A Note On Finding Minimum Mean Cycle

Mmanu Chaturvedi*, Ross M. McConnell

Department of Computer Science,
Colorado State University, Fort Collins, CO, 80523-1873, USA

Abstract

In a directed graph with edge weights, the *mean weight* of a directed cycle is the weight of its edges divided by their number. The *minimum cycle mean* of the graph is the minimum mean weight of a cycle. Karp gave a characterization of minimum cycle mean and an $O(nm)$ algorithm to compute it, where n is the number of vertices and m is the number of edges. However, an algorithm he suggested for identifying a cycle with this mean weight is not correct. We propose an alternative.

Keywords: analysis of algorithms, design of algorithms, graph algorithms

1. Introduction

Given a digraph, $G(V, E)$ and a weight function, $f : E \to \mathbf{R}$, let the *weight* of any "edge progression" (walk) $\sigma = (e_1, e_2, \ldots, e_p)$ be defined by $w(\sigma) = \sum_{i=1}^{p} f(e_i)$. Let $n = |V|$ and $m = |E|$. Let the *length* of the edge progression be p. Let the *mean weight* of a directed cycle be its weight divided by its number of edges, and let the *minimum cycle mean*, λ^*, of a digraph be the minimum mean weight of its directed cycles. Karp [1] gave a characterization of the minimum cycle mean over all the directed cycles in G.

Since any directed cycle is confined to a single strongly-connected component, the algorithm can be applied separately to the subgraph induced by each component and returning the minimum cycle mean over all the components. Henceforth, we assume that the graph is strongly connected.

Let s be an arbitrary *start vertex* of G. For each vertex $v \in V$, let $F_k(v)$ be the minimum weight of any edge progression of length exactly k from s to v, of ∞ if no such edge progression exists. Karp proves the following:

Theorem 1.

$$\lambda^* = \min_{v \in V} \max_{0 \le k \le n-1} \left[\frac{F_n(v) - F_k(v)}{n - k} \right]$$

*Corresponding author

Email addresses: mmanu@cs.colostate.edu (Mmanu Chaturvedi), rmm@cs.colostate.edu (Ross M. McConnell)

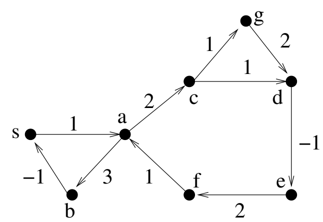

Figure 1: Counterexample to Karp's algorithm for constructing a minimum mean weight cycle.

$F_k(v)$ can be found in $O(nm)$ time for all $v \in G$ and all $k \in \{0, 1, \ldots, n\}$ using a dynamic programming algorithm that assigns $F_k(v)$ to an entry of the table indexed by (k, v). This gives an $O(nm)$ algorithm for finding the value of the minimum cycle mean, by Theorem 1.

For each $v \in V$ and $\{k | 0 < k \le n\}$, the dynamic programming algorithm can assign a backpointer from (k, v) to an entry $(k-1, w)$, giving the predecessor w on an edge progression of length k and weight $F_k(v)$ from s to v. This allows a minimum weight edge progression of length k from s to v to be reconstructed, by following backpointers.

2. Finding a cycle of minimum mean

Let a *minimizer* be a vertex v such that $\max_{0 \le k \le n-1}(F_n(v) - F_k(v))/(n - k) = \lambda^*$, and let a *minimizing pair* be a minimizer v and integer k such that $0 \le k \le n-1$ and k maximizes $(F_n(v) - F_k(v))/(n-k)$. Karp suggests the following for finding a cycle of minimum mean weight.

> If the actual cycle yielding the minimum cycle mean is desired, it can be computed by selecting the minimizing [pair] v and k, finding a minimum-weight edge progression of length n from s to v, and extracting a cycle of length $n - k$ occurring within that edge progression.

He does not supply a proof that this procedure is correct, and Figure 1 gives a counterexample. In the figure, the minimum cycle mean is 1, which is the mean weight of the cycles (s, a, b) and (a, c, d, e, f). The value of $F_n(g) = F_8(g)$ is 9, given by the edge progression (walk) $(s, a, c, d, e, f, a, c, g)$, and the value of $F_6(g)$ is 7, given by (s, a, b, s, a, c, g). Since $[F_8(g) - F_6(g)]/(8-6) = 1$, which is the minimum cycle mean, g and 6 are a minimizing pair. There is supposed to be a cycle of length $8 - 6 = 2$ on the walk $(s, a, c, d, e, f, a, c, g)$, but there is no cycle of length 2 in the graph.

Karp's proof of Theorem 1 shows that for some minimizing pair v and k, and some walk W of weight $F_n(v)$ from s to v, the last $n - k$ edges of W are a cycle of minimum mean weight. In Figure 1, d and 3 are such a pair; the last $n - 3 = 5$ edges of $(s, a, c, d, e, f, a, c, d)$ are a cycle of minimum mean weight. For v, the proof uses a vertex that lies on a cycle of minimum mean weight, and shows that there exists a walk from s to v of length n and weight $F_n(v)$ such that the last $n-k$ edges of the walk are a cycle of minimum mean weight. However, these conditions do not apply for all minimizing pairs. The example of g in Figure 1, explained above, shows that a minimizer need not lie on a cycle of minimum cycle mean. Even for a vertex v such that the assumption applies, it is not true for every minimizing pair that v is a part of: in the figure, $(d, 6)$ is a minimizing pair for which the assumption applies, but $(d, 7)$, given by $(s, a, c, d, e, f, a, c, d)$ and (s, a, b, s, a, c, d), is one where it is not.

One fix would be to apply his suggested algorithm to each minimizing pair, since the assumptions apply to at least one of them. However, even when the assumptions apply to a given minimizing pair (v, k), the existence of more than one cycle of minimum mean weight can mean that there is more than one minimum-weight edge progression of length n from s to v. It may be the case that not all of them satisfy the required conditions, and the dynamic programming algorithm might find one that does not. There is a way around this, but there are $\Theta(n^2)$ minimizing pairs in the worst case, and some care is needed to keep the time bound of this approach to $O(nm)$. The insights developed in the proof of Theorem 1 suggest better alternatives, and the following is particularly simple.

Lemma 1. *Let v be a vertex such that there exists k, where v and k are a minimizing pair. Every cycle on the length n edge progression from s to v of weight $F_n(v)$ is a cycle of minimum mean weight.*

Proof. Let W be a length n edge progression from s to v of weight $F_n(v)$. Subtracting λ^* from the weight of every edge of G reduces the mean weight of every cycle and edge progression by λ^* in the resulting graph G'. The cycles of minimum mean weight in G are those that have weight 0 in G', if v and k are a minimizing pair, they remain one in G', and W remains a minimum-weight edge progression of length n from s to v in G'. It suffices to show that every cycle on W has weight 0 in G'.

Suppose there's a cycle of positive weight on W. Omission of the cycles on W results in a path P from s to v of weight $w < F_n(v)$ in G'. Let $k' = |P|$. In G', $F_{k'}(v) < F_n(v)$, and $[F_n(v) - F_{k'}(v)]/(n - k') > 0$, which is the minimum cycle mean of G', contradicting that v must be a minimizer in G' by Theorem 1. □

If v is a minimizer, then following backpointers from (n, v) of the table gives a walk of weight $F_n(v)$ from s to v. Since it is longer than $n-1$, it has a cycle, and by the lemma, every cycle on it is a cycle of minimum mean weight. By traversing backpointers marking vertices visited by the walk until a previously marked vertex w is encountered, a cycle of minimum mean weight can be identified in $O(n)$ time.

References

[1] R. M. Karp, A characterization of the minimum cycle mean in a digraph, Discrete mathematics 23 (3) (1978) 309–311.