Università degli Studi “G. d’Annunzio” Chieti-Pescara

Facoltà di Economia

Corso di Laurea Specialistica in economia informatica

Tesi di Laurea

Solving SoftCSPs by using Nash’s equilibria

Laureanda:
Prencipe Daniela Maria

Relatore:
Prof. Stefano Bistarelli

Anno Accademico 2006-2007

Ai miei genitori e a Mario

Prefazione

Nella vita di tutti i giorni capita spesso di dover affrontare problemi che ci pongono davanti a delle scelte. Molto spesso le scelte non vengono effettuate in modo indipendente, infatti vari fattori limitano le scelte e pongono dei vincoli affinchè la nostra scelta ricada su determinate situazioni anzichè su altre.
La semplice attività di fare la spesa pone un individuo normalmente davanti ad un problema di massimizzazione della qualità e contemporanea minimizzazione del prezzo, che può essere più o meno accentuato a seconda di vari fattori che possono caratterizzare la sua vita: la disponibilità economica, la propensione al risparmio, le inclinazioni individuali nella scelta dei prodotto, l'influenza pubblicitaria,...
Invece per esempio per quanto riguarda la più complessa attività lavorativa di un manager, questi deve conciliare l'obiettivo profitto con i differenti vincoli esistenti: economici, temporali, spaziali,...

I SCSP (Soft Constraint Satisfaction Problem) rappresentano problemi di soddisfazione soggetti a vincoli, una più ampia categoria che racchiude in sè anche problemi come quelli appena citati.
L'obiettivo proposto è quello di trovare un modo alternativo per semplificare e velocizzare la ricerca delle soluzioni dei SCSP (oltre i già esistenti algoritmi) attraverso la teoria dei giochi.
Due rami di studio alquanto differenti ma straordinariamente vicini. Infatti attraverso la formulazione di un mapping per la trasformazione dei SCSP in giochi non cooperativi statici con informazione completa, si è evidenziata una relazione strettamente significativa tra le soluzioni dei SCSP e le soluzioni dei corrispondenti giochi, ossia gli equilibri di Nash.
Questo collegamento permette di sfruttare le ricerche e i risultati ottenuti nel campo della teoria dei giochi e di rivesarli nel campo di studio dei SCSP. Infatti l'utilizzo di software per la ricerca degli equilibri di Nash permette la ricerca automatica e informatizzata delle soluzioni dei problemi soggetti a vincoli.

Sommario

Indice

Prefazione i

Sommario iii

1 CSP 1

1.1 Introduzione . . . 1
1.2 Applicazioni . . . 1
1.2.1 Problemi di assegnamento . . . 2
1.2.2 Gestione del personale . . . 2
1.2.3 Problema di trasporto . . . 2
1.3 Vincoli CRISP . . . 2
1.3.1 I CSP . . . 3
1.3.2 Constraint Logic Programming (CLP) e tecniche di soluzione . . . 7
1.4 Vincoli NON-CRISP . . . 12

2 Soft Constraint Satisfaction Problems 15

2.1 C-semiring e le loro proprietà . . . 16
2.1.1 Sistemi di vincoli e problemi . . . 19
2.2 Functional constraint . . . 23
2.3 Istanze della struttura . . . 24
2.3.1 CSP classici . . . 24
2.3.2 Fuzzy CSP (FCSP) . . . 27
2.3.3 Probabilistic CSP (Prob-CSP) . . . 30
2.3.4 Weighted CSP . . . 33
2.3.5 Egualitarismo e utilitarismo . . . 35
2.4 Semiring con operatore divisione . . . 35
2.5 C-semiring n-dimensionali . . . 37

3 Giochi non cooperativi 39

3.1 Teoria dei giochi . . . 39

3.1.1 Cenni storici 39
3.1.2 Classificazione dei giochi 41
3.2 Soluzione di un gioco non cooperativo statico ad informazione completa 51
3.2.1 Equilibrio di dominanza 52
3.2.2 Equilibrio di Nash in strategie pure 57
3.2.3 Relazione tra equilibrio di Nash ed eliminazione iterata delle strategie strettamente dominate 59
3.2.4 Equilibrio di Nash in strategie miste 60

4 SCSP e giochi non cooperativi 65
4.1 Mapping di un SCSP in un gioco non cooperativo statico ad informazione completa 65
4.1.1 Soluzione SCSP ed equilibrio di Nash 69
4.1.2 Mapping di un problema di colorazione di un grafo . . 86

5 Conclusioni 97
5.1 Principali risultati 97
5.2 Future works 97

Elenco delle figure

1.1 il problema delle 8 regine. 4
1.2 soluzione di un CSP. 7
1.3 tuple ridondanti. 11

2.1 un CSP e il suo corrispondente SCSP. 26
2.2 combinazione e proiezione nei classici CSP. 27
2.3 esempio di un fuzzy CSP. 29
2.4 esempio di un Prob-CSP. 32
2.5 esempio di un WCSP. 34

3.1 esempio forma estesa. 45
3.2 struttura forma estesa. 46
3.3 forma estesa. 48

4.1 esempio di CSP. 67
4.2 esempio di fuzzy CSP. 74
4.3 esempio di weighted CSP. 78
4.4 esempio di CSP senza soluzioni. 84
4.5 cartina. 87
4.6 Relazioni di adiacenza tra le nazioni. 88
4.7 problema di colorazione della cartina della figura 4.5 sottoforma di CSP. 89
4.8 parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore verde. 90
4.9 parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore giallo. 91
4.10 parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore azzurro. 92
4.11 parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore viola. 93

4.12 equilibri di Nash corrispondenti alle soluzioni del CSP della figura 4.7 95

Elenco delle tabelle

2.1 soluzione del FCSP della figura 2.3. 28
2.2 soluzione del Prob-CSP della figura 2.4. 32
2.3 soluzione del WCSP della figura 2.5. 33

3.1 dilemma del prigioniero 49
3.2 gioco a somma costante 50
3.3 esempio di strategia strettamente dominata 53
3.4 esempio di strategia strettamente dominante 53
3.5 esempio di strategia dominata 54
3.6 esempio di strategia dominante 54
3.7 processo di eliminazione iterata delle strategie strettamente dominate (1) 55
3.8 processo di eliminazione iterata delle strategie strettamente dominate (2) 55
3.9 processo di eliminazione iterata delle strategie strettamente dominate (3) 55
3.10 processo di eliminazione iterata delle strategie strettamente dominate (4) 56
3.11 battaglia dei sessi 56
3.12 equilibrio dilemma del prigioniero 58
3.13 equiliibri battaglia dei sessi 58
3.14 equilibri puro coordinamento 58
3.15 esempio senza equilibri di Nash 59
3.16 matching pennies. 60
3.17 nessun equilibrio in matching pennies. 60

4.1 payoff di X calcolati partendo dal CSP della figura 4.1. 67
4.2 payoff di Y calcolati partendo dal CSP della figura 4.1. 69
4.3 gioco non cooperativo statico ad informazione completa corrispondente al CSP della figura 4.1. 69
4.4 equilibrio di Nash del gioco della figura 4.3. 72

4.5 soluzione del CSP rappresentato nella figura 4.1 73
4.6 soluzione del fuzzy CSP della figura 4.2. 74
4.7 le utilità del giocatore X calcolate partendo dal fuzzy CSP della figura 4.2. 75
4.8 le utilità per il giocatore Y calcolate partendo dal fuzzy CSP della figura 4.2. 76
4.9 gioco non cooperativo statico ad informazione completa corrispondente al fuzzy CSP della figura 4.2. 77
4.10 equilibrio di Nash del gioco non cooperativo della tabella 4.9. . 77
4.11 soluzione del weighted CSP della figura 4.3 78
4.12 le utilità del giocatore X calcolate partendo dal weighted CSP della figura 4.3. 81
4.13 le utilità del giocatore Y calcolate partendo dal weigthed CSP della figura 4.3. 83
4.14 equilibrio di Nash del gioco non cooperativo statico ad informazione completa corrispondente al weighted CSP della figura 4.3. 84
4.15 soluzioni del CSP della figura 4.4. 85
4.16 le utilità per il giocatore X nel gioco corrispondente al CSP della figura 4.4 85
4.17 le utilità per il giocatore Y nel gioco corrispondente al CSP della figura 4.4. 85
4.18 equilibrio di Nash del gioco non cooperativo statico ad informazione completa corrispondente al weighted CSP della figura 4.4. 86

Capitolo 1

CSP

1.1 Introduzione

La programmazione con i vincoli [4] è un emergente tecnologia software per le descrizioni dichiarative e per la risoluzione effettiva di molti problemi nelle aree della pianificazione e organizzazione. È emersa come un'area di ricerca che combina ricerche provenienti da diversi campi tra cui l'intelligenza artificiale e i linguaggi di programmazione.
Le origini della programmazione con i vincoli è iniziata vengono fatte risalite agli anni ottanta.
Nella programmazione con i vincoli, il processo di programmazione consiste nella generazione di richieste (vincoli) e soddisfazione di queste richieste.
L'interesse nella soddisfazione di problemi sottoposti a vincoli può essere facilmente giustificata dalle richieste provenienti dalla realtà. Infatti, molte situazioni della vita reale possono essere fedelmente descritte da un insieme di oggetti, rapportati tra loro da una serie di vincoli.
I vincoli normalmente sorgono in diverse aree di interesse dell'uomo: ad esempio, il fatto che la somma degli angoli di un triangolo è di 180 gradi si può esprimere attraverso vincoli sugli angoli del triangolo.
Pertanto i vincoli sono un mezzo naturale per l'uomo per rappresentare problemi di vita quotidiana.

1.2 Applicazioni

Molte importanti aziende sviluppano strumenti basati sulla programmazione con i vincoli per risolvere molti tipi di problemi tra cui problemi di assegnamento, di gestione del personale e di trasporto.

1.2.1 Problemi di assegnamento

I problemi di assegnamento furono il primo tipo di applicazione industriale che fu risolta attraverso la tecnologia CLP (constraint logic programming). Un esempio può essere quello dell'allocazione dei posti negli aeroporti, dove gli aerei (il primo tipo di risorsa) devono essere parcheggiati nei posti disponibili (il secondo tipo di risorsa) durante la permanenza in aeroporto.

1.2.2 Gestione del personale

Il problema di gestione del personale è uno speciale caso di problemi di assegnamento le cui risorse sono di tipo umano.
Questa peculiarità li rende così specifici da essere considerati separatamente. Infatti cambi di lavoro e regolamenti impongono difficoltà che si modificano nel tempo.

1.2.3 Problema di trasporto

Una varietà di problemi di trasporto possono essere trattati. Questi problemi sono spesso molto complessi ciò è dovuto alla loro dimensione, al numero e alla varietà dei vincoli, e alla presenza di vincoli complessi in base ai vari itinerari.

1.3 Vincoli CRISP

Il vincolo [2] è semplicemente una relazione logica tra diverse incognite (variabili), ognuna delle quali può assumere un valore in un dato dominio.
Un vincolo pertanto restringe i possibili valori che le variabili possono assumere all'interno di un determinato insieme.
I vincoli godono di interessanti proprietà:

– i vincoli possono specificare informazioni *parziali*;

– i vincoli sono *non-direzionali*, cioè tipicamente da un vincolo su due variabili X e Y si può dedurre il vincolo su X dato il vincolo su Y e viceversa;

– i vincoli sono *dichiarativi*, cioè specificano la relazione che dovrebbe essere mantenuta senza specificare una procedura per migliorare la relazione;

– i vincoli sono *additivi*, cioè non importa l'ordine di imposizione dei vincoli, quello che importa alla fine è la relazione tra i vincoli;

1.3.1 I CSP

I CSP sono problemi di soddisfazione con i vincoli, i quali possono rappresentare situazioni tipiche di vita quotidiana che sono descritte da insieme di variabili legate tra loro da vincoli differenti.
I CSP furono soggetti di ricerche nel campo dell'Intelligenza Artificiale per molti anni. In particolar modo in questa occasione si restringe lo studio ai soli CSP su domini finiti, cioè problemi sottoposti a vincoli dove le variabili possono assumere un numero finito di distinte configurazioni. Questi riescono in modo semplice ad analizzare problemi generali sottoposti a vincoli e non solo, infatti sono anche utili ad esprimere fedelmente o attraverso gradi di astrazione, i principali aspetti delle diverse classi di problemi della vita reale.
Prima di procedere alla definizione di CSP di seguito verranno riportati esempi caratteristici di CSP.

Il problema delle 8 regine

Questo è un problema abbastanza noto tra gli studiosi di CSP.
Il problema consiste nel piazzare le 8 regine sulla scacchiera facendo in modo che non si attacchino tra loro.
Secondo la formulazione CSP, la localizzazione delle regine è definita dalle variabili e la richiesta di una regina di non attaccare le altre è espressa in termini di vincoli.
Un modo per fare questo è di assegnare una variabile ad ogni regina. Dato che ogni regina deve essere posizionata su una differente colonna, si può identificare ogni regina con una differente colonna e rappresentare la sua posizione con una variabile che indica la riga della regina in questione. Si assegna ad x_i la riga della regina nell'i-esima colonna. Il dominio di ogni variabile $x_1, \ldots, x_8$ è $\{1, 2, \ldots, 8\}$.

Ogni due variabili dovrebbero essere rispettati i seguenti due vincoli:

$$x_i \neq x_j$$

e

$$|x_i - x_j| \neq |i - j|$$

cioè le regine devono essere posizionate in differenti righe e differenti diagonali.
Ci possono essere diversi modi di rappresentazione di un problema tramite CSP e diversi casi hanno dimostrato che la difficoltà nella risoluzione di un problema poteva essere superata modificandone la rappresentazione.

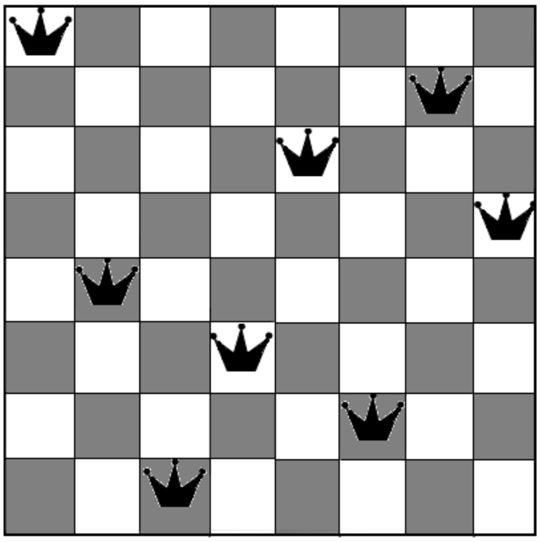


Figura 1.1: il problema delle 8 regine.

Colorazione del grafo

Un altro popolare problema è quello della colorazione del grafo. Questo problema consiste nel colorare i vertici di un determinato grafo usando k colori facendo in modo che vertici adiacenti ricevano colori differenti.
È abbastanza ovvio come convertire questo problema in CSP: ci sono tante variabili quanti i vertici, e il dominio per ogni variabile è $\{1, 2, \ldots, k\}$ dove k è il numero di colori che possono essere usati. Se c'è un arco che connette i vertici rappresentati dalle variabili x_i e x_j, allora c'è un vincolo relativo a queste variabili, cioè

$$x_i \neq x_j.$$

Sembrerebbe essere un gioco simile a quello delle 8-regine, ma ci sono delle differenze di base tra i due problemi. Primo, la colorazione del grafo è conosciuto come NP-completo, così non si attende un tempo polinomiale per l'algoritmo di ricerca. Secondo, è facile generare un grande numero di grafi da utilizzare come test con parametri sicuri, i quali sono più o meno difficili da colorare.

Dopo aver presentato alcuni popolari esempi di problemi di soddisfazione con i vincoli, si prosegue lo studio dei CSP riportatando innanzitutto la definizione di problema di soddisfazione di vincoli:

Definizione 1.3.1. *Un problema di soddisfazione dei vincoli è una tupla* $\langle V, D, C, con, def, a\rangle$ *dove*

– V *è l'insieme finito delle* variabili*, cioè* $V = \{v_1, \ldots, v_n\}$*;*

– D *è l'insieme dei valori possibili che le variabili possono assumere chiamati* domini*;*

– C *è l'insieme dei* vincoli*, cioè* $C = \{c_1, \ldots, c_m\}$*;*

– *con è chiamata la* funzione collegamento *tale che*

$$con : \bigcup_k (C_k \to V^k),$$

dove $con(c) = \langle v_1, \ldots, v_k\rangle$ *corrisponde alle tuple di variabili coinvolte in* $c \in C_k$*;*

– *def è chiamata la* funzione definizione *tale che*

$$def : \bigcup_k (C_k \to \wp(D^k)),$$

dove $\wp(D^k)$ *è la potenza di* D^k*, cioè tutti i possibili sottoinsiemi di* k*-tuple in* D^k*;*

– $a \subseteq V$*, rappresenta le* variabili distinte *del problema.*

In altre parole, la funzione *con* descrive le variabili che sono coinvolte dai vincoli, mentre la funzione *def* specifica il dominio delle tuple permesso dai vincoli. L'insieme a è usato per indicare le variabili di interesse di un dato CSP, cioè le variabili per cui si vuole conoscere i possibili assegnamenti compatibili con i vincoli.
Si noti che altre definizioni non danno la nozione di variabili distinte, come se tutte le variabili fossero rilevanti. Invece qui si tiene conto di questa definizione per due ragioni: primo, la classica definizione può essere banalmente simulata avendo un insieme di variabili distinte composto da tutte le variabili, e secondo, la definizione con l'insieme a si presta ad essere più realistica.
Le variabili distinte giocano un ruolo di variabili essenziali: si vuole assicurare che esse assumano valori compatibilmente ai vincoli.
Lo stato di un problema [8] è definito come l'assegnamento di valori ad alcune

o a tutte le variabili.
Il più semplice tipo di CSP coinvolge variabili che sono *discrete* e hanno *domini finiti*. Il problema di colorazione del grafo e delle 8 regine rispecchia queste caratteristiche.
Nei CSP con domini finiti sono compresi anche i *CSP booleani* dove le variabili possono assumere o il valore vero o il valore falso.
Inoltre variabili discrete possono avere anche *domini infiniti*, per esempio gli insiemi di numeri interi.
Speciali algoritmi di soluzione sono dati per risolvere problemi con vincoli lineari su variabili intere, e inoltre è dimostrato che non esistono algoritmi che risolvono vincoli non lineari su variabili intere. In alcuni casi si può ridurre problemi vincolati interi a problemi di dominio finito semplicemente limitando i valori di tutte le variabili.
Problemi di soddisfazione dei vincoli su *domini continui* sono molto comuni nei problemi di vita reale, per esempio problemi di programmazione dei tempi di osservazione di un qualche oggetto. La più conosciuta categoria di problemi CSP con domini continui è quella dei problemi di programmazione lineare, dove i vincoli possono essere disuguaglianze lineari che formano regioni convesse.

La soluzione *Sol(P)* di un CSP $P = \langle V, D, C, con, def, a \rangle$ è definita come l'insieme di tutte le istanze delle variabili in a le quali possono essere estese a tutte le istanze di tutte le variabili che sono coerenti con i vincoli in C.

Definizione 1.3.2. ***(proiezione delle tuple e soluzione CSP)***. *Data una tupla il cui dominio di valori sia $\langle v_1, \ldots, v_n \rangle$, si consideri una tupla di variabili $\langle x_{i1,}, \ldots, x_{im} \rangle$ tale che $\forall\ j = 1, \ldots, m$, allora esiste $k_j \in \{1, \ldots, n\}$ tale che $x_{ij} = x_{k_j}$. Allora la proiezione di $\langle v_1, \ldots, v_n \rangle$ su $\langle x_{i1}, \ldots, x_{im} \rangle$, scritto come segue $\langle v_1, \ldots, v_n \rangle_{|\langle x_{i1}, \ldots, x_{im} \rangle}$, è la tupla di valori $\langle v_{i1}, \ldots, v_{im} \rangle$. La soluzione Sol(P) di un CSP $P = \langle V, D, C, con, def, a \rangle$ è definita come*

$$\langle v_1, \ldots, v_n \rangle_{|a} \ \textit{tale che} \left\{ \begin{array}{l} v_i \in D \ \forall\ i \\ \forall\ c \in C, \langle v_1, \ldots, v_n \rangle_{|con(c)} \in def(c). \end{array} \right\}$$

La soluzione del CSP è quindi un assegnamento di valori dal suo dominio alla variabile, facendo in modo che ogni vincolo sia soddisfatto.
Per dare una rappresentazione grafica del problema CSP, si usa un ipergrafo etichettato che normalmente è chiamato *constraint graph.*

Definizione 1.3.3. ***(ipergrafo etichettato)***. *Dato un insieme di etichette L, un ipergrafo etichettato su L è una tupla $\langle N, H, c, l \rangle$, dove N è l'insieme di nodi, H è l'insieme di iperarchi, $c : \bigcup_k (H_k \to N^k)$, e $l : \bigcup_k (H_k \to \wp(L^k))$,* *cioè c restituisce le tuple di nodi collegati da ciascun iperarco, e l restituisce le etichette di ogni iperarco.*

Definizione 1.3.4. ***(dai CSP al grafo etichettato)***. *Si prenda in considerazione un CSP* $P = \langle V, D, C, con, def, a \rangle$. *Allora un ipergrafo etichettato corrispondente a P, scritto con* $G(P)$, *è definito come un ipergrafo* $G(P) = \langle V, C, con, def \rangle$ *etichettato su D.*

Nell'ipergrafo corrispondente ad un CSP, i nodi rappresentano le variabili del problema e gli iperarchi rappresentano i vincoli.
Come soluzione di un CSP di dominio finito, si consideri l'esempio della figura 1.2, dove ogni arco è etichettato dalla definizione del corrispondente vincolo, dato in termini di tuple di valori di dominio e variabili distinte (segnate con un $*$). La soluzione di questo problema è l'insieme $\{\langle a, b \rangle\}$.

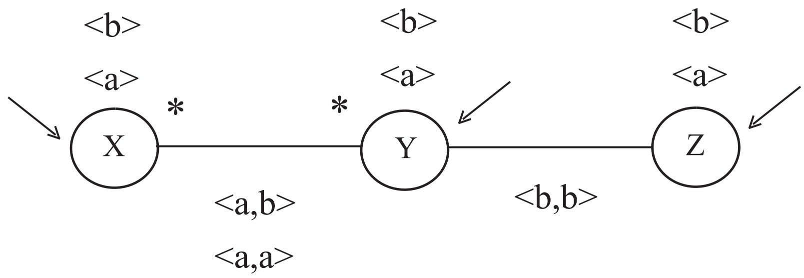


Figura 1.2: soluzione di un CSP.

1.3.2 Constraint Logic Programming (CLP) e tecniche di soluzione

Il termine "Constraint Logic Programming" sta ad indicare lo studio di sistemi computazionali basati sui vincoli. Il suo scopo è quello di risolvere problemi sottoposti a vincoli i quali possono essere soddisfatti dalle soluzioni. Soltanto nell'ultimo decennio, c'è stata una crescente crescita di queste idee le quali forniscono le basi all'approccio di programmazione, risoluzione e modellazione di problemi.
I diversi sforzi all'esplorazione di queste idee possono essere unificate sotto questo comune concetto: CLP.
Attualmente può essere individuato un principale campo di ricerca sul quale si fonda questo tipo di programmazione : *Constraint Satisfaction* [1].
Il problema di soddisfazione dei vincoli (CSP) è un problema composto da:

– un insieme finito di variabili;

una funzione che associa ad ogni variabile un dominio finito;

– un insieme finito di vincoli.

Come già è stato affermato, ogni vincolo restringe la combinazione dei valori che un insieme di variabili può assumere contemporaneamente.
Una soluzione di un CSP è un assegnamento ad ogni variabile di un valore dal suo dominio che soddisfa tutti i vincoli. La richiesta può essere quella di trovare tutte le soluzioni o soltanto una.
Esistono diversi tipi di ricerca:

– *Ricerca sistematica*, che effettua l'esplorazione di tutto lo spazio di ricerca.
  Un algoritmo di ricerca sistematica è il GT (Generate-and-Test). Questo metodo consiste nel generare assegnamenti completi alle variabili e per ognuno di questi testare se rispettano i vincoli. Questo algoritmo viene utilizzato come tecnica di risoluzione dei problemi di soddisfazione con i vincoli.
  La struttura del programma è abbastanza semplice, c'è la necessità di creare cicli che effettuino le stesse operazioni per tutte le variabili e che al loro interno controllino il rispetto dei vincoli per tutte le variabili.
  Questo metodo molto spesso è molto lento e la sua efficienza è mediocre in quanto il generatore degli assegnamenti non è informato ed esiste una scoperta tardiva delle inconsistenze.
  Ci sono due modi per migliorare l'efficienza di questo algoritmo:

  - con generatore di valutazione "smart" (informato), il quale genera valutazioni complete facendo in modo che il conflitto trovato dalla fase di test risulti minimizzato. Questa è l'idea base degli algoritmi di ricerca locale stocastica.
  - con generatore combinato con il tester, cioè il rispetto del vincolo viene testato nel momento in cui la variabile viene istanziata. Questa è l'idea sfruttata dagli algoritmi "backtracking".
    Questi algoritmi costruiscono una soluzione estendendo una parziale istanza passo dopo passo, affiancando differenti euristiche e usando più o meno intelligenti strategie di backtracking per trovare una soluzione. La riduzione del problema è vantaggiosa, risultante in un più piccolo spazio di ricerca.
    Le variabili sono etichettate sequenzialmente e quindi viene subito testata la validità del vincolo. Se una soluzione parziale viola ogni vincolo, la ricerca backtracking viene eseguita sulla variabile più recentemente istanziata che ha delle alternative disponibili.

Chiaramente se un'istanza parziale viola un vincolo, il backtracking è capace di eliminare il sottospazio dal prodotto cartesiano dei domini di tutte le variabili. Di conseguenza il backtracking è strettamente migliore del già descritto GT.
Ci sono tre principali svantaggi del backtracking:

1. la sconfitta, cioè l'insuccesso ripetuto dovuto per la stessa ragione;
2. lavoro ridondante, cioè valori di variabili in conflitto non sono ricordati;
3. scoperta ritardataria dei conflitti, il conflitto non è scoperto prima del momento in cui realmente ricorre.

– *Tecniche di consistenza*, questo approccio consiste nell'eliminazione dei valori inconsistenti dai domini delle variabili.
Ci sono differenti tecniche di consistenze, ma la maggioranza di loro non sono complete.
La più semplice tecnica di consistenza è riferita alla consistenza del nodo *(NC)* mentre quella maggiormente utilizzata è riferita alla consistenza dell'arco *(AC)*.

Concetti di consistenza

In generale ogni assegnamento che non viola nessun vincolo è chiamato *consistente o legale*.
Inoltre un *completo assegnamento* è quello in cui ogni variabile viene menzionata. Allora è possibile definire come soluzione di un CSP un completo assegnamento che soddisfa tutti i vincoli.

Definizione 1.3.5. ***(nodo-consistente)***. *Un CSP* è nodo-consistente*, se tutti i vincoli unitari sono mantenuti per tutti gli elementi del dominio.*

Il semplice algoritmo nodo-consistente (NC), che rimuove gli elementi ridondanti controllando i domini uno dopo l'altro, ha complessità di tempo $O(dn)$, dove d è il massimo della grandezza dei domini e n il numero di variabili.

Definizione 1.3.6. ***(k-consistente)***. *Un CSP è* arco-consistente*, se per ogni valore u dal dominio di ogni variabile x, ogni vincolo binario che riferisce ad x può essere soddisfatto.*

Ogni parziale soluzione contenente solo una variabile istanziata può essere estesa istanziando una seconda variabile ad un valore scelto. Applicando lo stesso principio per più variabili, si arriva al concetto di k-consistente.

Definizione 1.3.7. ***(k-consistente)***. *Un CSP è k-consistente, se ogni istanza consistente di ogni $k-1$ variabili può essere estesa istanziando ognuna delle rimanenti variabili.*

La definizione afferma che un CSP si dice k-consistente se, per ogni $k-1$ variabili e per ogni assegnamento consistente a queste variabili, un assegnamento consistente può essere assegnato ad ogni k-esima variabile. Per esempio, 1-consistenza significa che ogni variabile individuale è consistente, invece 2-consistente significa che la stessa cosa vale per un arco consistente, e infine, 3-consistente significa che ogni coppia di variabili adiacenti può essere estesa a una terza variabile adiacente; quest'ultimo è anche chiamato *percorso consistente*.
È importante comprendere chiaramente il significato di consistenza. Un CSP che è k-consistente non è necessariamente risolvibile, e in altro modo la risolvibilità di un problema non implica ogni livello di consistenza. La consistenza come caratteristica del problema non garantisce che certi valori, e i valori delle h-tuple che non sono proiezioni dell'insieme di soluzioni che sono state rimosse, formino i domini e i vincoli. Il livello di consistenza k indica per quali h-valori questo è fatto.

– *Propagazione dei vincoli*, è una tecnica di combinazione delle tecniche precedenti. Sia la Ricerca sistematica che le Tecniche di consistenza possono essere utilizzate singolarmente ma raramente ciò è fatto.
Eliminando valori ridondanti dalla definizione del problema, la grandezza dello spazio di soluzione decresce.
La riduzione dello spazio di soluzione può essere fatto una sola volta, come processo prima di un altro algoritmo, o passo dopo passo, interponendolo all'esplorazione dello spazio di soluzione che viene effettuata da un algoritmo di ricerca.
Gli algoritmi di riduzione eliminano valori con la propagazione dei vincoli. La quantità dei vincoli propagati è caratterizzata da livelli di consistenza dei vincoli, quindi questi algoritmi sono chiamati *algoritmi di consistenza*. In generale, la principale idea è migliorare la forza di backtracking riducendo lo spazio di ricerca. Questo è fatto trasformando l'insieme originale dei vincoli ad uno più esplicito (eliminando la

ridondanza), e poi effettuando la ricerca. Comunque una tupla di dominio di valori può essere considerata ridondante se la sua eliminazione non cambia la soluzione del problema.
Una condizione sufficiente per le tuple ridondanti è che le tuple devono essere inconsistenti con tutte le tuple in alcuni vincoli nel CSP. Si consideri per esempio, un CSP contenente le variabili x, y e z e vincoli come mostrati dalla figura 1.3. Allora la tupla $\langle a, a \rangle$ di vincoli tra x

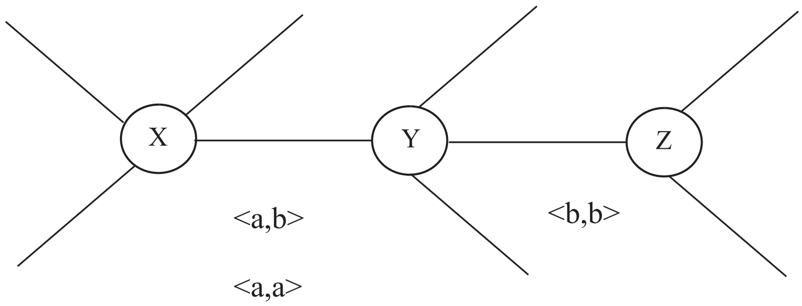


Figura 1.3: tuple ridondanti.

e y risulta inconsistente con il vincolo tra y e z. Infatti se x e y sono entrambi fissati ad a, allora non c'è modo di instanziare z in modo da soddisfare il vincolo tra y e z. Più precisamente dato un CSP P_1, lo scopo è ottenere un nuovo CSP P_2 tale che P_1 e P_2 hanno le stesse soluzioni e lo stesso grafo vincolato, e P_2 ha una più piccola definizione di vincoli.

– *Algoritmi stocastici ed euristici*, questi sono algoritmi di soddisfazione dei vincoli che estendono una etichetta parziale consistente a una piena etichetta che soddisfa tutti i vincoli.
Questi algoritmi modificano in modo incrementale assegnamenti inconsistenti a tutte le variabili ed inoltre utilizzano la metafora "riparazione" (repair) o "arrampicata" (hill-climbing) per moversi attraverso soluzioni complete. Inoltre per evitare di bloccarsi ad un minimo locale, essi sono dotati di varie euristiche per randomizzare la ricerca. La loro natura stocastica li libera dalle garanzie di una ricerca completa dei metodi di ricerca sistematica.
Hill-climbing è il più famoso algoritmo di ricerca locale. Esso parte da un'etichetta generata automaticamente, e ad ogni passo cambia il valore di alcune variabili facendo in modo che l'etichetta risultante soddisfi più vincoli. L'algoritmo si ferma quando un minimo globale viene trovato, cioè quando tutti i vincoli sono soddisfatti o quando tutte le risorse

sono esaurite. Si consideri che l'algoritmo deve esplorare un piccolo numero di vicini dello stato corrente prima di scegliere la mossa.

Forse il più famoso dominio di applicazione per CLP (Constraint Logic Programming) sono i problemi di programmazione. Dato un'insieme di risorse con determinate capacità, un'insieme di attività con differenti durate e necessità di risorse, e un insieme di vincoli temporali tra le attività, un problema di programmazione consiste nella decisione dei tempi di ciascuna attività, soddisfacendo la necessità delle risorse e i vincoli di tempo.

Gli algoritmi che indagano lo spazio di soluzione sistematicamente trovano una soluzione e quindi questi algoritmi risultano:

- *sicuri*, se terminano con una completa istanza della variabile che è la soluzione;

- *completi*, se sono capaci di investigare tutto l'intero spazio di ricerca e quindi trovano tutte le soluzioni.

Un CSP è difficile se l'algoritmo analizza l'intero spazio di ricerca prima di trovare una soluzione o conclude che il problema non ha soluzioni. Se lo spazio di ricerca è grande, allora un algoritmo sicuro e completo può impiegare giorni o settimane.
In tal caso, come già descritto con gli algoritmi di ricerca locale e stocastica, un compromesso viene trovato, usando un algoritmo che fornisce una risposta in breve tempo, ma non è garantito che questa sia la soluzione.

1.4 Vincoli NON-CRISP

In tutte le precedenti applicazioni (CRISP) vi sono alcuni importanti limitazioni, principalmente dovute al fatto che esse non si mostrano flessibili quando si prova a rappresentare gli scenari di vita reale dove la conoscenza non è nè disponibile nè chiara.
Per esempio, i CSP sono stati estesi con l'abilità di associare ad ogni tupla o ad ogni vincolo, un livello di preferenza e con la possibilità di combinare vincoli usando operazioni di min-max. Questo formalismo esteso fu chiamato *Fuzzy CSP* (FCSP).
I vincoli nel problema classico di soddisfazione dei vincoli sono crisp, cioè essi o permettono una tupla (di valori delle variabili coinvolte) o no. In alcune applicazioni della vita reale, questo non è un aspetto desiderato e quindi alcuni approcci alternativi alla soddisfazione dei vincoli furono proposti permettendo vincoli non-crisp.

Risolvere problemi di soddisfazione dei vincoli riferiti a problemi di vita reale può risultare abbastanza complicato, si vedrà in seguito come attraverso la teoria dei giochi è stato possibile ovviare a questa complicazione fornendo un metodo di trasformazione di un SCSP in un gioco non cooperativo statico ad informazione completa e di conseguenza risolverlo attraverso i software esistenti per la risoluzione di giochi non cooperativi.

Capitolo 2

Soft Constraint Satisfaction Problems

I problemi classici di soddisfazione dei vincoli (CSP) sono un formalismo molto espressivo e naturale nello specificare molti tipi di problemi della vita reale.
Essi comunque hanno evidenti limiti, principalmente dovuti al fatto che non appaiono molto flessibili a rappresentare situazioni della vita quotidiana dove la conoscenza nè è completamente disponibile nè chiara. Infatti in tale situazione, l'abilità di affermare se un'istanza di valori a variabili è permessa o no, non è abbastanza, o qualche volta non è per niente, possibile.
Per queste ragioni, è naturale provare ad estendere il formalismo CSP in questa direzione.
Si estente la classica nozione di soddisfazione dei vincoli per permettere anche:

– relazioni non-crisp;

– una più generale interpretazione delle operazioni su tali relazioni.

Si associa un semiring con la nozione standard di problema di soddisfazione di vincoli, così che differenti scelte di semiring (o meglio dei componenti del semiring) rappresentano differenti concreti schemi di soddisfazione di vincoli.

Definizione 2.0.1. *Un semiring è una tupla* $\langle A, sum, \times, \boldsymbol{0}, \boldsymbol{1} \rangle$ *tale che*

– *A è un insieme e* $\boldsymbol{0}$, $\boldsymbol{1} \in A$;

– *sum, è un'operazione generica con alcune importanti proprietà: chiusa, additiva, commutativa (cioè,* $sum(a, b) = sum(b, a)$*) e associativa (cioè* $sum(a, sum(b, c)) = sum(sum(a, b), c)$*) con* $\boldsymbol{0}$ *come elemento unità (cioè* $sum(a, \boldsymbol{0}) = a = sum(\boldsymbol{0}, a)$*);*

– $\times$, *è un'altra operazione generica in questo caso con le seguenti proprietà: chiusa, moltiplicativa, associativa tale che* $\boldsymbol{1}$ *è il suo elemento unità e* $\boldsymbol{0}$ *è l'elemento di assorbimento (cioè* $a \times \boldsymbol{0} = \boldsymbol{0} = \boldsymbol{0} \times a$*); e inoltre distributiva sulla somma (cioè per ogni* $a, b, c \in A$, $a \times sum(b, c) = sum((a \times b), (a \times c))$*).*

2.1 C-semiring e le loro proprietà

Di seguito verranno considerati semiring con proprietà addizionali per le due operazioni. Tali semiring sono chiamati *c-semiring* dove "c" sta per "constraint" (vincolo), ciò significa che c-semiring sono strutture usate quando si sta trattando con i vincoli.

Definizione 2.1.1. ***(c-semiring).*** *Un c-semiring è una tupla* $\langle A, +, \times, \boldsymbol{0}, \boldsymbol{1} \rangle$ *tale che*

– A *è un insieme e* $\boldsymbol{0}, \boldsymbol{1} \in A$*;*

– $+$ *è definito su gli insieme di elementi di* A *come segue:*

- $\forall\, a \in A, \sum(\{a\}) = a$*;*
- $\sum(\emptyset) = \boldsymbol{0}$ *e* $\sum(A) = \boldsymbol{1}$*;*
- $\sum(\bigcup A_i, i \in I) = \sum(\{\sum(A_i), i \in I\})$ *per tutti gli insiemi di indice* I *(flattening property).*

– $\times$ *è un'operazione binaria, associativa e commutativa tale che* $\boldsymbol{1}$ *è l'elemento unità e* $\boldsymbol{0}$ *è l'elemento assorbimento; ed è distributiva rispetto a* $+$.

È facile notare come un c-semiring è, infatti, un semiring con proprietà addizionali.
Si possono elaborare le seguenti considerazioni sull'operazione additiva ($+$). Il fatto che $+$ è definita su insiemi di elementi, e non su coppie o tuple, automaticamente rende tale operazione:

– *commutativa*;

– *associativa*;

– *idempotente.*

Inoltre grazie l'idempotenza di + (dimostrabile in base alla proprietà $\sum(\{a\}) = a$ facente parte della definizione 2.1.1) è possibile definire un'ordinamento parziale $\leq_S$ sull'insieme A, che permette di confrontare differenti elementi del semiring.
Tale ordinamento parziale è definito come segue:

$$a \leq_S b \text{ } sse \text{ } a + b = b.$$

Intuitivamente, $a \leq_S b$ significa che b è migliore di a, o da un altro punto di vista, tra a e b l'operazione + sceglie b.

Teorema 2.1.2. ***($\leq_s$ è un ordinamento parziale).*** *Per ogni c-semiring $S = \langle A, sum, \times, \mathbf{0}, \mathbf{1}\rangle$, si consideri la relazione $\leq_S$ su A tale che $a \leq b$ sse $a + b = b$. Allora $\leq_S$ è un ordinamento parziale.*

È importante che entrambe le operazioni, additiva e moltiplicativa, siano monotone su tale ordinamento.

Teorema 2.1.3. ***(+ e $\times$ sono monotone su $\leq_S$).*** *Dati qualsiasi c-semiring $S = \langle A, sum, \times, \mathbf{0}, \mathbf{1}\rangle$, si consideri la relazione $\leq_S$ su A. Allora + e $\times$ sono monotone su $\leq_S$. Cioè $a \leq_s a'$ implica $a + b \leq_S a' + b$ e $a \times b \leq_S a' \times b$.*

È possibile mostrare come l'elemento $\mathbf{0}$ risulta essere:

- l'*elemento unità* di +: infatti in base la flattening property della definizione 2.1.1 si ha

$$\sum(\{a, \mathbf{0}\}) = \sum(\{a\} \bigcup \emptyset) = \sum(\{a\}) = a;$$

- l'*elemento minimo* dell'ordimento parziale: il fatto che $\mathbf{0}$ è l'elemento unità dell'operazione additiva implica che $\mathbf{0}$ è il *minimo elemento* dell'ordinamento. Pertanto in base al teorema 2.1.2, dato che $a + \mathbf{0} = a$ allora si può affermare che $\mathbf{0} \leq a$ (quest'ultima disuguaglianza viene mantenuta per ogni $a \in A$).

Inoltre è possibile dimostrare che $\mathbf{1}$ sia:

- l'*elemento assorbimento* dell'operazione additiva. Ancora in base alla flattening property e in base al fatto che $\sum(A) = \mathbf{1}$

$$\sum(\{a, \mathbf{1}\}) = \sum(\{a\} \bigcup A) = \sum(A) = \mathbf{1}.$$

– l'*elemento massimo* dell'ordinamento parziale. Dovuto al fatto che **1** è anche l'elemento di assorbimento dell'operazione additiva, infatti in base al teorema 2.1.2 vale che $a + \mathbf{1} = \mathbf{1}$, si ha che $a \leq_S \mathbf{1}$ è valida per ogni $a \in A$. Pertanto l'elemento **1** risulta essere l'*elemento massimo* dell'ordinamento parziale.

In base a ciò che è stato appena affermato, il fatto che il massimo elemento dell'ordinamento parziale sia **1** implica che l'operazione $\times$ sia intensiva.

Teorema 2.1.4. ***($\times$ è intensiva).*** *Per ogni c-semiring $S = \langle A, +, \times, \mathbf{0}, \mathbf{1}\rangle$, si consideri la relazione $\leq_S$ su A. Allora $\times$ è intensiva, cioè per $a, b \in A$ implica che $a \times b \leq_S a$.*

In base al teorema 2.1.4 $a \times b \leq_S a$, in quanto il massimo valore che potrebbe assumere b è **1** che in tal caso porterebbe ad un risultato che rispetta la precedente disuguaglianza: $a \times \mathbf{1} = a$. Questo è importante poichè significa che la combinazione di più vincoli portarebbe ad un risultato peggiore.

Potrebbe essere necessario, quando si considerano i c-semiring, sfruttare la proprietà dell'idempotenza dell'operatore moltiplicativo, che risulta equivalente alla definizione di **lattice distributivo**.

Definizione 2.1.5. ***(lub, glb, lattice (completo)).*** *Si consideri un insieme ordinato parzialmente S e ogni sottoinsieme I di S. Si può definire:*

– *un limite superiore di I è ogni elemento x tale che, per ogni $y \in I, y \leq x$;*

– *il più piccolo limite superiore (lub)(o il più grande limite inferiore (glb)) di I è un limite superiore (o limite inferiore) x di I tale che, per ogni altro limite superiore (o limite inferiore) x' di I, si ha che $x \leq x'$ (o $x' \leq x$).*

Un lattice è un insieme parzialmente ordinato dove ogni sottoinsieme di due elementi ha un lub e un glb.
Un lattice completo è un insieme parzialmente ordinato dove ogni sottoinsieme ha un lub e un glb. Si noti che quando ogni sottoinsieme ha un lub, allora anche l'insieme vuoto ha il lub. Con ciò c'è sempre un minimo globale dell'ordine parziale (che corrisponde al lub dell'insieme vuoto).

Lemma 2.1.6. ***(lub $\Rightarrow$ glb).*** *Si consideri un ordine parziale $\langle A, \leq\rangle$ dove esiste un lub per ogni sottoinsieme I di A. Allora esiste anche il glb di I.*

Teorema 2.1.7. ***($\langle A, \leq_S\rangle$ è un lattice completo).*** *Dato un c-semiring $S = \langle A, +, \times, \mathbf{0}, \mathbf{1}\rangle$, si consideri l'ordinamento parziale $\leq_S$. Allora $\langle A, \leq_s\rangle$ è un lattice completo.*

In base al teorema precedente si può dedurre anche che l'operazione $+$ coincide con il lub del lattice.

Teorema 2.1.8. ***($\times$ idempotente).*** *Dato un c-semiring $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$, si consideri il corrispondente lattice completo $\langle A, \leq_S \rangle$. Se $\times$ è idempotente, si ha:*

– *$+$ distributiva su $\times$;*

– *$\times$ coincide con l'operazione glb del lattice;*

– *$\langle A, \leq_S \rangle$ è un lattice distributivo.*

Si noti che, nel caso particolare in cui $\times$ è idempotente e $\leq_S$ è totale, si ha che $a + b = \max(a, b)$ e $a \times b = \min(a, b)$.

2.1.1 Sistemi di vincoli e problemi

Intuitivamente, un sistema di vincoli specifica il c-semiring $\langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ usato, l'insieme di tutte le variabili e il loro dominio D.

Definizione 2.1.9. ***(sistema di vincoli).*** *Un sistema di vincoli è definito come una tupla $CS = \langle S, D, V \rangle$, dove S è un c-semiring, D è un insieme finito, e V è un insieme ordinato di variabili.*

Un vincolo su un dato sistema di vincoli specifica le variabili coinvolte e i valori che possono assumere. Più precisamente per ogni tupla di valori è assegnato alle variabili un corrispondente elemento di A. Questo elemento può essere interpretato come un peso per le tuple, o un costo, o un livello di confidenza, o altro.

Definizione 2.1.10. ***(vincolo).*** *Dato un sistema di vincoli $CS = \langle S, D, V \rangle$, dove $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$, un vincolo su CS è una coppia $\langle def, con \rangle$, dove:*

– *$con \subseteq V$, esso è chiamato "tipo" di vincoli;*

– *$def : D^k \rightarrow A$ (dove k è la cardinalità di con) è chiamato il "valore" dei vincoli.*

Un problema vincolato è composto da un insieme di vincoli su un dato sistema vincolato e da un insieme selezionato di variabili.

Definizione 2.1.11. ***(problema con i vincoli).*** *Si consideri un sistema di vincoli $CS = \langle S, D, V \rangle$. Un problema con i vincoli P su CS è una coppia $P = \langle C, con \rangle$, dove C è un insieme di vincoli su CS e $con \subseteq V$. Si può assumere che $\langle def_1, con' \rangle \in C$ e $\langle def_2, con' \rangle \in C$ implica che $def_1 = def_2$. In seguito si scriverà che SCSP (for Semiring-based CSP) si riferisce a tale problema di vincoli.*

Si noti che tale condizione non è restrittiva. Infatti, se in un problema si hanno due vincoli $\langle def_1, con' \rangle$ e $\langle def_2, con' \rangle$, si possono rimpiazzare entrambi con un singolo vincolo $\langle def, con' \rangle$ con $def_1(t) = def_1(t) \times def_2(t)$ per ogni tupla t.
Quando tutte le variabili sono distinte, come in molti approcci dei CSP classici, *con* contiene tutte le variabili coinvolte in ognuno dei vincoli di un dato problema, diciamo P. Tali insieme, chiamato $V(P)$, è un sottoinsieme di V che può essere recuperato osservando le variabili coinvolte in ogni vincolo. Cioè $V(P) = \bigcup_{\langle def, con' \rangle \in C} con'$.

I SCSP, come i CSP, possono anche essere definiti graficamente, rappresentati attraverso ipergrafi etichettati dove i nodi corrispondono alle variabili, gli iperarchi sono i vincoli, e ogni iperarco etichettato è la definizione del corrispondente vincolo. Le variabili distinte normalmente sono marcate nel grafico.
Nella struttura CSP, i valori specificati per le tuple di ogni vincolo sono usate per calcolare i corrispondenti valori per le tuple di valori delle variabili in *con*, in conformità alle operazioni del semiring: l'operazione moltiplicativa è usata per combinare i valori delle tuple di ogni vincolo e quindi per ottenere il valore di una tupla per tutte le variabili, mentre l'operazione additiva è usata per ottenere il valore delle tuple delle variabili per il tipo di problema. Più precisamente l'operazione di combinazione ($\otimes$) e proiezione ($\Downarrow$) sui vincoli.
Per definire l'operatore di proiezione sui vincoli c'è bisogno di adattare la definizione 1.3.2 alla proiezione di tuple con riferimento ai SCSP.

Definizione 2.1.12. ***(proiezione di tuple).*** *Dato un sistema di vincoli $CS = \langle S, D, V \rangle$ dove V è totalmente ordinato attraverso $\prec$, consideriamo ogni k-tuple $t = \langle t_1, \ldots, t_k \rangle$ di valori di D e due insiemi $W = \{w_1, \ldots, w_j\}$ e $W' = \{w'_1, \ldots, w'_m\}$ tale che $W' \subseteq W \subseteq V$ e $w_i \prec w_j$ se $i \leq j$ e $w'_i \prec w'_j$ se $i \leq j$. Allora la proiezione di t da W a W', scritto come $t \downarrow^W_{W'}$, è definita come la tupla $t' = \langle t'_1, \ldots, t'_m \rangle$ con $t'_i = t_j$ se $w'_i = w_j$.*

Definizione 2.1.13. ***(combinazione).*** *Dato un sistema di vincoli $CS = \langle S,$*

$D, V\rangle$ dove $S = \langle A, +, \times, \mathbf{0}, \mathbf{1}\rangle$, e due vincoli $c_1 = \langle def_1, con_1\rangle$ e $c_2 = \langle def_2, con_2\rangle$ su CS, la loro combinazione, scritta $c_1 \otimes c_2$ è il vincolo $c = \langle def, con\rangle$ con

$$con = con_1 \cup con_2$$

e

$$def(t) = def_1(t \downarrow^{con}_{con_1}) \times def_2(t \downarrow^{con}_{con_2})$$

Poichè $\times$ è sia commutativa che associativa, allora lo è anche $\otimes$. Pertanto questa operazione può essere facilmente estesa a più di due argomenti, con $C = \{c_1, \ldots, c_n\}$ ottenuto con $c_1 \otimes c_2 \otimes \ldots \otimes c_n$, che potra essere indicato con $\bigotimes C$.

Definizione 2.1.14. ***(proiezione).*** *Dato un sistema di vincoli $CS = \langle S, D, V\rangle$ dove $S = \langle A, +, \times, \mathbf{0}, \mathbf{1}\rangle$, un vincolo $c = \langle def, con\rangle$ su CS, e un insieme I di variabili ($I \subseteq V$), la proiezione di c su I, scritta con $c \Downarrow_I$, è il vincolo $\langle def', con'\rangle$ su CS con*

$$con' = I \cap con$$

e

$$def'(t') = \sum_{\{t | t \downarrow^{con}_{I \cap con} = t'\}} def(t)$$

Proposizione 2.1.15. ($c \Downarrow_I = c \Downarrow_{I \cap con}$)**.** *Dato un vincolo $c = \langle def, con\rangle$ definito su un sistema di vincoli CS e un insieme I di variabili ($I \subseteq V$), ha che $c \Downarrow_I = c \Downarrow_{I \cap con}$*

Una proprietà utile dell'operazione di proiezione è la seguente.

Teorema 2.1.16. ($c \Downarrow_{S_1} \Downarrow_{S_2} = c \Downarrow_{S_2}$)**.** *Dato un vincolo c su un sistema di vincoli CS, si ha che $c \Downarrow_{S_1} \Downarrow_{S_2} = c \Downarrow_{S_2}$ se $S_2 \subseteq S_1$.*

Teorema 2.1.17. *Dati due vincoli $c = \langle def_1, con_1\rangle$ e $c = \langle def_2, con_2\rangle$ su un sistema di vincoli CS, si ha che $(c_1 \otimes c_2) \Downarrow_{(con_1 \cup con_2) - x} = c_1 \Downarrow_{con_1 - x} \otimes c_2$ se $x \cap con_2 = \emptyset$.*

Usando le proprietà di $\times$ e $+$, è facile provare che: $\otimes$ è associativa e commutativa; $\otimes$ è monotona su $\sqsubseteq_S$. Comunque se $\times$ è idempotente allora anche $\otimes$ è idempotente.
Utilizzando le nozioni di combinazione e proiezione, si può definire la nozione di soluzione di un SCSP.

Definizione 2.1.18. ***(soluzione).*** *Dato un problema vincolato $P = \langle C, con\rangle$ su un sistema di vincoli CS, la soluzione di P è un vincolo definito come $Sol(P) = (\otimes C) \Downarrow_{con}$.*

La soluzione di SCSP è un vincolo indotto sulle variabili in *con* dall'intero problema. Tale vincolo fornisce, per ogni tupla di valori di D asseganate alle variabili in *con*, un corrispondente valore di A. Alcune volte è abbastanza conosciuto come "best value" associato a tali tuple. Questo è ancora un vincolo, che sarà chiamato come il "best level of consistency" dell'intero problema, dove il significato di "best" dipende dall'ordinamento $\leq_S$ definito all'operazione additiva.

Definizione 2.1.19. ***(best level of consistency).*** *Dato un SCSP* $P = \langle C, con \rangle$, *si definisce come* $blevel(P) \in S$ *tale che* $\langle blevel(P), \emptyset \rangle = (\bigotimes C) \Downarrow_{\emptyset}$. *Inoltre, si può dire che* P *è consistente se* $0 <_S blevel(P)$, *e che* P *è* α-*consistente se* $blevel(P) = \alpha$.

Informalmente il best level of consistency ci suggerisce l'idea di quanto si possono soddisfare i vincoli di un dato problema. Si noti che $blevel(P)$ può essere anche ottenuto per prima calcolando le soluzioni e poi proiettando tale vincolo sull'insieme vuoto di variabili.

Proposizione 2.1.20. *Dato un SCSP* P, *si ha che* $Sol(P) \Downarrow_{\emptyset} = \langle blevel(P), \emptyset \rangle$.

È possibile definire il concetto di ordinamento dei vincoli.

Definizione 2.1.21. ***(ordinamento dei vincoli).*** *Si consideri due vincoli* c_1, c_2 *su* CS,*e si assuma che* $con_1 = con_2$ *e* $|con_1| = k$. *Allora* $c_1 \sqsubseteq_S c_2$ *se e soltanto se, per tutte le* k-*tuple* t *di valori da* D, $def_1(t) \leq def_2(t)$.

Teorema 2.1.22. ***($\sqsubseteq_S$ è un ordinamento parziale).*** *La relazione* $\sqsubseteq_S$ *su un insieme di vincoli* CS *è un ordinamento parziale.*

Definizione 2.1.23. ***(problema di ordinamento ed equivalenza).*** *Si consideri due SCSP* $P_1 = \langle C_1, con \rangle$ *e* $P_1 = \langle C_2, con \rangle$ *su* CS. *Allora* $P_1 \sqsubseteq_P P_2$ *se* $Sol(P_1) \sqsubseteq_S Sol(P_2)$. *Se* $P_1 \sqsubseteq_P P_2$ *e* $P_2 \sqsubseteq_P P_1$ *allora essi hanno la stessa soluzione. Pertanto si dice che* P_1 *e* P_2 *sono equivalenti e si scrive che* $P_1 \equiv P_2$.

Teorema 2.1.24. ***($\sqsubseteq_P$ è un pre-ordinamento e $\equiv$ è un'equivalenza).*** *La relazione* $\sqsubseteq_P$ *su l'insieme dei problemi vincolati su* CS *è un pre-ordinamento. Inoltre,* $\equiv$ *è una relazione di equivalenza.*

L'ordinamento $\sqsubseteq_P$ può anche essere usato per ordinare insieme di vincoli, quando un insieme di vincoli è solo un problema dove *con* contiene tutte le variabili.

2.2 Functional constraint

Diverse formalizzazioni del concetto di *soft constraint* sono disponibili. In precedenza è stata mostrata la formalizzazione sui c-semiring, la quale può essere utilizzata per generalizzare ed esprimere molte altre formalizzazioni. Dato un semiring $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ e l'insieme ordinato di variabili V su il dominio D, si metterà in rilievo le proprietà necessarie per mappare i soft constraint su di un sistema vincolato “a la Saraswat”.

Definizione 2.2.1. ***(functional constraints).*** *Si definisce $\mathcal{C} = (V \to D) \to A$ come l'insieme di tutti i possibili vincoli che può essere costruito partendo da $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$, D e V.*

Una generica funzione descrive l'assegnamento di elementi del dominio alle variabili. Questa funzione sarà rappresentata da $\eta : V \to D$. Pertanto un vincolo è una funzione che, dato un assegnamento η di variabili, restituisce un valore del semiring.
Si noti che in questa *functional formulation*, ogni vincolo è una funzione e non una coppia che rappresenta la variabili coinvolte e le loro definizioni. Tale funzione, seppure coinvolge tutte le variabili in V, dipende solo dall'assegnamento di un insieme finito di loro, chiamato *support*.

Definizione 2.2.2. ***(constraint support).*** *Si consideri un vincolo $c \in \mathcal{C}$. Si definisce come suo supporto $supp(c) = \{v \in V \mid \exists\ \eta, d_1, d_2 . c\eta[v := d_1] \neq c\eta[v := d_2]\}$, dove*

$$\eta[v := d]v' = \begin{cases} d & \textit{if } v = v', \\ \eta v' & \textit{altrimenti.} \end{cases}$$

Si noti che $c\eta[v := d_1]$ significa che $c\eta$ dove η' è η modificato con $v := d_1$(cioè l'operatore [ ] ha la precedenza sull'applicazione).

Definizione 2.2.3. ***(functional mapping).*** *Per ogni soft constraint $\langle def, \{v_1, \ldots, v_n\} \rangle \in C$, si può definire la sua corrispondente funzione $c \in \mathcal{C}$ in modo che $c\eta[v_1 := d_1] \ldots [v_n := d_n] = def(d_1, \ldots, d_n)$. Chiaramente $supp(c) \subseteq \{v_1, \ldots, v_n\}$.*

Definizione 2.2.4. ***(combinazione e somma).*** *Dato un insieme $\mathcal{C}$, si può definire come funzioni di combinazione e di somma ripettivamente $\otimes, \oplus : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ come segue:*

$$(c_1 \otimes c_2)\eta = c_1\eta \times_S c_2\eta$$

e

$$(c_1 \oplus c_2)\eta = c_1\eta +_S c_2\eta$$

Si noti che la funzione $\otimes$ ha la stesso significato che già definito operatore $\otimes$ nei c-semiring.

Definizione 2.2.5. ***(funzioni costanti).*** *Si definisce una funzione $\bar{a}$ come funzione che restituisce il valore del semiring a per tutti gli assegnamenti η, cioè, $\bar{a}\eta = a$. Si scrive normalmente $\bar{a}$ come semplicemente a.*

Proposizione 2.2.6. *Il supporto di un vincolo $c \Downarrow_I$ è sempre un sottoinsieme di I (cioè $supp(c \Downarrow_I) \subseteq I$).*

Teorema 2.2.7. ***(higher-order semiring).*** *La struttura $S_C = \langle \mathcal{C}, \oplus, \otimes, \mathbf{0}, \mathbf{1} \rangle$ dove:*

- *$\mathcal{C} : (V \rightarrow D) \rightarrow A$ è l'insieme di tutti i possibili vincoli che possono essere costruiti partendo da S, D e V come definito nella definizione 2.2.1;*
- *$\otimes$ e $\oplus$ sono funzioni definite nella definizione 2.2.4;*
- *$\mathbf{0}$ e $\mathbf{1}$ sono le funzioni costanti definite nella definizione 2.2.5;*

è un c-semiring.

2.3 Istanze della struttura

Si mostrerà di seguito che diverse strutture per la risoluzione di vincoli possono essere presentate sotto forma di istanze di una struttura SCSP.
La formalizzazione che è utilizzata come riferimento per definire le diverse tipologie di SCSP è quella basata sui c-semiring. Questo significa che si può capire se si può ereditare le proprietà delle strutture generali solo guardando le proprietà delle operazioni di un semiring scelto, e riferendosi ai teoremi precedenti.

2.3.1 CSP classici

Come è stato già descritto un problema CSP classico è solo un insieme di variabili e vincoli, dove ogni vincolo specifica le tuple che sono permesse per le variabili coinvolte. Si assuma la presenza di un sottoinsieme di variabili distinte, la soluzione di CSP consiste un insieme di tuple che rappresentano gli assegnamenti alle variabili distinte, che può essere esteso ad assegnamenti totali quando soddisfano tutti i vincoli.
Quando i vincoli in CSP sono crisp, cioè essi o permettono una tupla o no, i problemi si possono modellare attraverso un dominio di semiring con solo due valori, indicati con 1 e 0: alle tuple permesse gli sarà associato il valore

1, e quelle non permesse il valore 0.
Inoltre, nei CSP, una combinazione di vincoli è ottenuta attraverso un'operazione di combinazione attraverso l'insieme di tuple. Questo può essere modellato qui assumendo che l'operazione moltiplicativa sia l'*and* logico ($\wedge$ ed interpretando 1 come vero e 0 come falso). Infine, modellare la proiezione su alcune variabili, come k-tuple (assumendo di proiettare k variabili), per cui esiste una consistente estensione a n-tuple, è abbastanza per assumere l'operazione additiva come *or* logico. Perciò, un CSP è solo un SCSP dove il semiring è

$$S_{CSP} = \langle \{0, 1\}, \vee, \wedge, 0, 1 \rangle.$$

Teorema 2.3.1. *$S_{CSP} = \langle \{0, 1\}, \vee, \wedge, 0, 1 \rangle$ è un c-semiring.*

Esempio 2.3.2. *Si consideri il CSP nella figura 2.1, dove le variabili distinte sono marcate e i vincoli sono archi etichettati dalle tuple permesse. La soluzione di questo CSP è un unico insieme contenente le tuple $\langle a, a \rangle$ (ciò significa che $x = a$ e $y = a$).*
Questo CSP può essere visto come un SCSP P'.
Viene mostrato, usando le operazioni del semiring, che la soluzione di P coincide con quella di P'.
Calcolando la soluzione di P', si assegna un valore ad ogni tupla di variabili distinte. Questo è fatto innanzitutto considerando tutte le 3-tuple e assegnando a ciascuna di loro un valore, e poi proiettando (attraverso l'operazione $\vee$) tali valori sulle variabili distinte. Per assegnare un valore a 3-tuple si deve combinare (attraverso $\wedge$) i valori di tutte le sottotuple corrispondenti agli insiemi di variabili che sono connessi dai vincoli.
In figura 2.2 per ogni 3-tuple si possono scrivere i valori delle sue sottotuple: le sottotuple sono indicate da un trattino che le sottolinea, e le sottotuple di un solo elemento non sono indicate poichè tutte hanno valore 1. Allora il valore alla destra può essere visto come combinazione di valori delle sottotuple (attraverso l'$\wedge$ logico), e il valore per ogni 2-tuple a sinistra che è una proiezione (attraverso l'$\vee$ logico) di alcune 3-tuple sulle variabili x e y può essere visto, come combinazione di valori delle due tuple che hanno la stessa proiezione su x e y.
È facile notare che una 3-tupla riceve il valore 1 solo quando soddisfa tutti i vincoli. Dalla figura 2.2, si può notare che solo un gruppo di tuple che riceve il valore 1 è il primo, pertanto la soluzione del CSP è la proiezione di questo gruppo di tuple su x e y, cioè, $x = a$ e $y = a$.

Dall'esempio precedente si può affermare che il c-semiring scelto permette facilmente di rappresentare un dato CSP. Se P è un dato CSP e P' è il cor-

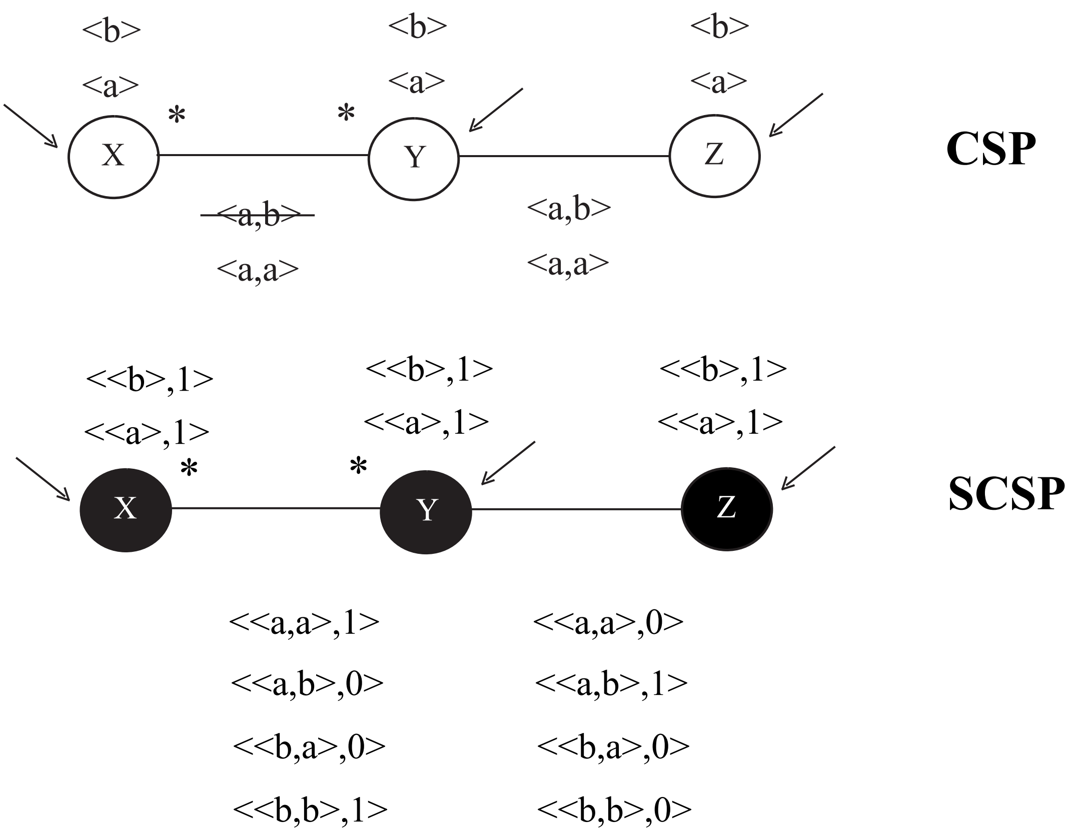


Figura 2.1: un CSP e il suo corrispondente SCSP.

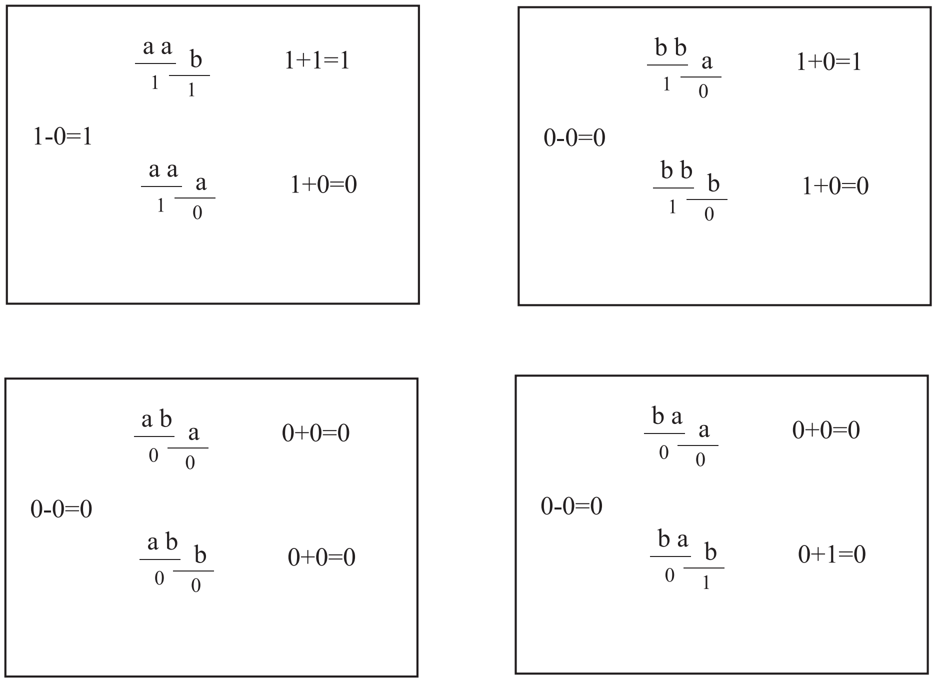


Figura 2.2: combinazione e proiezione nei classici CSP.

rispondente SCSP ottenuto applicando una trasformazione a P, le soluzioni di P e P' hanno lo stesso modulo.

2.3.2 Fuzzy CSP (FCSP)

I fuzzy CSP estendono la nozione di CSP classico permettendo vincoli non-crisp, cioè vincoli che associano un livello di preferenza con ogni tupla di valori. Tale livello è sempre compreso tra 0 e 1, dove 1 rappresenta il miglior valore e 0 il peggior valore.
La soluzione di un fuzzy CSP è allora definita come l'insieme di tuple di valori che hanno il massimo valore. Il modo in cui essi associano un valore con n-tuple è minimizzando il valore delle sue sottotuple: la ragione per cui una struttura max-min fa affidamento al tentativo di massimizzare il valore della minima tupla preferita.
I fuzzy CSP sono una significativa estensione dei CSP in quanto essi sono capaci di modellare la soddisfazione parziale dei vincoli, per ottenere una soluzione persino quando il problema è sovravincolato, e di rendere possibile l'ordinamento in base alla priorità dei vincoli con differenti livelli di importanza.

Definizione 2.3.3. ***(Fuzzy CSP).*** *Un problema di soddisfazione dei vincoli fuzzy (FCSP) è composto da:*

- *un insieme di variabili* $X = \{x_1, \ldots, x_n\}$*,*
- *un insieme finito* D_i *di possibili valori (il suo dominio), per ogni variabile* x_i*,*
- *un insieme di vincoli; ogni vincolo* c *è definito da una funzione di livello fuzzy* $fl(c, A)$ *che assegna un numero reale compreso tra* 0 *e* 1 *ad ogni tupla* A *dei valori del dominio.*

Una soluzione di un FCSP è un assegnamento di valori dal suo dominio ad ogni variabile tale che l'espressione $\min\{fl(c, A) \mid c$ è un vincolo in FCSP$\}$ è massimizzata lungo tutti i possibili assegnamenti ad A dei valori del dominio. Un fuzzy CSP può essere modellato semplicemente scegliendo il seguente semiring:

$$S_{FCSP} = \langle \{x \mid x \in [0, 1]\}, \max, \min, 0, 1\rangle.$$

Teorema 2.3.4. S_{FCSP} *è un c-semiring.*

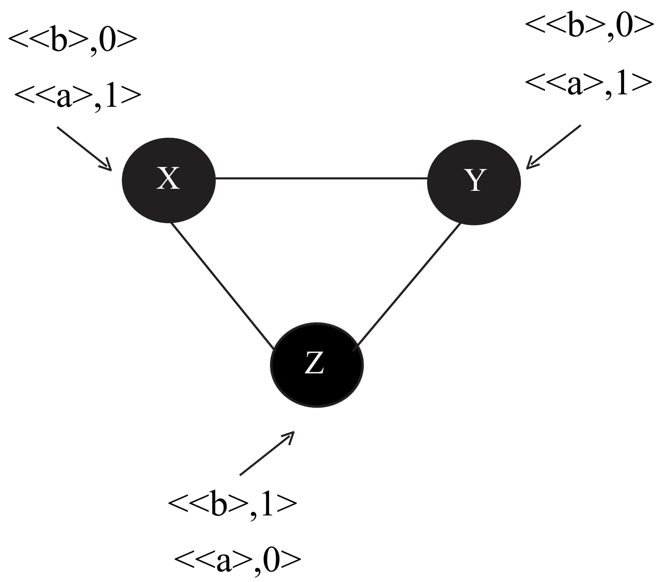
Si noti che ogni FCSP P può essere riscritto come un SCSP P' su il semiring S_{FCSP} tale che $sol(P) = sol(P')$.

Esempio 2.3.5. *Si prenda in esame il fuzzy riportato nella figura 2.3.*
In base al c-semiring definito nel teorema 2.3.4, si utilizzano le operazioni di $+$ *e* $\times$ *per determinare la soluzione del fuzzy considerato.*
Nella prima colonna della tabella 2.1 vengono riportate le tuple di valori

XYZ	*min*($\times$)	*max*($+$)
aaa	0	
aab	1	**soluzione**
aba	0	
abb	0	
baa	0	
bab	0	
bba	0	
bbb	0	

Tabella 2.1: soluzione del FCSP della figura 2.3.

che possono assumere le tre variabili in questione X*,* Y *e* Z*: ad esempio la*



XY	YZ	XZ
<<a,a>,1>	<<a,a>,0>	<<a,a>,0>
<<a,b>,0>	<<a,b>,1>	<<a,b>,1>
<<b,a>,1>	<<b,a>,1>	<<b,a>,0>
<<b,b>,0>	<<b,b>,1>	<<b,b>,0>

Figura 2.3: esempio di un fuzzy CSP.

prima riga della prima colonna sta ad indicare che in questo caso specifico si assume che X, Y e Z assumano il valore a.
Attraverso l'operazione min *si combinano i vincoli e attraverso l'operazione* max *si effettua le proiezione di tali valori ottenuti su tutte le variabili. Con ciò si evidenzia l'unica soluzione del fuzzy CSP in questione, cioè aab.*

2.3.3 Probabilistic CSP (Prob-CSP)

I Prob-CSP furono introdotti per modellare quelle situazioni dove ogni vincolo c ha una certa probabilità $p(c)$, indipendente dalla probabilità di altri vincoli, di far parte di un dato problema.
I livelli di probabilità sui vincoli danno ad ogni istanza, per tutte le variabili, una probabilità che è soluzione di un problema reale. Questo viene fatto, per prima, associando ad ogni sottoinsieme di vincoli la probabilità che esso appartenga al problema reale, e poi, ricapitolando tutte le probabilità dei sottoinsiemi di vincoli dove l'istanza considerata è una soluzione.
In modo alternativo, la probabilità associata alle n-tuple t può essere vista come la probabilità che tutti i vincoli che t viola sono veramente nel problema reale. Questo corrisponde al prodotto di tutti i fattori $1 - p(c)$ per ogni c violato da t. Infine, lo scopo è ottenere quelle istanze che massimizzano la probabilità.

Definizione 2.3.6. ***(Prob-CSP).*** *Un Prob-CSP comprende:*

– *un insieme di variabili $X = \{x_1, \ldots, x_n\}$,*

– *un insieme finito D_i di possibili valori (il suo dominio), per ogni variabile x_i.*

– *un insieme di vincoli ristretti ai valori che le variabili possono simultaneamente assumere; ad ogni vincolo c viene assegnata una certa probabilità $p(c)$ di appartenere al problema dato.*

Ci sono due approcci alternativi per definire la soluzione di un Prob-CSP. La soluzione è un assegnamento di valori dal suo dominio ad ogni variabile tale che

– l'espressione $Sum\{Product\{p(c) \mid c \;\; un\ vincolo\ in\ S\} \mid S \;\; unsottoinsieme\ dell'insieme\ di\ vincoli\ soddisfatti\ da\ A\}$ è massimizzata lungo tutti i possibili assegnamenti ad A di valori, o

– l'espressione $Product\{(1 - p(c)) \mid c\ un\ vincolo\ in\ FCSP\ violato\ da\ A\}$ è massimizzata lungo tutti i possibili assegnamenti A di valori.

La relazione tra Prob-CSP e SCSP è complicata dal fatto che i Prob-CSP contengono vincoli crisp con livelli di probabilità, mentre SCSP contengono vincoli non-crisp, cioè si associano valori alle tuple, e non ai vincoli.
Comunque è possibile modellare Prob-CSP, usando una trasformazione simile a quella proposta per modellare vincoli con priorità attraverso vincoli soft nella struttura FCSP. Più precisamente si assegna la probabilità alle tuple invece che ai vincoli. Si consideri ogni vincolo c con probabilità $p(c)$, e si prenda t che rappresenta ogni tupla di valori per le variabili coinvolte da c. Allora $p(t) = 1$ se t è permessa da c, altrimenti $p(t) = 1 - p(c)$.
Le ragioni per tale scelta sono le seguenti: se una tupla t è permessa da c e c è nel problema reale, allora t è permessa nel problema reale; questo accade con probabilità $p(c)$; se invece c non è nel problema reale, allora t è ancora permessa nel problema reale, questo accade con probabilità $1-p(c)$. Pertanto t è permessa con probabilità $p(c) + 1 - p(c) = 1$. Si consideri invece ora una tupla che non è permessa da c. Allora essa può essere permessa nel problema reale solo se c non è presente; questo accade con probabilità $1 - p(c)$.
Per assegnare il valore appropriato ad una n-tupla t, dati i valori di tutte le più piccole k-tuple, con $k \leq n$ e che sono sottotuple di t, si può solo realizzare il prodotto del valore di ogni sottotupla. Nella maniera in cui i valori sono stati assegnati alle tuple, questo coincide con il prodotto di tutte $1-p(c)$ per tutti c che sono stati violati da t. Infatti, se una sottotupla viola c, allora la costruzione del suo valore è $1 - p(c)$; se invece una sottotupla soddisfa c, allora il suo valore è 1. Per il fatto che 1 è l'elemento unità del $\times$, si ha che $1 \times a = a$ per ogni a. Pertanto si ha $\prod(1 - p(c))$ per tutte le c che t viola.
Come risultato, il semiring corrispondente al Prob-CSP è

$$S_{prob} = \langle \{x \mid x \in [0, 1]\}, \max, \times, 0, 1 \rangle$$

Teorema 2.3.7. *S_{prob} è un c-semiring.*

È facile notare come ogni Prob-CSP P può essere scritto come un n SCSP P' sul semiring S_{prob}, tale che $sol(P) = sol(P')$. Si noti che il fatto che P' sia α-consistente significa che in P esiste un n-tupla che ha probabilità α che è una soluzione del problema reale.

Esempio 2.3.8. *Si prenda in considerazione l'esempio del Prob-CSP rappresentato nella figura 2.4.*
In base al c-semiring definito nel teorema 2.3.7, si utilizzano le operazioni di $+$ e $\times$ per determinare le soluzioni del Prob-CSP. In questo caso l'operazione $\times$ viene utilizzata per combinare i diversi vincoli e l'operazione max *viene utilizzata per proiettare le soluzioni sulle variabili. In questo esempio risulta essere unica la soluzione* bab *(vedi tabella 2.2).*

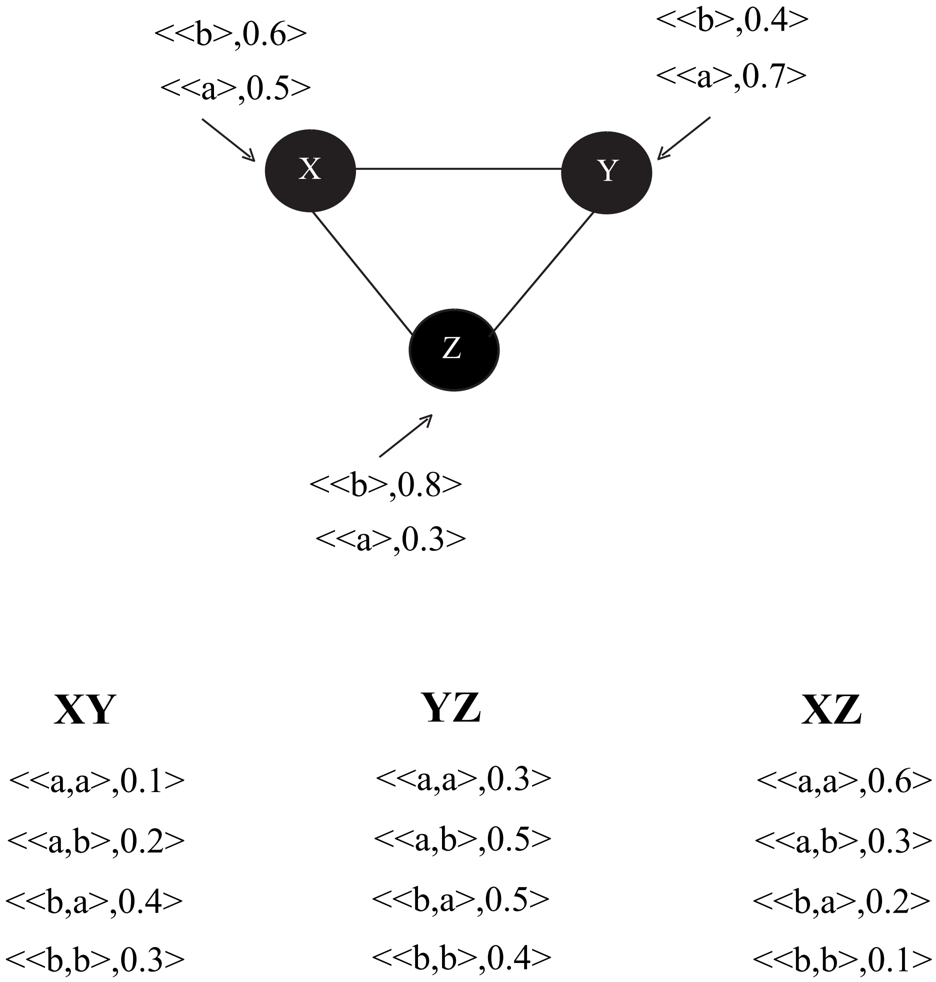


Figura 2.4: esempio di un Prob-CSP.

XYZ	$\times(\times)$	$max(+)$
aaa	0.00189	
aab	0.0042	
aba	0.0036	
abb	0.00384	
baa	0.003024	
bab	0.00672	**soluzione**
bba	0.00216	
bbb	0.002304	

Tabella 2.2: soluzione del Prob-CSP della figura 2.4.

2.3.4 Weighted CSP

Mentre per i fuzzy CSP si associa un livello di preferenza ad ogni tuple in ogni vincolo, nei weighted CSP alle tuple vengono associati dei costi.
Ciò permette di modellare problemi di ottimizzazione dove il fine è di minimizzare il costo totale delle soluzioni proposte. Perciò, nei WCSP la funzione costo è definita ricapitolando tutti i costi di tutti i vincoli. Il fine, quindi, è quello di trovare le n-tuple che minimizzano la somma totale dei costi delle loro sottotuple.
Un altro modo per comprendere la differenza tra WCSP e FCSP è che FCSP ha un approccio egualitario per ottimizzare i problemi (cioè, per tutti i vincoli), mentre WCSP ha un approccio utilitario (cioè, essi hanno lo scopo di ricevere il minimo costo globalmente, come se alcuni vincoli possano essere trascurati e perciò presentano un grande costo).
Secondo la descrizione informale appena data dei WCSP, il semiring associato è il seguente:

$$S_{WCSP} = \langle \mathcal{R}^+, \min, +, +\infty, 0 \rangle$$

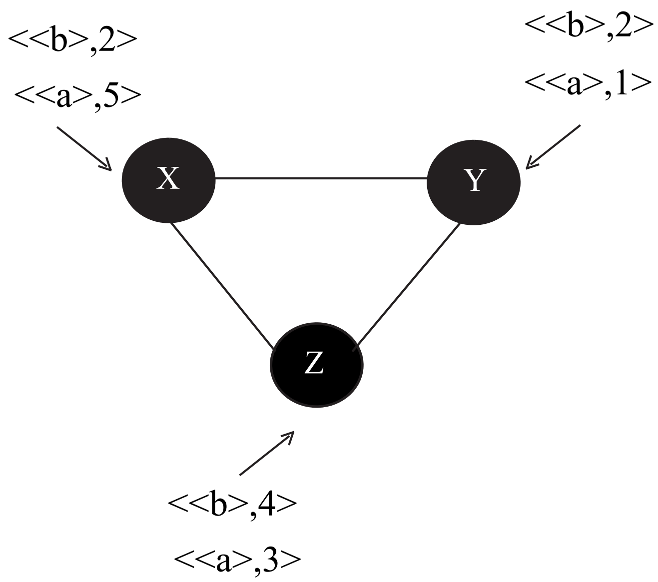
Teorema 2.3.9. *S_{WCSP} è un c-semiring*

Esempio 2.3.10. *Si prenda in considerazione l'esempio del WCSP rappresentato nella figura 2.5.*
In base al c-semiring definito nel teorema 2.3.9, si utilizzano le operazioni di $+$ e $\times$ per determinare la soluzione del WCSP riportato nella figura 2.5.

XYZ	$+(\times)$	$min(+)$
aaa	18	
aab	16	**soluzione**
aba	18	
abb	16	**soluzione**
baa	16	**soluzione**
bab	18	
bba	16	**soluzione**
bbb	18	

Tabella 2.3: soluzione del WCSP della figura 2.5.

In questo caso è l'operazione $+$ che viene utilizzata per combinare i diversi vincoli ed è l'operazione max *che viene utilizzata per proiettare le soluzioni. In questo esempio le soluzioni trovate sono le seguenti:* aab*,* abb*,* baa *e* bba *(vedi tabella 2.3).*



XY	YZ	XZ
<<a,a>,1>	<<a,a>,4>	<<a,a>,4>
<<a,b>,2>	<<a,b>,3>	<<a,b>,2>
<<b,a>,3>	<<b,a>,2>	<<b,a>,3>
<<b,b>,4>	<<b,b>,1>	<<b,b>,5>

Figura 2.5: esempio di un WCSP.

2.3.5 Egualitarismo e utilitarismo

Come già accennato, i sistemi FCSP e WCSP possono essere visti con due differenti approcci per definire un significato alla nozione di ottimizzazione. I due modelli corrispondono a due differenti definizioni di assistenza sociale nella teoria dell'utilità:

– *egualitarismo*, che corrisponde alla massimizzazione della minima utilità individuale;

– *utilitarismo*, che massimizza la somma dell'utilità individuali.

I FCSP sono basati sull'approccio egualitario mentre i WCSP sono basati sull'approccio utilitario.
Si crede che entrambi gli approcci presentino vantaggi e svantaggi, e pertanto la preferenza di uno rispetto ad un altro dipende dalla possibilità di modellare situazioni della vita reale.

2.4 Semiring con operatore divisione

Tra le possibili soluzioni tesi a garantire strutture rilevanti, il *symmetrisation* incastra il semiring in una grande struttura contenente un inverso per ogni elemento. Questa struttura è utile quando potrebbe essere necessario l'utilizzo dell'operatore divisione.
Questo paragrafo introduce la nozione di invertibilità per gli absorptive semiring, ossia una tipologia di semiring che per il rispetto di alcune proprietà supporta appunto la nozione di invertibilità.
Innanzitutto si definisce il concetto di absorptive semiring [5].

Definizione 2.4.1. ***(absorptive semiring).*** *Si consideri un commutative semiring (un semiring in cui entrambi gli operatori, additivo e moltiplicativo, godono della proprietà commutativa) $\mathcal{K}$. Allora $\mathcal{K}$ è un* absorptive semiring *se gode delle seguenti proprietà:*

– *absorptiveness:* $\forall\ a, b \in A.\ a + (a \times b) = a$;

– *top element:* $\forall\ a \in A.\ a + \mathbf{1} = \mathbf{1}$.

È possibile effettuare un collegamento tra la proprietà absorptiveness e la nozione di top element.

Definizione 2.4.2. ***(absorptive semiring).*** *Si consideri un commmutative semiring $\mathcal{K}$. Se $(\mathcal{K})$ è absorptive allora l'operatore somma è idempotente.*

Proposizione 2.4.3. ***(tropical semiring).*** *Si consideri un commmutative semiring $\mathcal{K}$. Allora* ($\mathcal{K}$) *è* tropical semiring *l'operatore somma è idempotente.*

Si noti che gli absorptive semiring sono proprio i tropical semiring con l'elemento unità.
In base alla definizione di c-semiring, si può equiparare tale definizione a quella dei tropical semiring con top element ed inoltre ciò suggerisce attraverso la proposizione 2.4.2 la proprietà implicita dell'absorptiveness.
La soluzione che si insegue è basata sulla residuation theory, che permette di ottenere uno operatore di divisione attraverso l'approssimazione dell'equazione $b \times x = a$. Si noti che usando gli operatori di divisione appena definiti, gli opportuni algoritmi di consistenza sono progettati anche per l'operatore $\times$ non idempotente.

Definizione 2.4.4. ***(invertible absorptive semiring).*** *Si consideri un absorptive semiring $\mathcal{K}$. Allora:*

- $\mathcal{K}$ *è* invertibile *se esiste un elemento $c \in A$ tale che $b \times c = a \ \forall\ a, b \in A$ dove $a \leq b$;*

- *è* weakly uniquely *invertibile se c è unico quando $a < b$;*

- *è* uniquely *invertible se c è unico se $b \neq \mathbf{0}$.*

Come si può facilmente constatare la definizione 2.4.4 non richiede l'esistenza per ogni elemento a di un inverso, cioè di un elemento a^{-1} che verifica $a \times a^{-1} = \mathbf{1}$. Infatti la condizione di absorptiveness per un absorptive semiring, dove $\forall\ a, b \in A.a + (a \times b) = a$ quindi $a \times b \leq a$, garantisce che nessun elemento, eccetto $\mathbf{1}$, ha un inverso.
Di seguito verranno mostrati due risultati che stabiliscono condizioni per assicurare l'invertibilità di un absorptive semiring.
La prima proposizione riguarda i moltiplicative idempotent semiring.

Proposizione 2.4.5. ***(absorptive moltiplicative idempotent semiring).*** *Dato un absorptive, multiplicative idempotent semiring $\mathcal{K}$. Allora, $\mathcal{K}$ è invertibile. Inoltre se $\leq$ è un ordinamento totale, allora $\mathcal{K}$ è weakly uniquely invertible.*

La seconda proposizione è per i *cancellative* semiring, cioè i semiring in cui viene rispettata la seguente proprietà: $\forall\ a, b, c \in A.(a \times c = b \times c) \wedge (c \neq \mathbf{0}) \Rightarrow a = b$.

Proposizione 2.4.6. *Dato un absorptive e invertible semiring $\mathcal{K}$. Allora $\mathcal{K}$ è uniquely invertible se e solo se esso è cancellative.*

Inoltre si noti che nei tropical semiring la disuguaglianza $b \times x \leq a$ ha sempre una soluzione.

Definizione 2.4.7. ***(residuated semiring)****. Dato un tropical semiring $\mathcal{K}$. Allora $\mathcal{K}$ è residuated se l'insieme $\{x \in A \mid b \times x \leq a\}$ ammette un massimo per tutti gli elementi $a, b \in A$, indicato con $a \div b$.*

La suddetta definizione in verità suggerisce un algoritmo euristico per approssimare l'elemento massimo, intuitivamente dato dalla somma di tutti gli elementi che soddisfano la disuguaglianza. Il punto chiave è che l'insieme delle sotto-soluzioni di un'equazione contiene le possibili soluzioni, quando esistono, e in questo caso anche l'elemento massimo è anche una soluzione. Diverse proprietà reggono se il semiring è absorptive, tale che $b \leq a \Rightarrow a \div b = \mathbf{1}$. Questo fatto permette la nozione di invertibilità.

Definizione 2.4.8. *Si consideri un absorptive e invertible semiring $\mathcal{K}$. Allora $\mathcal{K}$ è invertible attraverso residuation se l'insieme $\{x \in A \mid b \times x = a\}$ ammette un massimo per tutti gli elementi $a, b \in A$ tale che $a \leq b$.*

Con un abuso di notazione, il massimo elemento tra le soluzioni è indicato come $a \div b$.
In base alle definizioni date facilmente si può affermare che:

- il classico CSP, per il fatto che $\times$ è idempotente, è un semiring invertible (rientra nella definizione 2.4.5);
- il fuzzy CSP è un semiring invertible per la stessa motivazione del classico CSP;
- il weigthed CSP, seppure il $\times$ non è idempotente, è un cancellative semiring e quindi è uniquely invertible;
- il probabilistic CSP (come per il weighted CSP) seppure il $\times$ non è idempotente, essendo un cancellative semiring è uniquely invertible.

2.5 C-semiring n-dimensionali

Scegliere un'istanza del SCSP significa specificare un particolare c-semiring. Questo, come già enunciato, induce ad un parziale ordinamento, che può essere interpretato come linea guida per la scelta della migliore tra le differenti soluzioni.
Nelle situazioni della vita reale, una linea guida non è abbastanza, per esempio potrebbe essere necessario scegliere le soluzioni che permettono di raggiungere un buon compromesso tra tutti gli scopi.

Si consideri l'esempio di una rete di computer, dove si vorrebbe sia minimizzare il tempo totale che massimizzare il lavoro dei computer meno usati. Allora in questo caso si ha la necessità di considerare due c-semiring, uno per la minimizzazione dei costi e uno per la massimizzazione del lavoro.
Pertanto uno potrebbe lavorare con uno di questi c-semiring e poi con l'altro provando a combinare le soluzioni che sono migliori per entrambi.
Un approccio più semplice, comunque consiste nella combinazione dei due c-semiring e lavorare solo con la struttura risultante. Più precisamente il modo per combinare diversi c-semiring e ottenere un altro c-semiring consiste nel vettorizzare i domini e le operazioni di combinazione dei c-semiring.

Definizione 2.5.1. ***(composizione dei c-semiring).*** *Dati n c-semiring $S_i = \langle A_i, +_i, \times_i, \boldsymbol{0}_i, \boldsymbol{1}_i \rangle$ per $i = 1, \ldots, n$ pertanto si definisce la struttura composizione come segue: $Comp(S_1, \ldots, S_n) = \langle \langle A_1, \ldots, A_n, +, \times, \langle \boldsymbol{0}_1, \ldots, \boldsymbol{0}_n \rangle$, $\langle \boldsymbol{1}_1, \ldots, \boldsymbol{1}_n \rangle \rangle$. Dati $\langle a_1, \ldots, a_n \rangle$ e $\langle b_1, \ldots, b_n \rangle$ tale che con $a_i, b_i \in A_i$, per $i = 1, \ldots, n$, $\langle a_1, \ldots, a_n \rangle + \langle b_1, \ldots, b_n \rangle = \langle a_1 +_1 b_1, \ldots, a_n +_n b_n \rangle$ e $\langle a_1, \ \ldots, \ a_n \rangle \times \langle b_1, \ldots, b_n \rangle = \langle a_1 \times_1 b_1, \ldots, a_n \times_n b_n \rangle$.*

Teorema 2.5.2. ***(una composizione di c-semiring è un c-semiring).*** *Dati n semiring $S_i = \langle A_i, +_i, \times_i, \boldsymbol{0}_i, \boldsymbol{1}_i \rangle$, per $i = 1, \ldots, n$, la struttura $Comp(S_1, \ldots, S_n)$ è un c-semiring.*

Dati $\langle a_1, \ldots, a_n \rangle$ e $\langle b_1, \ldots, b_n \rangle$, tale che con $a_i, b_i \in A_i$ per $i = 1, \ldots, n$, si ha che $\langle a_1, \ldots, a_n \rangle \leq_S \langle b_1, \ldots, b_n \rangle$ se e solo se $\langle a_1 +_1 b_1, \ldots, a_n +_n b_b \rangle = \langle b_1, \ldots, b_n \rangle$. Ciò significa che tra le soluzioni di un problema di ordinamento, tale che un semiring contiene un incomparabile insieme di tuple, nessuna di queste ha un $blevel(P)$ come valore associato.
Comunque se si vuole ridurre il numero delle "migliori" tuple (o ottenere soltanto una), uno deve specificare alcune priorità attraverso le quali ordinare le componenti dei c-semiring.

Capitolo 3

Giochi non cooperativi

3.1 Teoria dei giochi

3.1.1 Cenni storici

La Teoria dei giochi [10] è la scienza matematica che analizza situazioni di conflitto e ricerca soluzioni competitive e cooperative tramite modelli ovvero corrisponde allo studio delle decisioni individuali in situazioni in cui vi sono interazioni tra i diversi soggetti, tali per cui le decisioni di un soggetto possono influire sui risultati conseguibili da parte di un rivale, secondo un meccanismo di retroazione.
La storia della Teoria dei giochi [3] è una storia recente.
Convenzionalmente essa inizia nel 1913 col teorema di Zermelo, che dimostra che i giochi (nel senso comune del termine) come gli scacchi, che hanno regole certe e informazione completa sulle mosse effettuate, hanno anche una strategia ottimale.
Giochi di questo tipo sono a somma zero, nel senso che ciò che uno vince - la partita a scacchi, per esempio - l'altro perde, e fino agli anni $'50$ la teoria si occupa solo di questo caso, ovviamente di importanza relativamente limitata.
Soltanto nel 1944 compare il primo trattato sistematico [13] che introduce la teoria dell'utilità attesa.
L'estensione della teoria ai giochi non a somma zero è dovuta in gran parte a John Nash. Nel 1950 egli introdusse il concetto di equilibrio di contrattazione, e nel 1951 quello di equilibrio di Nash, che è come si vedrà in seguito la "soluzione tipica" di un gioco statico ad informazione completa. Ancora studente a Princeton, Nash spiega la sua idea di fondere intimamente due concetti apparentemente assai lontani: quella di un punto fisso (equilibrio) in una trasformazione di coordinate, e quella della strategia più razionale che un giocatore può adottare, quando compete con un avversario anch'esso

razionale, estendendo la teoria dei giochi ad un numero arbitrario di partecipanti, o agenti. Egli dimostra che, sotto certe condizioni, esiste sempre una situazione di equilibrio, che si ottiene quando ciascun individuo che partecipa a un dato gioco sceglie la sua mossa strategica in modo da massimizzare la sua funzione di retribuzione, sotto la congettura che il comportamento dei rivali non varierà a motivo della sua scelta (vuol dire che anche conoscendo la mossa dell'avversario, il giocatore non farebbe una mossa diversa da quella che ha deciso).
Alla fine degli anni ′50, Luce e Raiffa [9] fornirono una nuova fondamentale trattazione sistematica, e Schelling [12] la usò per studiare appunto situazioni tipiche di conflitto. E a questo punto, misteriosamente, la teoria dei giochi perse di interesse fino più o meno alla fine degli anni '70.
Attualmente invece è in rapido sviluppo sia per quanto riguarda la teoria stessa che per quanto riguarda le sue applicazioni, ed è considerata fondamentale per tutte quelle discipline che riguardano le scelte individuali, come la sociologia, la psicologia e la biologia, oltre naturalmente che per l'economia.

Definizione 3.1.1. ***(teoria dei giochi).*** *La Teoria dei Giochi tratta le situazioni in cui il risultato dipende dalle scelte fatte da più persone, dette* giocatori*, che operano perseguendo determinati* obiettivi.

Gli elementi costitutivi di un gioco sono:

– i *giocatori* che si assume abbiano le seguenti caratteristiche, siano:

  - intelligenti, cioè significa che capiscano la situazione in cui si trovino e siano in grado di fare ragionamenti logici corretti;
  - razionali, cioè significa che abbiano preferenze coerenti sugli esiti finali del processo decisionale e che abbiano l'obiettivo di massimizzare queste preferenze;

– le *strategie*, cioè i piani di azione di un giocatore;

– le *azioni* a disposizione di ciascun giocatore;

– le *regole del gioco*, cioè le diverse caratteristiche che differenziano tra di loro i diversi tipi di giochi (caratteristiche dell'informazione, della ripetitività delle mosse, etc);

– gli *esiti* del gioco o meglio le vincite dei giocatori (payoff) in seguito alle azioni intraprese da tutti i giocatori;

– il *ragionamento strategico*, cioè l'elaborazione di congetture su strategie altrui al fine di elaborare miglior piano di azione per sè;

– gli *obiettivi* che possono risultare:

- comuni ma non identici;
- differenti ed eventualmente contrastanti;
- alle volte anche aleatori.

Ogni individuo ha una sua funzione di utilità sull'insieme dei beni. Per esempio, se l'insieme dei beni è una fetta di torta, un libro, una vacanza, un diploma, ognuno è in grado di quantificare numericamente la sua utilità per ciascuno dei beni.
Non è detto che le utilità di due persone distinte per lo stesso bene siano le stesse. La razionalità richiesta ai giocatori impone che valga la *proprietà transitiva nelle preferenze*: se il diploma è preferito alla vacanza e la vacanza al libro allora il diploma deve essere preferito al libro.

3.1.2 Classificazione dei giochi

In questa sezione verranno trattate alcune classificazione di particolare interesse ai fini di questo lavoro.

La prima suddivisione si può fare seguendo la classificazione di Harsanyi (1966) che distingue due classi di giochi:

– *Giochi non cooperativi*: non sono possibili accordi vincolanti tra i giocatori e quindi giochi di questo tipo regolano i meccanismi di decisione dei singoli, sulla base di ragionamenti individuali, senza alleanze fra individui. Attualmente si preferisce assumere, più restrittivamente, che in un gioco non cooperativo i giocatori non possano nemmeno comunicare in quanto ciò potrebbe alterare il risultato. Si deve a Nash l'idea di analizzare i giochi non cooperativi.

– *Giochi cooperativi*: sono possibili accordi vincolanti tra i giocatori affinchè questi possono essere di vantaggio a singoli componenti. Si deve, invece, soprattutto a Von Neumann l'idea di analizzare i giochi studiando il nascere delle coalizioni fra individui. I giochi cooperativi sono suddivisibili in due ulteriori sottoclassi:

  - *Giochi cooperativi a utilità non trasferibile (NTU) o senza pagamenti laterali*: in cui i giocatori ricevono un payoff preassegnato in base alle ripartizioni del gioco fissate inizialmente dagli stessi giocatori facenti parte della coalizione;

- *Giochi cooperativi a utilità trasferibile (TU) o a pagamenti laterali*: i giocatori di una coalizione possono ripartirsi in qualsiasi modo la vincita. Questi giochi costituiscono un caso particolare dei giochi cooperativi a utilità non trasferibile. In particolare per avere un gioco TU devono essere soddisfatte tre ipotesi:
  1. deve essere possibile (da un punto di vista normativo) trasferire l'utilità tra i giocatori;
  2. deve esistere un mezzo comune di scambio, ad esempio il denaro, con cui trasferire (da un punto di vista materiale) l'utilità;
  3. le funzioni di utilità dei giocatori devono essere equivalenti, ad esempio funzioni lineari della quantità di denaro.

È importante osservare da subito che questo non significa che nei giochi cooperativi si aspettano atteggiamenti più altruistici da parte dei giocatori: questo non si assume mai, perchè l'ipotesi base è che ognuno persegue il suo interesse (se poi l'interesse personale consiste nel fare del bene al prossimo, ciò è correlato al grado di soddisfazione personale nel fare una cosa piuttosto che un'altra).

Un'altra suddivisione possibile fra giochi riguarda la loro rappresentazione in forma matematica. Si parla allora:

– *Giochi in forma estesa o con memoria*: la descrizione del gioco è sotto forma di albero; si tratta di costruire un grafo che, partendo dalla radice, descrive il gioco mossa per mossa, fino ad arrivare a presentare tutte le situazioni finali, conseguenti ad una data serie di mosse. Ad ogni nodo si associa una possibile situazione del gioco, agli archi uscenti da ciascun nodo si associano le possibili mosse del giocatore che è chiamato a muovere in quella situazione e ai nodi terminali si associano i valori delle vincite (payoff) di ciascun giocatore. Il tutto fornisce un modo estremamente efficace ed esauriente per analizzare un gioco, ma ha il difetto di essere complicato.

Definizione formale di un gioco in forma estesa [11]

Un gioco finito in forma estesa è una n-pla $(V, D, r, N, \mathcal{P}, \mathcal{Q}, \mathcal{M}, E, h, (\succeq_k)_{k\in N})$, dove:

- (V, D, r) è un albero finito (V, D) con radice r. Il simbolo V denota l'insieme dei nodi (o vertici) dell'albero, mentre D indica i suoi lati (o rami). Indicheremo con Z l'insieme dei nodi (o vertici) finali dell'albero; pertanto $V \setminus Z$ saranno i nodi non terminali dell'albero. [INTERPRETAZIONE: i nodi non terminali, cioè gli elementi di $V \setminus Z$, rappresentano le varie situazioni in cui il giocatore (o la sorte) può (e deve) effettuare una scelta. Potrà essere utile distinguere tra situazioni che differiscono tra loro anche solo per via di una storia pregressa: l'esempio classico è dato dal gioco degli scacchi, in cui posizioni identiche sulla scacchiera possono derivare da diverse serie di mosse precedenti, nel qual caso saranno identificate con nodi divesi.]

- N è un insieme finito. Per evitare conflitti con quanto si dirà in seguito, si proibirà che $0 \in N$. [INTERPRETAZIONE: N rappresenta l'insieme dei giocatori. Farà comodo introdurre anche "un giocatore" speciale, che sarà indicato con il nome "sorte" e che sarà identificato col simbolo 0 ("zero"), per rappresentare situazioni in cui la "decisione", ovvero la scelta tra i diversi rami uscenti da un nodo, non è frutto della deliberazione di un giocatore ma è determinata da un fattore aleatorio "esterno".]

- $\mathcal{P} = (P_k)_{k \in N^*}$ è una suddivisione di $V \setminus Z$ in sottoinsiemi disgiunti, tanti quanti sono gli elementi di $N^* = N \cup \{0\}$. Con "suddivisione" si intende dire che $\cup_{k \in N^*} P_k = V \setminus Z$. Si noti che si ammette anche che alcuni di questi sottoinsiemi siano vuoti (ad esempio, P_0 potrebbe essere vuoto), per cui non si usa il termine "partizione" per indicare $\mathcal{P}$, in quanto il termine "partizione" è riservato al caso in cui un insieme venga suddiviso in insieme disgiunti, ognuno dei quali sia non vuoto. [INTERPRETAZIONE: i nodi di P_k, per $k \in N$, rappresentano i nodi di "pertinenza" del giocatore k, ovvero quei nodi in cui il giocatore k sarà chiamato a scegliere, se a tale nodo porterà lo svolgimento del gioco. I nodi di P_0 rappresentano invece le circostanze in cui si ha una "scelta" del caso (per esempio, vengono mescolate le carte, vengono lanciati i dadi, etc...).]

- $\mathcal{Q} = (Q_{k,j})_{k \in N^*, j \in J_k}$ rappresenta, per ogni k, una partizione di P_k in una famiglia di insiemi $Q_{k,j}$ (l'insieme J_k è semplicemente un insieme di indici). Gli insiemi di Q_{0j} sono tutti dei singleton. Si impone il vincolo che nessuno degli insiemi $Q_{k,j}$ possa contenere due vertici distinti appartenenti ad un cammino elementare che unisce la radice ad uno dei vertici terminali. [INTERPRETAZIONE: ogni $Q_{k,j}$ rappresenta un insieme di nodi, tutti pertinenti al giocatore k, caratterizzati dalla proprietà che, quando il giocatore k si trova in un nodo di $Q_{k,j}$, non sa in quale dei nodi di $Q_{k,j}$ si trova.

Per le scelte della "sorte", questa complicazione non è necessaria e pertanto richiediamo che gli insiemi $Q_{0,j}$ siano dei singleton.]

- $\mathcal{M}$ è, per $k \neq 0$, una famiglia di insiemi $M_{k,j}$, uno per ciascuno dei $Q_{k,j}$. Assumiamo che, per ogni nodo di $Q_{k,j}$, sia data da una corrispondenza biunivoca tra $M_{k,j}$ e l'insieme dei rami uscenti da tale nodo. Per ciascun $Q_{0,j}$ (se ve ne sono), è individuata un'applicazione che ad ogni ramo uscente da $Q_{0,j}$ associa un numero reale non negativo, in modo che la somma di tutti questi numeri, al variare di tutti i rami uscenti da $Q_{0,j}$, valga 1. [INTERPRETAZIONE: l'idea è che, in corrispondenza di un nodo di un dato insieme di informazione $Q_{k,j}$, il giocatore k debba fare una scelta da un certo insieme di alternative, che è dato da $M_{k,j}$. Tale insieme di scelte possibili deve essere lo stesso in ogni nodo dell'insieme di informazioni $Q_{k,j}$ se si vuole salvare la coerenza interpretativa con quanto detto prima. Naturalmente, scegliere un elemento di $M_{k,j}$ in un nodo o in un altro di $Q_{k,j}$ può avere effetti molto diversi. Per quanto riguarda i nodi della sorte, l'interpretazione dovrebbe essere evidente: la scelta tra i vari rami uscenti viene effettuata con estrazione a sorte con le probabilità assegnate ai rami.]
- E è un insieme. [INTERPRETAZIONE: si tratta dei diversi possibili esiti finali del gioco. Si noti che un esito finale potrebbe incorporare, nella sua descrizione, anche la storia di quanto è avvenuto nel gioco: in questo modo si può tenere conto di eventuali esiti "intermedi" (si pensi ai punteggi parziali delle varie "mani" di una partita a carte).]
- $h : Z \to E$ [INTERPRETAZIONE: si associa ad ogni vertice finale dell'albero un "esito". Si noti, in riferimento a quanto detto al punto precedente, che un vertice finale individua in modo univoco la storia pregressa, ovvero ciò che è avvenuto dall'inizio del gioco. Questo fatto ci permette di costruire h, se necessario, in modo che l'esito che si associa ad un vertice finale tenga conto, per l'appunto di tutta la storia del gioco.]
- $(\succeq_k)_{k \in N}$ è una famiglia di preordini totali su E [INTERPRETAZIONE: si tratta, chiaramente, delle preferenze dei giocatori rispetto agli esiti finali del gioco.]

Volendo si possono utilizzare funzioni di utilità per rappresentare le preferenze dei giocatori. In tal caso il gioco in forma estesa verrà descritto come: $(V, D, r, N, \mathcal{P}, \mathcal{Q}, \mathcal{M}, E, h, (u_k)_{k \in N})$.

Esempio 3.1.2. *Nella figura 3.1 si ha un albero con* 7 *nodi o rami.*

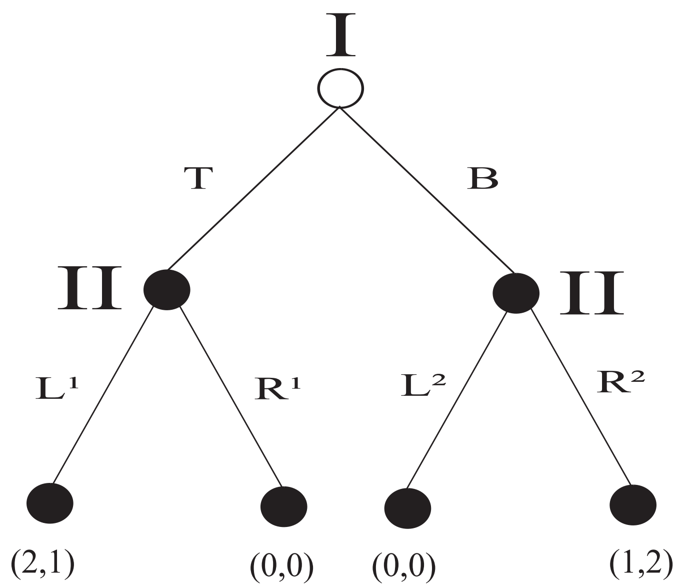


Figura 3.1: esempio forma estesa.

Si userà la figura 3.2 come fare riferimento alla struttura d'albero sottostante il gioco.
La radice r *è evidenziata graficamente per il fatto che è il nodo indicato con il pallino "vuoto", mentre tutti gli altri nodi sono rappresentati da un pallino "pieno".*
L'insieme N *dei giocatori è* $\{I, II\}$*. La partizione dei nodi (non terminali)* $\mathcal{P}$ *è individuata graficamente dal fatto che ad ogni nodo non terminale è posta vicino l'etichetta* I *o* II*: quindi* P_I *contiene solo la radice, mentre* P_{II} *contiene gli altri due vertici non terminali* $v1$ *e* $v2$*.*
In questo esempio ognuno degli insiemi di informazione è un singleton. Vi è quindi un solo insieme di informazione per I *(che contiene la radice* r*) e vi sono due insiemi di informazione per* II *(uno contiene il vertice* $v1$ *e l'altro per* $v2$*).*
Si ha quindi $J_I = \{1\}$ *e* $Q_{I,1} = \{r\}$*, e* $J_{II} = \{1, 2\}$ *e* $Q_{II,1} = \{v1\}$*,* $Q_{II,2} = \{v2\}$*.*
Inoltre $M_{I,1} = \{T, B\}; M_{II,1} = \{L_1, R_1\}$ *ed* $M_{II,2} = \{L_2, R_2\}$*. La corrispondenza biunivoca tra ciascuno di questi insiemi ed i rami "uscenti" dal corrispondente insieme di informazione è indicata graficamente semplicemente ponendo i nomi degli elementi di questi insiemi vicino ai*

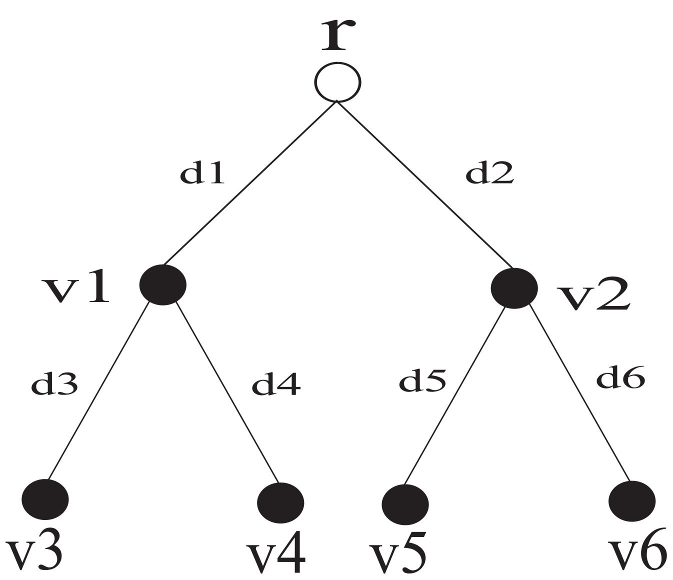


Figura 3.2: struttura forma estesa.

rami " corrispondenti ". Si può affermare che (ad esempio, per quanto riguarda $M_{I,1}$) a T è assegnato il ramo d_1, a B il ramo d_2.
Infine si è posto ai nodi finali una coppia ordinata di numeri reali che rappresentano i valori che le funzioni di utilità assegnano all'esito associato a tale nodo finale. Si noti che nella descrizione grafica è completamente ignorato quale sia l'esito finale. Dopotutto, ciò che interessa sono le preferenze dei giocatori rispetto a quegli esiti finali, non cosa essi siano.

Questa forma di rappresentazione è stata introdotta da von Neumann (1928) e formalizzata da Kuhn (1953). Normalmente un gioco in forma estesa si risolve attraverso l'*induzione a ritroso* in quanto partendo dalla fine del gioco si cerca di determinare una soluzione del gioco.

- *Giochi in forma strategica o normale o senza memoria*: di solito precisa il numero dei giocatori, lo spazio delle loro strategie, e la funzione di utilità di ciascuno di essi: una funzione a valori reali, definita sul prodotto cartesiano degli spazi di strategie.

Definizione 3.1.3. ***(Gioco non cooperativo in forma strategica).*** *Un gioco non cooperativo in forma strategica viene indicato attraverso una tripla $\langle N, S, U \rangle$ dove:*

- *$N = \{1, \ldots, n\}$ è l'insieme dei giocatori;*
- *$S = \{S_1, \ldots, S_k\}$ è l'insieme delle strategie possibili;*
- *$U = \{u_1, \ldots, u_n\}$ è l'insieme delle funzioni payoff associati a ciascun giocatore con $u_i : S^k \to R$.*

Una possibile interpretazione è che tutti i giocatori scelgono contemporaneamente la loro strategia e la u_i dice quale è il guadagno del giocatore i determinato dalle scelte fatte.
Gli elementi della forma strategica possono essere riassunti in una tabella dove ogni casella indica in successione i payoff di ciascun giocatore. Il punto fondamentale consiste nel fatto che le strategie in questa rappresentazione sono un dato del problema. Questa forma di rappresentazione del gioco è stata così chiamata da Shubik (1982) ma già definita forma normale da von Neumann e Morgenstern (1944).

Osservazione 3.1.4. *Il passaggio dalla forma estesa alla forma strategica è sempre possibile nei giochi finiti, ossia si possono sempre elencare tutte le combinazioni di strategie possibili. Il passaggio inverso invece non è sempre possibile, e in ogni caso presenta complicazioni non semplici.*

– *Giochi in forma caratteristica*: è usata nei giochi cooperativi e il suo nome è dovuto a von Neumann e Morgenstern (1944).

Definizione 3.1.5. ***(coalizione).*** *Definito N come l'insieme dei giocatori, ogni sottoinsieme S di N è detto* coalizione. *Se $S = N$ si ha la grande coalizione.*

Definizione 3.1.6. ***(funzione caratteristica).*** *Si dice* funzione caratteristica *di un gioco cooperativo TU ad n giocatori una funzione $v : \wp(N) \to R$ con $v(\emptyset) = 0$.*

Pertanto $v(S)$ è la somma delle utilità che i membri di S possono ottenere dal gioco, indipendentemente da come si comportino gli altri giocatori. In altre parole v assegna ad S la massima vincita possibile indipendentemente dal comportamento degli altri giocatori.

Esempio 3.1.7. ***Riepilogo delle rappresentazioni attraverso il dilemma del prigioniero [7].***
È uno dei problemi più noti della Teoria dei Giochi; è stato introdotto nel 1952 *da Dresher e Flood [6] e poi ripreso da Tucker.*

Due individui sospettati (I e II) sono stati arrestati e accusati di aver commesso un crimine; la polizia non ha prove sufficienti per condannare i sospettati a meno che uno di loro non confessi. La polizia tiene i sospettati in celle separate e a ciascuno di essi vengono spiegate le conseguenze che derivano dalle loro eventuali azioni. Ognuno può scegliere indipendentemente dall'altro di confessare (C) o non confessare (NC). Se entrambi non confessano, vengono condannati per reati minori a due anni ciascuno; se entrambi confessano vengono condannati a cinque anni ciascuno; se uno confessa e l'altro no quello che confessa ha uno sconto di pena e viene condannato a un anno, mentre l'altro ha un'aggravante e viene condannato a sei anni.

Forma estesa
Rappresentata nella figura 3.3.

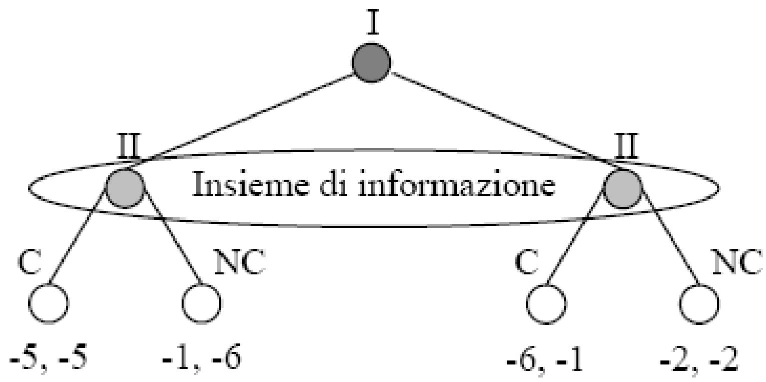


Figura 3.3: forma estesa.

Come già definito in precedenza il gioco in forma estesa si risolve attraverso il metodo delle induzione a ritroso: nella rappresentazione del gioco si considera che a scegliere sia per primo il primo giocatore e poi il secondo giocatore.
Indipendentemente dalla scelta del primo giocatore, il secondo giocatore preferisce sempre la strategia C a NC in quanto -5 *(payoff del secondo giocatore in caso di scelta della coppia di strategie (C,C))* > -6 *(payoff del secondo giocatore in caso di scelta della coppia di strategie (C,NC)) e* -1 *(payoff del secondo giocatore in caso di scelta della coppia di strategie (NC,C))* > -2 *(payoff del secondo giocatore in caso di scelta della coppia di strategie (NC,NC));*
Invece il primo giocatore sapendo che il secondo giocatore sceglierebbe

sempre e comunque la strategia C, di conseguenza sceglie anch'egli la strategia C in quanto -5 *(payoff del primo giocatore in caso di scelta della coppia di strategie (C,C))* > -6 *(payoff del primo giocatore in caso di scelta della coppia di strategie (NC,C)). Quindi il base al ragionamento appena descritto la soluzione del gioco risulta essere la coppia di strategia* (C, C).

Forma strategica
Le pene giudiziare alle quali i sospettati possono essere condannati sono riportate nella tabella 3.1 come coppia (I, II).

	C	**NC**
C	$-5, -5$	$-1, -6$
NC	$-6, -1$	$-2, -2$

Tabella 3.1: dilemma del prigioniero.

Ragionevolmente I sceglie C poichè consegue una condanna minore qualunque sia la scelta di II $(-5 > -6; -1 > -2)$ *e analogamente II sceglie C. La decisione attesa è quindi (C, C) a cui corrisponde una condanna a* 5 *anni per ciascuno, mentre per entrambi sarebbe più vantaggiosa la scelta (NC, NC) con* 2 *anni per ciascuno.*

$u_1 = \{C, C\} = -5,\ u_1 = \{C, NC\} = -1,\ u_1 = \{NC, C\} = -6,\ u_1 = \{NC, NC\} = -2$

$u_2 = \{C, C\} = -5,\ u_2 = \{C, NC\} = -6,\ u_2 = \{NC, C\} = -1,\ u_2 = \{NC, NC\} = -2$

Forma caratteristica

$N = \{1, 2\}$

$v(\varnothing) = 0;$

$v(1) = v(2) = -5;$ *(valore del gioco rispettivamente per il primo e per il secondo giocatore se ciascuno agisce in modo indipendente);*

$v(1,2) = -4$ *(valore del gioco per la coalizione* $(1,2)$*: cioè la somma degli anni di carcere ai cui ambedue i giocatori possono essere condannati mettendosi d'accordo sono* 4*).*

Un'altra importante distinzione riguarda il livello di informazione. Si parla in questo caso di:

– *Giochi con informazione completa*: tutti i giocatori sono a conoscenza di tutte le informazioni possibili. Un'informazione è di comune conoscenza tra due o più giocatori se questi ne sono al corrente, sanno che gli altri ne sono al corrente, sanno che gli altri sanno che i giocatori ne sono al corrente e così via. In verità non è un modello molto aderente alla realtà. Per esempio gli scacchi è un gioco dove l'informazione è completa.

– *Giochi con informazione incompleta*: non tutte le informazioni sono parimenti note a tutti i protagonisti del gioco. Pertanto un modello che tenga conto di questo fatto appare più verosimile. Un esempio di questo tipo di gioco è il poker.

Un'altra importante distinzione riguarda le caratteristiche dei payoff. Quindi il gioco può essere:

– *a somma costante* se e soltanto se ogni per ogni tupla di payoff associata ad una determinata scelta di strategie si verifica che la somma dei payoff risulta costante. Normalmente la costante che più spesso viene considerata come riferimento è 0, ed in questo caso il gioco si dice *a somma zero.*

Esempio 3.1.8. *Come si può notare dal gioco nella tabella 3.2 la somma dei payoff in ciascuna casella (che individua una coppia di strategie) è sempre* 9*.*

	A	**B**
A	$10, -1$	$9, 0$
B	$-5, 14$	$5, 4$

Tabella 3.2: gioco a somma costante.

– *a somma non costante*, dove differentemente dal caso precedente non esiste nessuna relazione tra le tuple di payoff associate alle strategie.

Un'altra distinzione riguarda la temporalità delle mosse. Le mosse in un gioco possono essere:

– *sequenziali (gioco dinamico)*: esiste una sequenza prestabilita nelle scelte dei giocatori, un giocatore non sceglie se non al proprio turno e non contemporaneamente ad un altro. Un giocatore nel momento in cui effettua la propria scelta è a conoscenza della storia del gioco, cioè delle scelte effettuate dagli altri giocatori. Per esempio nel gioco degli scacchi le mosse sono sequenziali.

– *simultanee (gioco statico)*: i giocatori effettuano le loro scelte contemporaneamente e perciò non sono a conoscenza delle scelte effettuate dagli altri giocatori.

Un'altra distinzione riguarda la ripetibilità del gioco. Un gioco può essere disputato:

– *una sola volta (gioco non ripetuto)*, nel caso di una situazione occasionale;

– *più volte (gioco ripetuto)*, nel caso per esempio di una situazione collaborativa tra i giocatori.

Dopo aver effettuato una presentazione delle diverse tipologie di giochi, è corretto individuare le caratteristiche dei giochi che saranno analizzati in questo lavoro. Lo studio sarà incentrato sui giochi:

– non cooperativi;

– in forma strategica;

– ad informazione completa;

– statici;

– non ripetuti.

3.2 Soluzione di un gioco non cooperativo statico ad informazione completa

Il fine ultimo dello studio dei giochi è quello di trovare almeno una soluzione ottima del problema, che viene chiamata anche in questo contesto equilibrio: cioè una n-upla di strategie che si prevedere verrà scelta da giocatori razionali, oppure tale per cui giocatori razionali, in assenza di interventi

esogeni, non avrebbero interesse a modificare la loro strategia una volta che tale n-upla sia stata scelta.
In conformità con lo scopo di questo lavoro, ciò che desta più interesse è lo studio dei criteri di soluzione per i giochi non cooperativi statici ad informazione completa non ripetuti. Sia storicamente che concettualmente, sono stati proposti tre criteri di soluzione:

– quello della dominanza;

– quello dell'equilibrio di Nash in strategie pure;

– quello dell'equilibrio di Nash in strategie miste;

Come si vedrà, le due caratterizzazioni del concetto di equilibrio non sono equivalenti.
La prima è alla base dell'equilibrio di dominanza e di maximin, invece la seconda è alla base di quello di Nash.

3.2.1 Equilibrio di dominanza

Si ammette che i giocatori escludano le strategie dominate, cioè quelle strategie tali per cui in base alla scelta degli avversari esiste almeno un'altra strategia che dia un guadagno maggiore (o in qualche caso, ma non in tutti, un guadagno eguale, nel caso di dominanza debole).

Definizione 3.2.1. ***(strategia strettamente dominata).*** *Dato un gioco in forma strategica e date $a, b \in S_i$ (due strategie ammisibili per il giocatore i), allora* a *si dice* strategia strettamente dominata *da* b *per il giocatore i se*

$$u_i(S_1, S_2, \ldots, S_{i-1}, a, S_{i+1}, \ldots, S_n) < u_i(S_1, S_2, \ldots, S_{i-1}, b, S_{i+1}, \ldots, S_n)$$

per ogni $(S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n) \in S$

Esempio 3.2.2. *Un'esempio di gioco con strategia strettamente dominata è riportato nella tabella 3.3.*
Se il giocatore II sceglie x*, al giocatore I conviene scegliere* a*; se II sceglie* y*, a I conviene scegliere* a*; se II sceglie* z*, a I conviene scegliere* c*. Quindi in nessun caso al giocatore I gli conviene scegliere* b*, che quindi risulta essere una strategia strettamente dominata.*

Definizione 3.2.3. ***(strategia strettamente dominante).*** *Dato un gioco in forma strategica e date due strategie $a, b \in S_i$, allora* a *si dice* strategia strettamente dominante *a* b *per il giocatore i se*

$$u_i(S_1, S_2, \ldots, S_{i-1}, a, S_{i+1}, \ldots, S_n) > u_i(S_1, S_2, \ldots, S_{i-1}, b, S_{i+1}, \ldots, S_n)$$

per ogni $(S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n) \in S$

	x	**y**	**z**
a	$5,7$	$6,8$	$7,9$
b	$4,7$	$5,8$	$9,9$
c	$3,7$	$2,8$	$10,9$

Tabella 3.3: esempio di strategia strettamente dominata.

Esempio 3.2.4. *Si consideri il gioco riportato nella tabella 3.4 che si differenzia da quello dell'esempio precedente per* $u_1(a, z)$ *dove in questo caso ha il valore* 17.

	x	**y**	**z**
a	$5,7$	$6,8$	$17,9$
b	$4,7$	$5,8$	$9,9$
c	$3,7$	$2,8$	$10,9$

Tabella 3.4: esempio di strategia strettamente dominante.

Indipendentemente dalla scelta del giocatore II, al giocatore I conviene sempre la strategia a, *che quindi è strettamente dominante (sia rispetto a* b *che rispetto a* c*).*

I concetti di strategia dominata e strategia dominante sono simili a quelli di *strategia strettamente dominata* e di *strategia strettamente dominante* con l'unica differenza che la relazione di disuguaglianza anzichè essere una disuguaglianza forte ($>$ o $<$) è una disuguaglianza debole ($\leq$ o $\geq$).

Definizione 3.2.5. ***(strategia dominata (debolmente)).*** *Dato un gioco in forma strategica e date due strategie* $a, b \in S_i$*, allora* a *si dice* strategia dominata *da* b *per il giocatore i se*

$$u_i(S_1, S_2, \ldots, S_{i-1}, a, S_{i+1}, \ldots, S_n) \leq u_i(S_1, S_2, \ldots, S_{i-1}, b, S_{i+1}, \ldots, S_n)$$

per ogni $(S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n) \in S$

Esempio 3.2.6. *Un'esempio di gioco con strategia dominata è riportato nella tabella 3.5. Questo gioco si differenzia da quello riportato nella figura 3.3 per* $u_1(c, y)$ *che in questo caso è uguale a* 8 *e per* $u_1(c, z)$ *che in questo caso è uguale a* 9.

	x	**y**	**z**
a	$5,7$	$6,8$	$7,9$
b	$4,7$	$5,8$	$9,9$
c	$3,7$	$8,8$	$9,9$

Tabella 3.5: esempio di strategia dominata.

Se il giocatore II sceglie la strategia x*, al giocatore I conviene scegliere la strategia* a*; invece se il giocatore II sceglie* y*, al giocatore I conviene scegliere* c*; invece se il giocatore II sceglie* z*, per il giocatore I è indifferente scegliere* b *o scegliere* c*.*
In questo esempio, quindi, si può affermare che b *è una strategia dominata in quanto solo in un caso viene scelta in modo alternativo rispetto ad un altra mentre negli altri casi la strategia* b *risulta essere dominata dalle altre strategie.*

Definizione 3.2.7. ***(strategia dominante (debolmente)).*** *Dato un gioco in forma strategica e date le due strategie $a, b \in S_i$, allora* a *si dice* strategia dominante *a* b *per il giocatore i se*

$$u_i(S_1, S_2, \ldots, S_{i-1}, a, S_{i+1}, \ldots, S_n) \geq u_i(S_1, S_2, \ldots, S_{i-1}, b, S_{i+1}, \ldots, S_n)$$

per ogni $(S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n) \in S$

Esempio 3.2.8. *Si consideri il gioco nella tabella 3.6.*
Questo gioco si differenzia da quello riportato nella tabella 3.4 per $u_1(b, x)$ che in questo caso è uguale a 5.

	x	**y**	**z**
a	$5,7$	$6,8$	$17,9$
b	$5,7$	$5,8$	$9,9$
c	$3,7$	$2,8$	$10,9$

Tabella 3.6: esempio di strategia dominante.

La strategia a *si dice strategia dominante (debolmente) in quanto è preferita dal giocatore I rispetto alle strategie* b *e* c *quando il giocatore II sceglie le strategie* y *e* z*, e scelta in modo indifferente alla strategia* b *quando il giocatore II sceglie la strategia* x*.*

Un concetto di soluzione collegato alle definizioni appena citate è quello dell'**eliminazione iterata delle strategie strettamente dominate**.

Esempio 3.2.9. *Si consideri il gioco della tabella 3.7.*

	a	**b**	**c**
a	$2, 2$	$4, 1$	$4, 0$
b	$1, 0$	$3, 3$	$7, 2$
c	$1, 0$	$3, 2$	$1, 4$

Tabella 3.7: processo di eliminazione iterata delle strategie strettamente dominate (1).

Si può iniziare il processo di eliminazione iterata della strategia strettamente dominata partendo dal giocatore I: egli non giocherebbe mai la propria strategia strettamente dominata, cioè la strategia c, *poichè gli garantirebbe sempre e comunque payoff inferiori rispetto alle altre due strategie.*

	a	**b**	**c**
a	$2, 2$	$4, 1$	$4, 0$
b	$1, 0$	$3, 3$	$7, 2$

Tabella 3.8: processo di eliminazione iterata delle strettamente strategie dominate (2).

Ma una volta che il giocatore I esclude la strategia c, *di conseguenza il giocatore II può escludere anch'egli la* c.

	a	**b**
a	$2, 2$	$4, 1$
b	$1, 0$	$3, 3$

Tabella 3.9: processo di eliminazione iterata delle strategie strettamente dominate (3).

Dopo la seconda fase del processo di eliminazione iterata della strategia strettamente dominata, arriva nuovamente il turno del primo giocatore. A *questo punto il giocatore I non sceglierebbe mai di giocare la strategia* b.

	a	**b**
a	$2,2$	$4,1$

Tabella 3.10: processo di eliminazione iterata delle strategie strettamente dominate (4).

In ultima analisi è chiamato a scegliere il giocatore II. Egli esclude la strategia b *e quindi gioca la strategia* a.
Quindi in base al ragionamento elaborato, l'equilibrio del gioco è dato dalla coppia (a, a).

Questo concetto di soluzione è intuitivamente valido; ma è inadeguato, per vari motivi:

– non è detto che un equilibrio di dominanza esista: per esempio, non esiste nella battaglia dei sessi. Il criterio è quindi incompleto;

  Esempio 3.2.10. *La battaglia dei sessi*.
  Due fidanzati devono scegliere tra andare al teatro (T) o alla partita (P).
  Il primo giocatore (I) preferisce andare al teatro mentre il secondo giocatore (II) preferisce la partita, ma entrambi non hanno interesse a restare da soli, come riportato nella tabella 3.11.
  In questo caso non esiste una strategia strettamente dominante per nessun giocatore .
  Il gioco è rappresentato nella tabella 3.11 con le coppie di payoff (I,II).

	T	**P**
T	$2,1$	$0,0$
P	$0,0$	$1,2$

Tabella 3.11: battaglia dei sessi.

– richiede calcoli troppo complessi da parte del decisore perchè possa essere un criterio non solo razionale, ma anche realistico;

– non sempre il risultato dell'eliminazione iterata delle strategie strettamente dominate è il risultato migliore, in quanto ci potrebbero essere casi in cui i giocatori ottengono maggiore utilità mettendosi d'accordo;

– ogni passo del gioco richiede un'assunzione aggiuntiva su ciò che i giocatori conoscono sulla razionalità dell'avversario. Per essere autorizzati ad applicare il procedimento per un numero arbitrario di volte è necessario assumere che la razionalità dei giocatori sia conoscenza comune.

3.2.2 Equilibrio di Nash in strategie pure

Ora si passa all'analisi di un altro concetto di equilibrio: l'equilibrio di Nash in strategie pure.
Le strategie finora considerate sono denominate *strategie pure* per differenziarle dalla differenti strategie miste che saranno analizzate nel paragrafo 3.2.4.
È importante effettuare una precisazione sulla convenzione che viene utilizzate nell'ambito di questo lavoro: con il termine strategia, senza alcuna specificazione, si intende una strategia pura, diversamente si indicherà per esteso la strategia alla quale si vorrà fare riferimento.

Definizione 3.2.11. ***(Equilibrio di Nash).*** *In un gioco non cooperativo in forma strategica $G=\langle N,S,U\rangle$ si dice* equilibrio di Nash *la strategia $(S_1^*,S_2^*,\dots,S_n^*)$ se*

$$u_1(S_1^*,S_2^*,\dots,S_n^*) \geq u_1(S_1,S_2^*,\dots,S_n^*) \quad \forall\ S_1 \in S$$
$$u_2(S_1^*,S_2^*,\dots,S_n^*) \geq u_2(S_1^*,S_2,S_3^*,\dots,S_n^*) \quad \forall\ S_2 \in S$$
$$\vdots$$
$$u_n(S_1^*,S_2^*,\dots,S_n^*) \geq u_n(S_1^*,S_2^*,\dots,S_{n-1}^*,S_n) \quad \forall\ S_n \in S.$$

Esempio 3.2.12. ***Dilemma del prigioniero.***
Il gioco del dilemma del prigioniero ha come equilibrio di Nash la strategia (C,C) come si può notare dalla tabella 3.12.
Nella tabella è sottolineato il payoff associato alla migliore risposta del giocatore 1 *in corrispondenzandi ciascuna strategia ammissibile del giocatore* 2 *e viceversa.*

Esempio 3.2.13. ***La battaglia dei sessi.***
Il gioco della battaglia dei sessi ha due equilibri di Nash (T,T) e (P,P) come viene evidenziato nella tabella 3.13. Questo gioco rappresenta un caso particolare dove l'equilibrio di Nash non risulta unico.

	C	**NC**
C	$\underline{-5}, \underline{-5}$	$\underline{-1}, -6$
NC	$-6, \underline{-1}$	$-2, -2$

Tabella 3.12: equilibrio dilemma del prigioniero.

	T	**P**
T	$\underline{2}, \underline{1}$	$0, 0$
P	$0, 0$	$\underline{1}, \underline{2}$

Tabella 3.13: equilibri battaglia dei sessi.

Esempio 3.2.14. ***Puro coordinamento.***
È una semplice, ma significativa, variante in cui entrambi i giocatori hanno la possibilità di scegliere tra la strategia A e la strategia B.
I giocatori hanno le stesse preferenze: se il giocatore II sceglie la strategia A anche il giocatore I sceglierà la strategia A. Così nel caso in cui il giocatore II scelga la strategia B, il giocatore I si accorderà alla scelta del giocatore II. Il risultato è lo stesso in cui il primo a decidere sia il giocatore I, quindi si ottengono due equilibri di Nash: la coppia (A,A) e la coppia (B,B) come si può notare nella tabella 3.14.

	A	**B**
A	$\underline{1}, \underline{1}$	$0, 0$
B	$0, 0$	$\underline{1}, \underline{1}$

Tabella 3.14: equilibri puro coordinamento.

Esempio 3.2.15. *Il gioco riportato nella tabella 3.15 è caso particolare in cui non vi è l'esistenza dell'equilibrio di Nash.*

I *pregi* dell'equilibrio di Nash:

– stabilità;

– validità per più giocatori.

Invece i *limiti* dell'equilibrio di Nash sono:

	A	**B**
A	$\underline{2}, 1$	$1, \underline{3}$
B	$1, \underline{3}$	$\underline{4}, 2$

Tabella 3.15: esempio senza equilibri di Nash.

– la non unicità dell'equilibrio di Nash, ad esempio la battaglia dei sessi riportata nella tabella 3.11;

– non necessaria esistenza dell'equilibrio di Nash, come nell'esempio della tabella 3.15;

– la non efficienza dell'equilibrio di Nash, come nel caso del dilemma del prigioniero riportato nella tabella 3.1.

3.2.3 Relazione tra equilibrio di Nash ed eliminazione iterata delle strategie strettamente dominate

Proposizione 3.2.16. *Dato un gioco in forma strategica $G = \langle N, S, U \rangle$ con n giocatori se l'eliminazione iterata di strategie strettamente dominate lascia come esito la n-pla $(S_1^*, \ldots, S_n^*)$ allora tale n-pla è l'unico equilibrio di Nash.*

In base alla proposizione precedente si può affermare che se l'eliminazione iterata delle strategie strettamente dominate riesce ad eliminare tutte le strategie tranne una, allora questa combinazione di strategie è l'unico equilibrio di Nash.
Però non è detto che l'equilibrio di Nash sopravviva all'eliminazione iterata delle strategie debolmente dominate.

Proposizione 3.2.17. *Dato un gioco in forma strategica $G = \langle N, S, U \rangle$ con n giocatori il metodo dell'eliminazione iterata di strategie dominate (debolmente) potrebbe eliminare l'equilibrio di Nash.*

Proposizione 3.2.18. *Dato un gioco in forma strategica $G = \langle N, S, U \rangle$ con n giocatori se la n-pla $(S_1^*, S_2^*, \ldots, S_n^*)$ è un'equilibrio di Nash allora sopravvive all'eliminazione iterata di strategie strettamente dominate.*

Tuttavia, in relazione alla proposizione precedente, si può affermare che vi sono strategie che sopravvivono all'eliminazione iterate delle strategie strettamente dominate ma che non sono parte di nessun equilibrio di Nash.

3.2.4 Equilibrio di Nash in strategie miste

In base a quanto è stato detto in precedenza non tutti i giochi presentano un equilibrio di Nash e per superare in parte questo limite si ricorrere ad un altro concetto: quello delle *strategie miste o correlate*.
Il concetto di strategia mista sarà interpretato in termini di incertezza di un giocatore su ciò che faranno gli altri giocatori.

Definizione 3.2.19. ***(strategia mista o correlata)****. Nel gioco in forma strategica $G = \langle N, S, U \rangle$ con n giocatori si supponga che $S_i = \{s_{i1}, \ldots, s_{iK}\}$. Una* strategia mista o correlata *per il giocatore i è una distribuzione di probabilità $p_i = (p_{i1}, \ldots, p_{iK})$, con $0 \leq p_{ik} \leq 1$ per $k = 1, \ldots, K$ e $p_{i1} + \ldots + p_{iK} = 1$.*

Esempio 3.2.20. ***(Matching pennies)****.*
Nel gioco rappresentato nella tabella 3.16 lo spazio di strategie di ogni giocatore è {Testa, Croce}.

	Testa	**Croce**
Testa	$-1, 1$	$1, -1$
Croce	$1, -1$	$-1, 1$

Tabella 3.16: matching pennies.

La storia che solitamente si racconta per giustificare i payoff della matrice è la seguente: ogni giocatore ha una moneta da un penny e deve scegliere se mostrare il lato su compare Testa o quello su cui compare Croce. Se i lati dei due penny sono gli stessi allora il giocatore 2 vince il penny del giocatore 1; se i lati dei penny non sono gli stessi allora 1 vince il penny del giocatore 2. Facilmente si può affermare che in questo gioco non esistono equilibri di Nash in strategie pure come viene evidenziato nella tabella 3.17.

	Testa	**Croce**
Testa	$-1, \underline{1}$	$\underline{1}, -1$
Croce	$\underline{1}, -1$	$-1, \underline{1}$

Tabella 3.17: nessun equilibrio in matching pennies.

La caratteristica distintiva di Matching pennies è che ogni giocatore vorrebbe battere in anticipo l'avversario: indovinare la strategia dell'altro senza

che la propria strategia sia indovinata dall'avversario.
Quest'esempio è costituito da due strategie pure, Testa e Croce; quindi una strategia mista per il giocatore i è la distribuzione di probabilità $(q, 1-q)$ *con* $0 \leq q \leq 1$*, dove q è la probabilità di giocare Testa e* $1-q$ *è la probabilità di giocare Croce.*
La strategia mista $(0,1)$ *è semplicemente la strategia pura Croce; allo stesso modo, la strategia mista* $(1,0)$ *è la strategia pura Testa.*

Per estendere la definizione di equilibrio di Nash in modo da tener conto delle strategie miste si impone semplicemente che la strategia mista di ogni giocatore sia la migliore risposta alla strategia mista degli altri giocatori.
Poichè ogni strategia pura può essere rappresentata come una strategia mista che assegna probabilità zero a tutte le strategie pure del giocatore, questa definizione estesa include quella precedente di equilibrio di Nash.
Si consideri un gioco con due giocatori.
Si indica con $S_1 = \{s_{11}, \dots, s_{1J}\}$ l'insieme delle strategie pure del giocatore 1 e $S_2 = \{s_{21}, \dots, s_{2K}\}$ l'insieme delle strategie pure del giocatore 2; inoltre si indica con s_{1j} e con s_{2k} arbitrarie strategie pure contenute in S_1 e in S_2.
Se il giocatore 1 crede che il giocatore 2 giocherà le strategie $(s_{21}, \dots, s_{2K})$ con probabilità $(p_{21}, \dots, p_{2K})$, allora il payoff atteso dal giocatore 1 ottenuto giocando la strategia pura $s_1 j$ è

$$\sum_{k=1}^{K} p_{2k} u_1(s_{1j}, s_{2k}),$$

e il payoff atteso dal giocatore 1 ottenuto giocando la strategia mista $(p_{11}, \dots, p_{1J})$ è

$$v_1(p_1, p_2) = \sum_{j=1}^{J} p_{1j} \Big\{ \sum_{k=1}^{K} p_{2k} u_1(s_{1j}, s_{2k}) \Big\} = \sum_{j=1}^{J} \sum_{k=1}^{K} p_{1j} \cdot p_{2k} u_1(s_{1j}, s_{2k}),$$

dove $p_{1j} \cdot p_{2k}$ è la probabilità che 1 giochi s_{1j} e 2 giochi s_{2k}.
Affinchè la strategia mista $(p_{11}, \dots, p_{1J})$ sia una risposta ottima del giocatore 1 alla strategia mista di 2, deve valere che $p_{1j} > 0$ solo se

$$\sum_{k=1}^{K} p_{2k} u_1(s_{1j}, s_{2k}) \geq \sum_{k=1}^{K} p_{2k} u_1(s_{1j'}, s_{2k})$$

per ogni $s_{1j'}$ in S_1.
Lo stesso ragionamento viene elaborato anche per il giocatore 2. Se il giocatore 2 crede che il giocatore 1 sceglierà le strategie $(s_{11}, \dots, s_{1J})$ con probabilità $(p_{11}, \dots, p_{1J})$ allora il payoff atteso dal giocatore 2 ricevuto giocando

le strategie $(s_{21},\dots,s_{2K})$ con probabilità $(p_{21},\dots,p_{2K})$ è

$$v_2(p_1,p_2)=\sum_{k=1}^{K}p_{2k}\Big\{\sum_{j=1}^{J}p_{1j}u_2(s_{1j},s_{2k})\Big\}=\sum_{j=1}^{J}\sum_{k=1}^{K}p_{1j}\cdot p_{2k}u_2(s_{1j},s_{2k}).$$

Dati $v_1(p_1,p_2)$ e $v_2(p_1,p_2)$ ora si possono riformulare la condizione richiesta dall'equilibrio di Nash che la strategia mista di ogni giocatore sia una risposta ottima alla strategia mista dell'altro giocatore: affinchè la coppia di strategie miste $(p_1^\star,p_2^\star)$ sia un equilibrio di Nash, $p_1^\star$ deve soddisfare

$$v_1(p_1^\star,p_2^\star)\geq v_1(p_1,p_2^\star)$$

per ogni distribuzione di probabilità p_1 su S_1 e $p_2^\star$ deve soddisfare

$$v_2(p_1^\star,p_2^\star)\geq v_2(p_1^\star,p_2)$$

per ogni distribuzione di probabilità p_2 su S_2.

Definizione 3.2.21. ***(equilibrio di Nash in strategie miste)****. Nel gioco in forma strategica con due giocatori, le strategie miste $(p_1^\star,p_2^\star)$ sono un* equilibrio di Nash *se la strategia mista di ogni giocatore è una risposta ottima alla strategia mista dell'altro giocatore, quindi devono essere rispettate le seguenti condizioni:*

$$v_1(p_1^\star,p_2^\star)\geq v_1(p_1,p_2^\star)$$

per ogni distribuzione di probabilità p_1 su S_1 e $p_2^\star$ deve soddisfare

$$v_2(p_1^\star,p_2^\star)\geq v_2(p_1^\star,p_2)$$

per ogni distribuzione di probabilità p_2 su S_2.

Come si può notare dalla definizione 3.2.21, gli equilibri di Nash in strategie miste dipendono dalle differenti distribuzioni di probabilità associate da ciascun giocatore a ciascuna strategia pura.

Ora si può concludere il discorso sull'equilibrio di Nash con il seguente teorema di Nash che detta le condizioni per l'esistenza dell'equilibrio di Nash, eventualmente anche in strategie miste.

Teorema 3.2.22. ***(Teorema di Nash)****. Nel gioco in forma strategica con n giocatori, se n è finito e S_i è finito per ogni i, allora esiste almeno un equilibrio di Nash, eventualmente in strategie miste.*

Quindi le due condizioni affinchè un gioco ammetta l'esistenza di almeno un'equilibrio di Nash sono le seguenti:

– il numero dei giocatori deve essere finito;

– lo spazio di strategie possibili per ogni giocatore anch'esso deve essere finito.

Inoltre le condizioni del teorema di Nash sono sufficienti ma non necessarie per l'esistenza di un equilibrio di Nash: vi sono nmerosi giochi che non soddisfano le ipotesi del teorema e che tuttavia ammettono uno o più equilibri di Nash.

Capitolo 4

SCSP e giochi non cooperativi

In base a quanto già enunciato, è abbastanza complicato risolvere i SC-SP manualmente sopratutto se occorre gestire più variabili contemporaneamente, e così si è pensato ad un modo alternativo agli algoritmi già descritti per risolvere i SCSP.
Attraverso l'ideazione di un mapping (presentato nel paragrafo successivo) finalizzato alla trasformazione di un SCSP in un gioco non cooperativo statico ad informazione completa si è evidenziata una relazione significativa tra le soluzioni di SCSP e gli equilibri di Nash in strategie pure del corrispondente gioco.
Quindi due sono i passaggi per raggiungere lo scopo prefissato:

- mapping di un SCSP in gioco non cooperativo statico con informazione completa (la forma di rappresentazione del gioco scelta è quella strategica);
- relazione tra le soluzioni di SCSP e gli equilibri di Nash in strategie pure del corrispondente gioco.

4.1 Mapping di un SCSP in un gioco non cooperativo statico ad informazione completa

Si riprenda la definizione di un gioco non cooperativo in forma strategica (definizione 3.1.3).
Il gioco non cooperativo in forma strategica è rappresentato da una tripla $\langle N, S, U \rangle$ dove:

– $N = \{1, \ldots, n\}$ è l'insieme dei giocatori;

– $S = \{S_1, \ldots, S_k\}$ è l'insieme delle strategie possibili;

– $U = \{u_1, \ldots, u_n\}$ è l'insieme delle funzioni payoff associati a ciascun giocatore con $u_i : S^k \to R$.

Affinchè si abbia un gioco non cooperativo, in base alla definizione appena ricordata, è necessario individuare le tre componenti nel SCSP a cui far corrispondere gli elementi costitutivi di un gioco non cooperativo in forma strategica N, S e U.

Il **mapping** ideato è il seguente:

– $N = V$,
cioè l'insieme delle variabili di un SCSP, V, viene equiparato all'insieme dei giocatori di un gioco non cooperativo in forma strategica, l'insieme N; praticamente le variabili di un SCSP rappresentano in un gioco non cooperativo statico ad informazione completa i diversi giocatori;

– $S_i = D_i \qquad \forall\, i = 1, \ldots, n$,
cioè l'insieme delle strategie ha disposizione di ciascun giocatore i, l'insieme S_i, è rappresentato dal rispettivo dominio della variabile X_i, ovvero l'insieme D_i;

– $u_i(d_1, \ldots, d_n) = \bigotimes C_i \qquad$ dove $C_i = \{c \in C : i \in supp(c)\}$,
cioè la funzione di utilità di ciascun giocatore i, indicata con u_i viene costruita attraverso la combinazione dei vincoli che incidono sulla variabile i.

Esempio 4.1.1. *Consideriamo il CSP rappresentato nella figura 4.1.*
Come facilmente si evidenzia dalla figura, il CSP è composto da:

– *due variabili X e Y, quindi $V = \{X, Y\} = N$; in base a ciò il corrispondente gioco non cooperativo sarà composto da due giocatori X e Y;*

– *due strategie di gioco* a *e* b*, quindi $D_X = D_Y = \{a, b\} = S_X = S_Y$; in base a ciò nel corrispondente gioco non cooperativo ciascun giocatore avrà la possibilità di scelta tra due strategie* a*e* b*;*

– *due funzioni utilità u_X e u_Y, una per ciascun corrispondente giocatore, dove in generale:*

$$u_X(d_1, d_2) = \bigotimes C_X$$

e

$$u_Y(d_1, d_2) = \bigotimes C_Y.$$

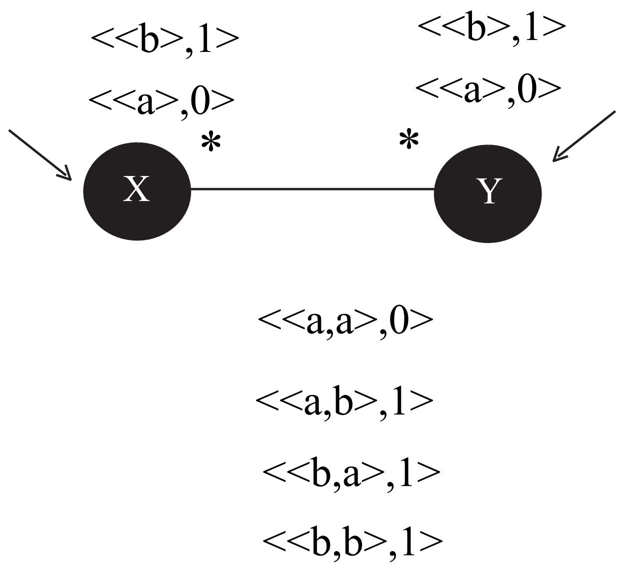


Figura 4.1: esempio di CSP.

XY	$\wedge(\otimes)$
aa	0
ab	0
ba	1
bb	1

Tabella 4.1: payoff di X calcolati partendo dal CSP della figura 4.1.

In particolare i payoff del giocatore X sono rappresentati nella tabella 4.1.
Innanzitutto la prima riga della tabella 4.1, in cui $X = a$ e $Y = a$, viene ottenuta eseguendo i seguenti calcoli:

- *si prende in considerazione il vincolo unario che incide sulla variabile x:* $\langle\langle a\rangle, 0\rangle$;
- *si prende in considerazione il vincolo binario che corrisponde al caso in esame e che incide sulla variabile x:* $\langle\langle a, a\rangle, 0\rangle$
- *infine si effettua la combinazione tra i diversi vincoli attraverso l'operatore* $\wedge$ *logico:* $u_X(a, a) = 0 \wedge 0 = 0$.

Con riferimento alla seconda riga, in cui $X = a$ e $Y = b$, il risultato si ottiene effettuando i seguenti calcoli:

- *si prende in considerazione il vincolo unario che incide sulla variabile x:* $\langle\langle a\rangle, 0\rangle$;
- *si prende in considerazione il vincolo binario che corrisponde al caso in esame e che incide sulla variabile x:* $\langle\langle a, b\rangle, 1\rangle$
- *infine si effettua la combinazione tra i diversi vincoli attraverso l'operatore* $\wedge$ *logico:* $u_X(a, b) = 0 \wedge 1 = 0$.

Con riferimento alla terza riga, in cui $X = b$ e $Y = a$, il risultato si ottiene attraverso i seguenti calcoli:

- *si prende in considerazione il vincolo unario che incide sulla variabile x:* $\langle\langle b\rangle, 1\rangle$;
- *si prende in considerazione il vincolo binario che corrisponde al caso in esame e che incide sulla variabile x:* $\langle\langle b, a\rangle, 1\rangle$
- *infine si effettua la combinazione tra i diversi vincoli attraverso l'operatore* $\wedge$ *logico:* $u_X(b, a) = 1 \wedge 1 = 1$.

Infine nell'ultima riga, in cui $X = b$ e $Y = b$, il risultato viene ottenuto attraverso i seguenti calcoli:

- *si prende in considerazione il vincolo unario che incide sulla variabile x:* $\langle\langle b\rangle, 1\rangle$;
- *si prende in considerazione il vincolo binario che corrisponde al caso in esame e che incide sulla variabile x:* $\langle\langle b, b\rangle, 1\rangle$

– *infine si effettua la combinazione tra i diversi vincoli attraverso l'operatore $\wedge$ logico:* $u_X(b,b) = 1 \wedge 1 = 1$.

Invece i payoff del giocatore Y indicati nella tabella 4.2 si calcolano attraverso lo stesso ragionamento appena descritto per il giocatore X naturalmente modificato e fatto riferire al giocatore Y.

XY	$\wedge(\otimes)$
aa	0
ab	1
ba	0
bb	1

Tabella 4.2: payoff di Y calcolati partendo dal CSP della figura 4.1.

In ultima analisi i payoff del giocatore X (riportati nella tabella 4.1) e i payoff del giocatore Y (riportati nella tabella 4.2) vengono utilizzati per costruire la tabella 4.3 attraverso la quale viene rappresentato il corrispondente gioco del CSP in forma strategica.

Gioco	**a**	**b**
a	$0,0$	$0,1$
b	$1,0$	$1,1$

Tabella 4.3: gioco non cooperativo statico ad informazione completa corrispondente al CSP della figura 4.1.

4.1.1 Soluzione SCSP ed equilibrio di Nash

Prima di indicare che corrispondenza esiste tra la soluzione di un SCSP e l'equilibrio di Nash (in strategie pure), può essere utile riprendere la definizione 3.2.11.

(Equilibrio di Nash). In un gioco non cooperativo in forma strategica $\langle N, S, U\rangle$ si dice *equilibrio di Nash* la strategia $(S_1^*, S_2^*, \ldots, S_n^*)$ se

$$u_1(S_1^*, S_2^*, \ldots, S_n^*) \geq u_1(S_1, S_2^*, \ldots, S_n^*) \quad \forall\ S_1 \in S$$

$$u_2(S_1^*, S_2^*, \ldots, S_n^*) \geq u_2(S_1^*, S_2, S_3^*, \ldots, S_n^*) \quad \forall\ S_2 \in S$$

$$\vdots$$

$$u_n(S_1^*, S_2^*, \ldots, S_n^*) \geq u_n(S_1^*, S_2^*, \ldots, S_{n-1}^*, S_n) \quad \forall\ S_n \in S.$$

La relazione che si può stabilire, grazie al mapping descritto nella sezione precedente, tra la soluzione di un dato SCSP e l'equilibrio di Nash del corrispondente gioco $G(C)$ si può riassumere nel teorema seguente.

Teorema 4.1.2. *Se un SCSP ha una soluzione ottima ≥ 0, allora il gioco $G(C)$ ha il corrispondente equilibrio di Nash.*

Dimostrazione 4.1.3. *La dimostrazione farà riferimento alla "functional formulation" indicata nel paragrafo 2.2 in quanto più utile al fine della dimostrazione in questione.*

Si assume che η sia la soluzione ottima per il SCSP.
Di conseguenza vale la seguente disuguaglianza:

$$\bigotimes C\eta \geq \bigotimes C\eta' \qquad \forall \eta'$$

cioè in base all'ipotesi iniziale dato che η è la soluzione ottima qualsiasi altra soluzione sarà minore o uguale a η.

L'equilibrio di Nash può essere ridefinito in base alla "functional formulation" nella seguente maniera:

$$\bigotimes C_1(\eta) \geq \bigotimes C_1 \Downarrow_{V-\{x_1\}} \eta$$

$$\vdots$$

$$\bigotimes C_\eta(\eta) \geq \bigotimes C_\eta \Downarrow_{V-\{x_n\}} \eta.$$

Come facilmente si può accurare, quest'ultima formulazione dell'equilibrio di Nash è equivalente a quella definizione 3.2.11 cambia solo la notazione. Infatti in base al mapping proposto dato che

$$u_i(d_1, \ldots, d_n) = \bigotimes C_i$$

di conseguenza si ha che l'espresione a sinistra della disuguaglianza, ossia la combinazione di vincoli che incidono sulla variabile i con riferimento alla soluzione ottima, risulta maggiore o uguale all'espressione a destra della disuguaglianza.
Ciò si può affermare in quanto l'espressione alla destra della disuguaglianza corrisponde all'espressione alla sinistra modificata proiettando la soluzione

ottima su tutte le variabili escluso la variabile in questione (è quest'ultima ad essere modificata).

Quindi se η è la soluzione ottima del SCSP, allora per il fatto che

$$\bigotimes C\eta \geq \bigotimes C\eta' \qquad \forall \eta'$$

si avrà

$$\bigotimes C\eta \geq \bigotimes C\eta[x_i := d_j] \qquad \forall\ x_i \in V, d_j \in D$$

dove il valore di $\bigotimes C\eta[x_i := d_j]$ è il valore $\bigotimes C\eta$ modificato attribuendo alla variabile $x_i \in V$ il valore $d_j \in D$.
Considerando C_i, l'insieme dei vincoli che incidono sulla variabile x_i, e $\bar{C}_i$, l'insieme dei vincoli che non incidono sulla variabile x_i (dove $C_i \cup \bar{C}_i = C$), scrivere l'espressione precedente equivale a scrivere

$$(\bigotimes \bar{C}_i \otimes \bigotimes C_i)\eta \geq (\bigotimes \bar{C}_i \otimes \bigotimes C_i)\eta[x_i := d_j] \qquad \forall\ x_i \in V, d_j \in D.$$

In base alla definizione 2.2.4 riferita alla combinazione di vincoli si ha che l'espressione precedente è equivalente a

$$\bigotimes \bar{C}_i\eta \times \bigotimes C_i\eta \geq \bigotimes \bar{C}_i\eta \times \bigotimes C_i\eta[x_i := d_j] \qquad \forall\ x_i \in V, d_j \in D.$$

Grazie al fatto che si può dividere per lo stesso valore in base all'operatore di divisione, applicabile qualora vengono rispettate alcune condizioni già enunciate in precedenza nelle definizioni da 2.4.4 a 2.4.8, allora

$$\bigotimes C_i\eta \geq \bigotimes C_i\eta[x_i := d_j] \qquad \forall\ x_i \in V, d_j \in D.$$

Quest'ultima disuguaglianza corrisponde alla definizione di equilibrio di Nash in quanto l'espressione alla destra della disuguaglianza equivale all'espressione alla sinistra modificata facendo variare i valori della variabile x_i.
Dato che la disuguaglianza vale per ogni $x_i \in V$ allora si ha la corrispondenza con l'equilibrio di Nash.

Di seguito saranno riportati degli esempi sui differenti tipi di SCSP tesi a confermare quanto enunciato dal teorema 4.1.2.

Esempio 4.1.4. ***CSP.***
Si riconsideri il gioco riportato nella tabella 4.3 corrispondente al CSP della figura 4.1.
Per individuare l'equilibrio o gli equilibri di Nash si elabora il seguente ragionamento:

- *per quanto riguarda il primo giocatore (X):*

  - *se il secondo giocatore (Y) sceglie di giocare la strategia* a*, il primo giocatore (X) tra* 0 *(payoff della strategia* a*) e* 1 *(payoff della strategia* b*) sceglie ovviamente di giocare la strategia* b *in quanto* $1 > 0$*;*
  - *se il secondo giocatore (Y) decide di giocare la strategia* b*, il primo giocatore (X) tra* 0 *(payoff della strategia* a*) e* 1 *(payoff della strategia* b*) sceglie ovviamente di giocare la strategia* b *in quanto* $1 > 0$*;*

- *per quanto riguarda il secondo giocatore (Y):*
  - *se il primo giocatore (X) sceglie di giocare la strategia* a*, il secondo giocatore (Y) tra* 0 *(payoff della strategia* a*) e* 1 *(payoff della strategia* b*) sceglie ovviamente di giocare la strategia* b *in quanto* $1 > 0$*;*
  - *se il primo giocatore (X) decide di giocare la strategia* b*, il secondo giocatore (Y) tra* 0 *(payoff della strategia* a*) e* 1 *(payoff della strategia* b*) sceglie ovviamente di giocare la strategia* b *in quanto* $1 > 0$*;*

Quindi come è possibile notare dalla tabella seguente, in cui vengono evidenziate le migliori risposte, in termini di strategie, di ciascun giocatore in corrispondenza delle strategie ammissibili dell'altro giocatore:

Gioco	**a**	**b**
a	$0, 0$	$0, \underline{1}$
b	$\underline{1}, 0$	$\underline{1}, \underline{1}$

Tabella 4.4: equilibrio di Nash del gioco della figura 4.3.

l'equilibrio di Nash è la coppia di strategie (b,b).
Per verificare la corrispondenza tra la soluzione del CSP e l'equilibrio di Nash del corrispondente gioco non cooperativo, seguendo quanto citato dal teorema, è necessario mostrare, innanzitutto, la soluzione del CSP rappresentato nella figura 4.1.
Nella tabella 4.5 vengono indicate le soluzioni del CSP.
Un'unica soluzione risulta dalla tabella, ossia la tupla di valori bb.
I calcoli che sono eseguiti per giungere a tale conclusione sono i seguenti:

- *per la tupla di valori aa* ($X = a$ *e* $Y = a$) *dopo aver individuato i seguenti vincoli:*

XY	$\wedge(\times)$	$\vee(+)$
aa	0	
ab	0	
ba	0	
bb	1	**soluzione**

Tabella 4.5: soluzione del CSP rappresentato nella figura 4.1.

- *il vincolo unario* $\langle\langle a\rangle, 0\rangle$ *sulla variabile* X*;*
- *il vincolo unario* $\langle\langle a\rangle, 0\rangle$ *sulla variabile* Y*;*
- *il vincolo binario* $\langle\langle a, a\rangle, 0\rangle$*;*

bisogna effettuarne la combinazione in base alla definizione di soluzione (2.1.18), quindi $\bigotimes C = 0 \wedge 0 \wedge 0 = 0$*;*

– *per la tupla di valori ab* ($X = a$ *e* $Y = b$) *vengono individuato i seguenti vincoli:*

- *il vincolo unario* $\langle\langle a\rangle, 0\rangle$ *sulla variabile* X*;*
- *il vincolo unario* $\langle\langle b\rangle, 1\rangle$ *sulla variabile* Y*;*
- *il vincolo binario* $\langle\langle a, b\rangle, 1\rangle$*;*

e combinati, infatti $\bigotimes C = 0 \wedge 1 \wedge 1 = 0$*;*

– *per la tupla di valori ab* ($X = b$ *e* $Y = a$) *vengono individuato i seguenti vincoli:*

- *il vincolo unario* $\langle\langle b\rangle, 1\rangle$ *sulla variabile* X*;*
- *il vincolo unario* $\langle\langle a\rangle, 0\rangle$ *sulla variabile* Y*;*
- *il vincolo binario* $\langle\langle b, a\rangle, 1\rangle$*;*

e combinati, infatti $\bigotimes C = 1 \wedge 0 \wedge 1 = 0$*;*

– *per la tupla di valori ab* ($X = b$ *e* $Y = b$) *vengono individuato i seguenti vincoli:*

- *il vincolo unario* $\langle\langle b\rangle, 1\rangle$ *sulla variabile* X*;*
- *il vincolo unario* $\langle\langle b\rangle, 1\rangle$ *sulla variabile* Y*;*
- *il vincolo binario* $\langle\langle b, b\rangle, 1\rangle$*;*

e combinati, infatti $\bigotimes C = 1 \wedge 1 \wedge 1 = 1$.

Con ciò l'unico caso in cui vengono rispettati tutti i vincoli è quando $X = b$ *e* $Y = b$.

Esempio 4.1.5. ***Fuzzy CSP.***
Si consideri il fuzzy CSP indicato nella figura 4.2.

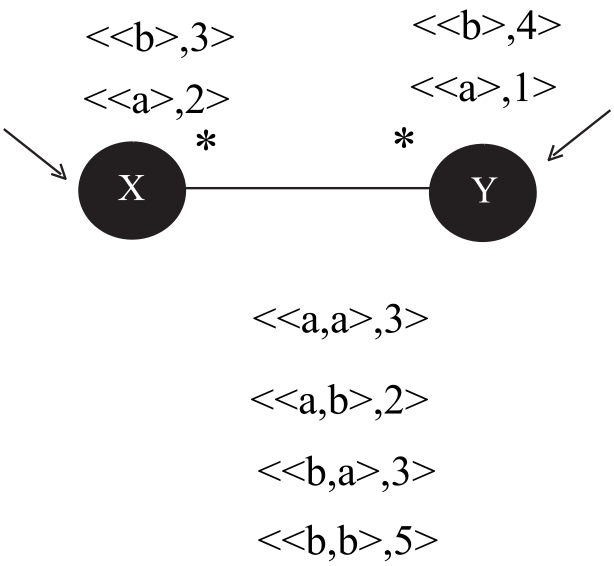


Figura 4.2: esempio di fuzzy CSP.

Per calcolare le soluzioni del fuzzy CSP si applica lo stesso ragionamento dell'esempio precedente dove in questo la combinazione dei vincoli avviene attraverso min *e la proiezione avviene attraverso* max.
L'unica soluzione del fuzzy CSP è indicata nella tabella 4.6 che infatti rappresenta il massimo dei minimi trovati con la combinazione dei vincoli.

XY	min(×)	max(+)
aa	1	
ab	2	
ba	1	
bb	3	**soluzione**

Tabella 4.6: soluzione del fuzzy CSP della figura 4.2.

Attraverso il mapping proposto si calcolano le utilità del giocatore X (tabella 4.7) e le utilità del giocatore Y (tabella 4.8).

XY	min(⊗)
aa	2
ab	2
ba	3
bb	3

Tabella 4.7: le utilità del giocatore X calcolate partendo dal fuzzy CSP della figura 4.2.

Per quanto riguarda il calcolo delle utilità del primo giocatore:

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* a*, l'utilità del giocatore X viene calcolata facendo la combinazione del:*

- *vincolo unario* $\langle\langle a\rangle, 2\rangle$*;*
- *e vincolo binario* $\langle\langle a, a\rangle, 3\rangle$*;*

attraverso min *ottenendo come risultato* 2*;*

– *nel caso in cui il primo giocatore sceglie di giocare la strategia* a *e il secondo giocatore sceglie di giocare la strategia* b*, l'utilità del giocatore X viene calcolata facendo la combinazione del:*

- *vincolo unario* $\langle\langle a\rangle, 2\rangle$*;*
- *e vincolo binario* $\langle\langle a, b\rangle, 2\rangle$*;*

attraverso min *ottenendo come risultato* 2*;*

– *nel caso in cui il primo giocatore sceglie di giocare la strategia* b *e il secondo giocatore sceglie di giocare la strategia* a*, l'utilità del giocatore X viene calcolata facendo la combinazione del:*

- *vincolo unario* $\langle\langle b\rangle, 3\rangle$*;*
- *e vincolo binario* $\langle\langle b, a\rangle, 3\rangle$*;*

attraverso min *ottenendo come risultato* 3*;*

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* b*, l'utilità del giocatore X viene calcolata facendo la combinazione del:*

- *vincolo unario* $\langle\langle b\rangle, 3\rangle$*;*
- *e vincolo binario* $\langle\langle b, b\rangle, 5\rangle$*;*

attraverso min *ottenendo come risultato* 3.

XY	min(⊗)
aa	1
ab	2
ba	1
bb	4

Tabella 4.8: utilità di Y calcolate partendo dal fuzzy CSP della figura 4.2.

Per quanto riguarda il calcolo delle utilità del secondo giocatore:

- *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* a*, l'utilità del giocatore Y viene calcolata facendo la combinazione del:*

  – *vincolo unario* $\langle\langle a\rangle, 1\rangle$*;*
  – *e vincolo binario* $\langle\langle a, a\rangle, 3\rangle$*;*

  attraverso min *ottenendo come risultato* 1*;*

- *nel caso in cui il primo giocatore sceglie di giocare la strategia* a *e il secondo giocatore sceglie di giocare la strategia* b*, l'utilità del giocatore X viene calcolata facendo la combinazione del:*

  – *vincolo unario* $\langle\langle b\rangle, 4\rangle$*;*
  – *e vincolo binario* $\langle\langle a, b\rangle, 2\rangle$*;*

  attraverso min *ottenendo come risultato* 2*;*

- *nel caso in cui il primo giocatore sceglie di giocare la strategia* b *e il secondo giocatore sceglie di giocare la strategia* a*, l'utilità del giocatore X viene calcolata facendo la combinazione del:*

  – *vincolo unario* $\langle\langle a\rangle, 1\rangle$*;*

  - *e vincolo binario* $\langle\langle b, a\rangle, 3\rangle$*;*

  attraverso min *ottenendo come risultato* 1*;*

- *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* b*, l'utilità del giocatore X viene calcolata facendo la combinazione del:*
  - *vincolo unario* $\langle\langle b\rangle, 4\rangle$*;*
  - *e vincolo binario* $\langle\langle b, b\rangle, 5\rangle$*;*

  attraverso min *ottenendo come risultato* 4*.*

Dopo aver calcolato le utilità rispettivamente per ciascun giocatore si ottiene il corrispondente gioco del fuzzy CSP rappresentato nella tabella 4.9.

Gioco	**a**	**b**
a	$2, 1$	$2, 2$
b	$3, 1$	$3, 4$

Tabella 4.9: gioco non cooperativo statico ad informazione completa corrispondente al fuzzy CSP della figura 4.2.

Le migliori risposte, in termini di strategie, di ciascun giocatore in riferimento alle scelte dell'altro giocatore sono evidenziate nella tabella 4.10.

Gioco	**a**	**b**
a	$2, 1$	$2, \underline{2}$
b	$\underline{3}, 1$	$\underline{3}, \underline{4}$

Tabella 4.10: equilibrio di Nash del gioco non cooperativo della tabella 4.9.

Si noti che l'equilibrio di Nash del gioco è la coppia (b,b) che corrisponde esattamente alla soluzione del suo corrispondente fuzzy CSP (tabella 4.6).

Esempio 4.1.6. ***Weighted CSP.***
Si consideri il weighted CSP indicato nella figura 4.3.
L'unica soluzione del weighted CSP è indicata nella tabella 4.11. La soluzione viene calcolata utilizzando come operazione di combinazione il $+$ *e come operazione di proiezione il* min*, cioè il fine è quello di minimizzare la somma*

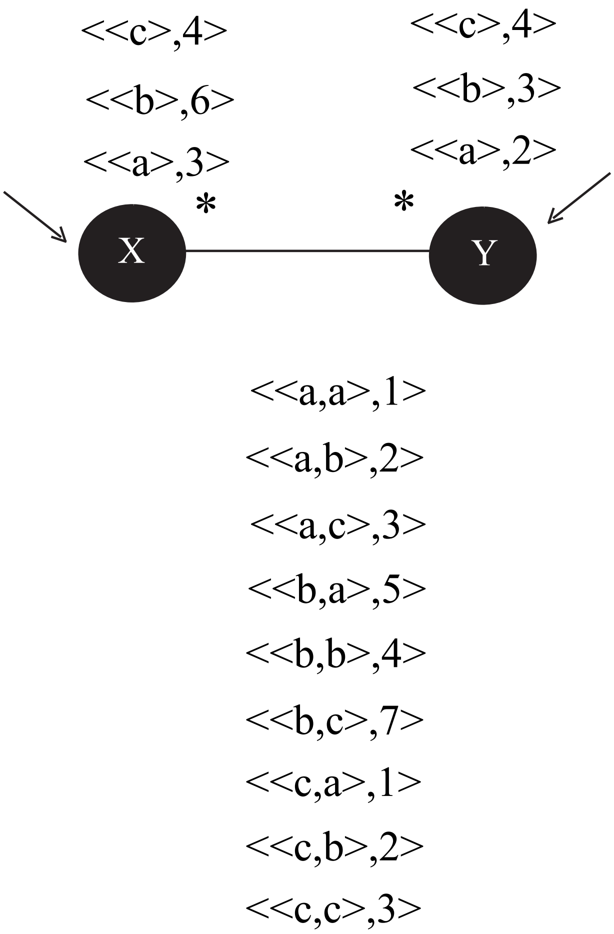


Figura 4.3: esempio di weighted CSP.

XY	$+(\times)$	$\min(+)$
aa	6	**soluzione**
ab	8	
ac	10	
ba	13	
bb	13	
bc	17	
ca	7	
cb	9	
cc	11	

Tabella 4.11: soluzione del weighted CSP della figura 4.3.

dei costi.

Attraverso il mapping proposto si calcolano le utilità del giocatore X e le utilità del giocatore Y.
Per quanto riguarda il giocatore X:

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* a*, l'utilità del giocatore X viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle a\rangle, 3\rangle$*;*
  - *e vincolo binario* $\langle\langle a, a\rangle, 1\rangle$*;*

  attraverso + ottenendo come risultato 4*;*

– *nel caso in cui il primo sceglie di giocare la strategia* a *e il secondo sceglie di giocare la strategia* b*, l'utilità del giocatore X viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle a\rangle, 3\rangle$*;*
  - *e vincolo binario* $\langle\langle a, b\rangle, 2\rangle$*;*

  attraverso + ottenendo come risultato 5*;*

– *nel caso in cui il primo sceglie di giocare la strategia* a *e il secondo sceglie di giocare la strategia* c*, l'utilità del giocatore X viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle a\rangle, 3\rangle$*;*
  - *e vincolo binario* $\langle\langle a, c\rangle, 3\rangle$*;*

  attraverso + ottenendo come risultato 6*;*

– *nel caso in cui il primo sceglie di giocare la strategia* b *e il secondo sceglie di giocare la strategia* a*, l'utilità del giocatore X viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle b\rangle, 6\rangle$*;*
  - *e vincolo binario* $\langle\langle b, a\rangle, 5\rangle$*;*

  attraverso + ottenendo come risultato 11*;*

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* b*, l'utilità del giocatore* X *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle b\rangle, 6\rangle$*;*
  - *e vincolo binario* $\langle\langle b, b\rangle, 4\rangle$*;*

  attraverso $+$ *ottenendo come risultato* 10*;*

– *nel caso in cui il primo sceglie di giocare la strategia* b *e il secondo sceglie di giocare la strategia* c*, l'utilità del giocatore* X *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle b\rangle, 6\rangle$*;*
  - *e vincolo binario* $\langle\langle b, c\rangle, 7\rangle$*;*

  attraverso $+$ *ottenendo come risultato* 13*;*

– *nel caso in cui il primo sceglie di giocare la strategia* c *e il secondo sceglie di giocare la strategia* a*, l'utilità del giocatore* X *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle c\rangle, 4\rangle$*;*
  - *e vincolo binario* $\langle\langle c, a\rangle, 1\rangle$*;*

  attraverso $+$ *ottenendo come risultato* 5*;*

– *nel caso in cui il primo sceglie di giocare la strategia* c *e il secondo sceglie di giocare la strategia* b*, l'utilità del giocatore* X *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle c\rangle, 4\rangle$*;*
  - *e vincolo binario* $\langle\langle c, b\rangle, 2\rangle$*;*

  attraverso $+$ *ottenendo come risultato* 6*;*

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* c*, l'utilità del giocatore* X *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle c\rangle, 4\rangle$*;*
  - *e vincolo binario* $\langle\langle c, c\rangle, 3\rangle$*;*

  attraverso $+$ *ottenendo come risultato* 7*;*

Le utilità calcolate vengono mostrate nella tabella 4.12.

XY	$+(\otimes)$
aa	4
ab	5
ac	6
ba	11
bb	10
bc	13
ca	5
cb	6
cc	7

Tabella 4.12: le utilità del giocatore X calcolate partendo dal weighted CSP della figura 4.3.

Invece per quanto riguarda il giocatore Y:

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* a*, l'utilità del giocatore Y viene calcolata effettuando la combinazione del:*

- *vincolo unario* $\langle\langle a\rangle, 2\rangle$*;*
- *e vincolo binario* $\langle\langle a, a\rangle, 1\rangle$*;*

attraverso $+$ *ottenendo come risultato* 3*;*

– *nel caso in cui il primo sceglie di giocare la strategia* a *e il secondo sceglie di giocare la strategia* b*, l'utilità del giocatore Y viene calcolata effettuando la combinazione del:*

- *vincolo unario* $\langle\langle b\rangle, 3\rangle$*;*
- *e vincolo binario* $\langle\langle a, b\rangle, 2\rangle$*;*

attraverso $+$ *ottenendo come risultato* 5*;*

– *nel caso in cui il primo sceglie di giocare la strategia* a *e il secondo sceglie di giocare la strategia* c*, l'utilità del giocatore Y viene calcolata effettuando la combinazione del:*

- *vincolo unario* $\langle\langle c\rangle, 4\rangle$*;*

  - *e vincolo binario* $\langle\langle a, c\rangle, 3\rangle$*;*

  attraverso + *ottenendo come risultato* 7*;*

– *nel caso in cui il primo sceglie di giocare la strategia* b *e il secondo sceglie di giocare la strategia* a*, l'utilità del giocatore* X *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle a\rangle, 2\rangle$*;*
  - *e vincolo binario* $\langle\langle b, a\rangle, 5\rangle$*;*

  attraverso + *ottenendo come risultato* 7*;*

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* b*, l'utilità del giocatore* Y *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle b\rangle, 3\rangle$*;*
  - *e vincolo binario* $\langle\langle b, b\rangle, 4\rangle$*;*

  attraverso + *ottenendo come risultato* 7*;*

– *nel caso in cui il primo sceglie di giocare la strategia* b *e il secondo sceglie di giocare la strategia* c*, l'utilità del giocatore* X *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle c\rangle, 4\rangle$*;*
  - *e vincolo binario* $\langle\langle b, c\rangle, 7\rangle$*;*

  attraverso + *ottenendo come risultato* 11*;*

– *nel caso in cui il primo sceglie di giocare la strategia* c *e il secondo sceglie di giocare la strategia* a*, l'utilità del giocatore* Y *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle a\rangle, 2\rangle$*;*
  - *e vincolo binario* $\langle\langle c, a\rangle, 1\rangle$*;*

  attraverso + *ottenendo come risultato* 3*;*

– *nel caso in cui il primo sceglie di giocare la strategia* c *e il secondo sceglie di giocare la strategia* b*, l'utilità del giocatore* Y *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle b\rangle, 3\rangle$;
  - *e vincolo binario* $\langle\langle c, b\rangle, 2\rangle$;

  attraverso $+$ *ottenendo come risultato* 5;

– *nel caso in cui sia il primo che secondo giocatore scelgono di giocare la strategia* c, *l'utilità del giocatore* Y *viene calcolata effettuando la combinazione del:*

  - *vincolo unario* $\langle\langle c\rangle, 4\rangle$;
  - *e vincolo binario* $\langle\langle c, c\rangle, 3\rangle$;

  attraverso $+$ *ottenendo come risultato* 7;

Le utilità appena calcolate vengono mostrate nella tabella tabella 4.13.

XY	$+(\otimes)$
aa	3
ab	5
ac	7
ba	7
bb	7
bc	11
ca	3
cb	5
cc	7

Tabella 4.13: le utilità del giocatore Y calcolate partendo dal weigthed CSP della figura 4.3.

Le utilità di entrambi i giocatori vengono utilizzate per costruire il gioco non cooperativo corrispondente al weigthed CSP. Nella tabella 4.14 saranno evidenziate le strategie scelte da ciascun giocatore in relazione alle decisioni dall'altro giocatore al fine di evidenziare gli equilibri di Nash.

Si noti che l'unico equilibrio di Nash del gioco (ottenuto minimizzando i costi) è la coppia (a,a) esattamente la soluzione del corrispondente weighted CSP (tabella 4.11).

La relazione opposta del teorema 4.1.2 non viene mantenuta, cioè potrebbero esserci casi in cui esistono più equilibri di Nash rispetto all'effettive

Gioco	**a**	**b**	**c**
a	$\underline{4}, \underline{3}$	$\underline{5}, 5$	$\underline{6}, 7$
b	$11, \underline{7}$	$10, \underline{7}$	$13, 11$
b	$5, \underline{3}$	$6, 5$	$7, 7$

Tabella 4.14: equilibrio di Nash del gioco non cooperativo statico ad informazione completa corrispondente al weighted CSP della figura 4.3.

soluzioni del SCSP o meglio il caso più immediato in cui il SCSP non ha soluzioni e il corrispondente gioco non cooperativo ha uno o più equilibri di Nash.

Esempio 4.1.7. ***SCSP senza soluzioni e corrispondente gioco non cooperativo con equilibrio di Nash.***
Si consideri il CSP mostrato nella figura 4.4.

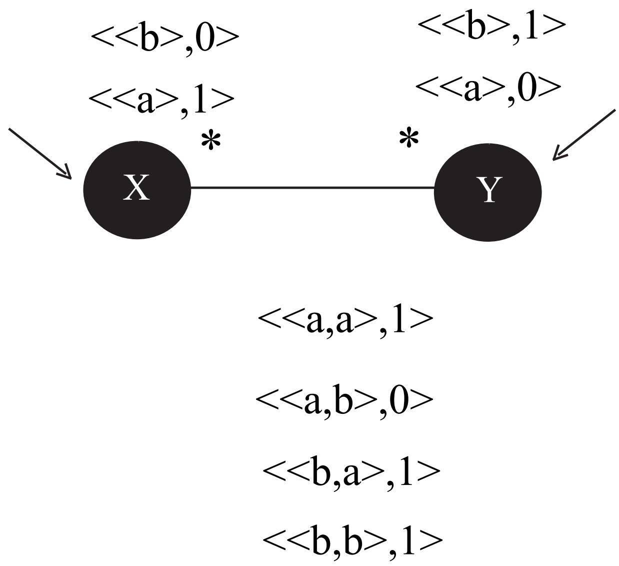


Figura 4.4: esempio di CSP senza soluzioni.

Come si può notare dalla tabella 4.15 il CSP non ha soluzioni in quanto in nessun caso vengono rispettati tutti i vincoli.

Nella tabella 4.16 e nella tabella 4.17 sono riportate le utilità rispettivamente del giocatore X e del giocatore Y calcolate attraverso il mapping.

XY	$\wedge(\otimes)$	**soluzioni**
aa	0	no
ab	0	no
ba	0	no
bb	0	no

Tabella 4.15: soluzioni del CSP della figura 4.4.

XY	$\wedge(\otimes)$
aa	1
ab	0
ba	0
bb	0

Tabella 4.16: le utilità di X nel gioco corrispondente al CSP della figura 4.4

XY	$\wedge(\otimes)$
aa	0
ab	0
ba	0
bb	1

Tabella 4.17: le utilità di Y nel gioco corrispondente al CSP della figura 4.4.

Le utilità mostrate vengono utilizzate per rappresentare il gioco corrispondente al CSP senza soluzione in forma strategica attraverso la tabella 4.18. Vengono messe in evidenza, attraverso sottolineatura, le scelte di ciascun giocatore in relazione alle scelte dell'altro giocatore al fine di indicare l'equilibrio di Nash.

Gioco	**a**	**b**
a	$\underline{1}, \underline{0}$	$\underline{0}, \underline{0}$
b	$0, 0$	$\underline{0}, \underline{1}$

Tabella 4.18: equilibrio di Nash del gioco non cooperativo statico ad informazione completa corrispondente al weighted CSP della figura 4.4.

Si può notare che seppure il CSP non ha soluzioni, il corrispondente gioco non cooperativo ottenuto attraverso il mapping (figura 4.18) ha tre equilibri di Nash: le coppie (a,a),(a,b) e (b,b).

4.1.2 Mapping di un problema di colorazione di un grafo

Questo paragrafo è finalizzato all'applicazione pratica del teorema 4.1.2 in modo da risolvere il problema di colorazione di un grafo, già presentato nel paragrafo 1.3.1, attraverso la Teoria dei giochi.
Si ricorda brevemente che lo scopo di questo problema è quello di utilizzare il minor numero possibile di colori per colorare i nodi di un grafo in modo tale che nodi adiacenti assumano colori differenti.

In questa occasione si pone come problema quello di colorare parte della cartina rappresentatata nella figura 4.5 attraverso il rispetto dei vincoli sopra indicati.
In particolare si prenderà in considerazione, come variabili del problema, soltanto le nazioni colorate di bianco.
La scelta è caduta su quelle nazioni poichè confinano contemporaneamente con più nazioni, è di conseguenza rispettano l'obiettivo preposto: mostrare un esempio ristretto ma allo stesso tempo rappresentativo che, in futuro eventualmente, può essere esteso per risolvere problemi più complessi in termini di numero di variabili.
Nella figura 4.6 sono evidenziate attraverso gli archi le relazioni di adiacenza tra le differenti nazioni.
Il problema proposto è rappresentato nella figura 4.7 sottoforma di CSP, nella quale sono elencati i vincoli unari e binari che devono essere rispettati.

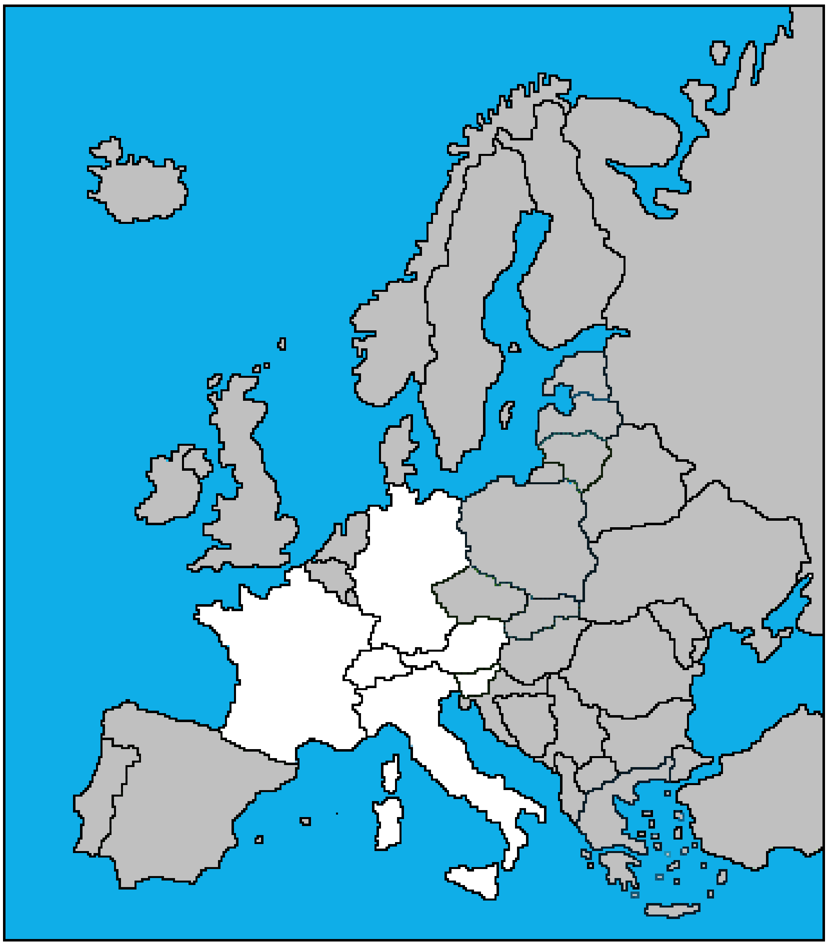


Figura 4.5: cartina.

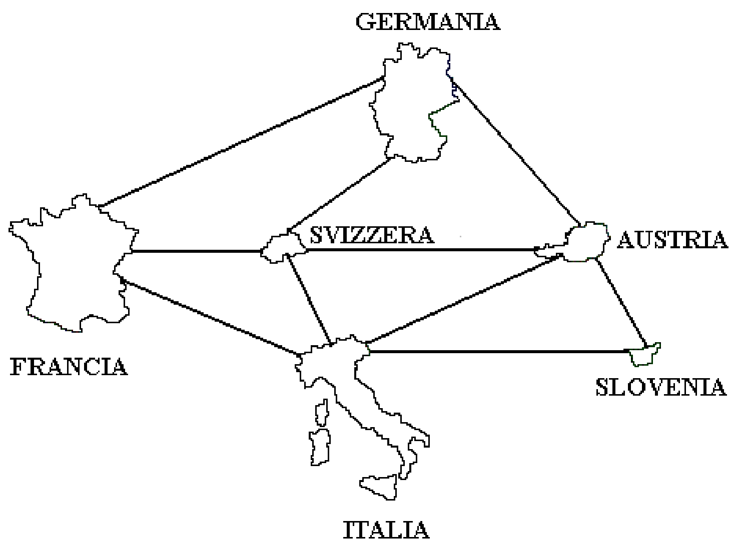
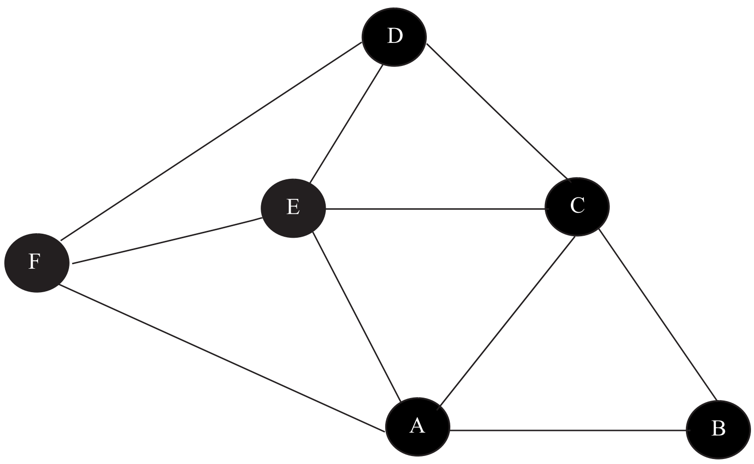


Figura 4.6: Relazioni di adiacenza tra le nazioni.

La nazione Italia è rappresentata dal nodo A, la nazione Slovenia dal nodo B, la nazione Austria dal nodo C, la nazione Germania dal nodo D, la nazione Svizzera dal nodo E e la nazione Francia dal nodo F.
Si assume che ciascuna nazione possa assumere uno dei seguenti colori: verde, giallo, azzurro e viola. Quest'ultimi saranno abbreviati per esemplificazione rispettivamente con le iniziali: VE, G, A e VI.

Il software utilizzato, per la creazione dell'interfaccia grafica del gioco corrispondente al CSP rappresentato, è stato il *gambit version* 0.2006.01.20. Nel gioco ottenuto le nazioni rappresentano i giocatori e i colori che ciascuna nazione può assumere rappresentano le strategie possibili.
Come prima cosa si è fornito in input le utilità calcolate attraverso il mapping proposto: ciò ha permesso di creare una tabella che rappresentasse in forma strategica le diverse combinazioni di colori per ciascuna nazione, con il rispetto contemporaneo di tutti i vincoli riportati nella figura 4.7.
La prima parte della tabella creata viene presa come riferimento illustrativo di ciò che è stato fornito in input attraverso gambit e viene mostrata attraverso le figure 4.8, 4.9, 4.10 e 4.11 rispettivamente differenti per il colore assunto dalla nazione Slovenia.



Vincoli unari per ogni nodo

<<VE>,1>

<<G>,1>

<<A>,1>

<<VI>,1>

Vincoli binari per ogni arco

<<VE,VE>,0>	<<G,VE>,1>	<<A,VE>,1>	<<VI,VE>,1>
<<VE,G>,1>	<<G,G>,0>	<<A,G>,1>	<<VI,G>,1>
<<VE,A>,1>	<<G,A>,1>	<<A,A>,0>	<<VI,A>,1>
<<VE,VI>,1>	<<G,VI>,1>	<<A,VI>,1>	<<VI,VI>,0>

Figura 4.7: problema di colorazione della cartina della figura 4.5 sottoforma di CSP.

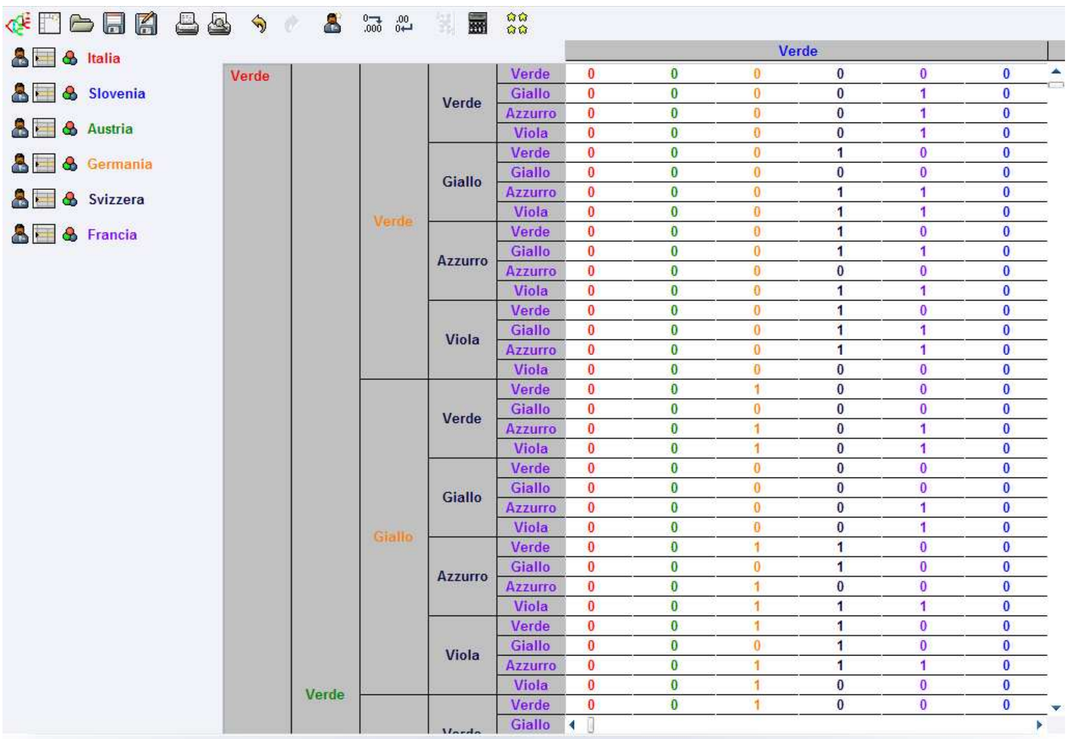
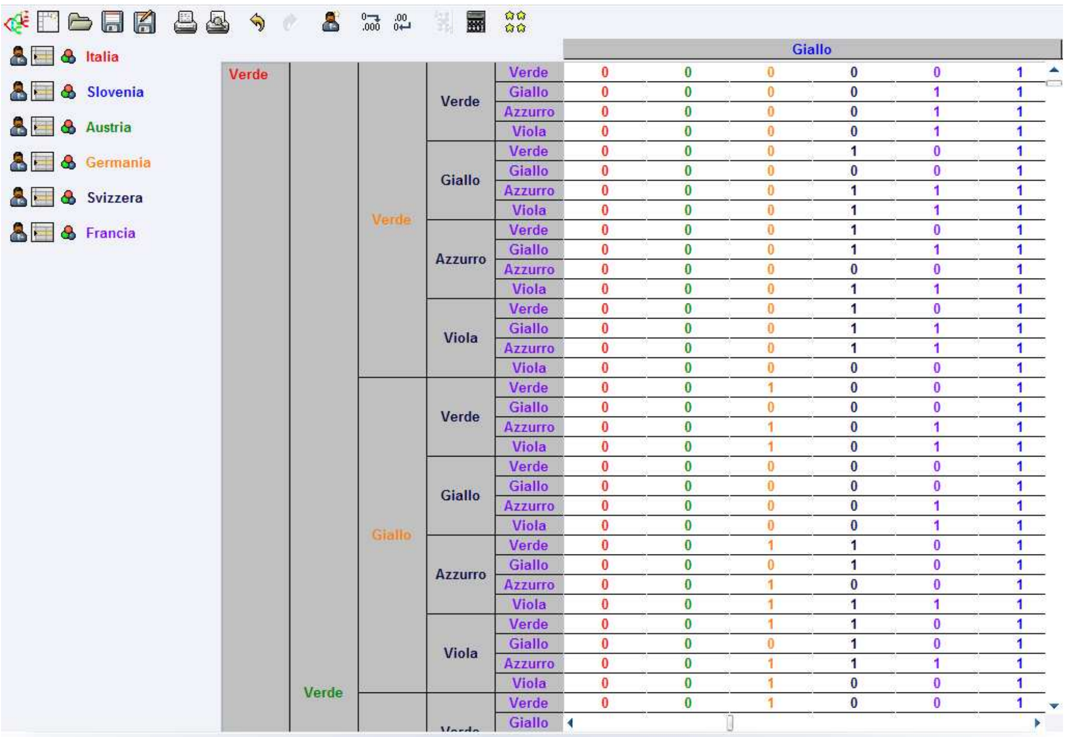
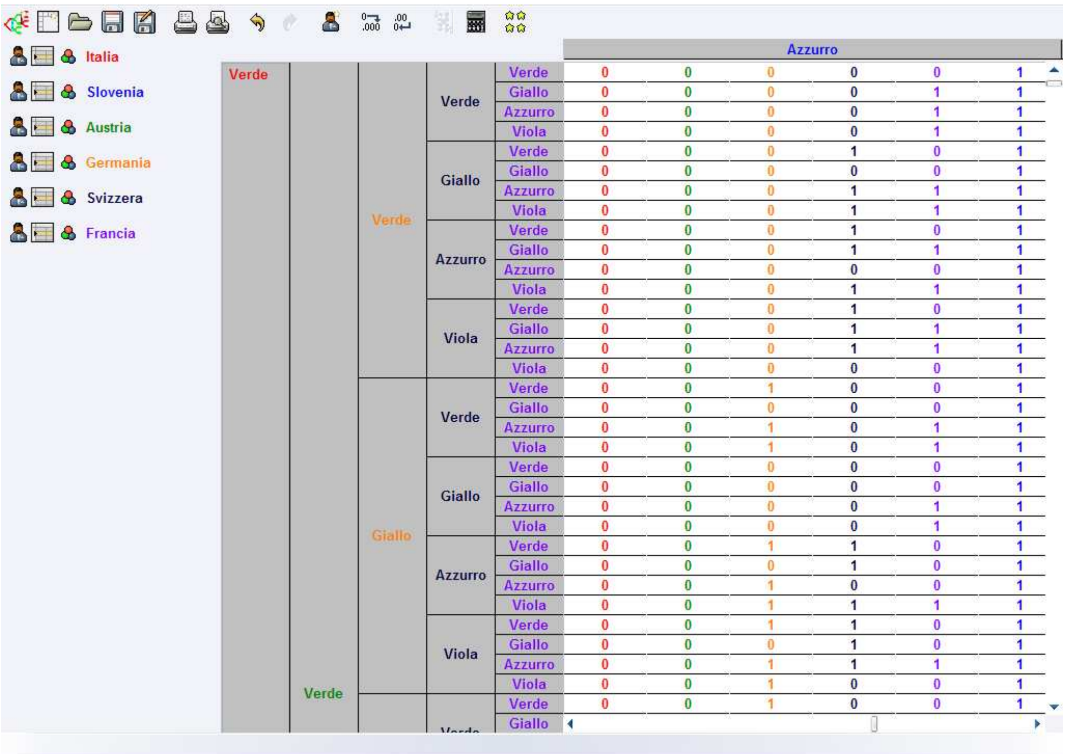


Figura 4.8: parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore verde.



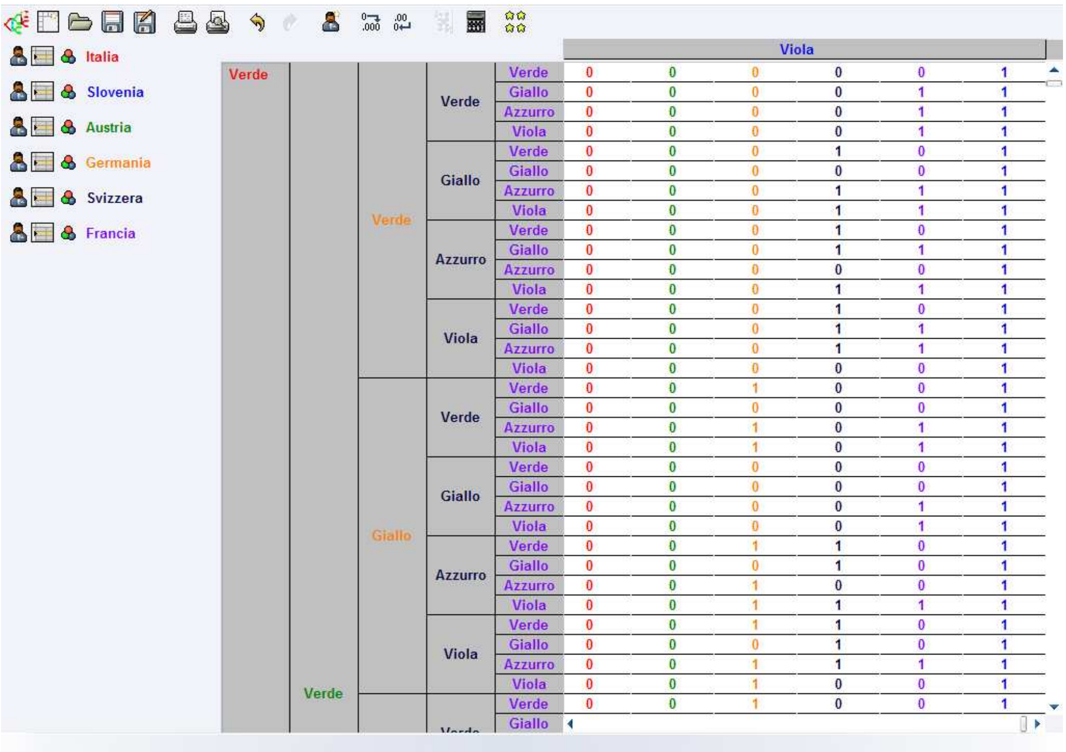
					Giallo					
Verde	Verde	Verde	Verde	Verde	0	0	0	0	0	1
				Giallo	0	0	0	0	1	1
				Azzurro	0	0	0	0	1	1
				Viola	0	0	0	0	1	1
			Giallo	Verde	0	0	0	1	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	0	1	1	1
				Viola	0	0	0	1	1	1
			Azzurro	Verde	0	0	0	1	0	1
				Giallo	0	0	0	1	1	1
				Azzurro	0	0	0	0	0	1
				Viola	0	0	0	1	1	1
			Viola	Verde	0	0	0	1	0	1
				Giallo	0	0	0	1	1	1
				Azzurro	0	0	0	1	1	1
				Viola	0	0	0	0	0	1
		Giallo	Verde	Verde	0	0	1	0	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	1	0	1	1
				Viola	0	0	1	0	1	1
			Giallo	Verde	0	0	0	0	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	0	0	1	1
				Viola	0	0	0	0	1	1
			Azzurro	Verde	0	0	1	1	0	1
				Giallo	0	0	0	1	0	1
				Azzurro	0	0	1	0	0	1
				Viola	0	0	1	1	1	1
			Viola	Verde	0	0	1	1	0	1
				Giallo	0	0	0	1	0	1
				Azzurro	0	0	1	1	1	1
				Viola	0	0	1	0	0	1
			Verde	Verde	0	0	1	0	0	1
				Giallo						

Figura 4.9: parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore giallo.



					Azzurro					
Verde	Verde	Verde	Verde	Verde	0	0	0	0	0	1
				Giallo	0	0	0	0	1	1
				Azzurro	0	0	0	0	1	1
				Viola	0	0	0	0	1	1
			Giallo	Verde	0	0	0	1	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	0	1	1	1
				Viola	0	0	0	1	1	1
			Azzurro	Verde	0	0	0	1	0	1
				Giallo	0	0	0	1	1	1
				Azzurro	0	0	0	0	0	1
				Viola	0	0	0	1	1	1
			Viola	Verde	0	0	0	1	0	1
				Giallo	0	0	0	1	1	1
				Azzurro	0	0	0	1	1	1
				Viola	0	0	0	0	0	1
		Giallo	Verde	Verde	0	0	1	0	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	1	0	1	1
				Viola	0	0	1	0	1	1
			Giallo	Verde	0	0	0	0	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	0	0	1	1
				Viola	0	0	0	0	1	1
			Azzurro	Verde	0	0	1	1	0	1
				Giallo	0	0	0	1	0	1
				Azzurro	0	0	1	0	0	1
				Viola	0	0	1	1	1	1
			Viola	Verde	0	0	1	1	0	1
				Giallo	0	0	0	1	0	1
				Azzurro	0	0	1	1	1	1
				Viola	0	0	1	0	0	1
				Verde	0	0	1	0	0	1
				Giallo						

Figura 4.10: parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore azzurro.



					Viola					
Verde	Verde	Verde	Verde	Verde	0	0	0	0	0	1
				Giallo	0	0	0	0	1	1
				Azzurro	0	0	0	0	1	1
				Viola	0	0	0	0	1	1
			Giallo	Verde	0	0	0	1	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	0	1	1	1
				Viola	0	0	0	1	1	1
			Azzurro	Verde	0	0	0	1	0	1
				Giallo	0	0	0	1	1	1
				Azzurro	0	0	0	0	0	1
				Viola	0	0	0	1	1	1
			Viola	Verde	0	0	0	1	0	1
				Giallo	0	0	0	1	1	1
				Azzurro	0	0	0	1	1	1
				Viola	0	0	0	0	0	1
		Giallo	Verde	Verde	0	0	1	0	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	1	0	1	1
				Viola	0	0	1	0	1	1
			Giallo	Verde	0	0	0	0	0	1
				Giallo	0	0	0	0	0	1
				Azzurro	0	0	0	0	1	1
				Viola	0	0	0	0	1	1
			Azzurro	Verde	0	0	1	1	0	1
				Giallo	0	0	0	1	0	1
				Azzurro	0	0	1	0	0	1
				Viola	0	0	1	1	1	1
			Viola	Verde	0	0	1	1	0	1
				Giallo	0	0	0	1	0	1
				Azzurro	0	0	1	1	1	1
				Viola	0	0	1	0	0	1
				Verde	0	0	1	0	0	1
				Giallo						

Figura 4.11: parte iniziale della tabella creata in gambit in cui la Slovenia assume il colore viola.

Dalle figure appena riportate si può notare che gambit associa a ciascuna nazione (giocatore) un colore differente utilizzato sia per rappresentare il payoff sia la strategia scelta da ciascun giocatore. Infatti è associato:

- alla nazione Italia il colore rosso;
- alla nazione Slovenia il colore blu;
- alla nazione Austria il colore verde;
- alla nazione Germania il colore arancio;
- alla nazione Svizzera il colore nero;
- alla nazione Slovenia il colore lilla;

Inoltre l'utilità associata ad una nazione (ad esempio l'Italia) in riferimento ad una determinata combinazione di colori assume il valore 1 se tutti i vincoli riferiti alle sole nazioni confinanti con la nazione presa in considerazione (l'Italia) sono rispettati.
Dopo aver costruito la tabella, attraverso gambit si sono calcolati gli equilibri di Nash tra i quali si evidenziano gli equilibri corrispondenti alle soluzioni del CSP: differentemente dagli altri in questi equilibri ad ogni nazione è associato il payoff 1 (quindi tutti i vincoli sono rispettati). Nella figura 4.12 sono riportati soltanto gli equilibri corrispondenti alle soluzioni del CSP.

Soluzioni CSP			
Italia=verde	**Italia=giallo**	**Italia=azzurro**	**Italia=viola**
<VE,G,A,VE,G,A>	<G,VE,A,VE,VI,A>	<A,VE,G,VE,VI,G>	<VI,VE,G,VE,A,G>
<VE,G,A,VE,G,VI>	<G,VE,A,G,VE,A>	<A,VE,G,A,VE,G>	<VI,VE,G,A,VE,G>
<VE,G,A,VE,VI,G>	<G,VE,A,G,VE,VI>	<A,VE,G,A,VE,VI>	<VI,VE,G,VI,VE,G>
<VE,G,A,VE,VI,A>	<G,VE,A,G,VI,VE>	<A,VE,G,A,VI,VE>	<VI,VE,G,VI,VE,A>
<VE,G,A,G,VI,A>	<G,VE,A,G,VI,A>	<A,VE,G,A,VI,G>	<VI,VE,G,VI,A,VE>
<VE,G,A,VI,G,A>	<G,VE,A,VI,VE,A>	<A,VE,G,VI,VE,G>	<VI,VE,G,VI,A,G>
<VE,G,VI,VE,G,A>	<G,VE,VI,VE,A,VI>	<A,VE,VI,VE,G,VI>	<VI,VE,A,VE,G,A>
<VE,G,VI,VE,G,VI>	<G,VE,VI,G,VE,A>	<A,VE,VI,G,VE,VI>	<VI,VE,A,G,VE,A>
<VE,G,VI,VE,A,G>	<G,VE,VI,G,VE,VI>	<A,VE,VI,A,VE,G>	<VI,VE,A,VI,VE,G>
<VE,G,VI,VE,A,VI>	<G,VE,VI,G,A,VE>	<A,VE,VI,A,VE,VI>	<VI,VE,A,VI,VE,A>
<VE,G,VI,G,A,VI>	<G,VE,VI,G,A,VI>	<A,VE,VI,A,G,VE>	<VI,VE,A,VI,G,VE>
<VE,G,VI,A,G,VI>	<G,VE,VI,A,VE,VI>	<A,VE,VI,A,G,VI>	<VI,VE,A,VI,G,A>
<VE,A,G,VE,A,G>	<G,A,VE,G,A,VE>	<A,G,VE,G,VI,VE>	<VI,G,VE,G,A,VE>
<VE,A,G,VE,A,VI>	<G,A,VE,G,A,VI>	<A,G,VE,A,G,VE>	<VI,G,VE,A,G,VE>
<VE,A,G,VE,A,VI>	<G,A,VE,G,A,VI>	<A,G,VE,A,G,VE>	<VI,G,VE,A,G,VE>
<VE,A,G,VE,VI,G>	<G,A,VE,G,VI,VE>	<A,G,VE,A,G,VI>	<VI,G,VE,VI,G,VE>
<VE,A,G,VE,VI,A>	<G,A,VE,G,VI,A>	<A,G,VE,A,VI,VE>	<VI,G,VE,VI,G,A>
<VE,A,G,A,VI,G>	<G,A,VE,A,VI,VE>	<A,G,VE,A,VI,G>	<VI,G,VE,VI,A,VE>
<VE,A,G,VI,A,G>	<G,A,VI,VE,A,VI>	<A,G,VE,VI,G,VE>	<VI,G,VE,VI,A,G>
<VE,A,VI,VE,G,A>	<G,A,VI,G,VE,A>	<A,G,VI,VE,G,VI>	<VI,G,A,VE,G,A>
<VE,A,VI,VE,G,VI>	<G,A,VI,G,VE,VI>	<A,G,VI,G,VE,VI>	<VI,G,A,G,VE,A>
<VE,A,VI,VE,,A,VI>	<G,A,VI,G,A,VE>	<A,G,VI,A,VE,G>	<VI,G,A,VI,VE,G>
<VE,A,VI,VE,A,G>	<G,A,VI,G,A,VI>	<A,G,VI,A,VE,VI>	<VI,G,A,VI,VE,A>
<VE,A,VI,G,A,VI>	<G,A,VI,A,VE,VI>	<A,G,VI,A,G,VE>	<VI,G,A,VI,G,VE>
<VE,A,VI,A,G,VI>	<G,VI,VE,G,A,VE>	<A,G,VI,A,G,VI>	<VI,G,A,VI,G,A>
<VE,VI,G,VE,A,G>	<G,VI,VE,G,A,VI>	<A,VI,VE,G,VI,VE>	<VI,A,VE,G,A,VE>
<VE,VI,G,A,VI,G>	<G,VI,VE,G,VI,VE>	<A,VI,VE,A,G,VE>	<VI,A,VE,A,G,VE>
<VE,VI,G,VE,VI,G>	<G,VI,VE,G,VI,A>	<A,VI,VE,A,G,VI>	<VI,A,VE,VI,G,VE>
<VE,VI,G,VE,VI,A>	<G,VI,VE,A,VI,VE>	<A,VI,VE,A,VI,VE>	<VI,A,VE,VI,G,A>
<VE,VI,G,A,VI,G>	<G,VI,VE,VI,A,VE>	<A,VI,VE,A,VI,G>	<VI,A,VE,VI,A,VE>
<VE,VI,G,VI,A,G>	<G,VI,A,VE,VI,A>	<A,VI,VE,VI,G,VE>	<VI,A,VE,VI,A,G>
<VE,VI,A,VE,G,A>	<G,VI,A,G,VE,A>	<A,VI,G,VE,VI,G>	<VI,A,G,VE,A,G>
<VE,VI,A,VE,G,VI>	<G,VI,A,G,VE,VI>	<A,VI,G,A,VE,G>	<VI,A,G,A,VE,G>
<VE,VI,A,VE,VI,G>	<G,VI,A,G,VI,VE>	<A,VI,G,A,VE,VI>	<VI,A,G,VI,VE,G>
<VE,VI,A,VE,VI,A>	<G,VI,A,G,VI,A>	<A,VI,G,A,VI,VE>	<VI,A,G,VI,VE,A>
<VE,VI,A,G,VI,A>	<G,VI,A,VI,VE,A>	<A,VI,G,A,VI,G>	<VI,A,G,VI,A,VE>
<VE,VI,A,VI,G,A>	<G,A,VE,VI,A,VE>	<A,VI,G,VI,VE,G>	<VI,A,G,VI,A,G>

Figura 4.12: equilibri di Nash corrispondenti alle soluzioni del CSP della figura 4.7

Capitolo 5

Conclusioni

5.1 Principali risultati

I principali risultati raggiunti in questo lavoro:

– il mapping per la trasformazione di un SCSP in gioco non cooperativo statico ad informazione completa;

– il teorema 4.1.2 che evidenzia la relazione tra le soluzioni dei SCSP e le soluzioni del corrispondente gioco.

sono contenuti nel capitolo 4.
Il mapping rappresenta il "ponte" tra due diversi mondi di ricerca: la teoria dei giochi e i SCSP. Più precisamente due mondi di ricerca apparentemente distanti tra loro ma straordinariamente in relazione.
Il teorema rappresenta, appunto, la constatazione teorica dimostrata di questa coesistenza tra i due campi di ricerca tale da permettere di riversare le conoscenze della Teoria dei giochi nel campo di studi dei SCSP.
Infatti in base a ciò è stato possibile l'utilizzo di un software di gestione e ricerca degli equilibri di Nash, il *gambit*, per l'individuazione delle soluzioni dei SCSP.
Mi auguro che questa "scoperta teorica" sia l'inizio di un più ampio rapporto collaborativo tra i due campi di studio al fine di ottenere risultati rivoluzionari utili per entrambi.

5.2 Future works

In base alle conoscenze acquisite ed in base agli obiettivi che si intendono raggiungere, ritengo di proporre una possibile direzione per future work: questa potrebbe essere diretta all'individuazione di una più immediata relazione

tra i SCSP e i giochi non cooperativi statici ad informazione completa.
Dato che il teorema elaborato non risulta valido nella relazione contraria, ossia non tutti gli equilibri di Nash rappresentano soluzioni dei SCSP, si potrebbe pensare all'elaborazione di un mapping o di un teorema, laddove fosse possibile, che comprenda questa corrispondenza contraria.
Così da poter individuare immediatamente le soluzioni dei SCSP che in tal caso corrisponderebbero a tutti gli equilibri di Nash.
Ciò potrebbe tradursi in un risparmio notevole di tempo nella ricerca delle soluzioni.

Bibliografia

[1] Roman Barták. Constraint programming: In pursuit of the holy grail.

[2] Roman Barták. *Guide to Constraint Programming*, 1998.

[3] Antonio Bernardo. Appunti di teoria dei giochi per il corso di modelli matematici per le scienze sociali, febbrary 2006.

[4] Stefano Bistarelli. *Semirings for Soft Constraint Solving and Programming (LECTURE NOTES IN COMPUTER SCIENCE)*. SpringerVerlag, 2004.

[5] Stefano Bistarelli and Fabio Gadducci. Enhancing constraints manipulation in semiring-based formalisms.

[6] M. M. Flood. *Some experimental games*. The Rand corporation, Santa Monica, CA, first edition edition, 1952.

[7] Robert Gibbon. *Teoria dei giochi*. Prentice Hall International.

[8] Giorgio Ingargiola. Constraint satisfaction problem. Introductory course in Artificial Intelligence for graduate computing students at Temple University, Room 1038, CIS Department, Temple University, Philadelphia.

[9] R. Duncan Luce and Howard Raiffa. *Games and decisions: introduction and critical survay*. Courier Dover Publications, 1989.

[10] Guillermo Owen. *GAME THEORY*. ACADEMIC PRESS, third edition edition, 1995.

[11] Fioravante Patrone. Aspetti formali delle teorie della contrattazione di nash e kalai-smorodinsky.

[12] Schelling. *Strategy of conflict*. Library of Congress, United States of America, 17 edition, 1999.

[13] John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1947.