Graph Algorithms

TD4 : Matchings

1 Consequences of Hall's Theorem

Let $H = (U, V, E)$ be a d-regular bipartite graph (all degrees in G equal d), for some $d \geq 1$.

1. *Show that G contains a perfect matching.*

   We first show that $|U| = |V|$. Indeed, each edge e has an extremity in U, and the other in V. Hence $d|U| = \sum_{u \in U} \deg(u)$ counts each edge $e \in E$ exactly once; likewise for V. So we have $d|U| = |E| = d|V|$, which implies that $|U| = |V|$.

   We now show that H satisfies Hall's condition. Let $X \subseteq U$, and let $Y = N(X)$. Let E_X and E_Y be the sets of edges incident to X and Y respectively. Note that every edge $e \in E_X$ is incident to Y, hence $E_X \subseteq E_Y$. Then, we have $d|X| = |E_X| \leq |E_Y| = d|Y|$, which implies that $|X| \leq |Y| = |N(X)|$.

   By Hall's theorem, there exists a matching M that saturates U, and since $|U| = |V|$, M also saturates V; this is a perfect matching of H.

2. *Show that $\chi'(H) = d$.*

   We show this by induction on d. If $d = 1$, then H is a matching and so $\chi(H) = 1$. Assume now that $d \geq 2$. We have seen that H contains a perfect matching M; let $H' \coloneqq H \setminus M$. Then H' is $(d-1)$-regular, so by the induction hypothesis there exists a proper $(d-1)$-edge-colouring c of H'. Set $c(e) \coloneqq d$ for every $e \in M$; this extends c into a proper d-edge-colouring of H.

2 Vertex Cover

Let G be a graph. We denote $\nu(G)$ the size of a maximum matching in G, and $\tau(G)$ the size of a minimum vertex cover of G.

1. *Show that $\nu(G) \leq \tau(G) \leq 2\nu(G)$.*

   Let M be a maximum matching of G, and let X be a minimum vertex cover of G. Since the edges in M share no end-points, each of them is covered by a distinct vertex in X, hence $\tau(G) = |X| \geq |M| = \nu(G)$. Since M is maximum, every edge $e \in E(G)$ is incident to at least one edge $e' \in M$, hence to one extremity of e'. So $V(M)$ is a vertex cover of G, of size $2|M| = 2\nu(G)$.

2. Write a polynomial algorithm that returns a 2-approximation of a minimal vertex cover of G.

Algorithm 1: VertexCover2Approx

```
Data: G: graph on n vertices
M ← maximum matching of G (cost O(n^2.5)
return V(M)
```

This algorithm returns a vertex cover of G, of size $2\nu(G) \leq 2\tau(G)$, so at most twice the size of an optimal solution.

3 More on Kőnig's Theorem

1. *Prove that the following is an equivalent statement of Kőnig's Theorem. For every bipartite graph H on n vertices, $\alpha(H) = n - \nu(H)$.*

   In order to show that both statements of Kőnig's Theorem are equivelent, let us show that $\alpha(H) = n - \tau(H)$.

   Let I be a maximum independent set of H, we first prove that $\overline{I}$ is a vertex cover of H, of size $n - \alpha(H)$, which implies that $\tau(H) \leq n - \alpha(H)$. Assume otherwise that some edge $e \in E(H)$ is not covered by $\overline{I}$. So both its extremities are in I, which contradicts that I is an independent set.

   Let now X be a minimum vertex cover of H, we prove that $\overline{X}$ is an independent set of H, of size $n - \tau(H)$, which implies that $\alpha(H) \geq n - \tau(H)$ and so that $\tau(H) \geq n - \alpha(H)$. Assume otherwise that there is an edge e induced by $\overline{X}$; this means that no extremity of e is in X, hence e is not covered by X, a contradiction.

2. *Write an algorithm that returns a maximum independent set of any given bipartite graph. We suppose that we have access to an algorithm `maxMatching` that returns a maximum matching of any (bipartite) input graph on n vertices in time $O(n^{2.5})$.*

   We follow the procedure described in the proof of Kőnig's Theorem in order to construct a minimum vertex cover of H, given a maximum matching M.

Algorithm 2: MaxIndependentSet

Data: $H = (X, Y, E)$: bipartite graph
$M \leftarrow$ maximum matching of H
$U \leftarrow X \setminus V(M)$
$R \leftarrow U$
while $N(R) \neq \varnothing$ **do**
  $A \leftarrow N(R)$ (by construction, $A \subseteq Y$)
  $B \leftarrow N_M(A)$ (by construction, $B \subseteq X$)
  $R \leftarrow R \cup A \cup B$
end
$S \leftarrow (X \setminus R) \cup (Y \cap R)$ (S is a minimum vertex cover of H)
return $V(H) \setminus S$

The total complexity of the while loop is that of an exploration of the graph (through alternating paths), so $O(|E(H)|)$. The complexity of the algorithm is therefore dominated by that of finding a maximum matching, hence $O(n^{2.5})$.