ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 10, No. 3, 2015, *pp*. 163-168

Improving Computing Performance for Algorithm Finding Maximal Flows on Extended Mixed Networks

Viet Tran Ngoc [1], Chien Tran Quoc [2] and Tau Nguyen Van [3]

[1] Vietnam Korea friendship IT college, Danang, Vietnam, trviet01@yahoo.com

[2] The university of Danang, Vietnam, tqchien@dce.udn.vn

[3] Gialai Department of Education and Training, Gialai, Vietnam, nguyentau2004@yahoo.com
(*Received november 23, 2014, accepted February 21, 2015)*

Abstract. Graph is a powerful mathematical tool applied in many fields as transportation, communication, informatics, economy, … In ordinary graph the weights of edges and vertexes are considered independently where the length of a path is the sum of weights of the edges and the vertexes on this path. However, in many practical problems, weights at a vertex are not the same for all paths passing this vertex, but depend on coming and leaving edges. The paper develops a model of extended network that can be applied to modelling many practical problems more exactly and effectively. The main contribution of this paper is a source-sink alternative algorithm, then improving computing performance for algorithm finding maximal flows on extended mixed networks.

Keywords: extended, graph, network, flow, maximal flow, algorithm.

1. Introduction

Graph is a powerful mathematical tool applied in many fields as transportation, communication, informatics, economy, … In ordinary graph the weights of edges and vertexes are considered independently where the length of a path is simply the sum of weights of the edges and the vertexes on this path. However, in many practical problems, weights at a vertex are not the same for all paths passing this vertex, but depend on coming and leaving edges. Therefore, a more general type of weighted graphs, called extended weighted graph, is defined in this work. The paper develops a model of extended network that can be applied to modelling many practical problems more exactly and effectively. Therefore, necessary to build a model of the extended network so that the stylization of practical problems can be applied more accurately and effectively. Based on the results of the study of the problem regarding finding the maximum flow [1], [2] and extended graphs [3], the main contribution of this paper is the revised Ford-Fulkerson algorithm finding maximal flows on extended mixed networks and improving computing performance.

2. Extended Mixed Network

A network is a mixed graph of the traffic $G = (V, E)$, circles V and roads E. Roads can be classified as either direction or non-direction. There are many sorts of means of transportation on the network. The non-direction shows two-way roads while the direction shows one-way roads. Given a group of the functions on the network as follows:

+The function of the route circulation possibility $c_E : \mathrm{E} \to \mathrm{R}^*$, $c_E(e)$ the route circulation possibility $e \in \mathrm{E}$.

+The function of the circle circulation possibility $c_V : \mathrm{V} \to \mathrm{R}^*$, $c_V(u)$ the circle circulation possibility $u \in \mathrm{V}$.

+$G = (V, E, c_E, c_V)$: extended mixed network.

3. Flow of The Extended Mixed Network

Given an extended mixed network $G = (V, E, c_E, c_V)$, a source point a and a sink point z.
Set: $\{f(x,y) \mid (x,y)\in G\}$, is called the flow of network G if the requirements are met:

(i) $0 \le f(x,y) \le c_E(x,y)\ \forall (x,y)\in G$

(ii) Any value of point r is referring to neither a sourse point nor a sink point

$$\sum_{(v,r)\in G} f(v,r) = \sum_{(r,v)\in G} f(r,v)$$

(iii) Any value of point r is referring to neither a sourse point nor a sink point

$$\sum_{(v,r)\in G} f(v,r) \le c_V(r)$$

Expression: $v(\mathrm{F}) = \sum_{(a,v)\in G} f(a,v)$, is called the value of flow F.

• ***The maximum problem:***

Given an extended mixed network $G = (V, E, c_E, c_V)$, a source point a and a sink point z. The task required by the problem is finding the flow which has a maximum value. The flow value is limited by the total amount of the circulation possibility on the roads starting from source points. As a result of this, there could be a confirmation on the following theorem.

• *Theorem 1:* Given an extended mixed network $G = (V, E, c_E, c_V)$, a source point a and a sink point z, then exist is the maximal flow [1].

4. Source-Sink Alternative Algorithm Finding Maximal Flows on Extended Mixed Networks [2]

+ ***Input:*** Given an extended mixed network $G = (V, E, c_E, c_V)$, a source point a and a sink point z. The points in graph G are arranged in a certain order.
+ ***Output:*** Maximal flow $F = \{f(x,y) \mid (x,y)\in G\}$.

(1) Start:

The departure flow: $f(x,y) := 0,\ \forall (x,y)\in G$.
Points from the source points and sink points will gradually be labelled L_1 for the first time including 5 components.

Form forward label:

$L_1(v) = [\uparrow, prev_1(v), c_1(v), d_1(v), bit_1(v)]$ and can be label $(\uparrow)$ for the second time

$L_2(v) = [\uparrow, prev_2(v), c_2(v), d_2(v), bit_2(v)]$,

Form backward label:

$L_1(v) = [\downarrow, prev_1(v), c_1(v), d_1(v), bit_1(v)]$ and can be label $(\downarrow)$ for the second time

$L_2(v) = [\downarrow, prev_2(v), c_2(v), d_2(v), bit_2(v)]$,

Put labeling $(\uparrow)$ for source point and labeling $(\downarrow)$ for sink point:

$$a[\uparrow,\phi,\infty,\infty,1]\ \&\ z[\downarrow,\phi,\infty,\infty,1]$$

The set S comprises the points which have already been labelled $(\uparrow)$ but are not used to label $(\uparrow)$, S' is the point set labelled $(\uparrow)$ based on the points of the set S. Begin $S := \{a\}, S' := \phi$

The set T comprises the points which have already been labelled $(\downarrow)$ but are not used to label $(\downarrow)$, T' is the point set labelled $(\downarrow)$ based on the points of the set T. Begin $T := \{z\}, T' := \phi$

(2) Forward label generate:

(2.1) Choose forward label point:

• Case $S \neq \varnothing$: Choose the point $u \in S$ of a minimum value. Remove the u from the set S, $S := S \setminus \{ u \}$.

Assuming that the forward label of u is $[\uparrow, prev_i(u), c_i(v), d_i(v), bit_i(v)]$, i = 1 or 2. A is the set of the points which are not forward label time and adjacent to the forward label point u. Step (2.2).

• Case $S = \varnothing$ and $S' \neq \varnothing$: Assign $S := S'$, $S' := \varnothing$. ***Step (3)***.

• Case $S = \varnothing$ and $S' = \varnothing$: The flow F is the maximum. ***End.***

(2.2) Forward label the points which are not forward label and are adjacent to the forward label points *u*

• Case $A = \varnothing$: Return to step (2.1).

• Case $A \neq \varnothing$: Choose $t \in A$ of a minimum value. Remove the *t* from the set *A*, $A := A \setminus \{ t \}$. Assign forward labeled point *t*:

If $(u,t) \in E,\ f(u,t) < c_E(u,t), bit_i(u) = 1$ put forward labeled point *t*: $prev_j(t) := u$;

$c_j(t) := \min\{c_i(u),\ c_E(u,t) - f(u,t)\}$, if $d_i(u) = 0$,

$c_j(t) := \min\{c_i(u),\ c_E(u,t) - f(u,t), d_i(u)\}$,

if $d_i(u) > 0$; $d_j(t) := c_V(t) - \sum_{(i,t)\in G} f(i,t)$;

$bit_j(t) := 1$, if $d_j(t) > 0$,

$bit_j(t) := 0$, if $d_j(t) = 0$.

If $(t,u) \in E,\ f(t,u) > 0$ put forward labeled point *t*: $prev_j(t) := u$; $c_j(t) := \min\{c_i(u), f(t,u)\}$,

$d_j(t) := c_V(t) - \sum_{(i,t)\in G} f(i,t)$; $bit_j(t) := 1$.

If *t* is not forward label, then return to step (2.2).

If *t* is forward label and *t* is backward label, then making adjustments in increase of the flow. Step (4).

If *t* is forward label and *t* is not backward label, then add *t* to *S'*, $S' := S' \cup \{ t \}$, and return to step (2.2).

(3) Backward label generate

(3.1) Choose backward label point:

• Case $T \neq \varnothing$: Choose the point $v \in T$ of a minimum value. Remove the *v* from the set *T*, $T := T \setminus \{v\}$.

Assuming that the backward label of *v* is $[\downarrow, prev_i(v), c_i(t), d_i(t), bit_i(t)]$, $i = 1$ or 2. B is the set of the points which are not backward label time and adjacent to the backward label point *v*. Step (3.2).

• Case $T = \varnothing$ and $T' \neq \varnothing$: Assign $T := T'$, $T' := \varnothing$. ***Return to step (2)***.

• Case $T = \varnothing$ and $T' = \varnothing$: The flow F is the maximum. ***End.***

(3.2) Backward label the points which are not backward label and are adjacent to the backward label points *v*

• Case $B = \varnothing$: Return to step (3.1).

• Case $B \neq \varnothing$: Choose $t \in B$ of a minimum value. Remove the *t* from the set B, $B := B \setminus \{t\}$. Assign backward labeled point *t*:

If $(t,v) \in E,\ f(t,v) < c_E(t,v), bit_i(v) = 1$ put backward label point *t*: $prev_j(t) := v$;

$c_j(t) := \min\{c_i(v),\ c_E(t,v) - f(t,v)\}$, if $d_i(v) = 0$,

$c_j(t) := \min\{c_i(v),\ c_E(t,v) - f(t,v), d_i(v)\}$, if $d_i(v) > 0$;

$d_j(t) := c_V(t) - \sum_{(i,t)\in G} f(i,t)$;

$bit_j(t) := 1$, if $d_j(t) > 0$,

$bit_j(t) := 0$, if $d_j(t) = 0$.

If $(v,t) \in E,\ f(v,t) > 0$ put backward label point *t*: $prev_j(t) := v$; $c_j(t) := \min\{c_i(v), f(v,t)\}$,

$d_j(t) := c_V(t) - \sum_{(i,t)\in G} f(i,t)$; $bit_j(t) := 1$.

If *t* is not backward label, then return to step (3.2).

If *t* is backward label and *t* is forward label, then making adjustments in increase of the flow. Step (4).

If *t* is backward label and *t* is not forward label, then add *t* to *T'*, $T' := T' \cup \{t\}$, and return to step (3.2).

(4) Making adjustments in increase of the flow

Suppose *t* is forward label $[\uparrow, prev_i(t), c_i(t), d_i(t), bit_i(t)]$ and *t* is backward label $[\downarrow, prev_i(t), c_i(t), d_i(t), bit_i(t)]$:

(***4.1***) Adjustment made from *t* back to *a* according to forward label

(4.1.1) Start

$$y := t,\ x := prev_1(t),\ \delta := c_1(t).$$

(4.1.2) Making adjustments

(i) Case (x, y) the road section whose direction runs from *x* to *y*: put $f(x,y) := f(x,y) + \delta$.

(ii) Case (y, x) the road section whose direction runs from y to x: put $f(y,x) := f(y,x) - \delta$.

(iii) Case (x, y) non-direction roads:

If $f(x,y) \geq 0$ and $f(y,x) = 0$ then put $f(x,y) := f(x,y) + \delta$.

If $f(y,x) > 0$ then put $f(y,x) := f(y,x) - \delta$.

(4.1.3) Moving

(i) Case $x = a$. Step (4.2).

(ii) Case $x \neq a$, put $y := x$ and $x:=h$, h is the second component of the forward labeled point y. Then return to step (4.1.2).

(***4.2***) Adjustment made from t back to z according to backward label

(4.2.1) Start

$$x := t,\ y := prev_1(t),\ \delta := c_1(t).$$

(4.2.2) Making adjustments

(i) Case (x, y) the road section whose direction runs from x to y: put $f(x, y) := f(x, y) + \delta$.

(ii) Case (y, x) the road section whose direction runs from y to x: put $f(y, x) := f(y, x) - \delta$.

(iii) Case (x, y) non-direction roads:

If $f(x, y) \geq 0$ and $f(y, x) = 0$ then put $f(x, y) := f(x, y) + \delta$.

If $f(y, x) > 0$ then put $f(y, x) := f(y, x) - \delta$.

(4.2.3) Moving

(i) Case $x = z$. Step (4.3).

(ii) Case $x \neq z$, put $x := y$ and $y:=k$, k is the second component of the backward labeled point x. Then return to step (4.2.2).

(***4.3***) Remove all the labels of the network points, except for the source point a and sink point z. Return to step (***2***).

• ***Theorem 2*:** If the value of the route circulation possibility and the circle circulation possibility are integers, then after a limited number of steps, the processing of the maximum network problem will end.

Proof

According to theorem 1, after each time of making adjustment of the flow, the flow will be increased with certain units (due to c_E is a whole number, c_V is a whole number, and δ is, therefore, a positive whole number). On the other hand, the value of the flow is limited above by the total amount of the circulation possibility at roads leaving the source points. So, after a limited number of steps, the processing of the maximum network problem will end.

• ***Theorem 3*:** Given an $F = \{f(x,y) \mid (x,y)\in G\}$ is the flow on extended mixed network G, a source point a and a sink point z:

$$\sum_{(a,x)\in G} f(a,x) = \sum_{(x,z)\in G} f(x,z)$$

Proof

The points of the set V. If x,y is not previous, assign $f(x, y) = 0$

$$\sum_{y\in V}\sum_{x\in V} f(x,y) = \sum_{y\in V}\sum_{x\in V} f(y,x)$$

$$\Leftrightarrow \sum_{y\in V}\left(\sum_{x\in V} f(x,y) - \sum_{x\in V} f(y,x)\right) = 0$$

$$\Leftrightarrow \sum_{y\in V\setminus\{a,z\}}\left(\sum_{x\in V} f(x,y) - \sum_{x\in V} f(y,x)\right) + \left(\sum_{x\in V} f(x,z) - \sum_{x\in V} f(z,x)\right) + \left(\sum_{x\in V} f(x,a) - \sum_{x\in V} f(a,x)\right) = 0$$

$$-\sum_{(a,x)\in G} f(a,x) + \sum_{(x,z)\in G} f(x,z) = 0 \quad \Leftrightarrow \sum_{(a,x)\in G} f(a,x) = \sum_{(x,z)\in G} f(x,z).$$

• ***The complexity of the algorithm:***

It is assumed that the road circulation possibility and the point circulation possibility are whole integer. After each round step, to find the roads to increase the amount of circulation on the flow, we have to approve to pass $|E|$ roads in maximum, and in order to adjust the flow we have to approve to pass $2.|V|$

roads, in maximum. As a result, the complexity of each time of increasing the flow is O(|*E* | + 2.|*V*|). Mark v^* is the value of the maximum flow. The number of times to increase the flow in maximum is v^*. So the complexity of the algorithm is O(v^*(|*E* | + 2.|*V*|)).

5. Result of The Experiment

Given an extended mixed network graph figure 1. The network has six circles, six direction roads and three non-direction roads. The road circulation possibility c_E and the circle circulation possibility c_V. The source point is l, the sink point is 6.

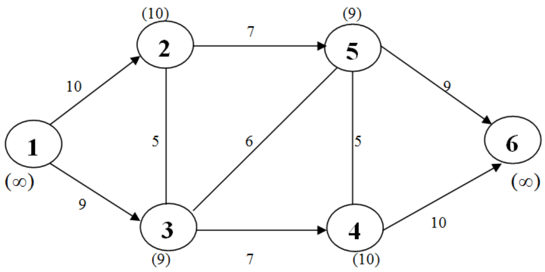

Fig.1: extended mixed network

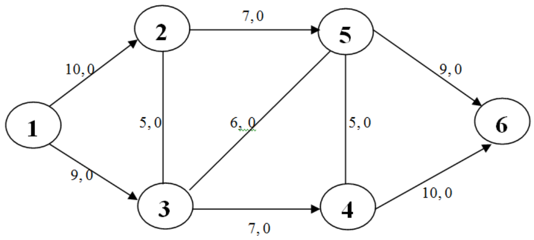

Fig.2: the departure flow 0

+Source point is **l**: [↑, ϕ, ∞, ∞, 1] and sink point is **6**: [↓, ϕ, ∞, ∞, 1]

Point **2**: forward label [↑, 1, 10, 10, 1]

Point **5**: backward label [↓, 6, 9, 9, 1]

Point **3**: forward label [↑, 1, 9, 9, 1] and backward label [↓, 4, 7, 9, 1]

Point **4**: forward label [↑, 3, 7, 10, 1] and backward label [↓, 6, 10, 10, 1]

+ Result of the flow increasing adjustment in figure 3 and the value of the increase *v(F)* = 7

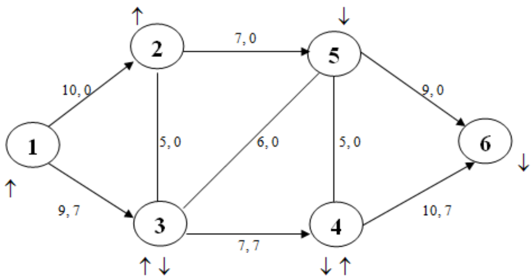

Fig.3: the value of the increase *v(F)* = 7.

+ Analog, result of the flow increasing adjustment in figure 4 and the value of the increase *v(F)* = 14

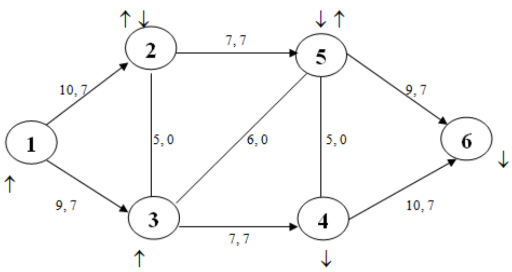

Fig.4: the value of the increase *v(F)* = 14.

+ Result of the flow increasing adjustment in figure 5 and the value of the increase *v(F)* = 16

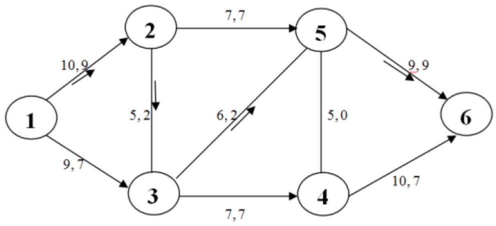

Fig.5: the value of the increase *v(F)* = 16.

This is the maximum flow, because in the following backward label and forward label is not labelled.

6. Conclusion

The article regarding building a model of an extended mixed network so that the stylization of practical problems can be applied more accurately and effectively. Next, improving computing performance for algorithm finding maximal flows on extended mixed networks is being built. Finally, a concrete example is presented to illustrate source–sink alternative algorithm finding maximal flows.

7. References

[1] Tran Quoc Chien, *The problem to find the maximum flow on the network*, the research project on the Ministry level, B2005-16-34,Vietnam, 2005.

[2] Tran Quoc Chien, *Weighted source-sink alternative algorithm*, The university of Danang-Vietnam Journal of science and technology, 3(26)/2008, 99-105.

[3] Tran Quoc Chien, Nguyen Mau Tue, Tran Ngoc Viet, *Shortest path algorithm on extended graphs*, Proceeding of the 6th National Conference on Fundamental and Applied Information Technology Research *(FAIR)*, Vietnam, 2013, 522-527.

[4] Taylor, M. A. P., editor: *Transportation and Traffic theory in the 21st Century,* Pergamon Press Amsterdam, The Netherlands, 2002.

[5] Ellis L. Johnson, Committee Chair, George L.Nemhauser: *Shortest paths and multicommodity network flow*, 2003