

Relazione per l'esame di Intelligenza Artificiale

Gabriele Stefani 6342913

10 Gennaio 2020

1 Introduzione

In questo elaborato si è implementato l'algoritmo Forward Checking per risolvere Constraint Satisfaction Problem (CSP) binari applicato come inferenza nell'algoritmo Backtracking Search. I 3 problemi presi in considerazione sono:

- Map Coloring
- Sudoku
- NQueens

Per ognuno di essi è stato codificato il modo in cui si ottengono le variabili, i domini di ciascuna variabile, le variabili unite da un vincolo (obbligatoriamente binario) e i vincoli.

2 CSP generico

La base del codice definisce una classe CSP che descrive un problema di CSP generico in base a 4 strutture dati in ingresso. La prima è la lista delle variabili del problema, la seconda e la terza sono entrambi dizionari con chiave ogni variabile del problema, e valori rispettivamente il dominio della variabile (come lista) e la lista delle variabili che hanno un vincolo con essa (che da ora in poi chiameremo “vicini”), la quarta è una funzione che restituisce vero se il vincolo tra due generiche variabili alle quali è stato assegnato un valore è rispettato e falso altrimenti. Internamente esiste un'altra struttura che contiene i domini di ogni variabile ridotti dall'inferenza del Forward Checking.

3 Algoritmi

L'algoritmo di Backtracking Search riceve in ingresso un'istanza della suddetta classe CSP e restituisce un dizionario con chiave ogni variabile e valore il valore associato come soluzione del problema. Internamente sceglie come prossima variabile a cui assegnare un valore la prima della lista delle variabili che non sia stata ancora assegnata e l'ordine con cui vengono assegnati i valori del dominio è proprio l'ordine con cui sono inseriti i valori nel dominio. L'algoritmo usa Forward Checking come inferenza ad ogni assegnazione di un valore ad una variabile riducendo così i domini delle variabili vicine in modo da rispettare i vincoli tra esse.

4 Problemi

4.1 Map Coloring

Il map coloring è stato codificato come problema di CSP binario utilizzando come variabili la lista degli stati da colorare, come dominio di ogni variabile la lista dei colori utilizzabili, come vicini gli stati confinanti tra loro e come vincolo la diversità tra i 2 valori assegnati a 2 variabili vicine.

4.2 Sudoku

Nel sudoku le variabili sono le celle, i domini di ogni variabile sono i valori da 1 a 9 tranne nei casi in cui a una cella sia già stato assegnato un valore, in tal caso il dominio è il valore stesso. Il vincolo è ancora una volta il vincolo di diversità per cui i vicini di ogni variabile (o cella) sono tutte le celle nello stesso quadrante, nella stessa riga e nella stessa colonna.

4.3 NQueens

Il problema delle n regine è stato codificato pensando che ogni regina deve essere disposta su una colonna diversa, altrimenti le regine si mangerebbero sicuramente tra loro, quindi ogni variabile corrisponde ad una colonna e il dominio sono le righe. Se usiamo delle coordinate (x, y) per indicare una posizione sulla scacchiera le variabili corrispondono alle y e i valori del dominio alle x . Il vincolo sfrutta proprio queste coordinate per capire se 23 regine si trovino sulla stessa diagonale o sulla stessa riga per cui ogni variabile ha come vicini tutte le altre variabili in modo che nessuna regina sia sulla stessa riga o sulla stessa diagonale.

5 Prestazioni

La risoluzione di questi problemi attraverso l'algoritmo Forward Checking non è molto efficiente, quindi la risoluzione di problemi difficili (NQueens con n grandi o sudoku con pochi valori iniziali) può portare a tempi di attesa notevoli.