STOCHASTIC SHORTEST PATH ALGORITHM
BASED ON LAGRANGIAN RELAXATION

BY

LESLIE K. HWANG

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Adviser:

Professor Martin D. F. Wong

ABSTRACT

In VLSI circuit design, graph algorithms are widely used and graph structure can model many problems. As technology continues to scale into nanometer design, the effects of process variation become more crucial and design parameters also change. Hence, taking stochastic variations into account, probability distributions are used as edge weights to form statistical graph structures. General applications in VLSI circuit design, such as timing analysis, buffer insertion, and maze routing, can be formulated as shortest path problems using a statistical graph model. The solution of any such graph problem will surely have a statistical distribution for its cost function value. The mean and variance, square of standard deviation, values are used as a pair of weight values on a graph to represent the stochastic distribution on each edge. For the stochastic shortest path problem, we observe that the objective functions can be formulated using mean and standard deviation values of the resulting probability distribution and general cost functions are nonlinear. To solve for the nonlinear cost function, we intentionally insert a constraint on the variance. Several candidate paths will be achieved by varying the bound value on the constraint. With fixed bound value, the Lagrangian relaxation method is applied to find the feasible solution to the constrained shortest path problem. During Lagrangian relaxation, a feasible solution close to the optimal is achieved through subgradient optimization. Among the candidate paths obtained, the best solution becomes the ultimate solution of our algorithm for the original cost function under parameter variation. The algorithm presented in this work can handle any graph structures, arbitrary edge weight distributions and general cost functions.

To my parents

ACKNOWLEDGMENTS

I would like to thank my adviser, Professor Martin Wong, for his admirable guidance and supportive advice. He has always encouraged me, and led me with patience whenever I was struggling with a new research area. This thesis could not have been finished without his support. I would also like to thank all my research group members for valuable advice and technical discussions. I came to understand the research field better with their help. Last but not least, I deeply appreciate my family's support, love and sacrifice. My parents have always encouraged me and wished me well in my study, and my siblings provided me with energy to boost myself up.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Variation is inevitable in modern technology and many current research works focus on improving the design and analysis methods involving parameter variation. These existing works are great achievements, but they still possess some limitations. In this thesis, we introduce an efficient method to handle previous limitations and improve the processing speed further.

1.1 Motivation

Manufacturing a chip is a process that involves many variables. As technology continues to scale into nanometer design, parameter variations become major challenges for circuit performance [1]. On-chip-variation (OCV) may come from fabrication processes such as mask alignment, etching process, and optical proximity correction [2]. Supply voltage, temperature, and other systematic and random process variations have a great impact on circuit and microarchitecture performance and can possibly result in yield loss [1], [3]. For example, the increases in operating frequency result in high junction temperature and die temperature variation and this will affect the power consumption. An illustrative example of die temperature variation is displayed in Figure 1 [4]. Thus, continued increase of performance variability makes the problem probabilistic rather than deterministic and emphasizes the importance of accurate circuit analysis. This will also help to avoid over-pessimistic design.

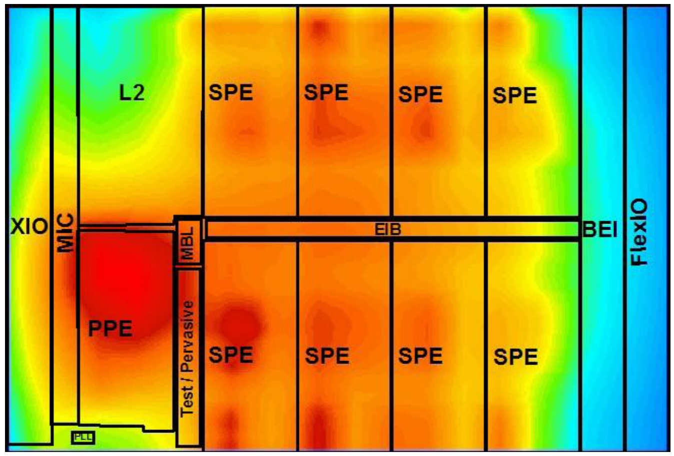


Figure 1 Die thermal map [4].

Graph algorithms are widely used in very large-scale integration (VLSI) computer aided design (CAD). Many different problems can be formulated as a shortest path problem, minimum spanning tree and so on. There has been much research on conventional graph algorithms with deterministic edge weights. Recently, the focus has been on shortest path problems using probabilistic edge weights. Probabilistic graphs have the same structure as general graphs, but have edge weights of the random variables with probability distributions. The shortest path on statistical graphs is a function of random events characterized by each probability distribution. For design and analysis under the parameter variation, graphs with probability distributions for edge weights should be used to model the uncertainty. Figure 2 illustrates a statistical graph with arbitrary probability distributions for edge weights.

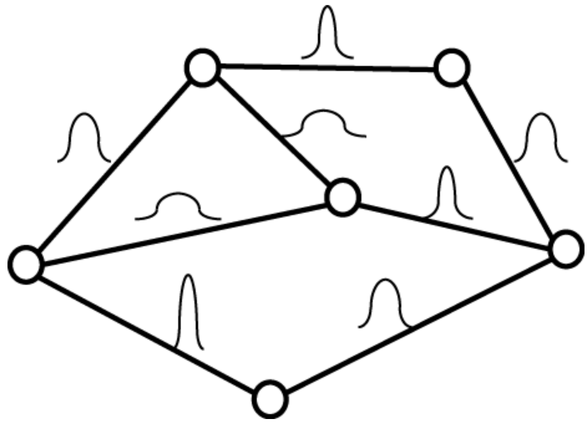


Figure 2 Statistical graph structure.

1.2 Problem Formulation

In a probability distribution, the portion of the distribution can be described as a function of certain parameters that describe the distribution. In the case of a Gaussian distribution, or normal distribution, drawn in Figure 3, the six-sigma range, between $\mu - 3\sigma$ and $\mu + 3\sigma$, covers the majority (> 99%) of the probability distribution, where μ and σ are mean and standard deviation respectively. The cost function $\mu \pm 3\sigma$ is widely used to measure the performance and yield in processes defined by Gaussian distribution [5]. Similarly, any part of the arbitrary distribution can be formulated as $\mu + k\sigma$ or $\mu - k\sigma$. Hence, mean and standard deviation, or variance σ^2, values represent the corresponding probability distribution and can be used as edge weights on the graph instead of the original probability distribution.

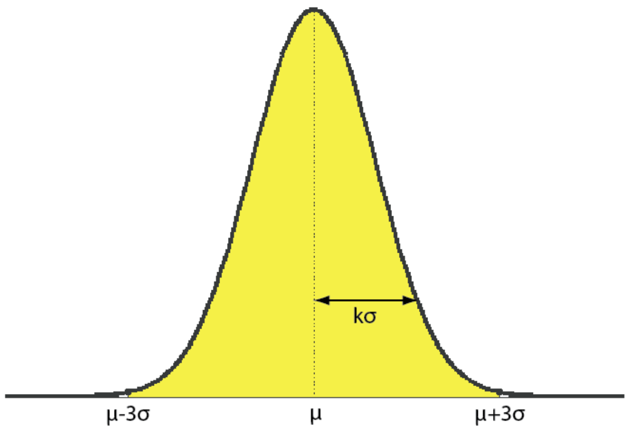


Figure 3 Gaussian distribution (Six sigma range: yellow shaded area).

Given a graph G, the weight for edge i is a random variable forming a distribution with mean μ_i and variance σ_i^2. The goal of the conventional shortest path problem is to minimize the total length on the path, $min_p\, \mu_p$, for total mean value μ_p on the path p, from the given source node to target node. However, if we take variation into consideration, the path with smallest mean value, $min_p\, \mu_p$, would not always correspond to the path with minimum of cost function with variation, $min_p(\mu_p + k\sigma_p)$, for overall standard deviation σ_p on the path p. An example is shown in Figure 4. Figure 4(a) is the statistical graph with weight values with a pair of (μ_i, σ_i^2) on every edge i. Figure 4(b) presents the solution using the traditional shortest path algorithm which minimizes the length, mean values only from the statistical graph, and Figure 4(c) is the resulting path under parameter variation. Thus, in the stochastic shortest path problem, the objective is to find a path p that minimizes the cost function $\mu_p + k\sigma_p$, where $\mu = \sum_{i \in P} \mu_i$, $\sigma = \sqrt{\sum_{i \in P} \sigma_i^2}$ for real number $k > 0$.

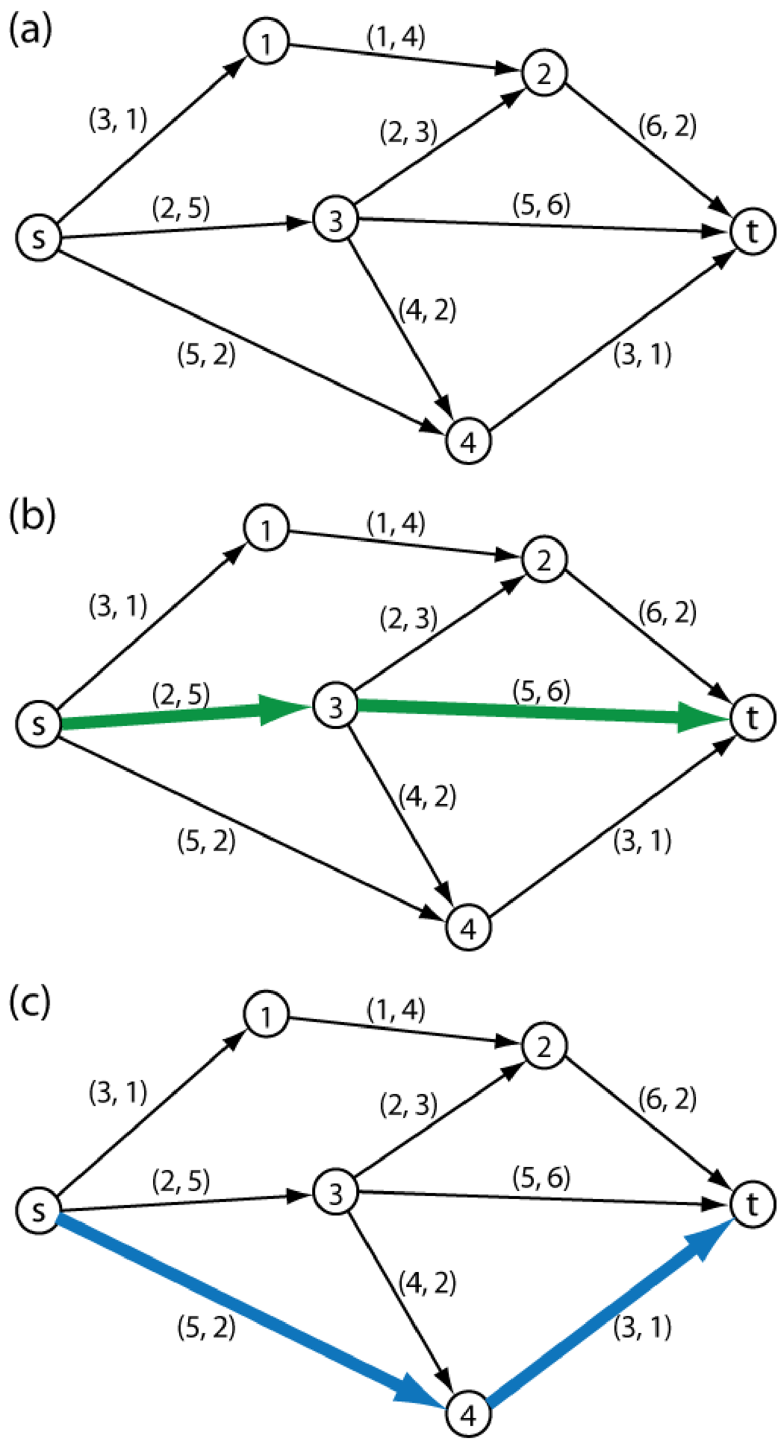


Figure 4 Example indicating the difference of the solutions from conventional and statistical method. (a) Simplified statistical graph. (b) Path with shortest length, minimum mean value, of 7 (labeled with bold green lines). (c) The worst case shortest path, $\boldsymbol{min}(\boldsymbol{\mu} + \mathbf{3}\boldsymbol{\sigma})$, of 9.73 assuming Gaussian distribution (labeled with bold blue lines).

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 Stochastic Shortest Path Problem

The shortest path problem is one of the most fundamental problems in graph theory. The objective of finding the path to minimize the cost function in the classical shortest path problem has been studied intensively. Various algorithms have been established for different implementations. A single-source, multiple-target algorithm has been developed by Dijkstra [6]. The Bellman-Ford algorithm is slower than Dijkstra's algorithm, but it is applicable for graphs with negative edge weights [7]. Another interesting algorithm is the Floyd-Warshall algorithm which is used for multiple-source, multiple-target, or all-pair shortest path problems [8].

Recently, variation has become an important factor in analysis and has become more crucial as technology scales down. Hence, many recent researches heavily focus on improving the method and algorithm for statistical analysis. However, there are still some limitations with current works.

Research on the shortest path problem in probabilistic graphs follows work by Frank [9]. In practical situations, the costs or time is often random. This work estimates the sum of the probability distributions of the shortest path through acyclic networks weighted with random lengths. Following Frank's work, Sigal et al. [10] addressed a shortest path problem through a directed, acyclic network where arc lengths are independent random variables. This work presented an analytic derivation of path optimality indices for directed, acyclic networks.

In addition, different types of cost functions on the stochastic shortest path problem have been studied. One of the works, by Loui [11], found computationally tractable formulations of stochastic and multidimensional optimal path problems. Similar problems with maximizing the expected cost with piecewise-linear concave utility function were studied by Murthy and Sarkar [12]. Lastly, Hall [13] and Fu and Rilett [14] studied the expected shortest path on stochastic shortest path problem. To combine the previous works, X. Ji proposed three models — expected shortest path, α-shortest path and the shortest path — and developed a hybrid intelligent algorithm combined with genetic algorithm to solve proposed models [15]. However, these early researches on the statistical shortest path problem were designed for directed acyclic graphs (DAG) with specific edge weight distributions, such as the Gaussian distribution. Moreover, the algorithms cannot handle the general cost function, which can be nonlinear.

To overcome the limitations of previous research, Deng and Wong found an exact algorithm [16] to find the optimal solution for the cost function $\mu_p + \Phi(\sigma_p^2)$ on the statistical shortest path problem. Unlike former works, this algorithm handles general graphs, arbitrary edge-weight distributions and general cost functions. To minimize the uncertainty in the final result of the statistical problem, Deng and Wong added the variance constraint $\sigma^2 \leq B$ to the problem. The main idea of their algorithm is to expand graph G into a larger graph G' by splitting each node into a number of nodes based on variance value. New edges are then added and mean values are assigned as edge weights. The expanded graph is guaranteed to be directed acyclic and the deterministic shortest path in G' gives an optimal path in G. For one of the implementations, Figure 5 shows the maze routing result using their algorithm. However, this method can only handle integer values for edge weights, and computational time increases significantly as the variance bound value *B* on constraint increases.

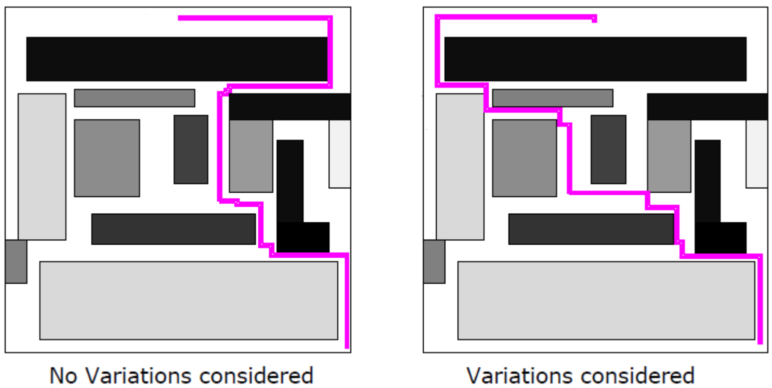


Figure 5 Comparison the maze routing with and without considering the temperature variations. (a) Shortest path found by classical shortest path problem. (b) Result by solving statistical shortest path problem [16].

2.2 Constrained Optimization Problem

To overcome existing limitations, we intentionally add a constraint to formulate a constrained optimization problem. We take advantage of the large number of variance bound values from the constraint to sample many candidate paths. It is sufficient to solve for the formulated constrained shortest path problems with diverse bound values and compute for the nonlinear objective function regarding the obtained candidates.

One of the early works that brought up the constrained shortest path problem was completed by Aneja and Nair [17]. The algorithm presented in [17] is similar to the Lagrangian multiplier technique but it requires, on average, a number of iterations which are polynomially bounded. Cai et al. [18] also studied a different approach to the constrained shortest path problem that is very similar to ours. The problem is formulated on a graph with two weight values — time and cost — on a directed graph with constraints. Three variants of the problem are examined: arbitrary waiting times, zero waiting times, and vertex-dependent upper bounds on the waiting times at each vertex.

The stochastic shortest path algorithm introduced in this thesis can handle any general graph, arbitrary probability distribution, general cost function and integers as well as other non-integer values. Finally, the computation runtime is very efficient compared to previous works.

CHAPTER 3

STOCHASTIC GRAPH AND OBJECTIVE

To form a graph model with variation, statistical graph structure is constructed. A pair of representative values on a probability distribution are used in objective function, and hence, assigned for edge weights. The objective function of the shortest path problem with random variables is typically nonlinear. Our approach for the nonlinear cost function is to formulate the problem into a series of constrained optimization problems with various bound values.

3.1 Statistical Graph

If the random variable on each edge forms a Gaussian distribution, then the overall distribution along the path, which results from adding the Gaussian distributions of each individual edge on the path, will also be Gaussian. To cover the majority of the path length distributions, the objective function may be to minimize $\mu_p \pm 3\sigma_p$ or to maximize $\mu_p \pm 3\sigma_p$. If the goal is to minimize the worst case of statistical path length from the given source node to the target node, then the objective will be minimizing the cost function as $min_p(\mu_p + 3\sigma_p)$ for path p. For other general edge-weight probability distributions, the objective function can be set as Equation (1) due to Chebyshev's inequality from Equation (2).

$$min_p(\mu_p \pm k\sigma_p) \text{ or } max_p(\mu_p \pm k\sigma_p) \tag{1}$$

$$P(|X - \mu_P| \geq k\sigma_P) \leq \frac{1}{k^2} \;\; for\ random\ variable\ X\ and\ real\ number\ k > 0 \tag{2}$$

From the above relation, any case of the given distribution can be considered by fixing the value of k. To solve for the stochastic shortest path problem, we only need mean μ and standard deviation σ values from the probability distribution. This allows us to simplify the statistical graph with two real numbers on each edge weight. Mean values of the cost function probability distribution are linear functions of mean values of each edge weight distribution. However, from the Equation (3), the function to compute the overall standard deviation from multiple probability distributions is nonlinear. Hence, the conventional shortest path algorithm cannot be directly applied. Instead, overall variance can be computed linearly as shown in Equation (4). Thus, instead of considering the entire random variation, we can simply represent the statistical distribution with mean and variance values for each edge weight on the statistical graph structure as shown in Figure 6.

$$\sigma_p = \sqrt{\sum_{i \in p} \sigma_i^2} \quad for\ path\ p \tag{3}$$

$$\sigma_p^2 = \sum_{i \in p} \sigma_i^2 \quad for\ path\ p \tag{4}$$

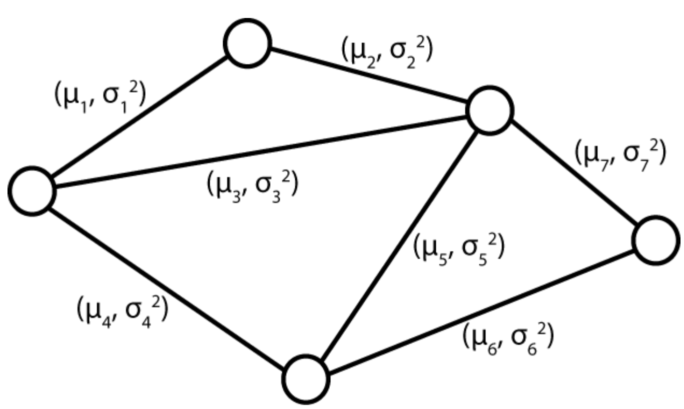


Figure 6 Simplified statistical graph with pair *(μ, σ^2)* for each weight.

Consider the case for minimizing the worst case of any Gaussian distribution, $min_p(\mu_p + 3\sigma_p)$. If there is an ideal path that has a minimum mean and minimum variance, then this path will always be the optimal solution for the objective function. However, the probability that both the minimum mean and minimum variance lie on the same path is very low. Therefore it is necessary to compare the value of the cost function between paths. To achieve the goal, it will be extremely time-consuming to compare all the paths as the graph grows larger. It is sufficient to sample several path candidates and compare the goal. Typically, the candidate paths will have a small mean value but a relatively large variance, a small variance but a relatively large mean value, or fairly small values for both mean and variance.

Similarly, for maximizing the reward function of the worst case of the variation, the possibility for the existence of an ideal path with both maximum mean and maximum variance is extremely low. There are some cases for which the number of paths from the source node to target node is unbalanced. For these special structured graphs, the ideal path is likely to exist; however, the greater part would not correspond to this rare case. Path candidates for the longest path will have either a large mean value but comparatively small variance, a large variance but small mean value, or both reasonably large mean value and variance. If we negate all mean values and variances, then the objective function becomes a minimizing problem. Then, we can sample paths with properties illustrated above by applying the same method to solve for minimizing the cost function.

3.2 Sampling Method Using Constraints

To sample the path candidates for the objective function, we will formulate the problem as a series of constrained shortest path problems shown in Equation (5). This problem finds a shortest path with minimum overall mean with respect to the variance constraint. As shown in Equation (6), varying the bound value B from the upper limit of variance to a lower limit creates a number of constrained optimization problems. Solutions for each constrained problem may result in different paths.

$$p^* = \begin{Bmatrix} \min\limits_{p}\ \mu_p \\ subject\ to\ \ \sigma_p^2 \le B \end{Bmatrix} \tag{5}$$

$$where \quad \mu_p = \sum_{i \in P} \mu_i,\ \ \sigma_p^2 = \sum_{i \in P} \sigma_i^2\ , and\ P = path\ from\ source\ to\ target$$

$$\min_p \sigma_p^2 < B \le \max_p \sigma_p^2 + \varepsilon \quad for\ small\ \varepsilon \tag{6}$$

Initially, the bound value will be set as the upper limit of variance; therefore, the constraint is very loose. This problem considers all the paths to obtain the shortest path regardless of the value of variance as long as it meets the constraint. As the bound value gets smaller, i.e., tighter constraint, the feasible path will have a smaller variance; however, the mean value is likely to increase due to the narrower feasible set. All these path solutions target either minimizing the mean, variance or both; hence, these are eligible as the path candidates for the original objective.

Depending on the value of the bound, some constrained problems might have the same solution path. To eliminate the redundancy, we set the bound value according to the variance of the current solution — for example assume σ_{B1}^2 is the variance of the solution path to the constrained problem with bound value B_1. Until the bound value reaches the variance of the current solution, $B = \sigma_{B1}^2$, this path stays in the feasible set and it is the best solution. We set the next bound value to be slightly smaller than the variance of the current solution, $B = \sigma_{B1}^2 - \varepsilon$ for small ε, in order to move this solution out of the feasible set. Then, a different solution path will be achieved for the newly formulated constrained shortest path problem.

CHAPTER 4

STOCHASTIC SHORTEST PATH ALGORITHM

As the focus is on statistical design and analysis, we have worked on an efficient method to find a reasonable approximate solution for the stochastic shortest path problem. The shortest path problem under variation was modeled as a nonlinear optimization problem. To sample the feasible and promising candidate paths for the objective function, we have converted the problem to a series of constrained optimization problems. In this chapter, we introduce a Lagrangian relaxation based algorithm to solve for the stochastic shortest path problem. Our algorithm can handle any general graph structures and any arbitrary probability distributions with positive, negative or even floating point random variables for edge weights. In addition, our algorithm guarantees convergence and extremely efficient runtime.

4.1 Lagrangian Relaxation

A well known approach for the constrained optimization problem is the Lagrangian relaxation method [19]. This method can also be applied for the shortest path problem with an additional side constraint. The basic concept of Lagrangian relaxation is to combine the constraint into the objective function by relaxing the constraint. Then, as shown in Equation (7), we can easily approach the constrained shortest path problem indirectly with the modified cost function without any constraints, $\mu_p + \lambda\left(\sigma_p^2 - B\right)$.

$$L(\lambda) = \ min \ \{\mu_p + \lambda(\sigma_p^2 - B)\} \quad for\, p \in P\,, \lambda \geq 0 \tag{7}$$

The Lagrangian relaxation problem is a function of λ, which is called the Lagrangian multiplier and it is also known as the Lagrangian multiplier problem. The Lagrangian multiplier problem is a dual problem to the primal problem, which is the constrained optimization problem [20]. One common property of the relationship between the primal and the dual problem is weak duality. Weak duality means the optimal objective function value g^* of the Lagrangian multiplier problem in Equation (8) is always a lower bound on the optimal objective function value of the primal problem p^* from Equation (5), i.e. $g^* \leq p^*$. With respect to the weak duality property, the solution for the Lagrangian multiplier problem will give a lower bound value for the primal problem. In Equation (9), this weak duality relationship is manifested.

$$g^* = \max_{\lambda \geq 0}\ L(\lambda) = \ \max_{\lambda \geq 0} \min_{p \in P}\{\mu_p + \lambda(\sigma_p^2 - B)\} \quad for\, path\, set\, P \tag{8}$$

$$\max_{\lambda \geq 0} \min_{p \in P}\{\mu_p + \lambda(\sigma_p^2 - B)\} = \boldsymbol{g}^* \leq \ \boldsymbol{p}^* = \begin{Bmatrix} \min\limits_{p \in P}\ \mu_p \\ subject\ to \ \ \sigma_p^2 \leq B \end{Bmatrix} \ for\, path\, set\, P \tag{9}$$

With fixed constant *B*, the modified cost function, $\mu_p + \lambda(\sigma_p^2 - B)$, of the Lagrangian multiplier problem can be used as weights on the graph as drawn on Figure 7. The Graph in Figure 7 is converted from the example shown in Figure 4(a). Now, the graph has only single weight value and it has the conventional graph structure. On this graph, we can first apply the shortest path algorithm to solve the Lagrangian multiplier problem after fixing the Lagrangian multiplier value λ. For general graphs with positive weights, Dijkstra's shortest path algorithm can be applied. Dijkstra's algorithm is a graph search algorithm that solves the single-source shortest path problem for a graph with nonnegative edge path costs [6]. The algorithm finds the path with lowest cost between a given source vertex and every other vertex. In the case with negative weights, the Bellman-Ford algorithm can be used or if the graph is directed and acyclic, the topological sort algorithm can be used for more efficient runtime.

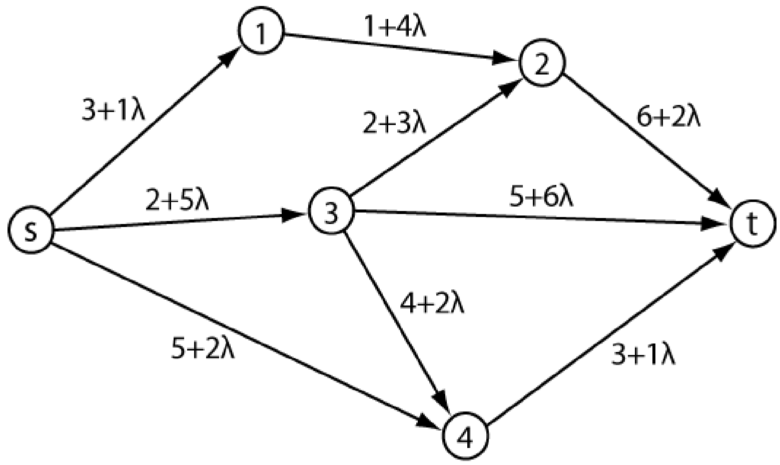


Figure 7 Graph with modified cost function as weight.

The resulting solution of the modified cost function from the graph in Figure 7 will form a piecewise linear concave function over the Lagrangian multiplier λ. All the points on this minimum envelop of the Lagrangian function are lower bounds for the optimal solution of the original problem. Among the points, the supremum of the Lagrangian multiplier problem or dual problem, least upper bound g^* which is equal to the peak value of the concave function, is the closest to the optimal and will be the best lower bound for the primal problem. The relationship between the primal and the dual problem is shown in Figure 8. The supremum point of the dual problem can be obtained by maximizing the Lagrangian multiplier function.

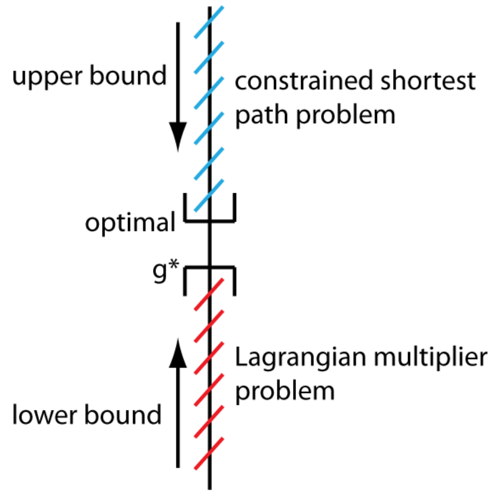


Figure 8 Relationship between the primal and the dual problem.

4.2 Subgradient Optimization

The Lagrangian multiplier problem, or the dual problem, is always a concave function but not necessarily differentiable. To account for this situation, the subgradient optimization technique has been implemented to maximize the Lagrangian multiplier problem, which is nondifferentiable. Subgradient optimization is a generalization of the steepest descent method, i.e. the gradient method. The idea of this optimization technique is to move to the direction d where directional derivative of function f is positive, $\nabla f(x)d > 0$, with small enough step length. Equation (10) is derived from Equation (7) and since λ is the variable, the gradient of equation (10) is shown in Equation (11). To the positive gradient direction, we move the lambda value to reach to the best value. The equation to update lambda during iteration is stated in Equation (12). From this equation, step size θ should be carefully assigned in order to guarantee the convergence. Practitioners of the Lagrangian relaxation method often use the following heuristic for selecting the step length stated in Equation (13).

$$L(\lambda) = \min_p \{ \mu_p + \lambda(\sigma_p^2 - B) \,|\, p \in P \} \tag{10}$$

$$\frac{dL(\lambda)}{d\lambda} = \sigma_p^2 - B \tag{11}$$

$$\lambda_{k+1} \leftarrow \lambda_k + \theta_k(\sigma_p^2 - B) \tag{12}$$

$$\theta_k = \frac{v_k[UB - L(\lambda_k)]}{\left\| \sigma_p^2 - B \right\|^2}, \;\; UB = upper\ bound, \;\; 0 < v_k \leq 2 \tag{13}$$

In the step size equation, the scalar v_k is any number between 0 and 2. Throughout the iteration, it will be reduced by a factor of 2 whenever the best Lagrangian objective function value found so far has failed to improve in a specified number of iterations. UB is the upper bound of the optimal objective function. The initial upper bound can be any known feasible solution to the problem. During the subgradient iteration, the upper bound will be updated if a smaller feasible solution has been generated. A feasible path that provides the upper bound approaches the optimal path as the subgradient iteration converges. Several constrained problems formed with different bound value B will sample different feasible paths which are the solution candidates for the original objective function. The value of the goal will be calculated and compared among the candidate paths. Afterwards, the best result will be chosen as the solution of the Lagrangian relaxation based stochastic shortest path algorithm.

4.3 Algorithm

Our stochastic shortest path algorithm based on the Lagrangian relaxation method described in previous sections is provided below. It is the global view of the algorithm, and the existing algorithm was implemented on shortest path. For the subgradient optimization, the initial value of v is chosen as 0.8 after several experiments on convergence. We defined the convergence criterion to be when the error between previous and current λ values is small enough within 10 000 iterations.

```
Begin
    Input given graph with (μ_i, σ_i^2) as weight;
    Calculate min σ_p^2 and max σ_p^2;
    Let initial UB = μ_p of the any known path;
    Set initial B = max σ_p^2 + ε;
    Set L(0) ← min μ_p;
    Set v = 0.8, θ and λ using Equation (12)-(13);
    Repeat until B < min σ_p^2,
        Topological sort shortest path for min{μ_p + λ(σ_p^2 − B)};
        σ_cur^2 ← σ_p^2 of shortest path for min{μ_p + λ(σ_p^2 − B)};
        μ_cur ← μ_p of shortest path for min{μ_p + λ(σ_p^2 − B)};
        If converges,
            Set next B = σ_cur^2 − ε;
            Set L(0) ← min μ_p;
            Reset v = 0.8, θ and λ using Equation (12)-(13);
        Else,
            Update v ← v/2;
            Set θ and λ using Equation (12)-(13) with updated v;
        End if
    Calculate μ ± kσ of the sampled paths with different B;
    Obtain min μ ± kσ or max μ ± kσ;
End
```

CHAPTER 5

EXPERIMENTAL RESULTS

5.1 Expected Results

With fixed bound B value, we can formulate a constrained problem and compute the feasible solution path. The outcome path of a certain constrained problem with fixed bound value is going to satisfy the following constrained problems with smaller bound value until the constraint becomes tighter and, hence, the path becomes infeasible. Therefore, we used the variance sum of the current solution path as the reference value for the bound for the next iteration. The bound value for the next constrained problem was set based on the variance of the current solution path. By repeating this procedure of varying bound value, numerous potential solutions can be expected.

Suppose the bound for constraint was $max_p\, \sigma_p^2 + \varepsilon$, and the output path had the minimum possible variance. Then, subsequent bound values will be less than the minimum variance and there will be no feasible solutions to the following constrained problems. Thus, only one path will be sampled for that graph, and it is the solution. However, this is a special case. Typically, if maximum variance was set for the bound, the problem becomes loose. The resulting path would have minimum mean but relatively large variance, $\mu \propto \sim \frac{1}{\sigma^2}$. As the bound decreases and problem becomes stricter, and the resulting mean value will get larger, i.e., worse for minimization. The expected trend curves are illustrated in Figure 9. The curves will have decreasing shape, but it will be hard to predict whether it will have convex, concave or linear structure.

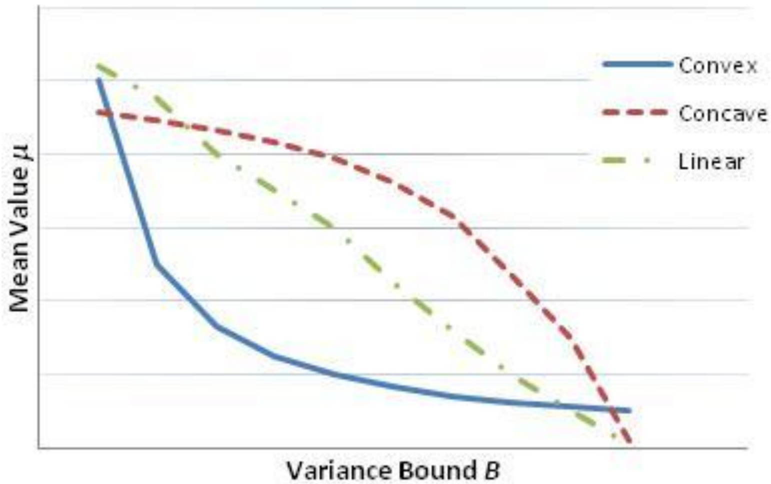


Figure 9 Expected sampled path curve.

(Convex curve: blue solid line, concave curve: red dash line, linear: green dash dot line.)

From the curves in Figure 9, the solution depends on the sign of the coefficient of standard deviation from the objective function. For example, suppose the objective is $min_p(\mu_p + k\sigma_p)$. Then, the entire sampled paths should be compared for the final best solution. However, if the objective function is $min_p(\mu_p - k\sigma_p)$, it is clear that the path with the rightmost property on the minimization curve will be the solution since it has minimum mean value with maximum standard deviation. Similarly, for the maximization problem, the leftmost point will always be the solution for the problem with objective function $max_p(\mu_p - k\sigma_p)$. For the objective $max_p(\mu_p + k\sigma_p)$, all the candidate paths should be compared in order to find the actual best solution.

5.2 Minimization Problem

The stochastic shortest path problem is formulated as constrained minimization problem to apply our algorithm. For convenience, all the weight values on the graphs, which represent the mean and variance value of the probability distribution, were assigned for positive values. Therefore, any shortest path algorithm can be applied during Lagrangian relaxation. In this work, we applied Dijkstra's algorithm for the minimization problem.

Maintaining the graph structure illustrated in Figure 10, three different sizes were generated and experimented. This balanced graph structure is generated to eliminate the effects of other irrelevant factors such as number of edges on the path from the source to target. If the graph is unbalanced, the effect of various ranges of mean and variance values on the edge weights is not going to be very apparent. The mean and the variance on each weight are randomly generated between the assigned ranges.

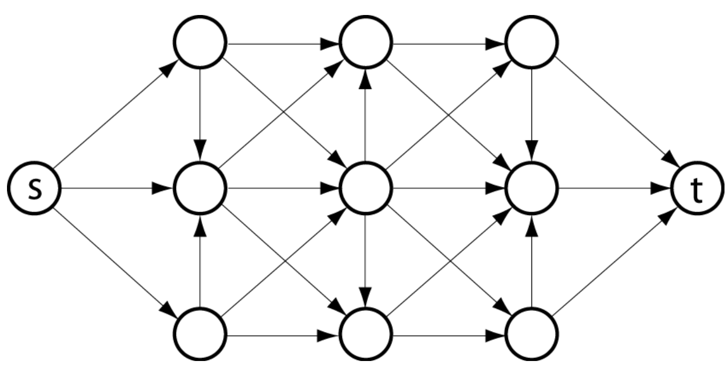
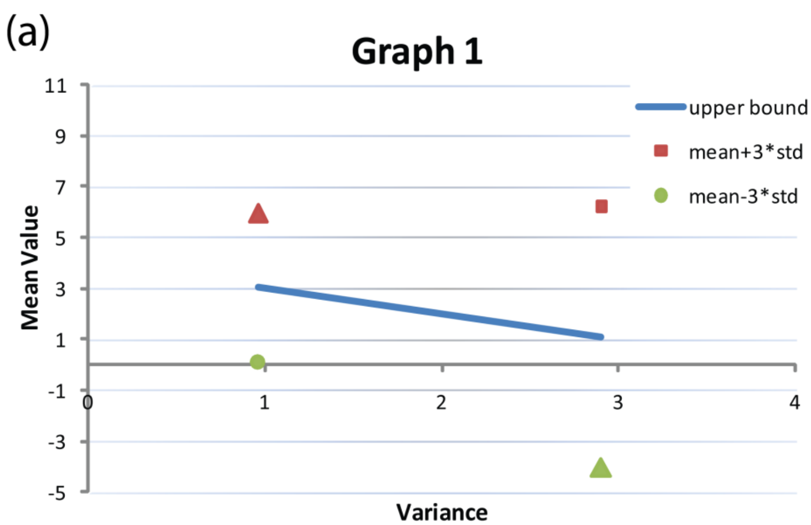


Figure 10 Minimization problem graph structure example.

The result curve of bound B verses mean μ of the sampled path is shown in the blue line in Figure 11; the curve is decreasing convex as expected in Chapter 5.1. The solution of the initial constrained problem with the loosest bound was positioned at the rightmost point of the curve. As the constraint tightens, the mean value of the solution increases and the point moves to the northwest. We can observe that the original optimal solution is not obvious among the sampled candidates. Some solutions have small variances with large mean values, while others have small mean values with large variances or middle range on both values. The calculated value of the cost function $\mu + 3\sigma$ is marked with red rectangles on the graph. The solution for the cost function $\mu - 3\sigma$ was calculated the same way and is marked as green circles. Finally, the path that minimizes $\mu \pm 3\sigma$ the most is marked as larger triangles.



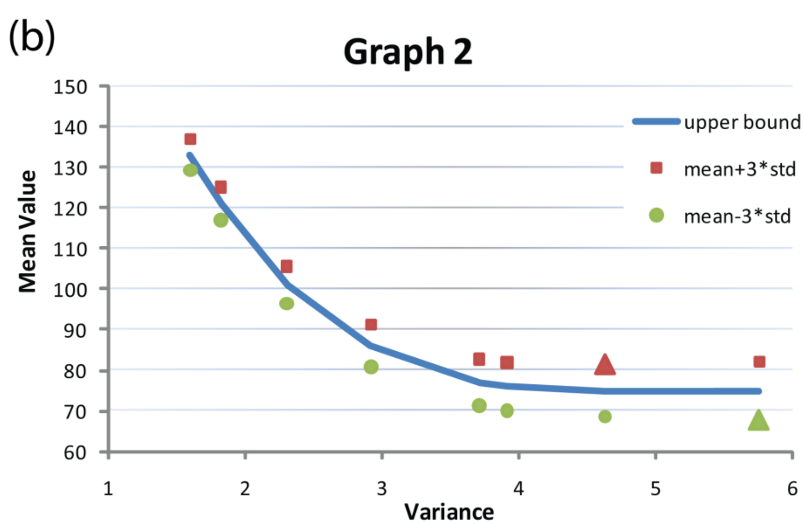


Figure 11 Result curve of minimization problem. (a) Graph 1. (b) Graph 2. (c) Graph 3. (Large triangles: the solution of the original objective function.)

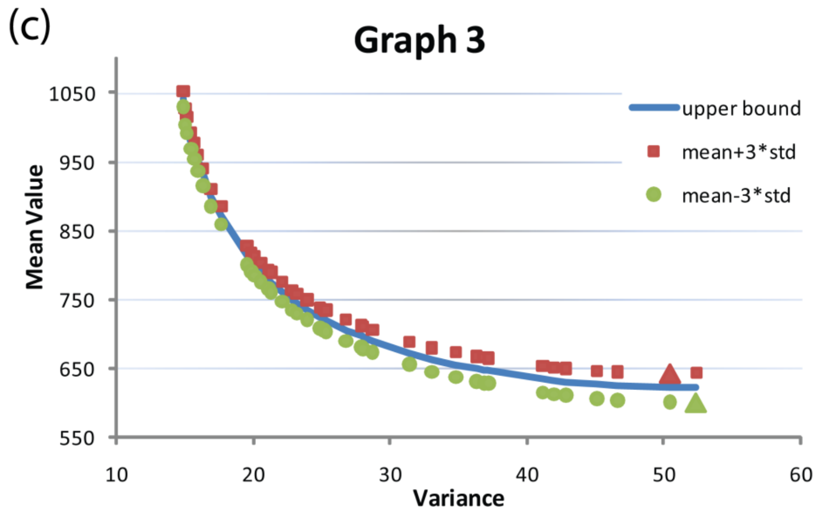


Figure 11 (continued)

The summary of the result is shown in Table 1. Three sizes of the graphs were varied between graph examples and the mean and variance value range was differed to examine the effect of the variance. When the variance is as large as mean value, it is no longer negligible. From Table 1, G_+^* and G_-^* are the solution from our algorithm for $min_p(\mu_p + k\sigma_p)$ and $min_p(\mu_p - k\sigma_p)$ respectively. For comparison, we performed 10 000 iterations of Monte Carlo analysis and G_{MC} is the solution of this method. During the Monte Carlo simulation, the probability distribution of random variables on each edge was assumed to be Gaussian. However, any arbitrary distribution, for example the uniform distribution, exponential distribution, etc., can also be applied in the stochastic shortest path algorithm and Monte Carlo analysis. Finally, the runtimes of the presented stochastic shortest path algorithm and Monte Carlo simulation are compared, and it is noticeable that our algorithm is more efficient.

Table 1 Minimization experiment results

	Minimization		
Examples	**Graph 1**	**Graph 2**	**Graph 3**
Nodes	11	102	10002
Edges	26	362	39602
Weights	(0-1, 0-1)	(5-15, 0-1)	(5-15, 0-1)
Paths	2	8	39
G_+^*	5.97893	81.4558	643.31056
G_-^*	-4.0114	67.7991	600.2874
Runtime	0 msec	1 msec	204 msec
G_{MC}	5.97893	81.4558	643.375
Runtime_MC	3 msec	30 msec	N/A

5.3 Maximization Problem

The maximization problem is similar to the minimization problem. Instead of applying the longest path algorithm directly on the problem described in Equation (14), we transform the problem into a minimizing problem and apply shortest path algorithm. We negate both mean and the variance values to model Equation (14) as a minimization problem written in Equation (15). Since Dijkstra's algorithm cannot deal with negative weights and the graph structure we constructed is directed acyclic, we applied the topological sort algorithm for finding the shortest path. The topological sort algorithm sorts the vertices in order and calculates the accumulated weight on each vertex. This algorithm is much faster in runtime than Dijkstra's algorithm and able to handle any real number on the weight.

$$\begin{Bmatrix} \max\limits_{p} \mu_p \\ subject\ to\ \ {\sigma_p}^2 \leq B \end{Bmatrix} \tag{14}$$

$$\begin{Bmatrix} \min_p(-\mu_p) \\ subject\ to\ \ -\sigma_p{}^2 \le -B \\ or\quad \sigma_p{}^2 \ge B \end{Bmatrix} \tag{15}$$

We used the same graph structure as in the minimization problem. The result curve of the sampled path is illustrated in Figure 12 and it is a decaying concave shape. Unlike the minimization problem, the initial constraint is when the bound B value is the minimum variance and the path will lie at the leftmost position. The constraint tightens as the bound value increases and the points move to the southeast. Among those sampled paths, the objective value should be calculated to find the best solution. In the case of Figure 12, similarly, some solutions have small variances with large mean values, while others have small mean values with large variances or middle range on both values. The calculated value of the cost function $\mu + 3\sigma$ is marked with red rectangles on the graph and the path that maximizes the most is marked as larger red triangles. The solution for the cost function $\mu - 3\sigma$ is calculated in the same way and marked as green circles, and the best maximizing solution is larger green triangles.

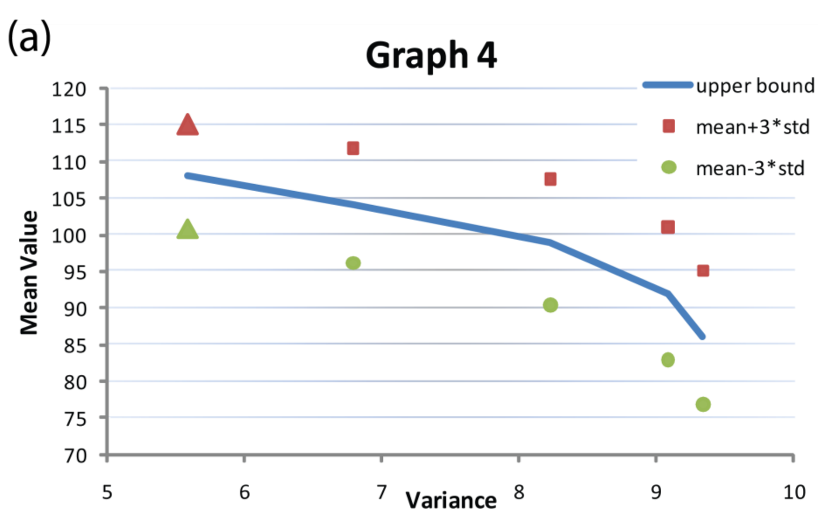
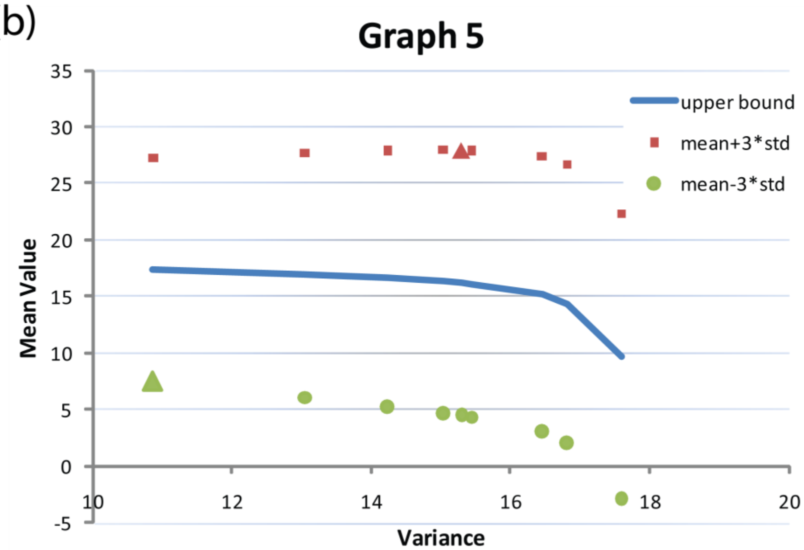


Figure 12 Result curve of maximization problem. (a) Graph 4. (b) Graph 5. (c) Graph 6. (d) Graph 7. (Large triangles: the solution of the original objective function.)

(b)



(c)

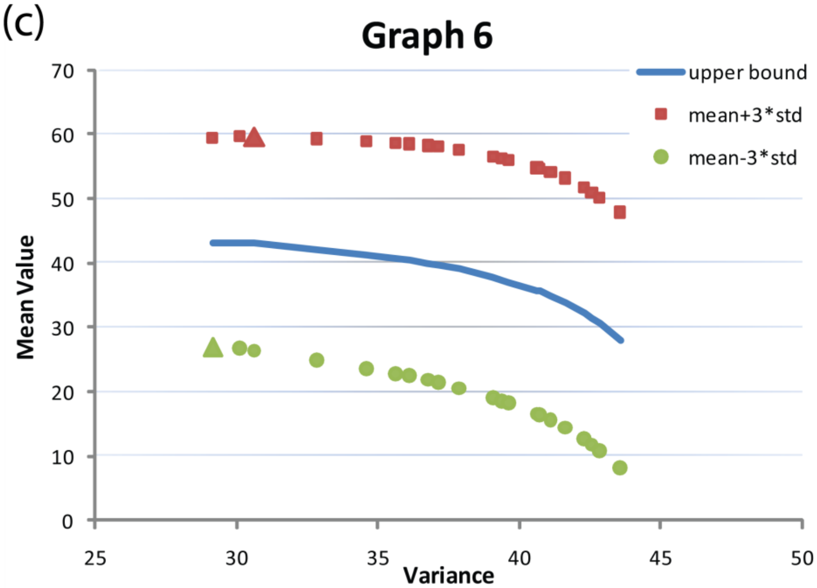


Figure 12 (continued)

(d)

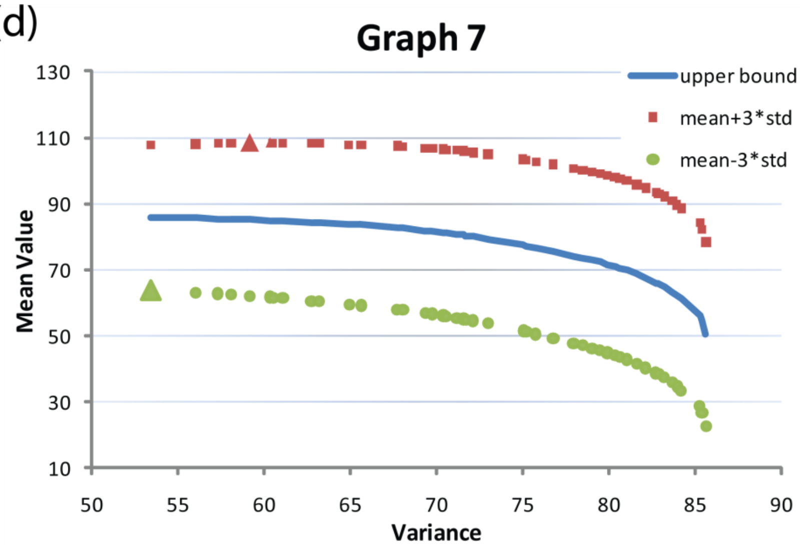


Figure 12 (continued)

The experimental results are summarized in Table 2. Several different graph sizes have been generated and tested. The solution of the objective function $max\ (\mu_p \pm 3\sigma_p)$ using our algorithm is indicated as G_+^* and G_-^*.

Table 2 Maximization experiment results

	Maximization			
Sample	**Graph 4**	**Graph 5**	**Graph 6**	**Graph 7**
Nodes	102	402	2502	10002
Edges	362	1522	9802	39602
Weights	(5-15, 0-1)	(0-1, 0-1)	(0-1, 0-1)	(0-1, 0-1)
Paths	5	9	21	46
G_+^*	115.0886	27.9605	59.6816	108.2846
G_-^*	100.9114	7.4538	27.0330	63.7365
Runtime (ms)	0 msec	6 msec	15 msec	265 msec

CHAPTER 6

CONCLUSION

As the impact of process variation increases, we cannot rely on the results from conventional algorithms that assume variations are negligible. The solution of the traditional algorithms might be very different from the statistical problem considering the variations that form a probability distribution for edge weights on the graph. There were several researches on the statistical shortest path; however, previous algorithms were not very efficient in runtime and the majority of them had limitations on practical applications. In this work, we have introduced an efficient way to obtain a reasonable solution based on Lagrangian relaxation. We intentionally insert a constraint and formulate a series of constrained problems by varying the variance bound value and sample candidate solutions for each formulated problem. At the end, we compare the objective value of the sampled candidates and attain the best solution.

The method we proposed in this work can be used in various applications in nanometer design that potentially have high parameter variations that are significant. Common applications in nanometer designs include timing analysis, maze routing, and buffer insertion. For timing analysis, precise timing information is necessary for circuit optimization to meet the yield or avoid over design. Maze routing finds the shortest path in the grid routing problem. The parameter variations cause the edge weights to be a probability distribution, and the cost functions are mostly related to the variations. Buffer insertion is a commonly used interconnection optimization technique. The possible buffer inserting location can be structured as nodes and the wire interconnection can be edges on the graph. Our algorithm can be implemented on the above listed nanometer circuit design applications to achieve efficient runtime.

REFERENCES

[1] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, “Parameter variations and impact on circuits and microarchitecture,” in *Proceedings of the 40th Design Automation Conference*, 2003, pp. 338-342.

[2] Incentia Design Systems, Inc., “Advanced On-chip-variation Timing Analysis for Nanometer Designs,” Tech. Rep., Jun. 2007.

[3] T. Karni, S. Borkar, and V. De, “Sub-90nm technologies—Challenges and opportunities for CAD,” in *Proceedings of the International Conference on Computer Aided Design*, 2002, pp. 203-206.

[4] D. Pham, S. Asano, M. Bolliger, M. N. Day, H. P. Hofstee, C. Johns, J. Kahle, A. Kameyama, J. Keaty, Y. Masubuchi, M. Riley, D. Shippy, D. Stasiak, M. Suzuoki, M. Wang, J. Warnock, S. Weitzel, D. Wendel, T. Yamazaki, and K. Yazawa, “The design and implementation of a first-generation CELL processor,” in *Proceedings of the International Solid-State Circuits Conference*, 2005, pp. 184-186.

[5] N. H. Chang, V. Kanevsky, O. S. Nakagawa, and S. Y. Oh, “Method and system for determining statistically based worst-case on-chip interconnect delay and crosstalk,” U.S. Patent 6,018,623, Jan. 25, 2000.

[6] E. W. Dijkstra, “A note on two problems in connection with graphs,” *Numerische Mathematik* , vol. 1, pp. 269-271, 1959.

[7] R. E. Bellman, “On a routing problem,” *Quarterly of Applied Mathematics*, vol. 16, no. 1, pp. 87-90, 1958.

[8] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms.* 2nd ed. Cambridge, MA: The MIT Press, McGraw-Hill, 2001.

[9] H. Frank, “Shortest paths in probabilistic graphs,” *Operations Research*, vol. 17, no. 4, pp. 583-599, Jul.-Aug. 1969.

[10] C. E. Sigal, A. A. B. Pritsker, and J. J. Solberg, “The stochastic shortest route problem,” *Operations Research Society of America*, vol. 28, no. 5, pp. 1122-1129, Sep.-Oct. 1980.

[11] R. P. Loui, “Optimal paths in graphs with stochastic or multidimensional weights,” *Communications of the ACM*, vol. 26, no. 9, pp. 670-676, Sep. 1983.

[12] I. Murthy and S. Sarkar, “Stochastic shortest path problems with piecewise-linear concave utility functions,” *Management Science*, vol. 44, no. 11, pp. S125-S136, Nov. 1998.

[13] R. Hall, “The fastest path through a network with random time-dependent travel time,” *Transportation Science*, vol. 20, no. 3, pp. 182-188, Aug. 1986.

[14] L. Fu and L. R. Rilett, “Expected shortest paths in dynamic and stochastic traffic networks,” *Transportation Research*, vol. 32, no. 7, pp. 499-516, Sep. 1998.

[15] X. Ji, “Models and algorithm for stochastic shortest path problem,” *Applied Mathematics and Computation*, vol. 170, no. 1, pp. 503-514, Nov. 2005.

[16] L. Deng and M. D. F. Wong, “An exact algorithm for the statistical shortest path problem,” in *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2006, pp. 965-970.

[17] Y. P. Aneja and K. P. K. Nair, “The constrained shortest path problem,” *Naval Research Logistics Quarterly*, vol. 25, no. 3, pp. 549-555, 1978.

[18] X. Cai, T. Kloks, and C. K. Wong, “Time-varying shortest path problems with constraints,” *Networks*, vol. 29, no. 3, pp. 141-150, Dec. 1998.

[19] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows, Theory, Algorithms, and Applications*. Upper Saddle River, NJ: Prentice Hall, 1993.

[20] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1999.