Optimal Weighted General Factors

Student: Lam Hoi Kei   Project supervisor: Shao Shuai

1 Introduction

An undirected graph $G = (V, E)$ consists of a set of vertices V and a set of edges E. A matching M in an undirected graph is a subset of the edge set E that have no vertices in common. It is called a perfect matching or 1-factor if each vertex in G is incident with an edge of M. General factor problem (GFP) is a generalization of matching problem. In which each vertex $v \in V$ has a set of feasible degree which is called a degree constraint $\pi(v) \subseteq \{0, 1, ..., \deg_G(v)\}$ where . The weighted general factor problem (WGFP) further establishes a weight $w(e) \in \mathbb{R}$ to every edge $e \in E$ and the goal is to find a optimal general factor that maximizes the total weight. Previous work has proved that GFP is NP-complete when a degree constraint has a gap of length at least 2. Polynomial time algorithms are discovered for some cases of WGFP for interval or parity interval degree constraints by reducing them to weighted matching or perfect matching problem using gadget constructions. In [1], an $O(mn^6)$ algorithm and a weakly polynomial time algorithm $O(\log Wmn^6)$ are introduced to solve the unweighted GFP and the weighted version respectively where $n = |V|,\ m = |E|,\ W = \max_{e \in E} w(e)$. Then, [2] proved that the method of gadget construction cannot be used to solve other cases of WGFP and also provided a strongly polynomial time algorithm for WGFP allowing degree constraints as intervals, parity intervals, $\{p_v, p_v + 1, p_v + 3\}$ and $\{p_v, p_v + 2, p_v + 3\}$ where $p_v \in \{0, ..., \deg_G(v) - 3\}$. In this project, we hope to solve WGFP with each degree constraint containing gap of length at most 1 in polynomial time which is the strongest generalizations of matching that was not proven NP-hard. This poster does not propose any complete algorithm for this problem, but will discuss some ideas to approach the problem.

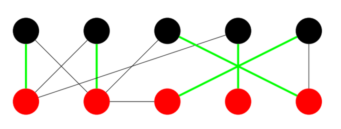
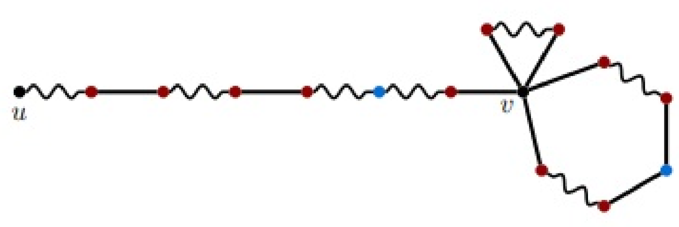


Fig 1: Assume all vertices have degree constraint $= \{1\}$. The set of green edges is a perfect matching or 1-factor, forming a bipartite graph.



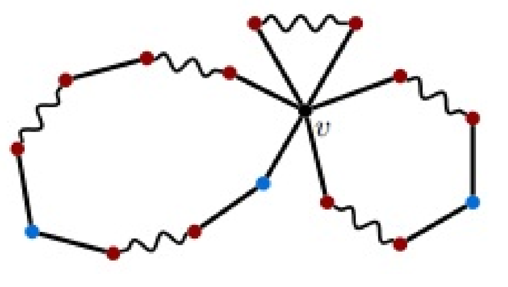


Fig 2&3: Examples of canonical paths. Solid and zavy edges belong to F&F' respectively. π(red vertices)$=\{1\}$. π(blue vertices) $= \{0, 2\}$. Fig 2: $\pi(v) = \{0, 1, 3, 5, 6\}$. Fig 3: $\pi(u) = \{0, 1\}$, $\pi(v) = \{0, 1, 3, 5\}$

2 Method

Firstly, let us recall some important concepts from [1] and [2] that are useful in this project. Then, we will state the main theorem that we are trying to prove in order to achieve a polynomial time algorithm. Let F be a factor of graph G.

Definition 1. *An alternating path with respect to (w.r.t.) F is a sequence of edges $P = ((v_1, v_2), (v_2, v_3), ..., (v_k, v_{k+1}))$ such that*

- *$\forall i$ such that $1 \leq i \leq k - 1$, exactly one of the edges $(v_i, v_{i+1}), (v_{i+1}, v_{i+2})$ belongs to M*
- *each edge of G occurs in P at most once*
- *if $v_1 = v_{k+1}$, then either both edges (v_1, v_2) and (v_k, v_{k+1}) are in F, or both are not in F*

Definition 2. *A meta cycle $\mathfrak{C}$ w.r.t. F is a sequence of alternating paths of the form $(P(v_1, v_2), P(v_2, v_3), ..., P(v_k, v_1))$ such that $v_1, ..., v_k$ are pairwise distinct.*

Definition 3. *A meta path $\mathfrak{P}(v_1, v_{k+1})$ w.r.t. F is a sequence of alternating paths of the form $(P(v_1, v_2), P(v_2, v_3), ..., P(v_k, v_{k+1}))$ such that $v_1, ..., v_{k+1}$ are pairwise distinct.*

Definition 4. *A factor F' is of neighbouring type of F if there exists a set W such that one of following is satisfied*

- *$|W| = 0$*
- *$|W| = 2$ and $\forall w \in W$, $deg_F(w)$ and $deg_{F'}(w)$ are adjacent, that is $\max(deg_F(w)) + 1 = \min(deg_{F'}(w))$ or $\max(deg_{F'}(w)) + 1 = \min(deg_F(w))$*
- *$|W| = 1$ and $\forall w \in W$, there exists k, such that $deg_k(w)$ is adjacent to $deg_F(w)$ and $deg_{F'}(w)$*

Definition 5. *A canonical path $S(v_1, v_k)$ w.r.t F in graph G consists of meta-cycles $\mathfrak{C}_1, \mathfrak{C}_2, ..., \mathfrak{C}_p$ incident to a vertex v_1 and $\mathfrak{C}_1', \mathfrak{C}_2', ..., \mathfrak{C}_p'$ incident to vertex v_k. In case $v_1 \neq v_k$, there is a meta-path $\mathfrak{P}(v_1, v_k)$. The application of all meta-cycles and the meta-path to M results in a factor F' of neighbouring type to F. S is a basic (canonical) path if no proper subset $S' \subsetneq S$ is a canonical path and $w(S') \geq w(S)$ or $w(S') > 0$.*

By the above construction, we can obtain 2 important results. Suppose any 2 factors F and F^*.

Firstly, the symmetric difference of F and F^*, denoted as $F \Delta F^* = \bigcup_{i=1}^{n} S_i \cup \bigcup_i^l C_i$, where C_i is an alternating cycle and S_i is a basic path w.r.t F_{i-1} if we denote $F_0 = F \Delta \bigcup_i^l C_i$ and $F_i = F_{i-1} \Delta S_i$. Also, $F_k = F^*$.

Secondly, if $\exists$ a factor of greater weight than F, then $\exists$ another factor of greater weight than F that is of neighbouring type to F.

Thus, the algorithm is obtained. Firstly, run Cornuejol's algorithm [3] to return an arbitrary factor F if there is one. Then, iteratively find neighbouring type to F with greater cardinality until it is maximum. There are at most 2 vertices whose degree is not restricted to $deg_F(v)$ so possible sets are $O(n^2)$. There are m edges and uniform matching can be found in $O(n^4)$. Thus, runtime $= O(mn^6)$. For weighted version, treat the weights as binary number and run the above algorithm for each digit which takes $O(\log W)$ times.

As for Shao's algorithm, denote $\{p_v, p_v + 1, p_v + 3\}$ as $\mathfrak{T}_1$ and $\{p_v, p_v + 2, p_v + 3\}$ as $\mathfrak{T}_2$. Denote $\mathfrak{T}_1 \cup \mathfrak{T}_2 = \mathfrak{T}$ and $T_\omega = \{v \in V | \pi(v) \in \mathfrak{T}\}$

Definition 6. *If $\pi(v) = \{p_v, p_v + 1, p_v + 3\}$, $D_v^0 = \{p_v + 1, p_v + 3\}$, $D_v^1 = \{p_v\}$. If $deg_F(v) = p_v$, $D_v^F = \{p_v\}$. Otherwise, $D_v^F = \{p_v + 1, p_v + 3\}$.*

If $\pi(v) = \{p_v, p_v + 2, p_v + 3\}$, $D_v^0 = \{p_v, p_v + 2\}$, $D_v^1 = \{p_v + 3\}$. If $deg_F(v) = p_v + 3$, $D_v^F = \{p_v + 3\}$. Otherwise, $D_v^F = \{p_v, p_v + 2\}$.

Definition 7. *("neighbouring types") $\pi_W^F(v) = \pi(v) \backslash D_v^F$ for $v \in W$; $\pi_W^F(v) = D_v^F$ for $v \in T_\omega \backslash W$; $\pi_W^F(v) = \pi(v)$ for $v \in V \backslash T_\omega$*

Notice that the newly defined degree constraints are all interval or parity interval. By the properties of subcubic graph, similar result can be obtained. Suppose F is a factor and F is optimal for some $u \in T_\omega$ and D_u^0 as its degree constraint. Then, a factor F' is optimal globally if and only if $w(F') \geq w(F)$ and $w(F') \geq$ optimal factor for every W where $u \in W \subseteq T_\omega$ and $|W| = 1$ *or* 2. As a result, we can first find an optimal neighbouring factor F^{opt} with all degree constraint as intervals or parity intervals by recursively picking $u \in T_\omega$ and set its degree constraint to be D_u^0 if there exists a factor, otherwise, set it to be D_u^1. Then, for all $v \in T_\omega$, set $W = u \cup v$ and set degree constraint as $\pi_W^{F^{opt}}$. Find optimal factor F^W in this case if exists. If $w(F^W) > w(F^{opt})$, then set F^{opt} to be F^W.

Being inspired by the above algorithms with structures like basic canonical paths, we hope to prove that some local optimal factor can be updated into global optimal factor by polynomial updates, which gives the intuition for the theorem that we are trying to prove.

Theorem 1. *Suppose $\mathfrak{T}_1' = \{0, 2, 4, \ldots, 2p, 2p + 1\}$, $\mathfrak{T}_1' = \{0, 1, 3, \ldots, 2q + 1\}$. Denote $\mathfrak{T}' = \mathfrak{T}_1' \cup \mathfrak{T}_2'$ and $T^{F \Delta F^*} = \{v \in V | \pi(v) \in \mathfrak{T}'$ and $deg_F(v) \neq deg_{F^*}(v)$ mod 2$\}$. Let $u \in F \cap F \Delta F^*$ such that $\pi(u) \in \mathfrak{T}_1'$. If F^* is optimal and $|T^{F \Delta F^*}| \geq 4$, then $\exists F'$ such that*

- *$w(F') > w(F)$*
- *F' contains at most 2 vertices with different parity as the corresponding vertices in F*
- *$deg_{F'}(u) \equiv deg_F(u)$ mod 2*

3 Discussion

The proof is not complete and this is just an idea. Should you have any queries or ideas, please feel free to contact hklamar@connect.ust.hk

4 Reference

[1] Szymon Dudycz and Katarzyna Paluch. Optimal general matchings. *arXiv* : 1706.07418v3, *version* 3, 2021.

[2] Shuai Shao, Stanislav Živný. A Strongly Polynomial-Time Algorithm for Weighted General Factors with Three Feasible Degrees. *International Symposium on Algorithms and Computation* (*ISAAC*) 2023.

[3] Gérard Cornuéjols. General factors of graphs. *Journal of Combinatorial Theory, Series B*, 45(2):185 – 198, 1988.