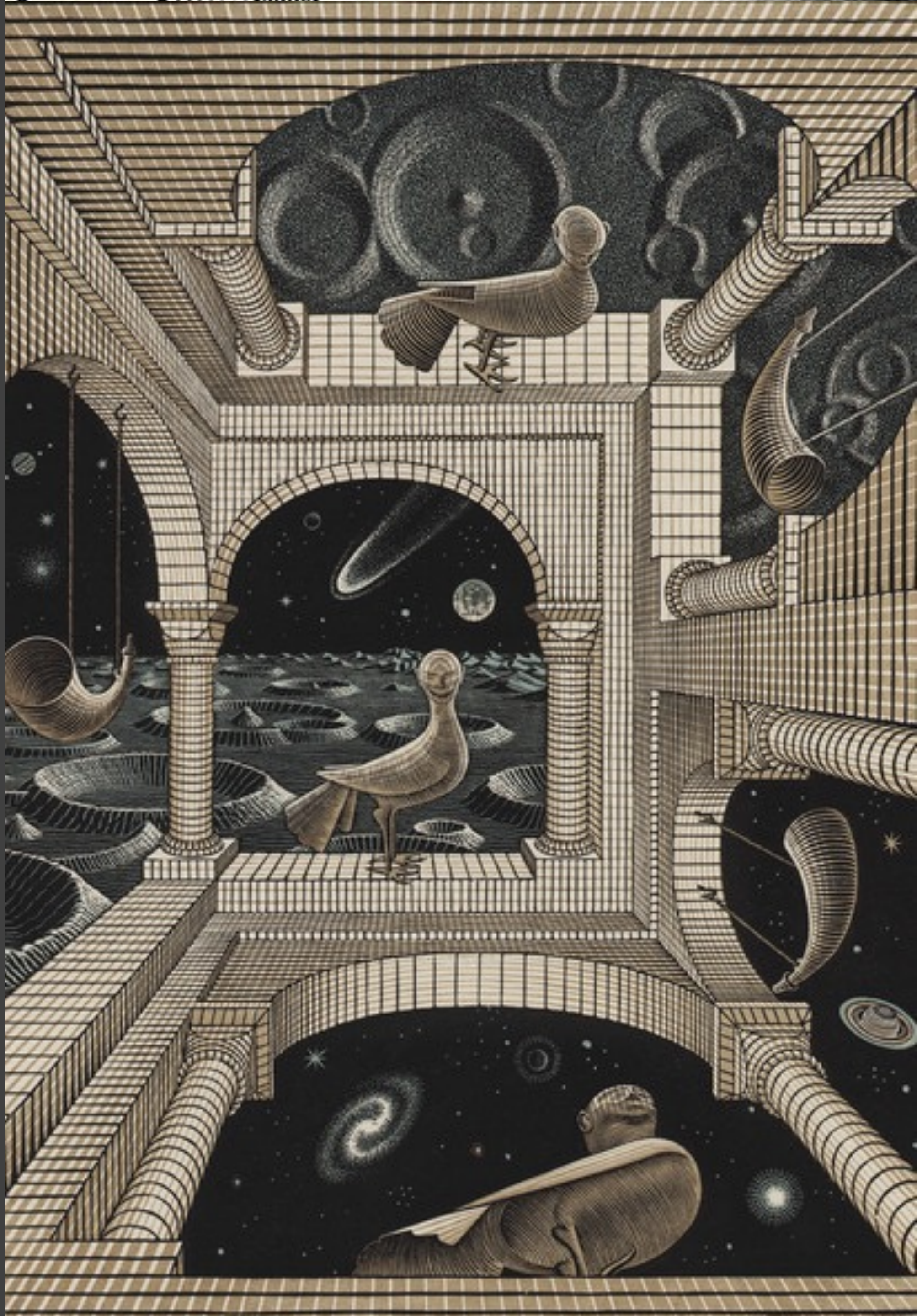
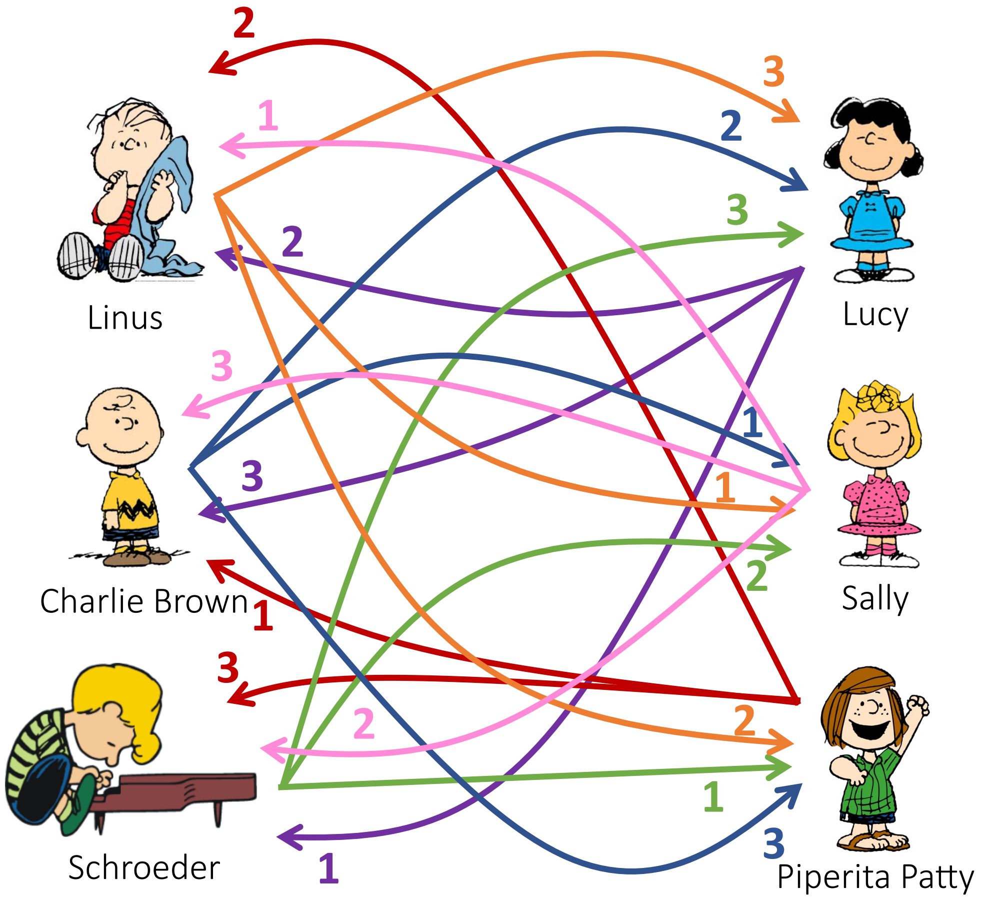
Università Roma Tre – Dipartimento di Matematica e Fisica
IN440 – Ottimizzazione Combinatoria
Prof. Marco Liverani

Problemi di matching e il matrimonio stabile



Un problema di «accoppiamento»

- Ad una festa un gruppo di ragazzi e di ragazze si conoscono e tra di loro nasce in interesse reciproco
- Certo, non tutti hanno la stessa preferenza per gli altri, non tutti sono ricambiati, ma ci sono i presupposti perché si vengano a creare delle coppie
- Possiamo schematizzare i sentimenti fra i ragazzi dei due gruppi con un grafo bipartito completo
- Sugli spigoli del grafo riportiamo il grado di preferenza (1 = massimo, 3 = minimo), che non sempre è «simmetrico»
- Facciamo qualche valutazione:
  - La coppia «Linus – Sally» è piuttosto stabile: entrambi hanno espresso la preferenza per l'altro al primo posto
  - Viceversa Lucy detesta Charie Brown e adora Schroeder; quest'ultimo però detesta Lucy e le preferisce Piperita Patty che però è follemente innamorata di Charlie Brown!
  - Trovare delle coppie in grado di resistere al tempo, non sembra facile...



Un problema di «accoppiamento»

- Semplifichiamo il contesto ed il problema che intendiamo affrontare con un altro esempio
- Se Alberto è sposato con Anna e Bruno è sposato con Beatrice il legame tra Alberto e Anna è **stabile** nel tempo se:
  - Alberto preferisce Anna ad ogni altra e al tempo stesso Anna preferisce Alberto ad ogni altro

  oppure se
  - Alberto preferisce Beatrice ad Anna, tuttavia Beatrice preferisce il suo Bruno ad Alberto
- Tuttavia, come si sa, «*al cuor non si comanda*» e quindi il legame tra Alberto e Anna è invece **instabile** (e potenzialmente a rischio) se:
  - Alberto preferisce Beatrice ad Anna ed anche Beatrice preferisce Alberto al suo Bruno

  oppure se
  - Anna preferisce Bruno ad Alberto ed anche Bruno preferisce Anna alla sua Beatrice

- Dati due insiemi di partner potenziali, che esprimano delle preferenze sui soggetti dell’altro insieme, vogliamo trovare, se esiste, **un accoppiamento (*matching*) stabile** tra gli elementi dei due gruppi

Costruiamo un modello formale

- Consideriamo due insiemi di elementi $X = \{x_1, ..., x_n\}$ e $Y = \{y_1, ..., y_n\}$
- Un ***matching*** M tra X e Y è un sottoinsieme di $X \times Y$ tale che per ogni $(x_h, y_k), (x_i, y_j) \in \mathrm{M}$ risulta $x_h \neq x_i$ e $y_k \neq y_j$; il matching è **massimale** se ha la massima cardinalità (non è possibile aggiungere altre coppie senza riusare almeno uno dei due elementi della coppia)
- Se $|X| = |Y| = n < \infty$ allora è sempre possibile creare un matching massimale di cardinalità n; se $|X| \neq |Y|$ allora il problema è meno banale
- Il problema della determinazione di un matching di cardinalità k fra tre insiemi finiti X, Y e Z è noto come 3DM (*three-dimensional matching*) ed è NP-completo
- Costruiamo un grafo bipartito completo $K_{n,\,n}$ i cui due insiemi indipendenti siano X e Y; associamo un peso $w(x_i, y_j)$ ad ogni coppia di vertici del grafo $K_{n,\,n}$ che rappresenti la preferenza di x_i rispetto a y_j
- Si noti che:
  - le relazioni sono asimmetriche: $(x_i, y_j) \neq (y_j, x_i)$
  - inoltre se $w(x, y_h) < w(x, y_k)$ allora x preferisce y_h rispetto a y_k (scriveremo in questo caso $y_h <_x y_k$)
- Il matching è **instabile** se esistono $x \in X$ e $y \in Y$ tali che $(x, y'), (x', y) \in M$ e $w(x, y) < w(x, y')$, $w(y, x) < w(y, x')$
- Cerchiamo, se esiste, un matching stabile tra X e Y

Un esempio di matrimonio stabile (*stable matching*)

- Consideriamo i seguenti insiemi: $X = \{$Alberto, Bruno, Carlo, Daniele$\}$, $Y = \{$Elena, Francesca, Giulia, Helga$\}$
- Supponiamo che valgano le seguenti preferenze:
  - Alberto: G, F, H, E
  - Bruno: F, E, G, H
  - Carlo: F, H, E, G
  - Daniele: G, E, H, F
  - Elena: A, B, D, C
  - Francesca: C, A, D, B
  - Giulia: C, B, D, A
  - Helga: B, A, C, D
- Il matching $M = \{$(Alberto, Elena), (Bruno, Francesca), (Carlo, Giulia), (Daniele, Helga)$\}$ è ***instabile***, perché Alberto e Francesca si preferiscono reciprocamente ai loro partner
- Un matching ***stabile*** è ad esempio $M = \{$(Alberto, Francesca), (Bruno, Elena), (Carlo, Helga), (Daniele, Giulia)$\}$
- Quanti sono i possibili matching su due insiemi da n elementi ciascuno?
  A ha n possibilità di scelta, B ha $n-1$ possibili scelte, C ha $n-2$ scelte, ...
  Complessivamente abbiamo $n!$ matching differenti

Strategia delle separazioni successive

- Si consideri il matching (A-E, B-F, C-G, D-H); è instabile perché Alberto e Francesca si preferiscono reciprocamente ai rispettivi partner Elena e Bruno, allora A-E e B-F possono lasciarsi, in modo da consentire a A-F di coronare il loro sogno d'amore
- Viene a crearsi il nuovo matching (A-F, B-E, C-G, D-H) che è instabile, perché Carlo preferisce Francesca a Giulia e Francesca preferisce Carlo ad Alberto
- Anche A-F e C-G si lasciano quindi, creando un nuovo matching: (A-G, B-E, C-F, D-H)
- Anche questo matching è instabile perché Giulia preferisce Daniele ad Alberto e Daniele preferisce Giulia a Helga
- A-G e D-H si lasciano dando luogo ad un nuovo matching: (A-H, B-E, C-F, D-G) che finalmente è **stabile**

Algoritmo 37 SEPARAZIONISUCCESSIVE(X, Y)

Input: I due insiemi X e Y e le liste di preferenza complete

Output: Un matching stabile tra X e Y

1: sia M un matching qualsiasi tra gli elementi di X e Y: $M = \{(x_1, y_1), \ldots, (x_n, y_b)\}$
2: **fintanto che** il matching M non è stabile **ripeti**
3:     sia $x \in X$ e $y \in Y$ una coppia che determina la condizione di instabilità su M; siano $(x, \tilde{y}), (\tilde{x}, y) \in M$, con $y <_x \tilde{y}$ e $x <_y \tilde{x}$
4:     definisci il nuovo matching $M = M \setminus \{(x, \tilde{y}), (\tilde{x}, y)\} \cup \{(x, y), (\tilde{x}, \tilde{y})\}$
5: **fine-ciclo**

- Alberto: G, F, H, E
- Bruno: F, E, G, H
- Carlo: F, H, E, G
- Daniele: G, E, H, F

- Elena: A, B, D, C
- Francesca: C, A, D, B
- Giulia: C, B, D, A
- Helga: B, A, C, D

Strategia delle separazioni successive

- Tuttavia la strategia delle separazioni successive non rappresenta un buon algoritmo perché non funziona per ogni istanza del problema
- Si consideri il matching tra gli insiemi $X = \{A, B, C\}$ e $Y = \{a, b, c\}$ con le seguenti preferenze:

• A: b, a, c	• a: A, C, B
• B: c, a, b	• b: C, A, B
• C: a, b, c	• c: A, B, C

- Procedendo per separazioni successive:
  - $(\mathbf{A}a, B\mathbf{b}, Cc) \rightarrow (Ab, Ba, Cc)$
  - $(A\mathbf{b}, Ba, \mathbf{C}c) \rightarrow (Ac, Ba, Cb)$
  - $(Ac, B\mathbf{a}, \mathbf{C}b) \rightarrow (Ac, Bb, Ca)$
  - $(\mathbf{A}c, Bb, C\mathbf{a}) \rightarrow (Aa, Bb, Cc) \rightarrow$ siamo tornati alla situazione iniziale!
- Il problema ammette comunque una soluzione stabile (come dimostreremo esiste *sempre* almeno una soluzione stabile): ad esempio (Ab, Bc, Ca)

Generalizzazioni del problema del matrimonio stabile

Prima generalizzazione: $|X| \neq |Y|$

- Vogliamo applicare il problema del matrimonio stabile alla scelta/assegnazione del medico di base ai pazienti di una ASL: supponiamo che $X = \{$insieme dei medici$\}$, $Y = \{$insieme dei pazienti$\}$, risulta ovviamente $|X| < |Y|$; il medico k-esimo può accettare al massimo n_k pazienti
- Anche nel caso della scelta di un corso di laurea a numero chiuso (X) da parte degli studenti che intendono immatricolarsi (Y); anche in questo caso $|X| < |Y|$; il corso di laurea x_k può accettare al massimo n_k studenti

- Il problema si riconduce al caso del matrimonio stabile definendo un nuovo insieme $X' = \{x_k^{(i)}, i = 1, 2, \ldots, n_k, k = 1, 2, \ldots, |X|\}$

Generalizzazioni del problema del matrimonio stabile

Seconda generalizzazione: liste di preferenza incomplete

- Se nella lista di preferenza di alcuni elementi $x_k \in X$ mancano alcuni elementi $y_h \in Y$ e/o viceversa si aggiungono ai due insiemi X e Y due nuovi elementi, i **vedovi**, $v_X \in X$ e $v_Y \in Y$
- Le liste vengono completate come segue, in modo tale da ricondursi al caso del problema con liste di preferenza complete:
  - si colloca v_X come ultima preferenza di v_Y e viceversa
  - si colloca v_X come ultima preferenza di ciascun elemento $y \in Y$, e viceversa
  - si aggiungono in un ordine arbitrario, dopo v_X tutti gli elementi mancanti alla lista di ciascun elemento $y \in Y$, e viceversa

Generalizzazioni del problema del matrimonio stabile

Seconda generalizzazione: liste di preferenza incomplete

- **Teorema**. Per la versione completa del problema esiste un matching stabile in cui v_X e v_Y sono associati se e solo se esiste un matching stabile nella versione incompleta del problema

  Dimostrazione.

  - Se esiste un matching stabile per la versione completa, in cui (v_X, v_Y) costituiscono una coppia, allora lo stesso matching è stabile anche eliminando v_X e v_Y, perché visto che v_X è la scelta peggiore per v_Y e viceversa, evidentemente perché l'istanza completa del problema abbia un matching stabile con v_X e v_Y accoppiati, significa che nessun altro elemento di X e Y preferisce rispettivamente v_Y e v_X al proprio partner
  - Al contrario se esiste un matching stabile per il problema con le liste incomplete allora lo stesso matching, con l'aggiunta della coppia (v_X, v_Y) è un matching stabile per il problema con le liste complete ■

Algoritmo di Gale e Shapley

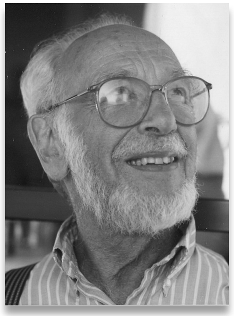
- Nell'algoritmo di Gale e Shapley (1962) viene introdotto un elemento fittizio Ω aggiunto all'insieme X: rappresenta il partner meno desiderabile per ciascun elemento di Y e viene collocato in fondo alla lista di ciascun $y \in Y$
- Inizialmente ogni elemento $y \in Y$ è associato a Ω
- L'algoritmo opera in modo asimmetrico sui due insiemi: fissato un elemento $x \in X$ procede selezionando gli elementi y in ordine decrescente di gradimento per x
- In questo modo il matching per x con l'elemento y sarà accettato solo se è migliorativo rispetto al partner del precedente matching per y

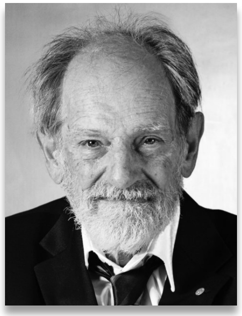
Algoritmo 38 STABLEMARRIAGE(X, Y)

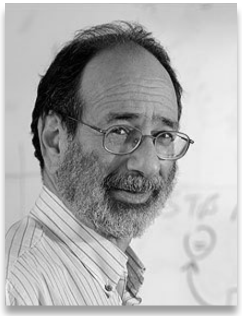
Input: I due insiemi X e Y e le liste di preferenza complete

Output: Un matching stabile tra X e Y

$k := 0$, associa a Ω ogni $y \in Y$
fintanto che $k < n$ **ripeti**
    sia $\bar{x} := x_{k+1}$
    fintanto che $\bar{x} \neq \Omega$ **ripeti**
        sia y la migliore scelta rimanente per $\bar{x}$ nella sua lista
        se y preferisce $\bar{x}$ al suo partner **allora**
            associa $(\bar{x}, y)$, sia $\bar{x}$ il precedente elemento associato a y
        fine-condizione
        se $\bar{x} \neq \Omega$ **allora**
            elimina y dalla lista di preferenze di $\bar{x}$
        fine-condizione
    fine-ciclo
    $k := k + 1$
fine-ciclo


David Gale
(1921 – 2008)


Lloyd Shapley
(1923 – 2016)


Alvin Roth
(1951)

Algoritmo di Gale e Shapley

- Alberto: G, F, H, E
- Bruno: F, E, G, H
- Carlo: F, H, E, G
- Daniele: G, E, H, F

- Elena: A, B, D, C, Ω
- Francesca: C, A, D, B, Ω
- Giulia: C, B, D, A, Ω
- Helga: B, A, C, D, Ω

- $X = \{A, B, C, D, \Omega\}$, $Y = \{E, F, G, H\}$, $n = 4$
- $M_1 = \{(\Omega, E), (\Omega, F), (\Omega, G), (\Omega, H)\}$
- $k = 1, x = A, y = G$
- $M_2 = \{(\Omega, E), (\Omega, F), (A, G), (\Omega, H)\}, x = \Omega$
- $k = 2, x = B, y = F$
- $M_3 = \{(\Omega, E), (B, F), (A, G), (\Omega, H)\}, x = \Omega$
- $k = 3, x = C, y = F$
- $M_4 = \{(\Omega, E), (C, F), (A, G), (\Omega, H)\}, x = B$
- $k = 3, x = B, y = E$
- $M_5 = \{(B, E), (C, F), (A, G), (\Omega, H)\}, x = \Omega$
- $k = 4, x = D, y = G$
- $M_5 = \{(B, E), (C, F), (D, G), (\Omega, H)\}, x = A$
- $k = 4, x = A, y = F, y = H$
- $M_5 = \{(B, E), (C, F), (D, G), (A, H)\}, x = \Omega$

Algoritmo 38 STABLEMARRIAGE(X, Y)

Input: I due insiemi X e Y e le liste di preferenza complete

Output: Un matching stabile tra X e Y

$k := 0$, associa a Ω ogni $y \in Y$
fintanto che $k < n$ **ripeti**
    sia $\bar{x} := x_{k+1}$
    fintanto che $\bar{x} \neq \Omega$ **ripeti**
        sia y la migliore scelta rimanente per $\bar{x}$ nella sua lista
        se y preferisce $\bar{x}$ al suo partner **allora**
            associa $(\bar{x}, y)$, sia $\bar{x}$ il precedente elemento associato a y
        fine-condizione
        se $\bar{x} \neq \Omega$ **allora**
            elimina y dalla lista di preferenze di $\bar{x}$
        fine-condizione
    fine-ciclo
    $k := k + 1$
fine-ciclo

Algoritmo di Gale e Shapley

- **Teorema 6.** L'algoritmo StableMarriage produce un matching stabile per ogni istanza del problema.

  Dimostrazione.

  - Se al passo 10 y viene rimosso dalla lista delle preferenze di x allora non esiste un matching stabile che contiene (x, y); infatti se y viene rimosso, vuol dire che y ha preferito un altro elemento $x' \in X$ a x: dunque y preferisce x' e x' preferisce y ad ogni altro elemento presente nella propria lista (la scelta per i matching avviene rispettando l'ordine di preferenza degli elementi di X)
  - Se ad un certo passo viene prodotta la coppia (x, y) e x preferisce y' a y, allora vuol dire che nei passi precedenti dell'algoritmo y' ha rifiutato x preferendo il proprio partner
  - Due elementi di Y non possono essere associati allo stesso elemento di X (a meno dell'elemento aggiuntivo Ω)
  - La situazione di accoppiamento di ciascun elemento di Y non peggiora mai durante l'esecuzione dell'algoritmo; infatti inizialmente risulta (Ω, y) per ogni $y \in Y$ (accoppiamento peggiore in assoluto); successivamente l'accoppiamento di y cambia solo se la nuova proposta migliora la precedente (condizione al passo 6)
  - La lista di preferenza di ogni elemento $x \in X$ non diventa mai vuota; se così fosse, infatti, vorrebbe dire che x è stato rifiutato da ogni $y \in Y$, il che non è possibile, visto che $|Y| = |X|$
  - Quindi l'algoritmo è ben definito e che termina sempre individuando un matching
  - Il matching con cui termina l'algoritmo è stabile; infatti per ogni coppia $(x,y) \in M$, se x preferisce y' a y, significa che y' ha preferito a x un altro elemento di X; dunque la preferenza di x per y' non compromette la stabilità del matching ■

Algoritmo di Gale e Shapley

- L'algoritmo produce la soluzione ottimale per ciascun elemento di X: ogni elemento $x \in X$ non potrebbe essere associato ad un diverso partner senza rendere instabile il matching
- Al tempo stesso l'algoritmo produce la soluzione peggiore (la meno gradita tra quelle accettabili dagli elementi di Y) per ciascun elemento di Y:
  - sia (x, y) una coppia del matching stabile prodotto dall'algoritmo; in un altro matching stabile, non prodotto dall'algoritmo, risultano le coppie (x, y') e (x', y)
  - ma siccome x preferisce y a y' (la soluzione prodotta dall'algoritmo è ottimale per gli elementi di X) allora significa che y preferisce x' a x, perché altrimenti il secondo matching non sarebbe stabile
- La complessità dell'algoritmo è $O(n^2)$, dove $n = |X| = |Y|$
- Non è vero quindi che «*in amor vince chi fugge*» (Ovidio?)
  - l'algoritmo di Gale & Shapley sembra dimostrare proprio il contrario: la strategia di «attacco» degli elementi di X li favorisce nella scelta del partner
  - gli elementi di *Y* sono invece penalizzati dalla loro strategia «attendista»

Riferimenti bibliografici

- Marco Liverani, «*Dispense del Corso di Ottimizzazione Combinatoria:* Partizionamento ottimo di grafi in componenti connesse» (http://www.mat.uniroma3.it/users/liverani/doc/disp_oc_10.pdf)
- Donald E. Knuth, *Stable Marriage and Its Relation to Other Combinatorial Problems – An Introduction to the Mathematical Analysis of Algorithms*, AMS – American Mathematical Society, 1997