Extended Shortest Path Problem
Generalized Dijkstra-Moore and Bellman-Ford Algorithms

Maher Helaoui
Higher Institute of Business Administration, University of Gafsa, Rades, Tunisia

Keywords: Combinatorial Optimization, Valuation Structure, Extended Shortest Path Problem, Longest Path Problem, Generalized Dijkstra-Moore Algorithm, Generalized Bellman-Ford Algorithm.

Abstract: The shortest path problem is one of the classic problems in graph theory. The problem is to provide a solution algorithm returning the optimum route, taking into account a valuation function, between two nodes of a graph G. It is known that the classic shortest path solution is proved if the set of valuation is $\mathbb{R}$ or a subset of $\mathbb{R}$ and the combining operator is the classic sum $(+)$. However, many combinatorial problems can be solved by using shortest path solution but use a set of valuation not a subset of $\mathbb{R}$ and/or a combining operator not equal to the classic sum $(+)$. For this reason, relations between particular valuation structure as the semiring and diod structures with graphs and their combinatorial properties have been presented. On the other hand, if the set of valuation is $\mathbb{R}$ or a subset of $\mathbb{R}$ and the combining operator is the classic sum $(+)$, a longest path between two given nodes s and t in a weighted graph G is the same thing as a shortest path in a graph $-G$ derived from G by changing every weight to its negation.
In this paper, in order to give a general model that can be used for any valuation structure we propose to model both the valuations of a graph G and the combining operator by a valuation structure S. We discuss the equivalence between longest path and shortest path problem given a valuation structure S. And we present a generalization of the shortest path algorithms according to the properties of the graph G and the valuation structure S.

1 INTRODUCTION

The shortest path problem is one of the classic problems in graph theory. The problem is to provide a solution algorithm returning the optimum route, taking into account a valuation function, between two nodes of a graph G.
In (Shimbel, 1955; Ford and Lester, 1956; Bellman, 1958; Sedgewick and Wayne, 2011) the classic shortest path solution is proved if

- the set of valuation is $\mathbb{R}$ or a subset of $\mathbb{R}$.
- the combining operator is the classic sum $(+)$

Many combinatorial problems like Fuzzy, Weighted, Probabilistic and Valued Constraint Satisfaction Problem (Schiex et al., 1995; Cooper, 2003; Cooper, 2004; Allouche et al., 2009) use a set of valuation E not subset of $\mathbb{R}$ and a combining operator $\oplus \neq +$ for weighted, fuzzy, probabilistic ... valuations. In (Cooper, 2003), the shortest path algorithm has been used to solve Fuzzy and Valued Constraint Satisfaction Problem.
In (Erickson, 2010), author observes that the classical maximum flow problem (Ford and Fulkerson, 1955; Ford and Fulkerson, 1962) in any directed planar graph G can be reformulated as a parametric shortest path problem in the oriented dual graph G^*. In (Cohen et al., 2004; Helaoui et al., 2013), a submodular decompositions approach has been presented to solve Valued Constraint Satisfaction Problem. This solution use the maximum flow algorithm.
Dijkstra-Moore and Bellman-Ford Algorithms are the most known algorithmic solutions for the shortest path problem.

- Since 1971, the Dijkstra-Moore Algorithm has been used if the set of valuation is $\mathbb{R}^+$ or a subset of $\mathbb{R}^+$ and the combining operator is the classic sum $(+)$.
- The Bellman-Ford Algorithm is the result of (Shimbel, 1955; Ford and Lester, 1956; Bellman, 1958) works. It is used if the set of valuation is $\mathbb{R}$ or a subset of $\mathbb{R}$ and the combining operator is the classic sum $(+)$.

As many combinatorial problems can be solved by using shortest path solution but use a set of valuation

Helaoui M.
Extended Shortest Path Problem - Generalized Dijkstra-Moore and Bellman-Ford Algorithms.
DOI: 10.5220/0006145303060313
In *Proceedings of the 6th International Conference on Operations Research and Enterprise Systems (ICORES 2017)*, pages 306-313
ISBN: 978-989-758-218-9

not a subset of $\mathbb{R}$ and/or a combining operator not equal to the classic sum $(+)$, then in (Gondran and Minoux, 2008), authors present new models and algorithms discussing relations between particular valuation structure: the semiring and diod structures with graphs and their combinatorial properties.
In (Sedgewick and Wayne, 2011), a longest path between two given nodes s and t in a weighted graph G is the same thing as a shortest path in a graph $-G$ derived from G by changing every weight to its negation. Therefore, if shortest paths can be found in $-G$, then longest paths can also be found in G. This result remains true if we have a valued graph G by a valuation structure S?
In this paper, we provide an answer to this question by discussing equivalence between longest path and shortest path problem given a valuation structure S. We present a generalization of Dijkstra-Moore Algorithm for a graph G with a $S^{\oplus}$ valuation structure. And we present a generalization of Bellman-Ford Algorithm with a more general valuation structure S.
We propose to model both the valuations of a graph G and the combining operator by a valuation structure S, in order to discuss the generalization of the shortest path algorithms according to the properties of the graph G and the valuation structure S:

- The valuation structure of G is $S^{\oplus}$.
- The graph G and the valuation structure S are arbitrary.

The paper is organized as follows: the next Section introduces definitions and notations needed in presenting the generalization of the shortest path algorithms. In Section 3 we study the extended Shortest Path Notion and the equivalence between longest path and shortest path problem. We propose a generalized shortest path algorithms in Section 4. The paper is concluded in Section 5.

2 DEFINITIONS AND NOTATIONS

2.1 A Directed Digraph G

The peculiarity of the shortest path problem requires to distinguish two directions between any two nodes. In this case, the connection between two nodes x and y can be defined by the directed connection between an original node for example x and a destination node y.

Definition 1. *A directed digraph $G = (E_S, E_{\vec{A}})$ is defined by a set of nodes E_S and a set of directed edges $E_{\vec{A}}$, each edge (arc) is the connection between an original node and a destination node.*
If x and y are two nodes:

- *the directed connection from x to y (denoted $\vec{xy}$), if it exists, is a directed connection (arc) of a graph G.*
- *An arc $\vec{xx}$: the directed connection from x to x is known as a loop.*
- *A p-graph is a graph wherein there is never more than p arcs $\vec{xy}$ between any two nodes.*
- *A Monograph is a graph wherein there is never more than 1 arc $\vec{xy}$ between any two nodes and there is never a loop.*

2.2 A Valuation Structure

We assume that E the set of all possible valuations, is a totally ordered set where $\perp$ denotes its minimal element and $\top$ its maximal element. In addition, we will use a monotone binary operator $\oplus$. These elements form a valuation structure defined as follows

Definition 2. *A valuation structure S of a graph G is the triplet $S = (E, \oplus, \preceq)$ such as*

- *E is the set of possible valuations;*
- *$\preceq$ is a total order on E;*
- *$\oplus$ is commutative, associative and monotone.*

We define below a fire and strictly monotone valuation structure.

Definition 3. *A valuation structure S is* fire *if for each pair of valuations $\alpha, \beta \in E$, such as $\alpha \preceq \beta$, there is a maximum difference between β and α denoted $\beta \ominus \alpha$.*
An aggregation operator $\oplus$ is strictly monotonic *if for any α, β, γ in E such as $\alpha \prec \beta$ and $\gamma \neq \top$, we have $\alpha \oplus \gamma \prec \beta \oplus \gamma$.*
A valuation structure S is strictly monotonic *if it has an aggregation operator* strictly monotonic.

In the remainder of this paper, we use only fire and strictly monotone valuation structures.
The fire and strictly monotone valuation structures satisfy the following two Lemmas, that has been proved in (Cooper, 2004), (Lemma 7 and Theorem 38).

Lemma 1. *Let $S = (E, \oplus, \preceq)$ a valuation structure fire and strictly monotone. Then for all $\alpha, \beta, \gamma \in E$ such as $\gamma \preceq \beta$, we have $(\beta \ominus \gamma) \preceq \beta$ and $(\alpha \oplus \gamma) \oplus (\beta \ominus \gamma) = \alpha \oplus \beta$.*

Lemma 2. *Let $S = (E, \oplus, \preceq)$ a valuation structure fire and strictly monotone. Then for all $\alpha, \beta, \gamma \in E$ such as $\gamma \preceq \beta$, we have $(\alpha \oplus \beta) \ominus \gamma = \alpha \oplus (\beta \ominus \gamma)$.*

Using both Lemmas (Lemma 1 and Lemma 2) presented above we can get Lemma 3:

Lemma 3. *Let* $\beta \prec \alpha$ *and* $\gamma \prec \alpha$
$\alpha \ominus \beta \prec \alpha \ominus \gamma$ *if and only if* $\gamma \ominus \alpha \prec \beta \ominus \alpha$.

Proof. ($\Rightarrow$) If we have $\alpha \ominus \beta \prec \alpha \ominus \gamma$
then $\alpha \ominus \beta \oplus (\beta \ominus \alpha \oplus \gamma \ominus \alpha) \prec \alpha \ominus \gamma \oplus (\beta \ominus \alpha \oplus \gamma \ominus \alpha)$
then $\gamma \ominus \alpha \prec \beta \ominus \alpha$
($\Leftarrow$) If we have $\gamma \ominus \alpha \prec \beta \ominus \alpha$
then $\gamma \ominus \alpha \oplus (\alpha \ominus \beta \oplus \alpha \ominus \gamma) \prec \beta \ominus \alpha \oplus (\alpha \ominus \beta \oplus \alpha \ominus \gamma)$
then $\alpha \ominus \beta \prec \alpha \ominus \gamma$. □

We define below a particular valuation structure, widely used in practice, that we will note $S^{\oplus}$

Definition 4. *A valuation structure* $S^{\oplus}$ *of a graph G is the triplet* $S^{\oplus} = (E^{\oplus}, \oplus, \preceq)$ *such as:*

- $E^{\oplus}$ *is the set of possible valuations such as for all* $\alpha, \beta, \lambda \in E^{\oplus}$ *if* $\alpha \preceq \beta$ *then* $\alpha \preceq \beta \oplus \lambda$*;*
- $\preceq$ *is a total order on E;*
- $\oplus$ *is commutative, associative and monotone.*

3 SHORTEST PATH NOTION

3.1 Extended Shortest Path Problem

In the beginning of this paragraph we formally define the shortest path between two nodes *x* and *y* of a graph *G*.
For this way, we start by defining the arc and path valuations.

Definition 5. *Let* $G = (E_S, E_{\vec{A}})$ *a valued directed graph. In each arc* $\vec{xy}$ *we associate a valuation function* $\varphi : E_S \times E_S \rightarrow E$ *such as* $\varphi(x,y)$ *is the valuation of* $\vec{xy}$ *arc. A path between two nodes x and y is denoted* $CH(x,y)$ *from the node x to a node y.*
For each path $CH(x,y)$ *we associate a valuation* $\Phi(CH(x,y))$.

$$\Phi(CH(x,y)) = [\bigoplus_{x_i\vec{x}_j \in CH(x,y)} \varphi(x_i, x_j)]$$

Now we can define the shortest path:

Definition 6. *Let* $G = (E_S, E_{\vec{A}})$ *a valued directed graph. The shortest path between x and y is the path* $\mu(x,y)$ *started from a node x and finished at y such as:*

$$\Phi(\mu(x,y)) = Min_{CH(x,y)}[\Phi(CH(x,y))]$$

3.2 Equivalent Problems of Shortest Path Problem (SPP)

If we have to find the Longest Path, is it possible to use the Shortest Path solution?
Finding the Longest Path is it equivalent to finding the Shortest Path?

Theorem 1. *Given a fire and strictly monotone valuation structure S, finding the Longest Path Problem is equivalent to finding the Shortest Path Problem.*

Proof Theorem 1. ($\Rightarrow$) We start from a shortest path problem. We replace each valuation ($\alpha \ominus \beta$) by the valuation ($\beta \ominus \alpha$) and we prove that a shortest path problem can be transformed in a longest path problem.
Let

$$\Phi(\mu(s,t)) = \min \Phi(CH(s,t))$$
$$\Phi(\mu(s,t)) = (\alpha \ominus \beta_0) \preceq \Phi(CH_1(s,t)) = (\alpha \ominus \beta_1) \preceq \dots \preceq \Phi(CH_n(s,t)) = (\alpha \ominus \beta_n)$$

If we replace each valuation ($\alpha \ominus \beta$) by the valuation ($\beta \ominus \alpha$)
We get

$$\Phi(\mu(s,t)) = (\beta_0 \ominus \alpha) \succeq \Phi(CH_1(s,t)) = (\beta_1 \ominus \alpha) \succeq \dots \succeq \Phi(CH_n(s,t)) = (\beta_n \ominus \alpha)$$

then we get

$$\Phi(\mu(s,t)) = \max \Phi(CH(s,t)) = \Phi(\mathcal{L}(s,t))$$

($\Leftarrow$) Now we start from a longest path problem. We replace each valuation ($\alpha \ominus \beta$) by the valuation ($\beta \ominus \alpha$) and we prove that a longest path problem can be transformed in a shortest path problem.
Let

$$\Phi(\mathcal{L}(s,t)) = \max \Phi(CH(s,t))$$
$$\Phi(\mathcal{L}(s,t)) = (\alpha \ominus \beta_0) \succeq \Phi(CH_1(s,t)) = (\alpha \ominus \beta_1) \succeq \dots \succeq \Phi(CH_n(s,t)) = (\alpha \ominus \beta_n)$$

If we replace each valuation ($\alpha \ominus \beta$) by the valuation ($\beta \ominus \alpha$)
We get by Lemma 3

$$\Phi(\mathcal{L}(s,t)) = (\beta_0 \ominus \alpha) \preceq \Phi(CH_1(s,t)) = (\beta_1 \ominus \alpha) \preceq \dots \preceq \Phi(CH_n(s,t)) = (\beta_n \ominus \alpha)$$

Then we get

$$\Phi(\mathcal{L}(s,t)) = \min \Phi(CH(s,t)) = \Phi(\mu(s,t))$$

□

Example 1. In February 2017, a large multinational X in Porto wishes to invest the sum of 3.000.000 € in a new project Π. To do this, each year X has d investment choices. One study allowed him to estimate the certainty of acceptable profitability (probabilities) according to the various decisions taken. In order to maximize the certainty of an acceptable overall yield of Π, X wishes to find the longest path in the monograph G_1, such that, $S_1 = (]0,1[,\times,\leq)$.

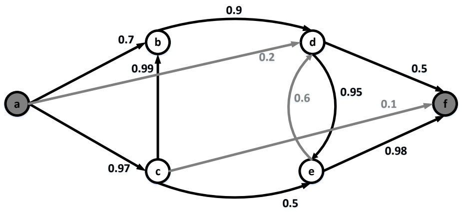

Figure 1: A directed monograph G_1.

By Theorem 1 and in order to maximize the certainty probabilities of an acceptable overall yield of Π, X can find the shortest path in the monograph G_2 minimize the uncertainty probabilities. Such that $S_2 = S_1$. For each valuation α in G_1 we associate a valuation β in G_2 such that $\beta = \ominus\alpha = 1-\alpha$ $\alpha \oplus \beta = \alpha \times (1-\alpha)$ $\Rightarrow$ **S_1 and S_2 are different from the semiring or diod structures**.

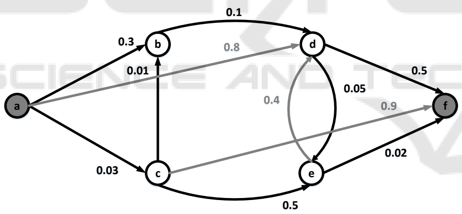

Figure 2: A directed monograph G_2.

3.3 Optimality Notion

The dynamic programming repose on the fundamental principle of optimality:
Given a directed graph G, a sub-path of a shortest path $\mu \in G$ is a shortest path in G_s a sub-graph of G.

Theorem 2. *Let:*

- *A directed graph $G = (E_S, E_{\vec{A}})$,*
- *A valuation function of G $\varphi : E_S \times E_S \rightarrow E$ with a fire and strictly monotone valuation structure S,*
- *a shortest path $\mu(x_1,x_k)$ from $x_1 \in E_S$ to $x_k \in E_S$ $\mu(x_1,x_k) = x_1 \rightarrow x_2 \rightarrow \ldots \rightarrow x_k = x_1 \overset{*}{\rightarrow} x_k$,*
- *A sub-path $CH(x_i,x_j)$ of $\mu(x_1,x_k)$ from x_i to x_j: $CH(x_i,x_j) = x_i \rightarrow \ldots \rightarrow x_j$ such as $x_i \overset{*}{\rightarrow} x_j$* [1].

Then $CH(x_i,x_j) = \mu(x_i,x_j)$ is the shortest path from $x_i \overset{*}{\rightarrow} x_j$.

Proof Theorem 2. We proceed by absurd reasoning. Assume that:

1. $\mu(x_1,x_k)$ is a shortest path
2. CH(x_i,x_j) is a sub path of $\mu(x_1,x_k)$
3. It $\exists$ CH'(x_i,x_j) of G such as $\Phi(\text{CH'}(x_i,x_j)) \prec \Phi(\text{CH}(x_i,x_j))$.

We proceed to get a contradiction.
Decompose $\mu(x_1,x_k)$ in CH(x_1,x_i), CH(x_i,x_j) and CH(x_j,x_k)
then
$\Phi(\mu(x_1,x_k)) = \Phi(\text{CH}(x_1,x_i)) \oplus \Phi(\text{CH}(x_i,x_j)) \oplus \Phi(\text{CH}(x_j,x_k))$.
As it $\exists$ CH'(x_i,x_j) such as $\Phi(\text{CH'}(x_i,x_j)) \prec \Phi(\text{CH}(x_i,x_j))$ and given that $\oplus$ is strictly monotone then
$\Phi(\mu(x_1,x_k)) \succ \Phi(\text{CH}(x_1,x_i)) \oplus \Phi(\text{CH'}(x_i,x_j)) \oplus \Phi(\text{CH}(x_j,x_k))$, which is a contradiction by the fact that the path $\mu(x_1,x_k)$ is a shortest path then CH$(x_i,x_j) = \mu(x_i,x_j)$.

□

4 GENERALIZATION OF DIJKSTRA-MOORE AND BELLMAN-FORD ALGORITHMS

4.1 Common Functions

The generalized Dijkstra-Moore and Bellman-Ford algorithms presented in this paper use

1. Algorithm 1 is an initialization marker algorithm of all nodes of G that we will denote INITMARK(G,s);
2. Algorithm 2 is an initialization finding shortest path algorithm that we will denote INITSPP(G,s);

[1]
- x_i is an immediate predecessor or a non immediate of x_j (it exists a path from x_i to x_j).
- x_i is an immediate successor or a non immediate of x_1 or $x_i = x_1$
- x_j is an immediate predecessor or a non immediate of x_k or $x_j = x_k$

3. The updating shortest path Algorithm, denoted UPDATESPP(x_i, x_j, φ).
   Updating the shortest path between two nodes x_i and x_j consists in updating the valuation of one of the arcs $\vec{x_i x_j}$:

   (a) The valuation spv$[x_j]$ of the shortest path until x_j;
   (b) The predecessor of x_j predspp$[x_j]$ in the shortest path until x_j.

Algorithm 1: InitMark(G,s):Mark.

---

for $(x_i = s \;\; Nbr_{Sommets_G})$ **do**
  Mark$[x_i] \;\leftarrow\; 0$;
Mark$[s] \;\leftarrow\; 2$;

Algorithm 2: InitSpp(G,s):spv,predspp.

---

for $(x_i = s \;\; Nbr_{Sommets_G})$ **do**
  spv$[x_i] \;\leftarrow\; \top$;
  predspp$[x_i] \;\leftarrow\; 0$;
spv$[s] \;\leftarrow\; \alpha_0$;

The updating process is based on the Theorem 2 where each sub-path of the shortest path is a shortest path in the sub-graph involving this sub-path.
The UpdateSpp(x_i, x_j, φ) function update the shortest path from one origin node to all other one if a shortest path is detected.

Algorithm 3: UpdateSpp(x_i, x_j, φ):spv,predspp.

---

if $(spv[x_j] \;\succ\; spv[x_i] \;\oplus\; \varphi[x_i][x_j])$ **then**
  spv$[x_j] \;\leftarrow\; spv[x_i] \;\oplus\; \varphi[x_i][x_j]$;
  predspp$[x_j] \;\leftarrow\; x_i$;

In the DIJKSTRA-MOORE Algorithm 4, each arc is updated exactly one way. In BELLMAN-FORD Algorithm, each arc can be updated many way.

4.2 Generalization of the Dijkstra-Moore Algorithm

If the arcs valuation, is in $\mathbb{R}$, can model for example

- A distance (kilometers)
- A cost (€)

In this case, the classic Dijkstra-Moore Algorithm can be used.
In this paper, we present a generalization of Dijkstra-Moore Algorithm 4 for a graph G with a $S^{\oplus}$ valuation structure.

Let G a valued directed graph given a valuation structure $S^{\oplus}$. We denote by s the origin node of G and x_i the destination node. For each node x_i of G, the Algorithm 4 associate

- The valuation spv$[x_j]$ for the shortest sub-path until x_i;
- The predecessor of x_i predspp$[x_i]$ in the shortest sub-path until x_i.
- The marker of x_i denoted Mark$[x_i]$ verifying if the distance from s to x_i has been updated.

Principe of the Algorithm:

1. initialization:
   For the node s
   - Mark$[s] \;\leftarrow 2$
   - predspp$[s] \;\leftarrow 0$
   - spv$[s] \;\leftarrow \alpha_0$

   for all other nodes x_i
   - Mark$[x_i] \;\leftarrow 0$
   - predspp$[x_i] \;\leftarrow 0$
   - spv$[x_i] \;\leftarrow \top$
2. Let $X =$ the set of non marked nodes;
   Do
   - **For** each non marked node i successor of y
     – UpdateSpp(y, i, φ)
   - Mark the node y if spv$[y] = \min_{x \in X} pcc[x]$.

   While $X \neq \emptyset$

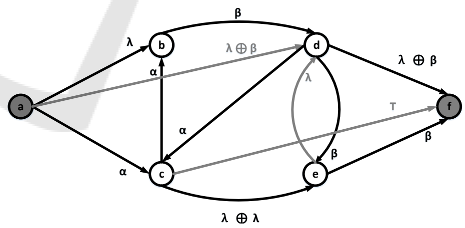

Figure 3: A directed monograph G.

Example 2. Let the directed monograph G presented by Figure 3. And let a valuation structure $S^{\oplus}$ (can be $\neq$ to the semiring or diod structures) such that

- $\{\alpha,\ \beta,\ \lambda,\ \gamma,\ \top\} \subset E^{\oplus}$
- $\alpha \prec \beta \prec \lambda$
- $\beta = \alpha \oplus \alpha$
- $\lambda = \alpha \oplus \beta$
- if $\gamma \succeq \lambda \oplus \lambda \oplus \lambda \oplus \beta$ then $\gamma = \top$

1. We apply the principle of Generalized Dijkstra algorithm on G: Figure 4
2. We present an algorithmic solution: Algorithm 4.

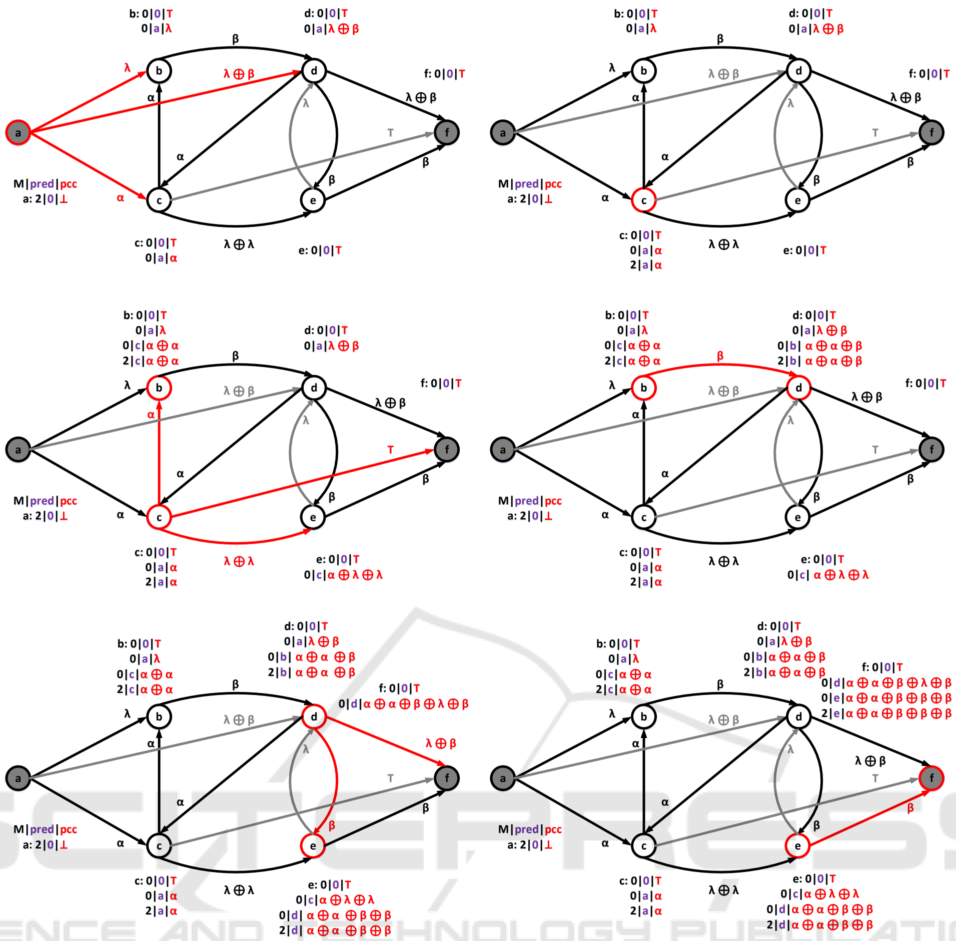

Figure 4: Applying Generalized-Dijkstra-Moore on G.

Algorithm 4: Generalized-Dijkstra-Moore(G,φ,s,t): spv,predspp.

InitMark(G,s);
InitSpp(G,s);
recent $\leftarrow$ s;
$t \leftarrow f$;
while $(Mark[t]=0)$ **do**
  $j \leftarrow 0$;
  while $(succ[recent][j])$ **do**
    if $(Mark[j]=0)$ **then**
      UpdateSpp$(recent, j, \varphi)$;
    $j \leftarrow j+1$;
  $y \leftarrow \min(spv)$;
  $Mark[y] \leftarrow 2$;
  $recent \leftarrow y$;

Theorem 3. *Given a directed monograph G with n nodes and a valuation structure $S^{\oplus}$, the shortest path from one started node to all others can be done in $O(n^2)$.*

Proof Theorem 3. *Given a directed monograph G with n nodes and a valuation structure $S^{\oplus}$, and referred to Theorem 2, the shortest path from one started node to all others can be done by applying Algorithm 4 to G. And the Algorithm 4 run in $O(n^2)$.* □

4.3 Generalization of Bellman-Ford Algorithm

Given a fire and strictly monotone valuation structure S we can model as example earnings and bounded costs!
Unfortunately, as nodes may be marked only once, the DIJKSTRA-MOORE algorithm does not guarantee the optimal solution if we consider the valuation structure not a subset of $S^{\oplus}$ (For example the bounded negative arcs). In fact, once the node is marked we cannot change the marking in subsequent iterations.
Fortunately, we can present an algorithm that ensures marking update until the program is not determined:

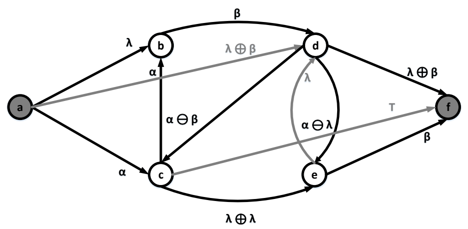

Figure 5: A directed monograph $G^{'}$.

a generalization of the BELLMAN-FORD Algorithm can be used for a fire and strictly monotone valuation structure S.

Theorem 4. *Given a fire and strictly monotone valuation structure S, the final values of the shortest paths are obtained by at most $n-1$ iterations.*

Proof Theorem 4. In the absence of absorbing circuitry, a shortest path from s to all other nodes is an element path, that is to say a path of at most $n-1$ arcs. By consulting the predecessors of all nodes Algorithm 5 must obtain the final values of the shortest paths by at most $n-1$ iterations.

□

Corollary 1. *If after n iterations, the values $spv[i]$ continue to be modified, is that the graph has an absorbent circuitry.*

Based on the results of Theorem 4 and Corollary 1 we can introduce the principle of the BELLMAN-FORD generalization algorithm.

Principle of the BELLMAN-FORD generalization algorithm:

1. InitSpp(G,s)
2. **Do**
   UpdateSpp(i,j,φ)
   While there is an edge to decrease spv[i].

Algorithm 5 presents an algorithmic solution for the generalized BELLMAN-FORD algorithm.

Example 3. Let the directed monograph $G^{'}$ given by Figure 5. And let a valuation structure S (can be $\neq$ to the semiring or diod structures) such that

- $\{\alpha,\ \beta,\ \lambda,\ \gamma,\ \top\} \subset E$
- $\alpha \prec \beta \prec \lambda$
- $\beta = \alpha \oplus \alpha$
- $\lambda = \alpha \oplus \beta$
- if $\gamma \succeq \lambda \oplus \lambda \oplus \lambda \oplus \beta$ then $\gamma = \top$

1. Present an algorithmic solution: Algorithm 5.

Algorithm 5: Generalized-Bellman-Ford(G,φ,s): spv,predspp.

```
InitSpp(G, s);
k ← 0;
t ← 1;
while (t ou k > N_SommetG − 1) do
    t ← 0;
    for (i de 1 N_SommetG) do
        j ← 0;
        while (predspp[i][j]) do
            UpdateSpp(j, i, φ);
            if (spv[i] ≻ spv[j] ⊕ φ[j][i]) then
                t ← 1;
            j ← j + 1;
    k ← k + 1;
```

Theorem 5. *Given a directed monograph G with n nodes and a fire and strictly valuation structure S, the shortest path from one started node to all others can be done in $O(n^3)$.*

Proof Theorem 5. *Given a directed monograph G with n nodes and a fire and strictly valuation structure S, and referred to Theorem 2, Theorem 4 and Corollary 1, the shortest path from one started node to all others can be done by applying Algorithm 5 to G. And the Algorithm 5 run in $O(n^3)$.*

□

5 CONCLUSION

This paper addressed combinatorial problems that can be expressed as shortest path solution but use a set of valuation not a subset of $\mathbb{R}$ and/or a combining operator not equal to the classic sum $(+)$.
Firstly, we have modeled the valuations of a graph G by using a general valuation structure S.
Secondly, given a general valuation structure S, we have discussed the equivalence between longest path and shortest path problem.
And finally, we have discussed the generalization of the shortest path algorithms according to the properties of the graph G and the valuation structure S:

1. The valuation structure of G is $S^{\oplus}$.
2. The graph G and the valuation structure S are arbitrary.

REFERENCES

Allouche, D., de Givry, S., Sanchez, M., and Schiex, T. (2009). Tagsnp selection using weighted csp and russian doll search with tree decomposition 1. In *In Proceedings of the WCB09. Lisbon Portugal, 1–8*.

Bellman, R. (1958). On a routing problem. In *Quarterly of Applied Mathematics. 16: 87–90*.

Cohen, D. A., Cooper, M. C., Jeavons, P. G., and Krokhin, A. A. (2004). A maximal tractable class of soft constraint satisfaction. In *Journal of Artificial Intelligence Research 22, 1–22*.

Cooper, M. C. (2003). Reduction operations in fuzzy or valued constraint satisfaction. In *Fuzzy Sets and Systems 134 311–342*.

Cooper, M. C. (2004). Arc consistency for soft constraints. In Artificial Intelligence *154, 199–227*.

Erickson, J. (2010). Maximum flows and parametric shortest paths in planar graphs. In *In Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms, 794–804*.

Ford, J. and Lester, R. (1956). Network flow theory. In *RAND Corporation, 923*, Santa Monica, California.

Ford, L. R. and Fulkerson, D. R. (1955). A simple algorithm for finding maximal network flows and an application to the hitchock problem. In *In Rand Report Rand Corporation*, Santa Monica, California.

Ford, L. R. and Fulkerson, D. R. (1962). Flows in networks. In *In Princeton University Press*.

Gondran, M. and Minoux, M. (2008). Graphs, dioids and semirings: New models and algorithms (operations research/computer science interfaces series). In *Springer Publishing Company*.

Helaoui, M., Naanaa, W., and Ayeb, B. (2013). Submodularity-based decomposing for valued csp. In *International Journal on Artificial Intelligence Tools*.

Schiex, T., Fargier, H., and Verfaillie, G. (1995). Valued constraint satisfaction problems: hard and easy problems. In *In Proceedings of the* 14^{th} *IJCAI, 631–637*, Montréal, Canada.

Sedgewick, R. and Wayne, K. D. (2011). Algorithms (4th ed.). In *Addison-Wesley Professional, 661–666*.

Shimbel, A. (1955). Structure in communication nets. In *Proceedings of the Symposium on Information Networks, 199–203*, New York, NY: Polytechnic Press of the Polytechnic Institute of Brooklyn.