# Maximum Probability Shortest Path Problem

Jeremy Krebs - Guillaume Soulié

Université Paris Saclay

4 novembre 2017

Shortest Path is a known problem and has many applications in "real life".

- Goods transport (industrial and private)
- Food Delivery (Deliveroo - Foodora - UberEats)

As this problem is well known, lots of scientists presented their work on related subject, most of the times with different hypotheses :

As this problem is well known, lots of scientists presented their work on related subject, most of the times with different hypotheses :

- Without resource constraints

As this problem is well known, lots of scientists presented their work on related subject, most of the times with different hypotheses :

- Without resource constraints
- With deterministic resource constraints

As this problem is well known, lots of scientists presented their work on related subject, most of the times with different hypotheses :

- Without resource constraints
- With deterministic resource constraints
- With stochastic resource constraints

As this problem is well known, lots of scientists presented their work on related subject, most of the times with different hypotheses :

- Without resource constraints
- With deterministic resource constraints
- With stochastic resource constraints

or also with a different optimization problem like utility functions to maximize or cost functions to minimize.

- Graph with weights on arcs, source node $s$ and sink node $t$,
- Stochastic resource consumptions with normal distribution,
- K resources,
- Threshold C of the cost function (maximum allowed weight of the path).

SRCSP can be formulated as this optimization problem :

$$max\ \mathbf{Pr}\{\tilde{a}_k^T x \leq d_k, k = 1..K\}$$

$$\begin{aligned} s.t.\ c^T &\leq C \\ Mx &= b \\ x &\in \{0,1\}^n \end{aligned}$$

where :

- $x(e) = 1$ if $x(e) \in$ path $P$
- The $a_k$ are multi-variate vectors with mean $\mu_k$ and known covariance matrix $V_k$
- M is the node-arc incidence matrix. $M(i,e) \in \{-1,0,1\}$
- b is a vector with 0 everywhere except $b(s) = 1$ and $b(t) = -1$

SRCSP can be reformulated as :

$$
\begin{aligned}
& max\ \ p \\
s.t.\ \ & p \leq \mathbf{Pr}\{\tilde{a}_k^T x \leq d_k, k = 1..K\} \\
& c^T \leq C \\
& Mx = b \\
& x \in \{0,1\}^n
\end{aligned}
$$

With $K = 1$ we have :

$$\begin{aligned} & max \ p \\ s.t. \ & p \leq \mathbf{Pr}\{\tilde{a}_1^T x \leq d_1\} \\ & c^T \leq C \\ & Mx = b \\ & x \in \{0,1\}^n \end{aligned}$$

Using the known multivariate distribution parameters we get :

$$
\begin{aligned}
& max \;\; p \\
& s.t. \;\; F^{-1}(p)(x^T V_1 x)^{\frac{1}{2}} \leq d_1 - \mu_1^T x \\
& \qquad c^T \leq C \\
& \qquad Mx = b \\
& \qquad x \in \{0, 1\}
\end{aligned}
$$

Relaxing the problem we get (SRCSPI) with $p \leq \frac{1}{2}$ :

$$
\begin{aligned}
&max \ 0 \\
&s.t. \ F^{-1}(p)(x^T V_1 x)^{\frac{1}{2}} \leq d_1 - \mu_1^T x \\
&\quad c^T \leq C \\
&\quad Mx = b \\
&\quad 0 \leq x \leq 1
\end{aligned}
$$

We can solve it using the binary search procedure. We take $p_1 \leq \frac{1}{2}$ a feasible solution, a lower bound of SRCSP. $p_l$ and $p_u$ are lower and upper bounds of SRCSPI. Then we iterate :

We can solve it using the binary search procedure. We take $p_1 \leq \frac{1}{2}$ a feasible solution, a lower bound of SRCSP. $p_l$ and $p_u$ are lower and upper bounds of SRCSPI. Then we iterate :

**Start** $p_l = \frac{1}{2}$ and $p_u = 1$. Iteration counter $t = 1$

We can solve it using the binary search procedure. We take $p_1 \leq \frac{1}{2}$ a feasible solution, a lower bound of SRCSP. $p_l$ and $p_u$ are lower and upper bounds of SRCSPI. Then we iterate :

**Start** $p_l = \frac{1}{2}$ and $p_u = 1$. Iteration counter $t = 1$

**Search** Solve SRCSPI with $p = p_t$. If SRCSPI has an optimal solution, set $p_l = p_t$, otherwise $p_u = p_t$.

We can solve it using the binary search procedure. We take $p_1 \leq \frac{1}{2}$ a feasible solution, a lower bound of SRCSP. $p_l$ and $p_u$ are lower and upper bounds of SRCSPI. Then we iterate :

**Start** $p_l = \frac{1}{2}$ and $p_u = 1$. Iteration counter $t = 1$

**Search** Solve SRCSPI with $p = p_t$. If SRCSPI has an optimal solution, set $p_l = p_t$, otherwise $p_u = p_t$.

**Stop** Stop when $\frac{p_u - p_l}{2} \leq \epsilon$. Otherwise $t++$ and $p_t = \frac{p_l + p_u}{2}$

Put the formulation Explain why we are looking for convexity, using the lectures Explain how we get the approximation (Theorem 4.1.2), saying that we used this method in classes

We formulate this problem taking individual probabilities constraints :

$$\begin{aligned} &max\ p \\ s.t.\ & p \leq \mathbf{Pr}\{\tilde{a}_k^T x \leq d_k\},\ \mathbf{k=1,..,K} \\ & c^T \leq C \\ & Mx = b \\ & x \in \{0,1\}^n \end{aligned}$$

We can relax the equation as we did before :

$$
\begin{aligned}
& max\ 0 \\
s.t.\ & F^{-1}(p)(x^T V_k x)^{\frac{1}{2}} \leq d_k - \mu_k^T x,\ \ k = 1, .., K \\
& c^T \leq C \\
& Mx = b \\
& 0 \leq x_i \leq 1,\ \ i = 1, .., n
\end{aligned}
$$

We use the relaxed equation using the Binary Search Procedure again :

**Start** $p_l = \frac{1}{2}$ and $p_u = 1$. Iteration counter $t = 1$

**Search** Solve SRCSPI with $p = p_t$. If SRCSPI has an optimal solution, set $p_l = p_t$, otherwise $p_u = p_t$.

**Stop** Stop when $\frac{p_u - p_l}{2} \leq \epsilon$. Otherwise $t++$ and $p_t = \frac{p_l + p_u}{2}$

Show results and analyze them (Why is this algorithm faster, why not, etc)