Dinitz conjecture, stable matching and graph painting

Raúl Penaguião
raul.penegas@gmail.com

2011 - 2012

Part I
Preamble

Since the invitation to the program *Novos Talentos* that I was looking for something in the area of Combinatorics. As such, I was led to Professor José Fachada, who mentioned two problems in this area, which even have solutions in "Proofs From The Book". One of them was the main target of my work this year, the *Dinitz Problem*.

The study of those two problems drove me to work on graphs and to establish some definitions on stable matching, which led to the solution of the Dinitz problem. A study of some propose-dispose algorithms for University admissions and a search for some deep results on stable matching applied to lattices, polytopes and linear programming complete my work.

Part II
Dinitz Conjecture

1 The problem

Forty years ago, Jeff Dinitz posted a question about coloring matrices that was left unsolved for 20 years. The history begins with a latin matrix: a matrix $m \times n$, in which no column or line has the same entry twice.

Filling in a latin square, Jeff decided to impose some restrictions: that's how he got to the definition of a Dinitz Matrix.

Definition 1. *By a Dinitz's Matrix, we understand a square matrix A, $m \times n$, where each* *entry is a set* $S_{i,j}$ *with* $\max\{m,n\}$ *elements, which we regard as colors.*

A Dinitz's Matrix is said to be **solvable** *if there is an* $m \times n$ *latin matrix* L*, such that* $L_{i,j} \in S_{i,j}$ *for each* i, j*. This means that we can pick some elements of the given sets in order to obtain a latin matrix.*

For example, here is a *Dinitz's Matrix*: $A = \begin{pmatrix} \{1,2\} & \{1,3\} \\ \{1,2\} & \{2,3\} \end{pmatrix}$

Conjecture 1. *Dinitz's Conjecture*

All Dinitz's Matrixes are solvable.

This is the typical "easy to state - hard to break" problem.

The question arose in 1978. At that time someone observed that the problem can be reduced to square matrixes and still be equivalent to the original one. This is the version of the Conjecture we are going to focus on.

It was, however, the last thing to be proven: in 1993, Jeannette Janson proved the Conjecture for rectangles, and only in 1995 did Fred Galvin prove it to square matrixes. He used a very similar solution to the one in Jeannette's paper. The world saw a beautiful adaptation of this proof in "Profs From The Book".

2 Familiarization with the Problem

Let's put our hands on the dirt with some examples:

Example 1. *Take the square* $A = \begin{pmatrix} \{1,2\} & \{1,3\} \\ \{1,2\} & \{2,3\} \end{pmatrix}$

At the light of Dinitz's Conjecture, we can say that there is a choice that transforms A in a latin square. Let's find it!

We pick the color 2 *in the* $(2,1)$[1] *entry, for the sake of the example. That forces us to chose color* 1 *in* $(1,1)$ *and color* 3 *in* $(2,2)$*, because we can have no repeated colors in each row or column, which leaves us with no color to paint the* $(1,2)$ *entry.*

Of course, if we had chosen color 1 *in* $(1,2)$*, that would lead us to a solution, namely* $L = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$.

For small cases, the trial-and-error approach is quick, but for larger matrices that is not the simplest choice:

Example 2. *Consider the square Dinitz's Matrix*

$$B = \begin{pmatrix} \{2,3,4,5\} & \{1,3,4,5\} & \{3,4,5,6\} & \{2,4,5,6\} \\ \{2,3,5,6\} & \{2,4,5,6\} & \{2,4,5,6\} & \{3,4,5,6\} \\ \{1,3,4,5\} & \{1,2,4,5\} & \{1,4,5,6\} & \{1,2,3,4\} \\ \{2,4,5,6\} & \{2,3,4,5\} & \{1,2,4,6\} & \{1,3,5,6\} \end{pmatrix}.$$

We want to find a latin coloring on this Dinitz's Matrix. The trial-and-error approach on this matrix may scare us a little bit, but fear not, because better methods are coming soon.

[1] Line 2, column 1

Let's represent the *Dinitz's Matrixes* in some other way. We construct a graph $L(\mathcal{K}_{n,n})$ where the vertices are the entries of the matrix $V = \{1, 2, \cdots, n\} \times \{1, 2, \cdots, n\}$, and we draw an edge between (i_1, j_1) and (i_2, j_2) if $i_1 = i_2$ or $j_1 = j_2$. We call it the line graph of $\mathcal{K}_{n,n}$ [2]

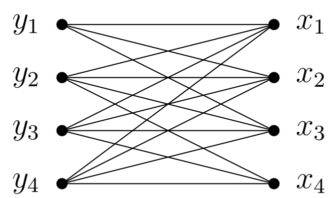

Figure 1: $\mathcal{K}_{4,4}$ graph

Painting matrices corresponds to painting vertices on this $L(\mathcal{K}_{n,n})$ graph, the same as painting edges on $\mathcal{K}_{n,n}$, with the required restrictions. Therefore, in this graph each edge corresponds to an entry on the *Dinitz's Matrix.*

Our goal in $\mathcal{K}_{n,n}$ is to paint the edges according to the restrictions, such that no two neighbor edges are monochromatic. If we consider only the subgraph D_1 of the edges that can be painted with, let's say, the color 1, we obtain a subgraph of $\mathcal{K}_{n,n}$, in which we would like to select a subset of non-neighbor edges, see figure 2. [3].

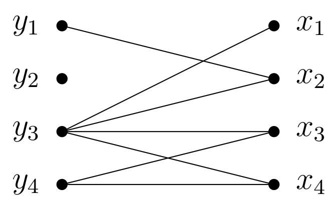

Figure 2: Graph D_1 corresponding to the color 1

Let's check that D_1 is the graph constructed by the color 1: $1 \in S_{1,2}$ so there is an edge (y_1, x_2) on D_1. Similarly, $1 \in S_{3,2}$ so we draw the edge (y_3, x_2) on D_1, and so on. An independent set would be $\{(y_1, x_2), (y_3, x_1), (y_4, x_4)\}$, or $\{(y_3, x_2), (y_4, x_4)\}$ or even $\{\}$, so our work from now on is to select from the pool of independent sets available, the one that better suits our needs.

[2]The "line graph" operation is a more general transformation on graphs. Roughly speaking, it regards the vertices in $L(G)$ as the edges in G, and links between two vertices of $L(G)$ if the corresponding edges in G share a vertex. The definition can be found later (see **Definition 8**).

[3]We will recall such sets as *independent* sets

Part III
Stable Matching

We are now going to study the **Matching Problem**. In this context we study a finite family $\mathcal{A} = \{a_1, a_2, \cdots\}$ of universities, or girls, that are going to evaluate and match someone in the finite family $\mathcal{B} = \{b_1, b_2, \cdots\}$ of students, or boys.

In this problem we consider a number of relations in the family $\mathcal{A}$ and in the family $\mathcal{B}$: For each $a \in \mathcal{A}$ we consider a total order in [a subset of, if the problem has **non-willing pairs**[4]] $\mathcal{B}$ and likewise for each $b \in \mathcal{B}$, consider a total order in [a subset of] $\mathcal{A}$. Those order relations intuitively explain that each student has is own preferences on each university, and each university ranks the students in its own way. By convenience we accept that there are no draws.

We represent that order relation by saying that b prefers a rather than a', or $a >_b a'$. We represent also the ordered list $A_a = \{b_1^*, b_2^*, \cdots\}$ of the preferences of a, and similarly $B_b = \{a_1^*, a_2^*, \cdots\}$. We assume that each university has $n_a > 0$ seats available.

This is the **Matching Problem**.

A **matching** is a subset S of $\mathcal{A} \times \mathcal{B}$ such that each $a \in \mathcal{A}$ occurs in S at most n_a times, and each $b \in \mathcal{B}$ occurs at most one time. In that sense, a **matching** either assigns each $b \in \mathcal{B}$ to exactly one $a \in \mathcal{B}$, or doesn't assign b to anything.

The abstract idea of a matching is a subgraph of $\mathcal{K}_{\mathcal{A},\mathcal{B}}$ [5] with $deg(a) \leq n_a$ and $deg(b) \leq 1$. This abstract notion in bipartite graphs is what distinguishes this approach, since the very same theory would have some flaws if a general graph structure were considered, as we will see later on.

We ask by now if there is an optimal way to choose a matching. We will give a precise meaning to the notion of a **Stable Matching** and an **Optimal Stable Matching**. The former will give us a fantastic way to establish the proof for Dinitz's Conjecture.

Definition 2. *Unstable pair, Stable Matching*

Given a matching S between $\mathcal{A}$ and $\mathcal{B}$, and a pair $(a, b) \in \mathcal{A} \times \mathcal{B}$, we say that (a, b) is unstable if the three following conditions are met:

- $(a, b) \notin S$
- *If $\forall a^* \in \mathcal{A}$, $(a^*, b) \notin S$, or there is some $a^* \in \mathcal{A}$ such that $(a^*, b) \in S$ and $a^* <_b a$*
- *If $\forall b^* \in \mathcal{B}$, $(a, b^*) \notin S$, or there is some $b^* \in \mathcal{B}$ such that $(a, b^*) \in S$ and $b^* <_a b$*

A matching is stable if there is no unstable pair.

[4] We consider this additional structure when regarding a problem where it is not possible to match a pair (a, b)

[5] We mean a $\mathcal{K}_{\#\mathcal{A},\#\mathcal{B}}$, which we assign a specific meaning to each vertex

Looking at the metaphor of matching girls and boys, we see a pair (a, b) of a girl and a boy as unstable if both the girl and the boy, currently in other relationships, would improve their status by breaking up with their matches and begin a relationship between themselves.

Additionally, we create the notion of an optimal matching. A matching S is **optimal** if, among all the stable matchings on a Matching Problem, any $b \in \mathcal{B}$ is not in a worst match in S than in S'.

Definition 3. *Optimal stable Matching*
A Stable Matching S is said to be $\mathcal{B}$**-Optimal** *if, for any stable matching S' and any $b \in \mathcal{B}$, there are no $a \in \mathcal{A}$ such that $(a, b) \in S'$ or, if there is, there is $a^* \in \mathcal{A}$ such that $(a^*, b) \in S$ and $a^* \geq_b a$.*

This definition imposes a clear uniqueness of the $\mathcal{B}$-Optimality.
If we set $n_a = 1$ for each a, there is a duality in this definition that may lead to a notion of $\mathcal{A}$**-Optimal**. We will pay some attention to that in due time.
One may wonder about the existence of stable matchings and optimal stable matchings. They do exist, and there is a precise construction.
However, their existence is not so general as it may seem at first glance. On this brief paragraph, we take a few moments to see the big picture and introduce the notion of matchings and stability on a general graph and lists of preferences. Take the **Roommate problem**, where there are four boys, a_1, a_2, a_3 and a_4, that have to share two double rooms. Here, we consider that:

- a_1 prefers a_2 over the others;
- a_2 prefers a_3 over the others;
- a_3 prefers a_1 over the others;
- Everyone classifies a_4 as the last one.

As one can check, whoever ends up with a_4, say wlog a_1, would rather be with some of the other guys; in fact, as a_3 prefers a_1 over the others, the pair (a_1, a_3) would be an unstable pair. Hence there is no stable matching in this case.
We will prove the existence of a stable matching on the case at hand (bipartite graphs), via the Gale-Shapley Algorithm.

1 The Gale-Shapley Algorithm

Consider now a Matching Game, in which $n_a = 1$, $\#\mathcal{A} = \#\mathcal{B}$ and there are no **non-willing pairs**. Note that we can change any Matching Game to attain these characteristics without losing any stability or optimality, by creating copies of universities with the same list of preferences, ghost-universities that no student would like to apply to, and ranking the ones no one wants to apply to as the last ones, so we don't lose generality.

The results proved henceforth will assume these circumstances, but will not depend on them unless stated otherwise.

In this set-up, a stable matching corresponds to a bijective function $f : \mathcal{B} \to \mathcal{A}$, i.e, any $a \in \mathcal{A}$ and any $b \in \mathcal{B}$ are matched to one (and only one) other person, because there are no non-willing pairs. This is called the **Matching function**.

Gale-Shapley Algorithm (or Proposal-Disposal Algorithm):

1. Begin with no matches (this matching is clearly unstable).

2. At each step, any $b \in \mathcal{B}$ that is unmatched selects the $a_b \in \mathcal{A}$ of his dreams, i.e., the one that is on the top of his list. Then, b "proposes" a matching between him and a_b.

3. When an $a \in \mathcal{A}$ receives some proposals, she selects the one that she likes the most, and rejects the others. The b selected keeps "on hold", until a better one appears, or until the algorithm ends.

4. When a "proposal" from $b \in \mathcal{B}$ get's rejected, he proceeds to delete a_b from his list, restarting from point 2.

5. The algorithm stops when all $b \in \mathcal{B}$ are matched, or when there are no more proposals to be done.

At the end, we will have a matching of the original Matching Problem. This matching is complete (i.e. all elements are matched to someone) when $\#\mathcal{A} = \#\mathcal{B}$ and when no **non-willing pairs** exists, because if some $b \in \mathcal{B}$ is not matched, he already proposed to every $a \in \mathcal{A}$, which means that every $a \in \mathcal{A}$ is matched.

2 Brief analysis of the Gale-Shapley Algorithm

Some notes on the algorithm will help us proving the existence of a stable matching, and we will talk about them right now. After those observations are stated, we will dive in the demonstration that an optimal-$\mathcal{B}$ stable matching exists, and is the one given by the Gale-Shapley Algorithm.

We note that, through the algorithm, every $b \in \mathcal{B}$ is going down his list until the end. Similarly, every $a \in \mathcal{A}$ is always moving up her list.

We could apply this algorithm in a symmetrical way: by making each $a \in \mathcal{A}$ propose to her preferred $b \in \mathcal{B}$. The result would not be the same in general: in fact, they are the same if and only if there is only one stable matching.

Lemma 1. *The Gale-Shapley algorithm gives a stable matching when $n_a = 1$, $\#\mathcal{A} = \#\mathcal{B}$ and there are no* **non-willing pairs**

Demonstration:

Consider an unstable pair $(a, b) \in \mathcal{A} \times \mathcal{B}$. Then $f(b) <_b a$.[6]

Since b is matched with some $f(b) \in \mathcal{A}$ such that $a >_b f(b)$, at some point b proposed to a and was rejected. This means a had a better proposal and ended up being matched with some $b^* \in \mathcal{B}$ such that $b^* >_a b$.

We conclude (a, b) is not an unstable pair, contradicting the hypothesis.

□

We could have broadened the scope of the Theorem, by not imposing any restrictions in the Matching Problem.

We will now prove the optimality condition:

Lemma 2. *The Gale-Shapley algorithm gives an optimal stable matching when* $n_a = 1$, $\#\mathcal{A} = \#\mathcal{B}$ *and there are no non-willing pairs.*

Demonstration:

For each $b \in \mathcal{B}$, we say that an $a \in \mathcal{A}$ is **b-possible** if there is a Stable Matching S' such that $(a, b) \in S'$. Define $a_b \in \mathcal{A}$ as the match of b given by the Gale-Shapley algorithm. We want to prove that $a_b \geq_b a$ for each $a \in \mathcal{A}$ that is b-possible.

We will prove inductively that, at each step of the algorithm, there were no $a \in \mathcal{A}$, $b \in \mathcal{B}$ such that b was declined by a, where a is b-possible: clearly, at the beginning, no such decline was done (there were no declines at all) so this is our base case.

Suppose now that we find ourselves in an arrangement that meets the induction hypothesis when a declines b. So there are $b^* \in \mathcal{B}$ such that

- b^* was turned down by every $a^* >_{b^*} a$
- $b^* >_a b$

Suppose by sake of contradiction that a is b-possible, so there is a Stable Matching S' such that $(a, b) \in S'$. We will manage to prove that (a, b^*) is an unstable pair, contradicting the statement that S' is a Stable Matching. [7]

Since b^* was turned down by every $a^* >_{b^*} a$, by induction hypothesis, none of the $a^* >_{b^*} a$ are b^*-possible, and since a is matched with b, it can't be matched with b^*. Then b^* is matched with some $a^{**} <_b a$ in S'. This concludes the instability property of (a, b^*), contradicting that S' is unstable, and hence contradicting that a is b-possible.

We conclude the induction hypothesis.

□

[6] The case where b is not matched is still applied here.

[7] Then $1 = 0$, and we wouldn't want that, would we?

Again, we claim that we could broaden the scope of the Theorem, by not imposing any restrictions in the Matching Problem.

There were no restrictions on the order of the proposals: we didn't bother imposing a particular one, as long as it respected the conditions mentioned. It is so because now that we know that the Gale-Shapley algorithm gives a $\mathcal{B}$-Optimal Stable Matching, the uniqueness of the optimality implies that all the orders may lead to the same Stable Matching: the $\mathcal{B}$-Optimal Stable Matching.

We conclude this section with some non-trivial observations that we leave as a complementary exercise, very useful for getting used to the details of the demonstrated lemmas.

Exercise 1. *Soundness of the Gale-Shapley Algorithm*

Show that Gale-Shapley Algorithm attains always the same result, independently of the order in which the proposals are done.

Exercise 2. *Prove that the number of proposals done along the Gale-Shapley algorithm is independent of the order in which the proposals are done.*

Exercise 3. *Suppose that we have a Matching Problem with n girls and boys, the boys being represented by $A_1, \cdots, A_n$, with no non-willing pairs and $n_i = 1$. Suppose that for each j, a_j is A_j's preferred girl, among the ones that are A_j-possible.*

Prove that if $i \neq j$ then $a_i \neq a_j$.

Exercise 4. *Suppose that the men's preference matrix M is a latin square (where we arrange the preferences of each men in a line). Show that all columns of M give a Stable Matching if and only if the women's preference matrix W is the dual of M, this is, if A_i ranks a_j in position k, then a_j ranks A_i in position $n+1-k$.*

3 Corrupting the algorithm

We will now suppose that there is a player $M \in \mathcal{B}$ that wants to improve his match by lying when asked for its own preference list. The goal will be to prove that the lie is in vain, and that M will not improve his match by lying.

Theorem 1. *Let $M \in \mathcal{B}$, and consider a Matching Game between $\mathcal{A}$ and $\mathcal{B}$, solved by the Gale-Shapley Algorithm.*

Then, if everyone besides $M \in \mathcal{B}$ is truthful in his preferences, M cannot achieve a better match by lying.

Demonstration: See [3], Theorem 9.

□

The more general statement can be found later in the same paper:

Theorem 2. *Take a subset $\mathcal{M} \subseteq \mathcal{B}$ of players, and consider a Matching Game between $\mathcal{A}$ and $\mathcal{B}$, solved by the Gale-Shapley Algorithm.*

Then, if everyone besides the elements $b \in \mathcal{M} \subseteq \mathcal{B}$ is truthful in his preferences, there is some $b \in \mathcal{M}$ that does not improve his status.

Demonstration:
See [3], Theorem 17.

□

We give an example of a coalition that, although doesn't improve the status of every coercer, improves the status of some of them, and no coercers get a worst status:

Take $\mathcal{A} = \{A, B, C\}$ and $\mathcal{B} = \{a, b, c\}$. Consider the real preferences as:

$$A_A = (b, a, c)$$
$$A_B = (a, c, b)$$
$$B_a = (A, B, C)$$
$$B_b = (B, A, C)$$
$$B_c = (B, C, A)$$

For the following analysis, the preferences list of C doesn't matter. The final Matching, obtained by the Gale-Shapley Algorithm, is:

$$S = \{(A, b), (B, a), (C, c)\}$$

Suppose now that b and c form a coalition, lying about c's preference list. They say that his list is:

$$E_c = (C, B, A)$$

The new Matching obtained by the Gale-Shapley Algorithm is:

$$S' = \{(A, a), (B, b), (C, c)\}$$

Note that a and b have improved their statuses, but c (who was the one who lied) has not, respecting Theorem 1. This match seems very good, since no $x \in \mathcal{B}$ has lost status, but is not stable, because (B, c) forms an unstable pair.

We will see that the Gale-Shapley Algorithm gives to $\mathcal{A}$ the worst stable matching. Also, by lying, those from $\mathcal{A}$ can attain better results.

Part IV
Solution to the original Problem

1 Some definitions

The solution presented here is a slight adaptation from [1].

Firstly, we will enrich our notions in graph theory with some definitions.

Definition 4. *$K_{A,B}$*

By $K_{A,B}$ we mean a graph with vertices $V = A \uplus B$ and edges $E = \{(a,b) \mid a \in A,\ b \in B\}$.

Definition 5. *Colorings*

By a coloring of the vertices of a graph G, we mean a function $C : V(G) \to \mathbb{Z}_{\geq 1}$.

A coloring is **admissible** if there is no edge $(a,b) \in E(G)$ such that $C(a) = C(b)$.

The chromatic number of a graph, $\chi(G)$, is the minimum $m \in \mathbb{Z}_{\geq 0}$ such that there is an admissible coloring function f that never exceeds m in any vertex: $f(v) \leq m$, $\forall v \in V$.

Definition 6. *Coloring restricted to functions on the vertices*

Let $f : V(G) \to \mathbb{Z}_{\geq 1}$ be a function.

We say that a graph G is **list-colorable to** f, or f**-colorable** if, for any family of sets $\{C_v\}_{v \in V(G)}$ with $\#C_v = f(v)$, there is an admissible coloring function g of G such that $g(v) \in C_v$ for each vertex v.

We call **list-chromatic number** of a graph $\chi_l(G)$ to the minimum m such that G is m-colorable. [8]

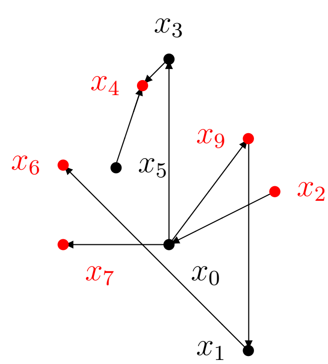

Figure 3: In red, one kernel of the graph.

[8] Here we regard m as the constant function with value m.

Definition 7. *Kernel*

In an oriented graph G, a kernel is a non-empty subset $K \subseteq V(G)$ such that:

- $I[S] = 0_n$, i.e. no pair of vertices $a, b \in S$ has an edge $(a, b) \in E(G)$ between them[9];
- $\forall_{x \in V(G) \setminus S} \ \exists_{s \in S} : \ (x, s) \in E(G)$. I.e., every vertex in G either is in S or points to some vertex in S.

Later, we will call a graph such that every generated subgraph has a kernel, a **Kernelized graph**. For the sake of an example, Figure 3 shows a kernel of a graph.

Definition 8. *Line graph*

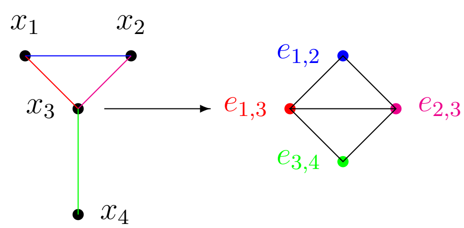

Figure 4: A graph and its line graph.

Given a graph G, its line graph $L(G)$ is a graph in which each vertex represents an edge of G such that

- $V(L(G)) = E(G)$
- $(a, b) \in E(L(G))$ if a and b share a vertex in G.

Figure 4 shows an example of a graph and its line graph.

2 Main Theorem

We can now state and prove the Dinitz Conjecture. Its breakdown will begin with a simple construction of a graph ϕ, and two lemmas. The first one proves that ϕ is a kernelized graph using stable matchings. The second one proves a very natural but tricky fact about coloring an oriented graph, providing a sufficient condition on a function f in order to ϕ be f-colorable.

The link between these two lemmas and the conjecture is very thin so the theorem follows naturally.

[9]The notation $I[S]$ stands for the subgraph generated by the set of vertices S

Theorem 3. *Fred Galvin's Theorem*

Let Q be a Dinitz's Matrix, with choice sets $C_{i,j}$. Then Q is solvable.

Demonstration

We already observed that this problem is equivalent to finding a coloring c of the vertices of $L(K_{n,n})$ such that $c(a,b) \in C_{a,b}$. In our notation it boils down to prove that $L(K_{n,n})$ is n-colorable.

Consider an ordering of the edges in $L(K_{n,n})$, henceforth denoted by ϕ, that fulfills the following:

- $\delta_+(v) = n-1 \ \forall v \in V$, i.e., the outer degree of each vertex is $n-1$.
- For each line i, if $(i, j_1) \to (i, j_2)$ and $(i, j_2) \to (i, j_3)$ are arrows in ϕ, then $(i, j_1) \to (i, j_3)$ is also an arrow in ϕ. This means that each line i defines a total order $>_i$ in the set $\{1, \cdots, n\}$ of the columns.
- For each column j, if $(i_1, j) \to (i_2, j)$ and $(i_2, j) \to (i_3, j)$ are arrows in ϕ, then $(i_1, j) \to (i_3, j)$ is also an arrow in ϕ. This means that each column j defines a total order $>_j$ in the set $\{1, \cdots, n\}$ of the lines.

This order relation is given by $(a, b) \to (a, c) \Leftrightarrow c >_a b$

For the sake of an example, note that the orientation given below, named once and for all ϕ, achieves the envisaged.

Columns: link $(i, j) \to (i', j)$ if $(i + j - 2) \mod n < (i' + j - 2) \mod n$.
Rows: link $(i, j) \to (i', j)$ if $(i + j - 2) \mod n > (i' + j - 2) \mod n$.

Lemma 3. *ϕ defines in $L(K_{n,n})$ a kernelized graph.*

Demonstration:

Consider a subgraph S, induced by the vertices V of $L(K_{n,n})$. Since V is a set of vertices of $L(K_{n,n})$, consider the set E of corresponding edges in $K_{n,n}$.

Regarding the orders defined by ϕ, and considering the possible matches given by E (hence there may be some non-willing matches), with $n_i = 1$ we are in the presence of a Matching Problem. The Gale-Shapley Algorithm guarantees that there is some stable Matching E^o. Let V^o be the corresponding set of vertices in $L(K_{n,n})$.

We claim that V^o is a kernel of S and we verify the two necessary conditions. The imposition $n_i = 1$ implies that each matching is disjoint: if $a, b \in E^o$ share a vertex t then it violates a condition to be a solution of our matching problem, namely, the number of occurrences of t in the matching should be one or none. Hence no two edges in E^o share a vertex, which means that they are not connected in $L(K_{n,n})$ and then V^o is an independent set.

Take some $v \in V \setminus V^o$ and the respective $e \in E \setminus E^o$. We will prove that there is some edge $(v, x) \in E(S)$[10] where $x \in V^o$. Since $e \notin E^o$, and E^o is a stable matching, $e = (a, b)$

[10]The edges in the graph S

is not an unstable pair so there is an edge (wlog, we say that the stability comes from a) $e^* = (a, b^*) \in E^o$ such that $b^* >_a b$, which means the arrow $(a, b) \to (a, b^*)$ is regarded in ϕ, which translates to and edge $e \to x \in \phi$ where $x = (a, b^*) \in V^o$ as we want.

We conclude that V^o is a kernel in S.

■

This lemma shows that kernels aren't so complicated, because kernels are equivalent to stable matchings. The following lemma isn't much of a surprise:

Lemma 4. *If G is a kernelized graph, and $f : V(G) \to \mathbb{Z}_{\geq 0}$ such that for each vertex $v \in V(G)$ we have $f(v) > \delta_+(v)$, then G is f-colorable.*

Demonstration: We will use induction in the number of available colors, this is, in $\#\bigcup_{v \in V} C_v$. Note that we are taking into account the inductive structure of $\{C_v\}_{v \in V}$ but also of the functions f and the graphs G, so a precise induction will not be demonstrated here.

For the base case, there is only one available color, so $1 = f(v) > \delta_+(v)$, then $\delta_+(v) = 0$ and the graph is empty. Any colouring is trivially admissible.

For the induction step, consider a specific color $r \in \bigcup_{v \in V} C_v$. We will color some of the vertices of G with r and the others by induction hipotesys by considering a new f', a new graph G' and new lists C'_v such that $\bigcup_{v \in V(G')} C'_v = \bigcup_{v \in V(G)} C_v \setminus \{r\}$.

Consider all the vertices V that can be colored by r. Since G is kernelized, $I[V]$ has a kernel S^o: those are the vertices that will be painted with the color r.

Consider the graph G' inducted by $V(G) \setminus S^o$. Consider the function $f' : V(G') \to \mathbb{Z}_{\geq 0}$ such that for $v \in V$, $f'(v) = f(v) - 1$, and for $v^* \notin V$, $f'(v^*) = f(v^*)$. Consider also the lists $C'_v = C_v \setminus \{r\}$.

Name δ'_+ the function that counts the outer degree in the graph G', maintaining the notation δ_+ for G.

The graph G' in these conditions fulfills the Lemma requirement: we remark the condition $f'(v) > \delta'_+(v)$ is verified because

- When $v \notin V$, $f(v) = f'(v)$ and $\delta'_+(v) \leq \delta_+(v)$ so we have $f'(v) = f(v) > \delta_+ \geq \delta'_+(v)$

- When $v \in V$, $f(v) = f'(v) + 1$ and, in G, there is an arrow (v, x) for some $x \in S^o$ which vanishes in G', so $\delta'_+(v) \leq \delta_+(v) - 1$. Then $f'(v) = f(v) - 1 > \delta_+(v) - 1 \geq \delta'_+(v)$

In conclusion, by induction hypothesis, G' is f'-colorable and there is an admissible coloring function in $V(G')$. Coloring the other vertices in S^o with r will not change the acceptability of C (notice that S^o is independent), concluding our induction step.

■

We can now conclude the demonstration of the Theorem: since ϕ is kernelized by Lemma 3, and $\delta_+(v) = n - 1 < n \; \forall v \in V$ by definition of ϕ, Lemma 4 gives us that $L(\mathcal{K}_{n,n}) = \phi$ is n-colorable.

□

3 Some related open problems

1. Let G, H be graphs, and suppose $G = L(H)$. Then $\chi(G) = \chi_l(G)$.

   We know that the very same question regarding general graphs G is false. A possible counter-example is given below, considering the graph $K_{4,2}$ with the following lists:

   $L_1 = \{1,2\}$; $L_2 = \{3,4\}$; $L_3 = \{1,3\}$; $L_4 = \{4,2\}$; $L_5 = \{1,4\}$; $L_6 = \{3,2\}$;

   Naturally, $\chi(K_{4,2}) = 2$ since $K_{4,2}$ is bipartite. However, there is no admissible list-colouring to this presentation, hence $\chi_l(K_{4,2}) > 2$

2. Define k_n as the least integer k such that $G = L(K_{n,n})$ is f-colorable, where $f(v) = n$, $\forall v \in V(G) \setminus \{v_0\}$ and $f(v_0) = k$, where $v_0 \in V(G)$ is arbitrary.

   Is it true that $k_n < n$, where $n > 2$ is an integer?

   For this, we know that $k_3 = 2$, and that $k_n > \frac{n}{2}$ (of course we know that $k_n \leq n$, by Fred Galvin's Theorem). This question was pointed out by Fred Galvin in his paper with the proof of the Dinitz's Conjecture.

Part V
Following work

The following work isn't available in English yet. In it, we present a new algorithm, a general treatment of the stable matchings and a definition of the stable matching polytope and of the stable matching lattice.

A closer relationship between the stable matching polytope and the stable matching lattice is then proven at the end of the work.

Part VI
Acknowledgments

I would like to thank my tutor José Fachada for guiding me in this new theme. I really appreciated the opportunity the *Calouste Gulbenkian* foundation gave me to work in this beautiful project.

I would also like to acknowledge the *Seminário Daigonal* team, from the Mathematical Department of Lisbon Tech, the help and the opportunity, the patience and the encouragement by letting me bring this work to those who were interested.

Part VII

References

[1] Martin Aigner, *Proofs From The Book*;

[2] David Gale and L. Shapley, *College Admissions and the Stability of Marriages*;

[3] L.E. Dublin and D.A. Freedman, *Machiavelli and the Gale-Shapley Algorithm*;

[4] David Gale and Marilda Sotomayor, *Machiavelli and the stable-matching problem*;

[5] Michael Balinski and Guillaume Raiter, *Of Stable Mariages and Graphs, and strategy, and polytopes*;

[6] Michael Balinski and Guillaume Raiter, *Graphs And Mariages*;

[7] Fred Galvin, *The List Cromatic Index of a Bipartite Multigraph*;

[8] Donald E. Knuth, *Martiages stables et leurs relations avec d'autres problèmes combinatoires*; The exercises 3 and 4 were selected from this reference.