



Criterion - Algoritmo di Assegnazione Automatica

1	INTRODUZIONE	1
2	FORMULAZIONE DEL PROBLEMA	1
3	IL PROBLEMA DELL'ASSEGNAZIONE	3
4	UN ESEMPIO DI SELEZIONE SEMPLIFICATA	5
5	LA SOLUZIONE DEL PROBLEMA DI SELEZIONE DEI PRODOTTI PER LA VQR3	6
5.1	FASE 1: GENERAZIONE DELL'ASSEGNAZIONE INIZIALE	7
5.2	FASE 2: RIDUZIONE ED OTTIMIZZAZIONE DELL'ASSEGNAZIONE	7
6	VALUTAZIONE DEI RISULTATI.....	9

Versione	Data	Descrizione
1.0	14 luglio 2023	Versione iniziale.

1 Introduzione

Per supportare il processo di selezione dei prodotti da conferire nell'ambito della VQR, CRITERIUM fornisce un algoritmo che suggerisce le selezioni più promettenti per l'Istituzione. L'algoritmo genera suggerimenti che hanno due caratteristiche:

- rispettano la regola della VQR, in termini di numero di prodotti da conferire per ciascuna struttura dipartimentale, numero minimo e massimo di prodotti da conferire per ciascun soggetto valutato, numero massimo di copie dei prodotti conferibili nell'ambito dell'Istituzione;
- massimizzano il valore per l'Istituzione di una funzione obiettivo, detta "funzione di vantaggio", discussa in dettaglio nel manuale d'uso del sistema.

Questa sezione descrive l'algoritmo utilizzato dal sistema per suggerire le assegnazioni. Dal momento che l'Istituzione può configurare liberamente i parametri della funzione di vantaggio, considereremo una generica funzione V che, dato un prodotto P ed un soggetto valutato SV , restituisce un numero reale $V(P, SV)$ che misura il vantaggio del conferimento del prodotto P da parte del soggetto valutato SV .

2 Formulazione del Problema

Supponiamo che l'Istituzione che seleziona i prodotti sia composto di numero N_{sv} di soggetti valutati, con $N_{sv} > 1$. Nel seguito indicheremo i soggetti valutati (ovvero i soggetti accreditati per la VQR) con i simboli $SV_1, SV_2, \dots, SV_{N_{sv}}$. Per ciascun soggetto è necessario selezionare un numero di prodotti compreso tra 0 e 4.

Nel seguito, indichiamo con P_1, P_2, \dots, P_{N_p} gli N_p prodotti valutati di cui i soggetti dell'Istituzione sono coautori.



È utile pensare al problema della selezione dei prodotti come al problema di “riempire un certo numero di posizioni (slot)” per ciascun soggetto, selezionando per ciascuno slot uno dei prodotti valutati di cui il soggetto è coautore.

Chiameremo **assegnazione** l’operazione che corrisponde a “riempire” uno degli slot relativi al soggetto valutato SV con il prodotto P di cui SV è coautore.

Data l’assegnazione $A = (P, SV)$ del prodotto P al soggetto valutato SV coautore di P, indicheremo con $V_A = V(P, SV)$ il valore del vantaggio dell’assegnazione A.

L’algoritmo genera *suggerimenti di assegnazione*. Un **suggerimento di assegnazione** è una collezione di assegnazioni **consistenti**, ovvero una collezione di assegnazioni che rispettano le regole del bando:¹

1. ciascun soggetto conferisce tra 0 e 4 prodotti;
2. ciascun soggetto conferisce esclusivamente prodotti di cui è coautore;
3. per ciascuna struttura dipartimentale D è necessario conferire al più un numero fissato di prodotti N_D , pari al più a 3 per il numero di soggetti valutati che afferiscono alla struttura;
4. ciascun prodotto può essere conferito solo una volta nell’ambito di ciascuna struttura dipartimentale;
5. i prodotti con un numero di coautori inferiore a 6 possono essere conferiti al più 2 volte dall’Istituzione, da strutture dipartimentali distinte;
6. i prodotti con un numero di coautori pari o superiore a 6 possono essere conferiti al più 3 volte dall’Istituzione, da strutture dipartimentali distinte.

Dato un suggerimento di assegnazione S indichiamo con V_S il *vantaggio totale dell’Istituzione relativo alle selezioni suggerite in S*, ovvero la somma dei vantaggi delle singole assegnazioni suggerite in S.

Il problema della selezione dei prodotti è quindi un problema di ottimizzazione: consiste nell’individuare i suggerimenti di assegnazione che massimizzano il vantaggio totale per l’Istituzione.

Si noti che discutiamo la soluzione del problema nel caso tipico, ovvero nell’ipotesi che:

- in almeno una struttura dipartimentale il numero di prodotti valutati N_p sia strettamente maggiore del numero di slot da riempire;
- esista almeno un prodotto valutato di cui due o più dei soggetti dell’Istituzione sono coautori.

Se le due ipotesi riportate non sopra sono rispettate, il problema presenta una soluzione banale. Viceversa, ci troviamo di fronte ad un problema di ottimizzazione combinatoria, che definiamo **problema della selezione dei prodotti per la VQR3**, che possiamo formalizzare come segue:

“Dati i seguenti dati in ingresso:

- le strutture dipartimentali D_1, D_2, \dots, D_{Nd} dell’Istituzione;

¹ Si noti che, oltre a quelle discusse nel documento, le regole della VQR3 prevedono anche che alcuni tipi di monografia possono essere conferiti come “prodotto doppio”, ovvero in modo da occupare due slot. Per contenere la complessità, l’algoritmo di assegnazione suggerisce sempre e solo conferimenti che occupano un unico slot. I selezionatori possono intervenire per assegnare manualmente ad un prodotto il doppio slot. Di conseguenza questo aspetto non è ulteriormente discusso.



- la collezione dei soggetti valutati $SV_1, SV_2, \dots, SV_{N_{sv}}$, e la loro afferenza alle strutture dipartimentali dell'Istituzione;
- la collezione dei prodotti valutati P_1, P_2, \dots, P_{N_p} ;
- il numero di assegnazioni (slot) attese per ciascuna struttura dipartimentale $S_{D1}, S_{D2}, \dots, S_{D_{Nd}}$
- la funzione di vantaggio V

individuare il suggerimento di assegnazione che massimizza il vantaggio totale per l'Istituzione.”

Sottolineiamo due aspetti importanti:

- a. Il problema risolto dal sistema a disposizione degli Atenei è ancora più complesso, perché prevede una modalità di lavoro cooperativa, in cui l'algoritmo riceve in ingresso anche eventuali selezioni manuali effettuate dai selezionatori individuati dall'Istituzione. Per evitare di complicare ulteriormente la trattazione nel seguito non faremo ulteriormente riferimento a questo aspetto, e ci limitiamo a dire che le assegnazioni manuali effettuate dai selezionatori vengono assunte come dati immutabili, e recepite tal quali.
- b. Il problema della selezione dei prodotti per la VQR3 è diverso dal corrispondente problema di assegnazione per la VQR 2011-2014 (la VQR2). Nel caso della VQR precedente, infatti, per ciascun soggetto accreditato era fissato a priori il numero di prodotti da conferire (nella VQR2 tipicamente 2, e 1 o 0 nei casi particolari), e quindi il numero di slot da riempire. Chiameremo **problema della selezione dei prodotti semplificato** questa variante del problema in cui per ciascun soggetto è disponibile l'informazione aggiuntiva relativa al numero di slot da riempire. Il termine semplificato intende sottolineare il fatto che la complessità del problema della selezione per la VQR3 è molto più elevata rispetto a quella del problema della selezione per la VQR2. Intuitivamente, infatti, l'algoritmo non deve solo individuare quali sono i prodotti migliori per riempire gli slot, ma anche qual è il numero di slot per ciascun soggetto che massimizza il vantaggio.

3 Il Problema dell'Assegnazione

Entrambe le istanze del problema della selezione dei prodotti individuate nel paragrafo precedente sono esempi di **problemi di assegnazione non bilanciati** (“unbalanced assignment problem”)² con vincoli aggiuntivi, un caso particolare del più noto **problema di assegnazione** (“assignment problem”)³.

Il problema dell'assegnazione nella sua formulazione classica è un problema di ottimizzazione combinatoria notevole che è possibile risolvere in tempo polinomiale. Il problema può essere formulato come segue:

“Dato un numero n di agenti (“agents”) e di compiti da svolgere (“task”), supponiamo che ciascun agente possa eseguire ognuno dei compiti, e che assegnare all'agente il compito j abbia un costo pari a $c_{ij} > 0$. Assegnare a ciascun agente esattamente un compito in modo da minimizzare il costo complessivo dell'assegnazione”.

² <http://statistics-assignment.com/unbalanced-assignments>

³ https://en.wikipedia.org/wiki/Assignment_problem



Il problema può essere formalizzato come un problema su grafi, supponendo di avere un grafo bipartito completo G con n vertici per gli agenti, ed n vertici per i task, e ciascun arco dall'agente i al task j etichettato con il costo c_{ij} .

Trascurando i dettagli tecnici (per i quali si rimanda ai riferimenti a piè di pagina), osserviamo che il problema coincide con quello di trovare un accoppiamento perfetto ("perfect matching") di costo minimo. Per la soluzione del problema è possibile utilizzare il noto **algoritmo ungherese** ("hungarian algorithm")⁴, del quale esistono varie implementazioni disponibili in rete.

È noto che il problema della selezione dei prodotti semplificato (ovvero la variante semplificata valida per la VQR2) può essere ridotto ad un problema di assegnazione in cui gli agenti sono i prodotti e i task sono gli slot da riempire. **Nel seguito discutiamo questa riduzione nel caso della VQR2, ovvero nel caso in cui il numero di slot per ciascun soggetto sia pari a 2.** Sottolineiamo che la VQR3 è basata su un contesto di regole diverso, la cui gestione verrà discussa in dettaglio nelle sezioni successive.

Dal momento che, come detto, il problema della selezione dei prodotti semplificato è riducibile ad un problema di assegnazione, non è possibile utilizzare direttamente le tecniche algoritmiche pensate per il problema dell'assegnazione. Questo per due complicazioni:

1. Per ipotesi, il numero di prodotti N_p è superiore al numero di slot, $2 * N_{sv}$.
2. Il problema presenta il vincolo aggiuntivo per cui non è possibile assegnare ciascun prodotto a ciascuno degli slot. Infatti, agli slot relativi al soggetto SV è possibile assegnare esclusivamente prodotti di cui SV è coautore.

In altri termini, nella costruzione del grafo bipartito relativo ad un problema di selezione dei prodotti il numero di vertici nella parte sinistra, quelli che rappresentano i prodotti, è maggiore del numero di vertici nella parte destra, che rappresentano gli slot, e in aggiunta il grafo non è completo.

È possibile però utilizzare tecniche standard⁵ per ridurre un problema di assegnazione non bilanciato con vincoli ad un problema di assegnazione standard. Nel nostro caso, il problema viene costruito come segue:

- Ogni prodotto diventa un agente, ovvero un vertice nella parte sinistra del grafo.
- Ogni slot diventa un task, ovvero un vertice nella parte destra del grafo.
- Per ciascun prodotto P_k del soggetto valutato SV_i con vantaggio $V(P_k, SV_i)$ si definisce il costo dell'assegnazione di P_k a SV_i come $maxV - V(P_k, SV_i)$, dove $maxV$ è il valore massimo di vantaggio tra le possibili assegnazioni dell'Istituzione.
- Per ciascun prodotto P_k del soggetto SV_i si aggiunge un arco incidente nei due slot relativi al soggetto, $S_{i,1}, S_{i,2}$, con costo pari al costo dell'assegnazione di P_k a SV_i .
- Per eliminare la complicazione n. 1 descritto sopra, si aggiunge un numero di slot fantoccio ("dummy") pari a $N_p - 2 * N_{sv}$, in modo da equiparare il numero di vertici sulla sinistra a quelli sulla destra. Per ciascun prodotto, si aggiunge un arco incidente in ciascuno slot fantoccio con costo superiore al costo massimo dei prodotti ordinari. Intuitivamente, tutti i prodotti sono associabili a questi slot, ma l'algoritmo preferirà ogni volta che è possibile prodotti ordinari.

⁴ https://en.wikipedia.org/wiki/Hungarian_algorithm

⁵ <https://books.google.it/books?id=mNUjTXLuBS4C&pg=PA35&lpg=PA35&dq=unbalanced+hungarian+method>

- Per eliminare la complicazione n. 2, si completa il grafo aggiungendo archi di costo molto maggiore del costo massimo degli altri archi (in questo caso, molto maggiore di $maxV$). Intuitivamente, questo evita che queste specifiche assegnazioni siano utilizzate per costruire l'assegnazione di costo minimo.

La tecnica di costruzione del grafo bipartito è esemplificata nella sezione seguente.

Il problema ottenuto è un problema di assegnazione standard, per il quale è possibile utilizzare l'algoritmo ungherese. Dalla soluzione del problema standard è facilmente possibile ottenere una soluzione del problema di assegnazione dei prodotti semplicemente ignorando le assegnazioni relative agli slot fantoccio.

4 Un Esempio di Selezione Semplificata

Consideriamo, a puro titolo di esempio, un gruppo di quattro soggetti chiamati Rossi (SV_1), Bianchi (SV_2), Verdi (SV_3) e Neri (SV_4). Per ciascuno dei quattro è necessario selezionare 2 dei prodotti valutati. Supponiamo anche che i prodotti valutati siano i seguenti, e che il vantaggio non dipenda dal coautore a cui viene assegnato il prodotto (la dizione "et al" viene utilizzata per fare riferimento a coautori che non appartengono all'Istituzione):

- P_1 : Rossi, Verdi et al, vantaggio 1
- P_2 : Rossi, Bianchi et al, vantaggio 0,7
- P_3 : Rossi et al, vantaggio 0,4
- P_4 : Verdi, Bianchi et al, vantaggio 0,4
- P_5 : Verdi et al, vantaggio 0,1
- P_6 : Verdi et al, vantaggio 0
- P_7 : Bianchi, Rossi et al, vantaggio 0,4
- P_8 : Neri et al, vantaggio 0,1
- P_9 : prodotto non conferito da Neri, considerato come prodotto virtuale con vantaggio 0 (necessario perché Neri è autore di un unico prodotto, e non può riempire entrambi gli slot)

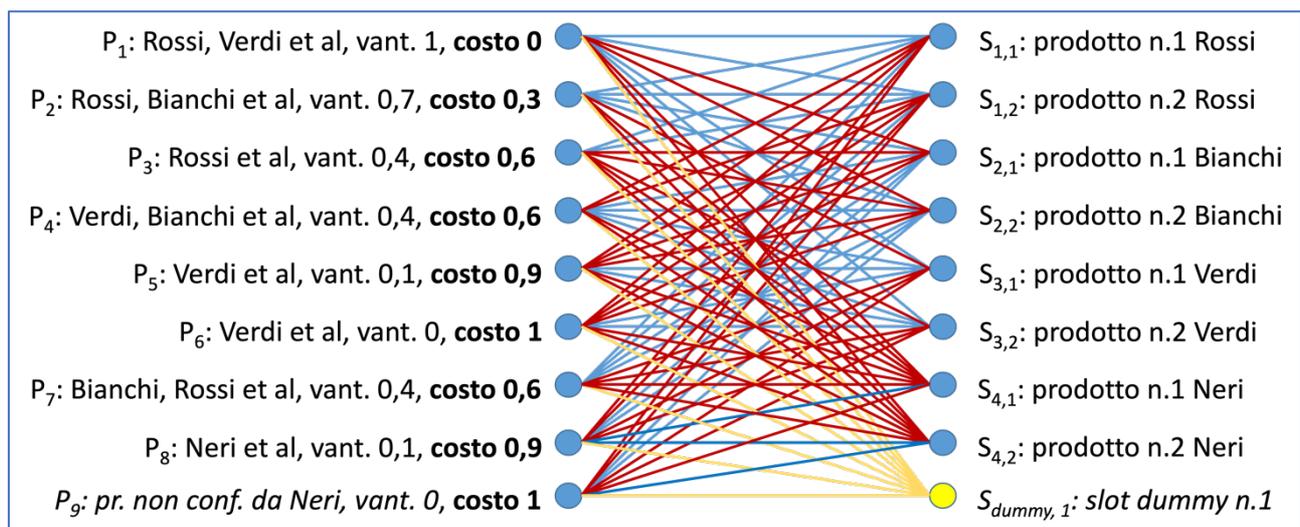


Figura 1: Esempio di grafo bipartito per il problema dell'assegnazione

Il grafo bipartito costruito secondo la tecnica descritta nella sezione precedente è rappresentato nella **Figura 1**.

Gli archi di colore azzurro uniscono i prodotti agli slot dei corrispondenti autori. Gli archi di colore rosso agli slot di soggetti che non sono autori del prodotto e sono etichettati con un costo molto superiore agli altri, pari a 100 in questo esempio. Gli archi di colore giallo uscenti dai nodi che corrispondono a slot “dummy” sono etichettati con costo 10, un costo superiore al costo massimo dell’assegnazione di un prodotto ordinario. La matrice di adiacenza relativa al grafo in figura è riportata di seguito:

	S _{1,1}	S _{1,2}	S _{2,1}	S _{2,2}	S _{3,1}	S _{3,2}	S _{4,1}	S _{4,2}	S _{d,1}
P ₁	0	0	100	100	0	0	100	100	10
P ₂	0,3	0,3	0,3	0,3	100	100	100	100	10
P ₃	0,6	0,6	100	100	100	100	100	100	10
P ₄	100	100	0,6	0,6	0,6	0,6	100	100	10
P ₅	100	100	100	100	0,9	0,9	100	100	10
P ₆	100	100	100	100	1	1	100	100	10
P ₇	0,6	0,6	0,6	0,6	100	100	100	100	10
P ₈	100	100	100	100	100	100	0,9	0,9	10
P ₉	100	100	100	100	100	100	1	1	10

5 La Soluzione del Problema di Selezione dei Prodotti per la VQR3

Abbiamo quindi appreso che il problema di selezione dei prodotti semplificato può essere efficacemente risolto attraverso una riduzione al problema classico di assegnazione e l’utilizzo dell’algoritmo ungherese.

Come discusso nelle sezioni precedenti questo non è vero per il problema della selezione dei prodotti per la VQR3, che comporta l’ulteriore complicazione di dover scegliere non solo i prodotti ma anche il numero di slot da riempire per ciascun soggetto.

In questa sezione viene descritto l’algoritmo implementato nel sistema per risolvere questa variante più complessa del problema. L’algoritmo è una combinazione di due tecniche algoritmiche:

- **Fase 1 – Assegnazione Iniziale:** nella prima fase viene costruito e risolto un problema di assegnazione semplificato in cui per tutti i soggetti vengono riempiti 4 slot, cioè il numero massimo previsto dal bando della VQR.
- **Fase 2 – Riduzione ed Ottimizzazione dell’Assegnazione:** nella seconda fase viene utilizzata una tecnica algoritmica di tipo “**tabu search**”⁶ per correggere l’assegnazione ottenuta al passo precedente e generare il suggerimento di assegnazione.

Le due fasi sono descritte nei paragrafi che seguono.

⁶ https://en.wikipedia.org/wiki/Tabu_search



5.1 Fase 1: Generazione dell'Assegnazione Iniziale

La fase di generazione dell'assegnazione iniziale risolve il problema della selezione semplificato con numero di prodotti da conferire per ciascun soggetto pari a 4. Si noti che le assegnazioni generate in questa fase NON sono consistenti, dal momento che non rispettano le regole relative alla numerosità dei prodotti da conferire. L'assegnazione di 4 prodotti per ciascun soggetto, infatti, supera certamente il numero massimo di assegnazioni per ciascuna struttura dipartimentale, pari a 3 per il numero di soggetti valutati afferenti.

Nonostante questo fatto, però, questa fase è preziosa, per due ragioni:

- può essere risolta in modo efficiente attraverso un'implementazione dell'algoritmo ungherese;
- consente di individuare un soprainsieme di assegnazioni, che chiameremo **assegnazioni candidate**, su cui concentrarsi nei passi successivi; queste assegnazioni, pari a 4 per il numero di soggetti dell'Istituzione, sono comunque molto meno numerose di tutte quelle possibili nell'Istituzione, che per Atenei numerosi possono essere molte migliaia.

Trascurando quindi la regola relativa alla numerosità delle assegnazioni per struttura (e quindi per l'Istituzione) che sarà oggetto della fase successiva, il risultato di questa fase viene costruito in modo da rispettare tutte le altre regole del bando.

Al fine di rispettare le regole n.5 e n.6 sul numero massimo di copie dei prodotti conferibili da strutture diverse, il sistema partiziona i prodotti assegnabili per struttura dipartimentale, creando copie distinte dei prodotti in modo che le assegnazioni generate al termine di questa fase rispettino le regole. In particolare:

- per i prodotti i cui coautori dell'Istituzione appartengono ad un'unica struttura dipartimentale, viene considerata un'unica copia, conferibile esclusivamente come parte delle assegnazioni della struttura;
- per i prodotti i cui coautori dell'Istituzione appartengono a strutture dipartimentali distinte, vengono create 2 o 3 copie, a seconda del numero di coautori; più precisamente, detto N il numero di strutture distinte a cui appartengono i coautori del prodotto dell'Istituzione:
 - se N è minore o uguale del numero di copie consentite (2 o 3), vengono create N copie, ciascuna conferibile esclusivamente nell'ambito di una delle strutture;
 - se N è maggiore del numero di copie consentite, alcune delle copie vengono considerate conferibili da più strutture, e sarà l'algoritmo ungherese a decidere in quale delle strutture conferire la copia.

5.2 Fase 2: Riduzione ed Ottimizzazione dell'Assegnazione

L'obiettivo della Fase 2 è correggere l'assegnazione iniziale generata nel corso della Fase 1 per generare un suggerimento di assegnazione, ovvero una nuova assegnazione consistente con le regole del bando.

Come primi passi, la procedura di Fase 2 effettua due operazioni.

La prima operazione è la **riduzione delle assegnazioni** generate dalla Fase 1. Per ciascuna struttura dipartimentale D , le assegnazioni di Fase 1 vengono ordinate in ordine decrescente di vantaggio e ridotte fino a N_D , cioè fino al numero di slot da conferire, considerando quelle di vantaggio maggiore.



Il suggerimento di assegnazione ottenuto in questo modo per l'Istituzione rispetta le regole del bando. Si noti però che, a differenza di quanto intuitivamente si possa pensare, nella stragrande maggioranza dei casi NON rappresenta la soluzione ottima. La ragione di questo fatto è che l'algoritmo ungherese utilizzato nella Fase 1, che ha come obiettivo riempire 4 slot per tutti i soggetti, potrebbe avere fatto scelte che riducendo il numero di slot non ottimizzano il vantaggio. Si consideri questo esempio, in cui supponiamo che l'algoritmo ungherese debba riempire solo 1 slot per ogni soggetto:

- il prodotto P ha due coautori SV_1, SV_2 ; i vantaggi sono diversi: $V(P, SV_1) = 10, V(P, SV_2) = 5$;
- contemporaneamente, il soggetto SV_1 ha un secondo prodotto P' , di vantaggio $V(P', SV_1) = 7$

È facile vedere che, durante la Fase 1, è estremamente probabile che l'algoritmo ungherese, che cerca di massimizzare il punteggio occupando tutti e 4 gli slot disponibili, suggerisca queste assegnazioni:

- $A_1(P, SV_2)$ con vantaggio $V_{A1} = 5$
- $A_2(P', SV_1)$ con vantaggio $V_{A2} = 7$

generando un vantaggio totale di 12. Se, però, durante la Fase 2 l'assegnazione del prodotto P al soggetto SV_2 venisse tagliata, il risultato non sarebbe più ottimo, perché invece del prodotto P' (con vantaggio 7) al soggetto SV_1 sarebbe possibile assegnare il prodotto P (con vantaggio 10).

Di conseguenza, l'operazione di riduzione delle assegnazioni di Fase 1 genera un suggerimento di assegnazione consistente, ma in generale questo non è quello ottimo.

L'obiettivo della Fase 2 è ottimizzare il vantaggio modificando questo suggerimento iniziale. Per farlo viene utilizzata una classica tecnica di ricerca in uno spazio di stati con strategia di tabu search. In particolare:

- gli stati sono rappresentati da assegnazioni per l'Istituzione; lo **stato iniziale** è quello ottenuto dalla riduzione delle assegnazioni di Fase 1, come discusso sopra;
- le transizioni tra stati sono rappresentate da operazioni che modificano l'assegnazione per l'Istituzione in due possibili modi:
 - eliminando l'assegnazione di un prodotto ad uno dei coautori;
 - aggiungendo l'assegnazione di un prodotto ad uno dei coautori.

Trattandosi, come più volte detto, di un problema di ottimizzazione combinatoria di alta complessità, è impossibile pensare di esplorare integralmente lo spazio degli stati per cercare la soluzione ottima. Sottolineiamo che l'algoritmo di assegnazione viene lanciato nel sistema centinaia di volte al giorno (varie volte per ciascuna Istituzione che utilizza il sistema), ed è indispensabile che restituisca i risultati nel giro di pochi secondi, al più pochi minuti per gli Atenei più grandi.

Di conseguenza, vengono adottate alcune euristiche che abbassano drasticamente la complessità della ricerca.

Per cominciare, il sistema individua l'insieme delle **operazioni candidate**, cioè tutte le possibili modifiche che è possibile apportare ad uno stato per generarne uno nuovo; per farlo procede in questo modo:

1. considera tutti i prodotti assegnati nella Fase 1;



2. genera tutte le possibili assegnazioni di questi prodotti, cioè le assegnazioni di ciascuno dei prodotti a tutti i coautori dell'Istituzione;
3. per ciascuna assegnazione (P, SV) individuata al passo precedente genera:
 - a. l'operazione di aggiunta dell'assegnazione di P a SV, purché il vantaggio $V(P, SV)$ sia maggiore di 0.
 - b. l'operazione di eliminazione dell'assegnazione di P a SV.

Si noti che non tutte le operazioni sono considerate applicabili a tutti gli stati. In altri termini, le operazioni hanno precondizioni:

- l'operazione che aggiunge l'assegnazione di P a SV è applicabile allo stato S solo se in S P non è assegnato a nessuno dei suoi coautori;
- analogamente, l'eliminazione dell'assegnazione di P a SV è applicabile allo stato S solo se in S P è effettivamente assegnato a SV.

È facile vedere che i due tipi di operazioni hanno effetti diversi sul vantaggio totale di uno stato. In particolare, le aggiunte aumentano sempre il vantaggio, le eliminazioni lo riducono. Questo è consentito nelle tecniche di tabu search. D'altro canto, per limitare i tempi di calcolo, la Fase 2 utilizza una forma estremamente limitata di tabu search, con queste regole:

- le eliminazioni possono diminuire temporaneamente il vantaggio, ma solo se l'operazione immediatamente successiva nella sequenza (che deve essere un'aggiunta) supera la perdita di vantaggio collegata all'eliminazione;
- le transizioni possono essere composte da un numero massimo K di operazioni. Sulla base di test di prestazioni, il valore di K configurato sul sistema in produzione è pari a 8.

Viene quindi considerata come **transizione ammissibile** una sequenza di operazioni di eliminazione/aggiunta di assegnazioni con le seguenti caratteristiche:

- la sequenza ha lunghezza pari al più a K;
- ogni eliminazione con perdita di vantaggio $V(P, SV)$ è seguita da un'aggiunta con vantaggio $V(P', SV') > V(P, SV)$;
- la transizione parte da uno stato consistente e genera uno stato consistente.

La Fase 2 esegue un algoritmo di punto fisso che parte dallo stato iniziale ed applica transizioni ammissibili per generare nuovi stati, finché raggiunge uno stato al quale non è possibile applicare ulteriori transizioni ammissibili. Questo stato finale viene restituito come suggerimento di assegnazione per l'Istituzione.

E' facile vedere che, per definizione di transizione ammissibile, la sequenza dei valori di vantaggio totale calcolati per gli stati generati dalla Fase 2 è monotona crescente. In altri termini, tutte le volte che la Fase 2 genera un nuovo stato, il vantaggio totale per l'Istituzione cresce.

6 Valutazione dei Risultati

Viste le euristiche adottate per abbassare la complessità, il suggerimento di assegnazione finale generato dall'algoritmo di assegnazione discusso in questo documento non è necessariamente l'ottimo globale.



Ci siamo quindi posti il problema di valutare la qualità dei suggerimenti generati dall’algoritmo, e per farlo abbiamo svolto numerosissimi test su configurazioni diverse di prodotti.

Ci teniamo a sottolineare che il problema di studiare la qualità dei risultati dell’algoritmo è di per sé un problema algoritmicamente non banale. Infatti:

- utilizzando configurazioni composte da pochi prodotti e pochi coautori – quelle per le quali con un algoritmo di forza bruta è possibile determinare l’assegnazione ottima – è praticamente impossibile fare emergere casi complessi in cui l’algoritmo non restituisce a sua volta l’assegnazione ottima;
- utilizzando configurazioni composte da migliaia di prodotti e coautori, simili a quelle che il sistema analizza in produzione, diventa impossibile individuare l’assegnazione ottima in tempi accettabili.

Di conseguenza, nei nostri test ci siamo concentrati esclusivamente su configurazioni complesse, e i test sono stati condotti senza conoscere la configurazione ottima. Piuttosto, l’assegnazione suggerita dall’algoritmo è stata confrontata con quella ottenibile da varianti dell’algoritmo in cui le varie euristiche sono rilassate – al prezzo di tempi di esecuzione significativamente più alti, ma con il vantaggio della copertura di una porzione più ampia dello spazio degli stati.

In oltre il 99% dei test condotti non sono stati osservati miglioramenti nel valore di vantaggio complessivo. Nei pochissimi casi in cui varianti dell’algoritmo hanno generato suggerimenti con valori di vantaggio più alto rispetto a quelli suggeriti dall’algoritmo, le differenze sono state di pochi punti decimali (su valori di vantaggio complessivo per l’Istituzione nell’ordine delle migliaia).