The Stable Marriage Theorem

Jonathan L.F. King
University of Florida, Gainesville FL 32611-2082, USA
squash@ufl.edu
Webpage http://squash.1gainesville.com/
4 April, 2018 (at *07:34*)

(jk: This theorem and proof were shown to me by Steve Kalikow. See ~/Lisp/stable-marriage.lsp for an implementation of this algorithm.)

Entrance. Let $\mathcal{B}$ and $\mathcal{G}$ be [disjoint-pair of] equal-cardinality sets of boys and girls. Initially, we study the case where the cardinality $N := |\mathcal{B}| = |\mathcal{G}|$ is finite.

Each boy $b \in \mathcal{B}$ has a girl he like best, second best, ..., $[N-1]^{st}$ best, N^{th} best. That is, he has a strict total-order $\succ^b$ on $\mathcal{G}$:

Expression $x \succ^b y$ *means that* b *likes girl* x *better than he likes girl* y.

Similarly, each girl g has a total-order $\succ^g$ on $\mathcal{B}$.

A matching is ***unstable*** if there are distinct married couples *Alan*↔*Ann* and *Bert*↔*Betty* so that *Ann* and *Bert* are a ***covetous-pair***. That is, each likes the other *more* than he likes his own spouse. Symbolically,

$$\textit{Bert} \succ^{\textit{Ann}} \textit{Alan} \qquad \underline{\text{and}} \qquad \textit{Ann} \succ^{\textit{Bert}} \textit{Betty}\,.$$

Finally, a matching is ***stable*** if there are no covetous neighbors.

1: Stable Marriage Theorem. *If* $|\mathcal{B}| = |\mathcal{G}|$ *is finite, then there is a stable matching.* ◊

Proof. Steve describes his algorithm as taking place at HS prom night. The boys are on one side of the gym, bashfully, and the girls are on the other side, also bashful, waiting to be asked to dance..

As the first dance is announced, every boy walks across the room and stands in a cluster next to the girl he likes best. Some girls may have several boys clustered around them, other girls having none. Now, each girl selects the boy she likes best *from her cluster*, and dances with him. The rejected boys go back across the room.

When the k^{th} *dance is announced*, this happens: Each boy clusters around the girl he likes best *from among the girls that have* <u>*never*</u> *rejected him.* As before, each girl picks the boy she likes best from among her cluster —this may be the boy she just danced with.

Let K be the earliest dance where EITHER:

i: The procedure is about to become ill-defined: A boy has just been rejected by the last girl on his list and so has no one to stand next to for the $[K+1]^{\text{st}}$ dance.

ii: No boy is rejected; each is the only boy in his cluster. Thus every girl has exactly one boy standing next to her. (Consequently, all partner switching has stopped; all later dances will be identical to dance K.) ♦

2: Lemma. *Condition* (i) *will not occur and so* (ii) *will. Moreover, the correspondence of* (ii) *is a stable matching.* ◊

Proof of (i). If *Ann* gets a partner at dance k, then she will thenceforth have a partner for subsequent dances, although her partner may change.

Let $N := |\mathcal{B}| = |\mathcal{G}|$. Were a boy, *Bert*, to be rejected by the last girl on his list, then at that moment *all* the girls would have to have partners, and therefore there would be at least N boys *other than Bert* —which is not the case. ♦

Proof of (ii). Consider a pair of couples *Alan*↔*Ann* and *Bert*↔*Betty*. If *Ann* likes *Bert* more than her own husband then –since she married *Alan* and not *Bert*– it must be that *Bert* **never** stood in her cluster. Thus *Bert* never got so far down his list as to get to *Ann* and therefore *Bert* must like his own wife, *Betty*, *more* than he likes *Ann*. So *Bert*|*Ann* is not a covetous-pair [relative to the matching] —and all is well. ♦

Notation. Use **BAG-*alg*** for the *Boy-ask-Girl algorithm*, and use **GAB-*alg*** for the *Girl-ask-Boy version*. I'll abbreviate the algorithms as BAG and GAB.

Many rejections.

```
% (print-stable-both (LongRun)) ;; Girl is (Ig)Nora.

   Prefs of girls: Best-to-Worst on each line.
Pref(G0)   = (B0 B1 B2 B3 B4)
Pref(G1)   = (B1 B2 B3 B4 B0)
Pref(G2)   = (B2 B3 B4 B0 B1)
Pref(G3)   = (B3 B4 B0 B1 B2)
Pref(Nora) = (B0 B1 B2 B3 B4) ;Nora's prefs irrelevant.

   Prefs of boys: Best-to-Worst on each line.
Pref(B0) = (G3 G2 G1 G0 Nora)
Pref(B1) = (G0 G3 G2 G1 Nora)
Pref(B2) = (G1 G0 G3 G2 Nora)
Pref(B3) = (G2 G1 G0 G3 Nora)
Pref(B4) = (G3 G2 G1 G0 Nora) ;Only B4 ever asks Nora.
==================================================

   The guys ask the gals.
     Dance  1:  One guy rejected.
     Dance  2:  One guy rejected.
     Dance  3:  One guy rejected.
     Dance  4:  One guy rejected.
     Dance  5:  One guy rejected.
     Dance  6:  One guy rejected.
     Dance  7:  One guy rejected.
     Dance  8:  One guy rejected.
     Dance  9:  One guy rejected.
     Dance 10:  One guy rejected.
     Dance 11:  One guy rejected.
     Dance 12:  One guy rejected.
     Dance 13:  One guy rejected.
     Dance 14:  One guy rejected.
     Dance 15:  One guy rejected.
     Dance 16:  One guy rejected.

  Men:     Women:   (Took 16 dances)
  B0       G0
  B1       G1
  B2       G2
  B3       G3
  B4       Nora
--------------------------------------------------

   The gals ask the guys.
     Dance  1:  One gal rejected.
     Dance  2:  One gal rejected.
     Dance  3:  One gal rejected.
     Dance  4:  One gal rejected.

  Men:     Women:   (Took 4 dances)
  B0       G0
  B1       G1
  B2       G2
  B3       G3
  B4       Nora
```

How many rejections? For a given girl/boy pref-structure $\mathcal{P}$, let $\mathbf{R}_{\mathcal{P}}$ be number of rejections [when boys ask girls], and use $\mathbf{D}_{\mathcal{P}}$ for the number of dances; so $\mathbf{D}_{\mathcal{P}} \leqslant \mathbf{R}_{\mathcal{P}}$.

Let $\mathbf{R}(N)$ be the maximum of $\mathbf{R}_{\mathcal{P}}$ over all preferences of N girls and N boys, and ditto $\mathbf{D}(N)$. [Note that rejections/dances could get maximized at different $\mathcal{P}$.]

Easily

$$\mathbf{D}(N) \leqslant \mathbf{R}(N) \leqslant N \cdot [N-1],$$

since there are N boys, and each can only be rejected at most $[N-1]$ times.

The preceding example, call it $\mathcal{P}_N$, shows that

$$\mathbf{D}_{\mathcal{P}_N} = \mathbf{R}_{\mathcal{P}_N} = [N-1]^2.$$

Q1: Are these the maxima?

Is there always an answer? Problems similar to Stable-Marriage may not have a solution.

3: Stable-roommate Problem. *An even number, N, of people, share a house with $\frac{N}{2}$ two-person rooms. Each resident has a preference-list on the other $N-1$ people. Then there exists a stable roommate-assigment.* ◊

Ouch! N=4 CEX. People $\{A, B, C, \text{Hitler}\}$ have prefs:

$$A\colon\begin{bmatrix} C \\ B \\ \text{Hitler}\end{bmatrix}, \quad B\colon\begin{bmatrix} A \\ C \\ \text{Hitler}\end{bmatrix}, \quad C\colon\begin{bmatrix} B \\ A \\ \text{Hitler}\end{bmatrix}, \quad \text{Hitler}\colon\begin{bmatrix} ? \\ ? \\ ?\end{bmatrix}.$$

In the $[A \leftrightarrow \text{Hitler},\ B \leftrightarrow C]$ matching, note $A|B$ is a covetous-pair. By A,B,C-symmetry, *every* matching has a covetous-pair. ♦

Lattice of stable matchings

A matching λ is a bijection $\mathcal{B} \hookrightarrow\!\!\!\rightarrow \mathcal{G}$; write $\lambda(b)$ for the girl that b is matched with; but if just the relation is needed, then write $b \overset{\lambda}{\longleftrightarrow} g$.

Let $\mathcal{S} = \mathcal{S}_{\mathcal{P}}$ be the set [indeed, a lattice] of stable-matchings for prefs $\mathcal{P}$. One $\mathcal{S}$-member is the stable BAG-matching; call it $\omega = \omega_{\mathrm{BAG}} = \omega_{\mathrm{BAG},\mathcal{P}}$.

Say that a boy-girl pair (b, g) is "**$\mathcal{P}$-*possible***" if *there exists* a stable-matching λ with $b \overset{\lambda}{\longleftrightarrow} g$. Define

$$\mathrm{PosW}(b) = \mathrm{PosW}_{\mathcal{P}}(b) := \{g \in \mathcal{G} \mid \exists \lambda \in \mathcal{S}_{\mathcal{P}} \text{ with } b \overset{\lambda}{\longleftrightarrow} g\}.$$

So $\mathrm{PosW}(b)$ is the set of potential stable-wives for b. Define similarly $\mathrm{PosH}(g)$, the set of potential stable-husbands for girl g.

Partial-order on $\mathcal{S}_{\mathcal{P}}$. Define $\sqsupseteq_{\mathcal{P}}$, a partial-order on the set $\mathcal{S} = \mathcal{S}_{\mathcal{P}}$, by [$\lambda$=larger, σ=smaller]

†: $$\lambda \sqsupseteq_{\mathcal{P}} \sigma \quad \text{IFF} \quad \Big[\forall b \in \mathcal{B}\colon\ \lambda(b) \succeq^{b} \sigma(b)\Big].$$

4: Order-inversion thm. *Suppose two stable matchings have $\lambda \sqsupseteq_{\mathcal{P}} \sigma$. Then*

‡: $$\Big[\forall g \in \mathcal{G}\colon\ \lambda^{-1}(g) \ {}^{g}\!\!\preccurlyeq\ \sigma^{-1}(g)\Big]. \qquad \diamondsuit$$

Pf. FTSOC, suppose $\exists \mathit{Ann}$ with strict inequality

∗: $$\mathit{Bert} := \lambda^{-1}(\mathit{Ann}) \succ^{\mathit{Ann}} \sigma^{-1}(\mathit{Ann}) =: \mathit{Alan}.$$

Note $B := \sigma(\mathit{Bert})$ is not Ann, since $\sigma(\mathit{Alan}) = \mathit{Ann}$, and $\mathit{Alan} \neq \mathit{Bert}$. Thus $\boxed{\mathit{Ann} >^{\mathit{Bert}} B}$, courtesy (†), which tells us that $\mathit{Ann} \geq^{\mathit{Bert}} B$.

Unfortunately, (∗) says $\boxed{\mathit{Bert} \succ^{\mathit{Ann}} \mathit{Alan}}$ and so $\mathit{Bert}|\mathit{Ann}$ is a covetous-pair for the $\mathit{Alan} \overset{\sigma}{\longleftrightarrow} \mathit{Ann}$, $\mathit{Bert} \overset{\sigma}{\longleftrightarrow} B$, stable matching. ※ ♦

5: Max-Thm. *Poset $(\mathcal{S}_{\mathcal{P}}, \sqsupseteq_{\mathcal{P}})$ has a [unique] maximum element. Moreover, this maximum is $\omega_{\mathrm{BAG},\mathcal{P}}$. Indeed, this $\omega = \omega_{\mathrm{BAG},\mathcal{P}}$ satisfies*

∗: $\forall b \in \mathcal{B}$: *Girl $\omega(b)$ is the $\succeq^{b}$–max of $\mathrm{PosW}(b)$.* ♢

Proof. Assertion (∗) is equivalent to saying that BAG never has a girl reject a possible-husband.

FTSOC, suppose *Ann* rejects possible-husband *Alan*, and that this happens at the *earliest dance*, say dance 9, that any boy is rejected by a possible-wife.

So at the 9th dance, *Ann* dances with, say, *Bert*, rejecting *Alan*. By hyp, there exists a stable λ with $\lambda(\mathit{Alan}) = \mathit{Ann}$. Let $\mathit{Betty} := \lambda(\mathit{Bert})$.

Note $\mathit{Bert} \succ^{\mathit{Ann}} \mathit{Alan}$, since *Ann* chose *Bert* over *Alan*. Pair $\mathit{Bert}|\mathit{Ann}$ is not λ-covetous [no pair is], since λ is stable. Hence $\boxed{\mathit{Betty} >^{\mathit{Bert}} \mathit{Ann}}$. Thus in BAG-alg,

Bert visited Betty <u>*before*</u> *visiting Ann.* ♦

Since *Bert* visited *Ann* at dance 9, he must have been rejected by *Betty* earlier; say, at dance 7. But *Betty* is a possible-wife for *Bert*, and the earliest such rejection happened at dance 9. ※

Max of matchings

For matchings, $\alpha,\beta:\mathcal{B}\hookrightarrow\!\!\!\rightarrow\mathcal{G}$, the pointwise maximum f

$$*: \qquad f(b) \;:=\; \succeq^b\text{-Max}\Big\{\alpha(b)\,,\;\beta(b)\Big\}$$

need not be 1-to-1. But if α and β are *stable*...

6: Join lemma. *Suppose α and β are stable matchings. Then the f from $(*)$*

6†: *...is a bijection.*

6‡: *...is a stable matching.* ♢

Pf of (6†). WLOG $\alpha\neq\beta$. ISTShow that f is 1-to-1.

FTSOC, suppose ∃ boys *Alan*≠*Bert* and girl U [the lovely *Ultima*] s.t $f(\textit{Alan})=U=f(\textit{Bert})$. Hence, WLOG

$$\alpha(\textit{Alan}) \;=\; U \;\succ^{\textit{Alan}}\; \beta(\textit{Alan}) =: A \quad \textit{and}$$
$$\beta(\textit{Bert}) \;=\; U \;\succ^{\textit{Bert}}\; \alpha(\textit{Bert}) =: B. \quad \begin{bmatrix}\text{Possibly}\\ A=B.\end{bmatrix}$$

[Evidently $\alpha(\textit{Alan}) \succeq^{\textit{Alan}} \beta(\textit{Alan})$, but why is it *strict*? Equality $\beta(\textit{Alan})=U \stackrel{\text{note}}{=\!=} \beta(\textit{Bert})$ would force $\textit{Alan}=\textit{Bert}$.]

In the α-matching, *Bert*↔B and *Alan*↔U. But our $U \succ^{\textit{Bert}} B$ forces $\boxed{\textit{Alan} \succ^{U} \textit{Bert}}$; otherwise *Bert*|$U$ would be an α-covetous-pair.

The same argument using the β-matching, yields $\boxed{\textit{Bert} \succ^{U} \textit{Alan}}$. ※ ♦

Proof of (6‡). FTSOContradiction, suppose *Alan*≠*Bert* are the husbands in an f-covetous-pair. Let $\textit{Ann} := f(\textit{Alan}) \stackrel{\text{WLOG}}{=\!=\!=} \alpha(\textit{Alan})$. Since α is stable, covetousness forces that $f(\textit{Bert}) = \beta(\textit{Bert})$. Thus

$$\textit{Ann} \;:=\; \alpha(\textit{Alan}) \succeq^{\textit{Alan}} \beta(\textit{Alan}) =: A\,,$$
$$£: \qquad \textit{Betty} \;:=\; \beta(\textit{Bert}) \succeq^{\textit{Bert}} \alpha(\textit{Bert}) \;=:\; B\,.$$

Recall f is 1-to-1, so *Ann*≠*Betty*.

FTSOC, suppose *Bert*|*Ann* is an f-covetous-pair. Thus

$$\textit{Bert} \;\succ^{\textit{Ann}}\; \textit{Alan}$$
$$\textit{Ann} \;\succ^{\textit{Bert}}\; \textit{Betty} \;\succeq^{\textit{Bert}}\; B\,,$$

where the last inequality comes from (£). But the α-matching is *Alan*↔*Ann* and *Bert*↔B, which has now been shown to be covetous. ♦

7: Theorem. *Our $(\boldsymbol{\mathcal{S}}_{\mathcal{P}}, \sqsupseteq_{\mathcal{P}})$ is a lattice, with **join** $\vee$, and **meet** $\wedge$, defined by*

$$[\lambda\vee\sigma](b) \;:=\; \succeq^b\text{-Max}\Big\{\lambda(b)\,,\;\sigma(b)\Big\} \quad \textit{and}$$
$$[\lambda\wedge\sigma]^{-1}(g) \;:=\; \succeq^g\text{-Max}\Big\{\lambda^{-1}(g)\,,\;\sigma^{-1}(g)\Big\}. \quad ♢$$

Proof. The Join lemma showed that $(\boldsymbol{\mathcal{S}},\sqsupseteq)$ is a join-semilattice. And Order-inversion together with Join lemma shows that $(\boldsymbol{\mathcal{S}},\sqsupseteq^{-1})$ is a join-semilattice, i.e $(\boldsymbol{\mathcal{S}},\sqsupseteq)$ is a meet-semilattice. Hence $(\boldsymbol{\mathcal{S}},\sqsupseteq)$ is a lattice. ♦

Q2. Does (†) define the same operation that

$$‡: \qquad [\lambda\wedge\sigma](b) \;:=\; \succeq^b\text{-Min}\Big\{\lambda(b)\,,\;\sigma(b)\Big\}$$

defines? [Equivalently, is $\boldsymbol{\mathcal{S}}$ sealed under (‡)?] If yes, then $(\boldsymbol{\mathcal{S}},\sqsupseteq)$ will automatically be a *distributive* lattice. □

Q3. As a function of $N := |\mathcal{B}| = |\mathcal{G}|$, what the maximum cardinality of $\boldsymbol{\mathcal{S}}$?

What is the maximum ***height***, the number of steps from the min-elt to the max-elt, of lattice $\boldsymbol{\mathcal{S}}$? □

Filename: `Problems/GraphTheory/Matching/stable-marriage.latex`
As of: *Friday 30Mar2018.* Typeset: *4Apr2018 at 07:34.*