad arrivare alla soluzione che si trova ad una profondità variabile *d*, tipicamente maggiore di *k*. Occupazione memoria $O(b^d)$. Tempo $O(b^d)$. Completo e ottimale (nota che il costo azione è costante).
Ricerca in profondità: Esiste un limite alla profondità se si spezzano i cicli. Il limite è dato dal numero di parole di lunghezza *k*. L'ottimalità non è garantita. Si potrebbe allora usare l'approfondimento iterativo, che garantisce completezza e ottimalità. Occupazione memoria $O(bd)$.
Ricerca bidirezionale: È possibile utilizzare la ricerca bidirezionale e in tal modo ridurre tempo e occupazione di memoria rispetto alla ricerca in ampiezza in una sola direzione.

d)
Un’euristica che stima la distanza dalla soluzione potrebbe essere “contare il numero di lettere diverse rispetto alle corrispondenti nella parola obiettivo”. È un’euristica ammissibile: infatti se non avessi il vincolo della significatività della parola (se tutte le combinazioni di lettere avessero significato) sarebbe un oracolo. Se usata con A “ad albero”, garantisce completezza e una soluzione ottimale.

e)
Si noti che se l’euristica fosse usata con Hill-climbing greedy la completezza non sarebbe garantita perché in certe situazioni può succedere che ci si debba “allontanare” dalla parola obiettivo per trovare una catena.

Esempio: nel problema di trovare una catena da PIGRO a PEGNO.
PIGRO dista 2 da PEGNO; ma non esistono parole di senso compiuto ottenibili cambiando la I in E o la R in N (almeno io non le ho trovate; se ci fossero basta restringere il vocabolario). Questo non significa che la catena non esiste, ma che ci si deve temporaneamente allontanare. Una catena potrebbe essere
PIGRO [2] > MIGRO [3] > MAGRO [3] > MAGNO [3] > RAGNO [2] > REGNO [1] > PEGNO [0]
In questo caso Hill-climbing greedy invece si arrenderebbe.

Il cavallo

Supponiamo una scacchiera infinita ed un cavallo nella posizione iniziale (1,1). Sia data inoltre una posizione (*m*, *n*) di arrivo per il cavallo. Il problema consiste nello spostare il cavallo dalla posizione iniziale alla posizione di arrivo con il numero minimo di mosse legali (la figura 1 mostra le mosse legali per il cavallo in un caso particolare).

1. Si formuli il problema come un problema di ricerca in uno spazio di stati.
2. Si trovi una euristica ammissibile, il più possibile informata, per il problema.
3. Si trovi una soluzione tracciando l'andamento dell'algoritmo A* sull'istanza del problema in figura 2 (obiettivo in (5, 4)).

Fig 1

	1	2	3	4	5	6
1		X		X		
2	X				X	
3			C			
4	X				X	
5		X		X		
6						

Fig 2

	1	2	3	4	5	6
1	C					
2						
3						
4						
5				G		
6						

Soluzione

1. Stati: coordinate (x, y) del cavallo.
   Stato iniziale: (1, 1)
   Stato goal: (5, 4)
   Funzione successore: (x, y) → {(x+1, y-2), (x+2, y-1), (x+2, y+1), (x+1, y+2), (x-1, y+2), (x-2, y+1), (x-2, y-1), (x-1, y-2)}
   Costo: 1 per ogni mossa

2. Una buona euristica ammissibile per il problema è h = $\lceil MD/3 \rceil$ arrotondato per eccesso. Ad esempio, la stima dallo stato iniziale allo stato goal sarebbe $\lceil 7/3 \rceil$ = 3. In questo caso il goal è direttamente raggiungibile in tre mosse (meglio non si può fare) ma tipicamente ne serviranno di più, quindi l'euristica è una sottostima.

3. Lasciato a propria soluzione

Confronto di euristiche ammissibili

Confronto di euristiche ammissibili basato su proprietà matematiche*

Siano date due euristiche ammissibili: h_1 e h_2. Quali delle seguenti euristiche sono ammissibili? Motivare le risposte.

a. $h(s) = h_1(s) + h_2(s)$
b. $h(s) = |h_1(s) – h_2(s)|$, dove | . | indica il valore assoluto
c. $h(s) = max(h_1(s), h_2(s)) – 2$
d. $h(s) = 2h_1(s) + h_2(s)/2$
e. $h(s) = (h_1(s) + h_2(s))/2$

Soluzione

a. Anche se $h_1(s) \leq h^*(s)$ e $h_2(s) \leq h^*(s)$ non è detto che la loro somma (o il loro prodotto) lo sia e quindi in generale la somma (o il prodotto) di due euristiche ammissibili non è ammissibile.
b. $|h_1(s) – h_2(s)| \leq h_1(s) \leq h^*(s)$ e quindi ammissibile nel caso $h_1(s) \geq h_2(s)$
$|h_1(s) – h_2(s)| \leq h_2(s) \leq h^*(s)$ e quindi ammissibile nel caso $h_1(s) \leq h_2(s)$
c. Non è una buona stima di distanza perché sul goal non vale 0 ma -2.
d. Non è detto che sia ammissibile. Ad esempio se fosse $h_1(s) = h_2(s)$,

$$h(s) = 2h_2(s) + h_2(s)/2 = 2.5\ h_2(s)$$

e quindi potrebbe sovrastimare. Potrebbe succedere in un caso specifico che trova la soluzione ottimale ma non abbiamo alcun conforto dal risultato teorico.
e. La media delle due euristiche è sicuramente inferiore della più grande e quindi ancora ammissibile, ma comunque meno informata della maggiore.

Combinazione lineare di euristiche ammissibili *

Questo caso merita un discorso a parte.
Sia h_1 l'euristica della somma delle distanze Manhattan di ogni numero dalla sua destinazione per il gioco dell'otto. Si consideri l'euristica $h_2=2*h_1+3$

a. h_2 è una euristica ammissibile?
b. Possiamo garantire che troverà una soluzione ottimale se usata con un algoritmo A (con costo azioni > e > 0)?
c. Se si è risposto SI alle prime due domande, la ricerca sarà più efficiente che con h_1?

Motivare le risposte.

Soluzione

1	2	3
4	5	6
7		8

a. L'euristica h_2 non è ammissibile.
Di fatto non è nemmeno una stima di distanza essendo diversa da zero sul goal. Non è difficile nemmeno far vedere un caso in cui sovrastima la distanza dalla soluzione. Per esempio, nel gioco dell'8, se h_1 è il numero di caselle fuori posto e $h_2=2*h_1+3$, nella seguente configurazione la soluzione dista 1 mossa ma h_2 vale 5.

Ma potrebbe venirci il dubbio che le due euristiche ordinino i nodi sulla frontiera nello stesso modo e producano esattamente le stesse scelte. Siano n e n' due nodi sulla frontiera e confrontiamo i due ordinamenti secondo

$$f_1(n) = g(n) + h_1(n) \qquad e \qquad f_2(n) = g(n) + h_2(n) = g(n) + 2*h_1(n) + 3$$

Supponiamo

$$g(n) + h_1(n) \qquad > \qquad g(n') + h_1(n')$$

è anche sempre vero che

$$g(n) + 2h_1(n) + 3 \qquad > \qquad g(n') + 2h_1(n') + 3 \text{ ?}$$

Chiaramente non è detto: se moltiplichiamo per 2 uno solo degli addendi il risultato può cambiare e quindi la nuova euristica potrebbe indurre scelte diverse.

Ma esistono euristiche non ammissibili che ci danno la garanzia di ottimalità?
Possibile: il risultato teorico ci dice, su ricerca "ad albero", Ammissibile => Ottimale (condiz. suff)
Ma un'euristica potrebbe essere comunque ottimale e anche essere dimostrata ottimale per altra via.

Algoritmo del percorso euristico

L'algoritmo del **percorso euristico** è un algoritmo *best first* in cui la funzione di valutazione è
$f(n) = (2 - w)\, g(n) + w\, h(n)$, con h ammissibile

a. Quale ricerca implementa l'algoritmo quando $w = 0$, $w = 1$ e $w = 2$?
b. Per quali valori di w l'algoritmo è completo, assumendo che h sia una stima di distanza?
c. Per quali valori di w l'algoritmo è ottimale con ricerca "ad albero"?

Soluzione

a. Con $w = 0$, $f(n) = 2g(n)$. Un algoritmo *best-first* con questa euristica si comporta esattamente come la ricerca di costo uniforme — il fattore due non fa differenza nell'ordinamento dei nodi.
Con $w = 1$ abbiamo la ricerca A*.
Con $w = 2$ abbiamo $f(n) = 2h(n)$, cioè la ricerca *greedy best-first*.
b. Possiamo riscrivere la definizione come: $f(n) = (2 - w)\, [g(n) + w/(2-w)\, h(n)]$ mettendo in evidenza $(2 - w)$. Questo ci dice che l'algoritmo si comporta come A* con una componente euristica $w/(2-w)\, h(n)$. Pertanto l'algoritmo è completo quando $0 \le w < 2$. Per valori di w inferiori a 0 o superiori a 2 l'euristica sarebbe negativa, e quindi non sarebbe più una buona stima di distanza (perdendo la completezza). Per $w = 2$, sarebbe indefinita.
c. Se $0 \le w \le 1$, l'euristica è inferiore ad $h(n)$ e quindi ammissibile, visto che $h(n)$ è ammissibile per ipotesi.

Euristica di Gasdching

Nel gioco dell'otto tradizionale una casella numerata A può essere spostata in B se:

1. B è adiacente ad A
2. A è vuota

Si pensi ad un gioco dell'otto in cui A può essere spostata in B, *se B è vuota*, anche se non adiacente. Sia Hg una stima esatta della distanza della soluzione minima per questo gioco "meno vincolato".

a. Hg (detta l'euristica di Gasdching) è ammissibile per il gioco dell'otto tradizionale?
b. Hg è un'euristica migliore (più informata) rispetto ad H1 (l'euristica che conta il numero di caselle fuori posto) per il gioco dell'otto tradizionale?
c. Esistono casi in cui Hg è più accurata di H2 (l'euristica che somma le distanze Manhattan delle caselle dalla loro collocazione finale)?

Soluzione

a. Hg è ammissibile per il gioco dell'otto tradizionale in quanto tutte le mosse possibili nel gioco dell'otto tradizionale sono anche possibili nel gioco dell'otto meno vincolato. Quindi tipicamente ci vogliono più mosse e nel caso migliore ce ne vuole lo stesso numero.

b. Per ogni n H1(n) $\leq$ Hg(n). Infatti le caselle fuori posto possono essere messe a posto con una o due mosse: una se la casella di destinazione è vuota, due se va svuotata prima (non sempre la mossa di svuotamento può servire a mettere a posto un'altra casella)

c. Si, esistono casi in cui Hg è più accurata di H2. Si consideri il seguente esempio:

2	1	3
4	5	6
7	8	

1	2	3
4	5	6
7	8	

H2=2 le due caselle fuori posto sono la 1 e la 2
Hg=3 per mettere le cose a posto nel gioco meno vincolato di Hg servono 3 mosse

NOTA: Relazione tra euristica consistente e ammissibile

Perché ci siamo occupati di ammissibilità e non di consistenza.
Un altro modo per trovare un'euristica ammissibile è trovare un'euristica consistente e facendo appello al risultato teorico *consistente => ammissibile*.
Si noti comunque che:

1. Non è detto che sia più facile trovare un'euristica consistente
2. "Quasi tutte" le euristiche ammissibili che possono venire in mente sono anche consistenti (ma si veda il controesempio a fine esercitazione 1)