Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 21, no. 2, pp. 177–181 (2017)
DOI: 10.7155/jgaa.00410

On the Shoshan-Zwick Algorithm for the All-Pairs Shortest Path Problem

Pavlos Eirinakis [1] *Matthew Williamson* [2] *K. Subramani* [3]

[1] Athens University of Economics and Business, Athens, Greece
[2] West Virginia University Institute of Technology, Montgomery, WV
[3] West Virginia University, Morgantown, WV

Abstract

The Shoshan-Zwick algorithm solves the all-pairs shortest paths problem in undirected graphs with integer edge costs in the range $\{1,2,\ldots,M\}$. It runs in $\tilde{O}(M \cdot n^{\omega})$ time, where n is the number of vertices, M is the largest integer edge cost, and $\omega < 2.3727$ is the exponent of matrix multiplication. It is the fastest known algorithm for this problem. This paper points out and corrects an error in the Shoshan-Zwick algorithm.

Submitted:	Reviewed:	Revised:	Accepted:	Final:
August 2016	November 2016	December 2016	December 2016	December 2016
		Published: January 2017		
		Article type: Concise paper	Communicated by: J. S. B. Mitchell	

The first author was co-financed by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Thales. Investing in knowledge society through the European Social Fund. (MIS: 380232) The second and third authors were supported in part by the National Science Foundation and Air Force of Scientific Research through Awards CNS-0849735, CCF-1305054, and FA9550-12-1-0199.

E-mail addresses: peir@aueb.gr (Pavlos Eirinakis) matthew.williamson@mail.wvu.edu (Matthew Williamson) ksmani@csee.wvu.edu (K. Subramani)

1 Introduction

In this paper, we revise the Shoshan-Zwick algorithm [2] to correct an error. Recall that the Shoshan-Zwick algorithm solves the all-pairs shortest paths (APSP) problem in undirected graphs, where the edge costs are integers in the range $\{1,2,\ldots,M\}$. This is accomplished by computing $O(\log(M\cdot n))$ distance products of $n\times n$ matrices with elements in the range $\{1,2,\ldots,M\}$. The algorithm runs in $\tilde{O}(M\cdot n^{\omega})$ time, where $\omega < 2.3727$ is the exponent for the fastest known matrix multiplication algorithm [3]. This paper identifies and resolves an error with the algorithm. Additional details including a description of the algorithm, a counter-example that identifies the error in the algorithm, and a discussion concerning the efficacy of the algorithm can be found in [1].

2 The Errors in the Algorithm

In this section, we describe what causes the erroneous behavior of the Shoshan-Zwick algorithm. Recall that Δ is the matrix that contains the costs of the shortest paths between all pairs of vertices after the algorithm terminates. Moreover, let $\delta(i,j)$ denote the cost of the shortest path between nodes i and j. After the termination of the algorithm, we must have $\Delta_{ij}=\delta(i,j)$ for any $i,j\in\{1,\ldots,n\}$. However, it may be the case that $\Delta_{ij}\neq\delta(i,j)$ for some i,j at termination. The exact errors of the algorithm are as follows:

1. $\mathbf{R}$ is not computed correctly.

2. $\mathbf{B}_0$ is not computed correctly.

3. Δ is not computed correctly, since $M\cdot\mathbf{B}_0+\mathbf{R}$ is part of the sum producing it.

In the rest of this section, we illustrate what causes these errors. When we compute $\Delta = M\cdot\mathbf{B}_0+\mathbf{R}$, observe that the matrices $\mathbf{B}_k$ (for $0\leq k\leq l$) represent the $\lceil\log_2 n\rceil$ most significant bits of each distance. That is,

$$(\mathbf{B}_k)_{ij} = \begin{cases} 1 & \text{if } 2^k\cdot M \text{ must be added to } \Delta_{ij} \text{ so that } \Delta_{ij}=\delta(i,j) \\ 0 & \text{otherwise} \end{cases},$$

while $\mathbf{R}$ represents the remainder of each distance modulo M. This is also illustrated in [2, Lemma 3.6], where for every $0\leq k\leq l$, $(\mathbf{B}_k)_{ij}=1$ if and only if $\delta(i,j) \mod 2^{k+m+1}\geq 2^{k+m}$, while $\mathbf{R}_{ij}=\delta(i,j) \mod M$. Hence, for every i,j, we must have

$$(M\cdot\mathbf{B}_0+\mathbf{R})_{ij} = \delta(i,j) \mod 2^{m+1}. \tag{1}$$

The first error of the algorithm arises immediately from the key observation that $\mathbf{P}_0$ can have entries with negative values. This means that $\mathbf{R}_{ij}=(\mathbf{P}_0)_{ij} \mod M$ is not correctly calculating $\mathbf{R}_{ij}=\delta(i,j) \mod M$, since $\delta(i,j)\geq 0$ by definition, while $(\mathbf{P}_0)_{ij}$ can be negative.

A closer examination of how $\mathbf{P}_0$ obtains its negative values reveals another error of the algorithm. The following definitions are given in [2, Section 3]. Consider a set

$Y \subseteq [0, M \cdot n]$. Note that $[0, M \cdot n]$ includes any value that $\delta(i,j)$ can take, since n is the number of nodes, and M is the maximum edge cost. Let $Y = \cup_{r=1}^{p}[a_r, b_r]$, where $a_r \leq b_r$, for $1 \leq r \leq p$ and $b_r < a_{r+1}$, for $1 \leq r < p$. Let $\blacksquare_Y$ be an $n \times n$ matrix, whose elements are in the range $\{-M, \ldots, M\} \cup \{+\infty\}$, such that for every $1 \leq i, j \leq n$, we have

$$(\blacksquare_Y)_{ij} = \begin{cases} -M & \text{if } a_r \leq \delta(i,j) \leq b_r - M \text{ for some } 1 \leq r \leq p, \\ \delta(i,j) - b_r & \text{if } b_r - M < \delta(i,j) \leq b_r + M \text{ for some } 1 \leq r \leq p, \\ +\infty & \text{otherwise.} \end{cases} \tag{2}$$

By [2, Lemma 3.5], $\mathbf{P}_0 = \blacksquare_{Y_0}$, where $Y_0 = \{x | (x \mod 2^{m+1}) = 0\}$. Recall that $2^m = M$. Note that by definition of Y_0, when calculating $\mathbf{P}_0 = \blacksquare_{Y_0}$, it can only be the case that $a_r = b_r$. Moreover, $b_r = 2^{m+1} \cdot (r-1)$ for $1 \leq r \leq p$, where p is such that $2^{m+1} \cdot (p-1) \leq M \cdot n < 2^{m+1} \cdot p$. But then:

$$\left(\cup_{r=1}^{p}[b_r - M, b_r + M]\right) \supset [0, M \cdot n]$$

That is, $(\cup_{r=1}^{p}[b_r - M, b_r + M])$ covers all possible values that $\delta(i,j)$ may take for any i, j. Hence,

$$(\mathbf{P}_0)_{ij} = \begin{cases} \delta(i,j) & \text{for } r = 1 \text{ (i.e., if } \delta(i,j) \leq 2^m), \\ \delta(i,j) - b_r & \text{for } 2 \leq r \leq p, \text{ such that } b_r - 2^m < \delta(i,j) \leq b_r + 2^m. \end{cases} \tag{3}$$

Let us examine the values that $(\mathbf{P}_0)_{ij}$ takes by equation (3):

- For $0 \leq \delta(i,j) \leq 2^m$, we have $(\mathbf{P}_0)_{ij} = \delta(i,j) \mod 2^{m+1}$.
- For $2^m < \delta(i,j) < 2^m + 2^m$, we have $(\mathbf{P}_0)_{ij} = (\delta(i,j) \mod 2^{m+1}) - 2^{m+1}$.
- For $2^{m+1} \leq \delta(i,j) \leq 2^{m+1} + 2^m$, we have $(\mathbf{P}_0)_{ij} = \delta(i,j) \mod 2^{m+1}$.
- For $2^{m+1} + 2^m < \delta(i,j) < 2^{m+2} + 2^m$, we have $(\mathbf{P}_0)_{ij} = (\delta(i,j) \mod 2^{m+1}) - 2^{m+1}$.
- And so forth...

More formally, equation (3) can be rewritten as follows:

$$(\mathbf{P}_0)_{ij} = \begin{cases} \delta(i,j) \mod 2^{m+1} & \text{if } \delta(i,j) \mod 2^{m+1} \leq 2^m, \\ (\delta(i,j) \mod 2^{m+1}) - 2^{m+1} & \text{if } \delta(i,j) \mod 2^{m+1} > 2^m. \end{cases} \tag{4}$$

Moreover, equation (4) implies that

$$\text{for } i, j \text{ such that } \delta(i,j) \mod 2^{m+1} \leq 2^m, \text{ we have } 0 \leq (\mathbf{P}_0)_{ij} \leq M, \tag{5}$$

while

$$\text{for } i, j \text{ such that } \delta(i,j) \mod 2^{m+1} > 2^m, \text{ we have } -M < (\mathbf{P}_0)_{ij} < 0. \tag{6}$$

Recall now that we must have $(\mathbf{B}_0)_{ij} = 1$ if and only if $\delta(i,j) \mod 2^{m+1} \geq 2^m$. However, from equations (5) and (6), this does not hold (as claimed in the proof of [2, Lemma 3.6]) for $\mathbf{B}_0 = (0 \leq \mathbf{P}_0 < M)$. Therefore, the algorithm does not compute $\mathbf{B}_0$ correctly.

It is clear that in the presence of these two identified errors (in calculating $\mathbf{R}$ and $\mathbf{B}_0$), the algorithm is not computing Δ correctly.

3 The Revised Algorithm

In this section, we present a new version of the Shoshan-Zwick algorithm that resolves the problems illustrated in Section 2. The first three steps from [2, Figure 2] remain unchanged. We make the following changes in Step 4:

1. We replace $\mathbf{B}_0$ with $\hat{\mathbf{B}}_0$ and set $\hat{\mathbf{B}}_0$ to $(-M < \mathbf{P}_0 < 0)$.
2. In the line $\mathbf{R} = \mathbf{P}_0 \mod M$, we replace $\mathbf{R}$ with $\hat{\mathbf{R}}$ and set $\hat{\mathbf{R}}$ to $\mathbf{P}_0$.
3. We set Δ to $M \cdot \sum_{k=1}^{l} 2^k \cdot \mathbf{B}_k + 2 \cdot M \cdot \hat{\mathbf{B}}_0 + \hat{\mathbf{R}}$.

Note that we have replaced $\mathbf{B}_0$ and $\mathbf{R}$ with $\hat{\mathbf{B}}_0$ and $\hat{\mathbf{R}}$, respectively. The purpose of the change in notation is to show that these matrices no longer represent the incorrect versions from the original (erroneous) algorithm. Step 4 of the revised algorithm is illustrated in Algorithm 3.1.

for $(k \leftarrow 1 \text{ to } l)$ **do**
  $\mathbf{B}_k = (\mathbf{C}_k \geq 0)$
end for
$\hat{\mathbf{B}}_0 \leftarrow (-M < \mathbf{P}_0 < 0)$
$\hat{\mathbf{R}} \leftarrow \mathbf{P}_0$
$\Delta \leftarrow M \cdot \sum_{k=1}^{l} 2^k \cdot \mathbf{B}_k + 2 \cdot M \cdot \hat{\mathbf{B}}_0 + \hat{\mathbf{R}}$
return Δ

Algorithm 3.1: The revised Step 4 of the Shoshan-Zwick algorithm

We now prove that the revised version of the Shoshan-Zwick algorithm is correct.

Theorem 1 *The revised Shoshan-Zwick algorithm calculates all the shortest path costs in an undirected graph with integer edge costs in the range* $\{1, \ldots, M\}$.

Proof: It suffices to show that $2 \cdot M \cdot \hat{\mathbf{B}}_0 + \hat{\mathbf{R}}$ represents what the original algorithm intended to represent with $M \cdot \mathbf{B}_0 + \mathbf{R}$. That is, by equation (1), it suffices to show that $(2 \cdot M \cdot \hat{\mathbf{B}}_0 + \hat{\mathbf{R}})_{ij} = \delta(i,j) \mod 2^{m+1}$, for every $1 \leq i, j \leq n$.

First, we consider the case where $\delta(i,j) \mod 2^{m+1} \leq 2^m$. Equation (5) indicates that $0 \leq (\mathbf{P}_0)_{ij} \leq M$. Hence, $(\hat{\mathbf{B}}_0)_{ij} = 0$ (by $\hat{\mathbf{B}}_0 \leftarrow (-M < \mathbf{P}_0 < 0)$ in the revised algorithm). Moreover, since $\hat{\mathbf{R}}_{ij} = (\mathbf{P}_0)_{ij}$ (by $\hat{\mathbf{R}} \leftarrow \mathbf{P}_0$ in the revised algorithm), we have that $\hat{\mathbf{R}}_{ij} = \delta(i,j) \mod 2^{m+1}$ by equation (4). Thus, $(2 \cdot M \cdot \hat{\mathbf{B}}_0 + \hat{\mathbf{R}})_{ij} = \delta(i,j) \mod 2^{m+1}$.

We next consider the case where $\delta(i,j) \mod 2^{m+1} > 2^m$. Equation (6) indicates that $-M < (\mathbf{P}_0)_{ij} < 0$. Hence, $(\hat{\mathbf{B}}_0)_{ij} = 1$ (by $\hat{\mathbf{B}}_0 \leftarrow (-M < \mathbf{P}_0 < 0)$ in the revised algorithm). Further, $\hat{\mathbf{R}}_{ij} = (\delta(i,j) \mod 2^{m+1}) - 2^{m+1}$ by equation (4). Therefore, $(2 \cdot M \cdot \hat{\mathbf{B}}_0 + \hat{\mathbf{R}})_{ij} = 2 \cdot 2^m \cdot 1 + (\delta(i,j) \mod 2^{m+1}) - 2^{m+1} = \delta(i,j) \mod 2^{m+1}$, which completes the proof. □

References

[1] P. Eirinakis, M. D. Williamson, and K. Subramani. On the Shoshan-Zwick algorithm for the all-pairs shortest path problem. *CoRR*, abs/1603.08627, 2016. URL: `http://arxiv.org/abs/1603.08627`.

[2] A. Shoshan and U. Zwick. All pairs shortest paths in undirected graphs with integer weights. In *40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA*, pages 605–615, 1999. `doi:10.1109/SFFCS.1999.814635`.

[3] V. V. Williams. Multiplying matrices faster than Coppersmith-Winograd. In *STOC*, pages 887–898, 2012. `doi:10.1145/2213977.2214056`.