A Non-Heuristic Reduction Method For Graph Cut Optimization

F. Malgouyres*, N. Lermé †

January 23, 2014

Abstract

Graph cuts optimization permits to minimize some Markov Random Fields (MRF) by computing a minimum cut (min-cut) in a relevant graph. Graph-cuts are very efficient and are now a well established field of research. However, due to the large amount of memory required for storing the graph, there application remains limited to the minimization of MRF involving a relatively small number of variables. An existing strategy to reduce the graph size restricts the graph construction to a subgraph, called reduced graph, whose nodes satisfy a predefined local condition. The test of the condition is evaluated on the fly during the graph construction. In this manner, the nodes of the reduced graph are typically located in a thin band surrounding the min-cut.

In this paper, we propose a local test similar to the already existing tests for reducing large graphs. The advantage of the proposed condition is that we are able to provide theoretical guarantee that the min-cut in the reduced graph permits to construct a global minimizer of the MRF. This is a significant advantage over the already existing strategies.

Once the theoretical guarantees are established, we present numerical experiments in the context of image segmentation. They confirm that the min-cut in the reduced graph provide a global minimum of the MRF and show empirically that the new test leads to memory gains similar to ones obtained with the existing heuristic test.

Keywords: discrete optimization, graph cuts, segmentation, Markov Random Fields.

AMS classification: 68U10, 94A08, 62M40, 90C27.

*F. Malgouyres works at the IMT, CNRS UMR 5219, Université de Toulouse, 118 route de Narbonne, F-31062 Toulouse Cedex 9, France. E-mail: francois.malgouyres@math.univ-toulouse.fr

†N. Lermé works at the Institut Supérieur d'Électronique de Paris, 21 rue d'Assas, 75006 Paris, France. E-mail: nicolas.lerme@isep.fr

1 Introduction

The introduction of efficient combinatorial optimization tools based on min-cut / max-flow has deeply modified the landscape of computer vision. Indeed, a wide spectrum of ill-posed problems such as segmentation, restoration or dense field estimation are solved by minimizing a MRF involving a large number of variables. These minimization problems can be solved with a moderate empirical complexity using graph cuts. As a consequence, graph cuts have increased the quality and the quantity of low-level analysis tools.

Although graph cuts had a limited impact during one decade, they become more attractive thanks to a fast max-flow algorithm [2] and efficient heuristics for multi-labels problems [3].

In parallel, technological advances have both increased the amount and the diversity of the data to process. Processing and analyzing these data amounts to solve large scale optimization problems. Despite their ability to provide a global minimizer at a small computational cost, graph cuts sometimes fail to solve such problems because of their memory requirement. This is due to the fact that the graphs in which the max-flow is computed usually contains as many nodes as the number of pixels and as many edges as the number of neighboring pixels in the image.

An example of a reduction strategy designed to solve a graph cutting problem (its purpose is not to save memory) is presented in [6]. It contracts the edges passing a given test. However, it requires to build the entire graph and is not applicable in the context we have in mind. The same idea has been also published in [18].

To our best knowledge, this problem seems to be first addressed in the context of image segmentation in [14]. The heuristic strategy adopted in this paper computes a graph cut in a graph built from a pre-segmentation. While this approach greatly increases the performance of standard graph cuts, the results (in particular, the regularity of the boundaries of the segments) depend on the algorithm used for computing the pre-segmentation and better results are obtained when over-segmentation occurs.

Band-based heuristic methods have also been proposed in [15, 19, 7]. A low-resolution version of the image is first segmented. Then, the solution is propagated to the finer resolution level by restricting the construction of the graph to a narrow band surrounding the interpolated foreground/background interface at that resolution. Although this strategy clearly improves the performance of standard graph cuts, it is less accurate to segment thin structures like blood vessels or filaments. Notice that this problem is notably reduced in [19] but still present for non-contrasted details. Also, the multi-scale strategy developed in [15, 19] applies to different MRF than the strategy proposed in the current paper. In [7], smaller graphs are obtained by associating an uncertainty measure to each pixel.

Other non-heuristic methods have been investigated (see [9, 5, 20]). In [9], binary energy functions designed for the shape fitting problem are minimized with graph cuts. The graph is restricted to a narrow band which evolves in order to ensure the optimality on the solution. More precisely, one makes the band evolve around the boundary of the object/segment to delineate by expanding it when the min-cut touches band's boundary. This process is iterated until the band no longer evolves. When it is properly initialized, this method finds a global minimizer.

In [5], a parallel max-flow algorithm yielding a near-linear speedup with the number of processors is presented. While this method achieves good performance on large scale problems, the algorithm is relatively sensitive to the available amount of physical memory and remains less efficient on small graphs.

In [20], the problem is decomposed into small sub-problems. The sub-problems are solved independently. The boundary conditions of the sub-problem are enforced using Lagrange multipliers and guarantee the global optimality of the solution. The Lagrange multipliers are iteratively updated according to the results of the adjacent sub-problems until convergence to a global solution.

Finally, another band-based method was proposed for reducing graphs in binary image segmentation [12], a preliminary version had been published in [11]. The graph is progressively built by only adding the nodes which satisfy a local test. In the manner of previous band-based methods, the graph nodes are typically located in a narrow band surrounding the boundary of the segment/object. This method is able to segment large volumes of data when standard graph cuts fail while providing very accurate or exact results. When the reduction is significant, the time for reducing the graph is compensated by the gain obtained thanks to the non-allocation of the useless nodes and the gain in the computation of the min-cut in a smaller graph. It is empirically observed in [12] that the value of the max-flow in the reduced graph is equal to the value of the max-flow in the original graph. However, there is no formal proof of this statement. This strategy has been applied to an energy model designed for interactive 3D tumor segmentation in Computerized Tomography (CT) images in [13]. An alternative heuristic permitting to reduce graphs whose nodes linked to the source and the sink are intertwined (e.g. when segmenting noisy images) has been proposed in [10].

In this paper, we pursue the work of [12] and propose a local test (similar to the test in [12]) to reduce these graphs by discarding a large amount of nodes during the graph construction. While the cost for evaluating this test is slightly larger compared to the test in [12], we give a formal proof that any node satisfying the new local test can be safely removed without modifying the max-flow value. The strategy therefore preserves the optimality of the solution. The main result of the current paper has already been announced in [16].

The rest of this document is organized as follows. In Section 2, we first define some notations about flows and cuts. We then present, in Section 3, the new local test for reducing the graph as well as the main theorem of this paper. It states that, despite the reduction of the graph, the min-cut in the reduced graph corresponds to a global minimum of the MRF. The proof of this theorem is detailed in the next sections. In Section 6, the theorem is completed with some experiments in the context of image segmentation. They empirically validate the theorem and show that the new test permits to obtain significant memory gains.

2 Framework and notations

We consider a set of pixels $\mathcal{P} \subset \mathbb{Z}^d$, for a positive integer d. We consider two terminal nodes s and t and the set of nodes $\mathcal{V} = \mathcal{P} \cup \{s, t\}$. We also consider a set of directed edges $\mathcal{E} \subset (\mathcal{V} \times \mathcal{V})$ such that $(\mathcal{V}, \mathcal{E})$ is a simple directed graph. We distinguish the edges involving a terminal node, which are called "t-links", and those only involving pixels, which we call "n-links". We also assume that for every $p \in \mathcal{P}$,

$$(p, s) \notin \mathcal{E} \text{ and } (t, p) \notin \mathcal{E}. \tag{1}$$

We denote the neighbors of any node $p \in \mathcal{V}$ by

$$\mathcal{N}_{\mathcal{E}}(p) = \{q \in \mathcal{V}, (p, q) \in \mathcal{E} \text{ or } (q, p) \in \mathcal{E}\}.$$

We define the capacities as a mapping $c : (\mathcal{V} \times \mathcal{V}) \to \mathbb{R}^+$ and denote the capacity of any couple $(p, q) \in (\mathcal{V} \times \mathcal{V})$ by $c_{p,q} \geq 0$. For simplicity, we have defined c for any $(p, q) \in (\mathcal{V} \times \mathcal{V})$, however we always have

$$c_{p,q} = 0, \text{ when } (p, q) \notin \mathcal{E}. \tag{2}$$

We assume, without loss of generality (see [8]), that the capacities are such that for every $p \in \mathcal{P}$

$$c_{s,p} \neq 0 \quad \Rightarrow \quad c_{p,t} = 0. \tag{3}$$

We therefore summarize the capacities of the t-links and set for all $p \in \mathcal{P}$

$$c_p = c_{s,p} - c_{p,t}. \tag{4}$$

For any $S \subset \mathcal{P}$, we denote the value of the s-t cut $(S \cup \{s\}, (\mathcal{P} \setminus S) \cup \{t\})$ in $\mathcal{G}$ by $\mathrm{val}_{\mathcal{G}}(S)$. We remind that

$$\mathrm{val}_{\mathcal{G}}(S) = \sum_{\substack{p \in S \cup \{s\} \\ q \notin S \cup \{s\}}} c_{p,q}.$$

We also define flows as any mapping $f : (\mathcal{V} \times \mathcal{V}) \to \mathbb{R}^+$ satisfying the capacity constraints

$$0 \leq f_{p,q} \leq c_{p,q} \quad , \text{for all } (p, q) \in (\mathcal{V} \times \mathcal{V}), \tag{5}$$

and the flow conservation

$$\sum_{q \in \mathcal{N}_{\mathcal{E}}(p)} f_{q,p} = \sum_{q \in \mathcal{N}_{\mathcal{E}}(p)} f_{p,q} \quad , \text{for all } p \in \mathcal{V}. \tag{6}$$

As usual, the value[1] of the flow f in $\mathcal{G}$ is defined by

$$\mathrm{val}_{\mathcal{G}}(f) = \sum_{p \in \mathcal{N}_{\mathcal{E}}(t)} f_{p,t}. \tag{7}$$

[1] Notice that we use the same notation for the value of a flow and the value of a s-t cut in $\mathcal{G}$. This abuse of notation will never be ambiguous once in context.

As for capacities, we summarize the flow passing through the t-linked and set for all $p \in \mathcal{P}$

$$f_p = f_{s,p} - f_{p,t}. \tag{8}$$

We call max-flow any solution f^* of the linear program

$$\begin{cases} \max_f \quad \mathrm{val}_{\mathcal{G}}\,(f)\,, \\ \text{under the constraints (5) and (6).} \end{cases}$$

As is well known (see [4]), the value of the max-flow is equal to the value of the min-cut:$\mathrm{val}_{\mathcal{G}}\,(f^*) = \min_{S\subset\mathcal{P}} \quad \mathrm{val}_{\mathcal{G}}\,(S)$.

Throughout the paper, we consider a fixed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, c)$, with $\mathcal{V}$, $\mathcal{E}$ and c as above.

Before going ahead, let us remind the reasons motivating the study of these graph problem. When minimizing a pairwise MRF of the form

$$E(u) = \beta \sum_{p\in\mathcal{P}} E_p(u_p) + \sum_{(p,q)\in(\mathcal{P}\times\mathcal{P})} E_{p,q}(u_p, u_q), \qquad \beta \in \mathbb{R}^+, \tag{9}$$

among $u \in \{0,1\}^{\mathcal{P}}$ and when the terms $E_{p,q}(.)$ are submodular, [8] proves that (9) can be globally minimized. The proof provides the construction of a graph $\mathcal{G}$, similar to the graph considered in the current paper, and a constant $K \in \mathbb{R}$ such that for any $S \subset \mathcal{P}$, we have

$$\mathrm{val}_{\mathcal{G}}\,(S) = E(u^S) + K, \tag{10}$$

where $u^S \in \{0,1\}^{\mathcal{P}}$ is defined by

$$u_p^S = \begin{cases} 0 & \text{if} \quad p \notin S \\ 1 & \text{if} \quad p \in S \end{cases}, \qquad \forall p \in \mathcal{P}. \tag{11}$$

Since (11) makes a one to one correspondence between cuts in $\mathcal{G}$ and $\{0,1\}^{\mathcal{P}}$, (10) guarantees that a min-cut in $\mathcal{G}$ corresponds to a global minimizer of (9). The latter can therefore be efficiently computed with a max-flow algorithm such as described in [2]. The graph constructed in [8] and [2] satisfies all the hypotheses made in this paper.

Finally, all along this paper, we also denote $B \subset \mathbb{Z}^d$ and denote

$$B_p = \{p + q, q \in B\}. \tag{12}$$

In practice, we typically think of B_p as a ball centered at p.

We also use, throughout the paper, the convention that the empty sum is null: $\sum_{n\in\emptyset} a_n = 0$, whatever $(a_n)_{n\in\mathbb{N}} \in \mathbb{R}^{\mathbb{N}}$.

3 Main result of the paper

Theorem 1 *Let $\mathcal{G}$ be the graph defined in Section 2, let $B \subset \mathbb{Z}^d$ and $p \in \mathcal{P}$ satisfy*

$$(\mathcal{N}_{\mathcal{E}}(p) \cap \mathcal{P}) \quad \subset \quad B_p, \tag{13}$$

and

$$\begin{cases} \textit{either} & \left(\forall q \in B_p, c_q \geq \sum_{\substack{q' \in \mathcal{N}_\mathcal{E}(q) \\ q' \notin B_p}} c_{q,q'}\right), \\ \textit{or} & \left(\forall q \in B_p, c_q \leq -\sum_{\substack{q' \in \mathcal{N}_\mathcal{E}(q) \\ q' \notin B_p}} c_{q',q}\right). \end{cases} \tag{14}$$

Then, there exists a max-flow f in $\mathcal{G}$ such that

$$\forall q \in \mathcal{N}_\mathcal{E}(p), \qquad f_{p,q} = f_{q,p} = 0. \tag{15}$$

As a consequence, removing the node p from the graph $\mathcal{G}$ does not modify its max-flow value.

Notice first that if the first condition of (14) holds, we have for all $q \in B_p$: $c_q \geq 0$. Indeed, for all $q \in B_p$, $c_q \geq \sum_{\substack{q' \in \mathcal{N}_\mathcal{E}(q) \\ q' \notin B_p}} c_{q,q'} \geq 0$. Similarly, $c_q \leq 0$, for all $q \in B_p$, if the second condition of (14) holds.

Notice also that the test (14) concerns B_p. Said differently, if (14) holds for p it also holds for any $q \in B_p$ such that $(\mathcal{N}_\mathcal{E}(q) \cap \mathcal{P}) \ \subset \ B_p$ (we just translate B). As a consequence, all the nodes in the interior of B_p (i.e. those satisfying $(\mathcal{N}_\mathcal{E}(q) \cap \mathcal{P}) \ \subset \ B_p$) can be removed from the graph.

The proof of this theorem is contained in Section 4 and Section 5. For simplicity, we only prove the theorem when the node p satisfies the first condition of (14).

The statement for the second condition of (14) is easily deduced from the other statement by considering a graph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}', c')$, whose edges are such that

$$\text{for all } (q', q) \in (\mathcal{P} \times \mathcal{P}), \qquad (q, q') \in \mathcal{E}' \Longleftrightarrow (q', q) \in \mathcal{E}$$

and

$$(s, q) \in \mathcal{E}' \Longleftrightarrow (q, t) \in \mathcal{E} \quad \text{and} \quad (q, t) \in \mathcal{E}' \Longleftrightarrow (s, q) \in \mathcal{E};$$

and whose capacities are such that

$$c'_{q,q'} = c_{q',q} \quad , \forall (q, q'); \text{ and } c'_{s,q} = c_{q,t} \text{ and } c'_{q,t} = c_{s,q} \quad , \forall q \in \mathcal{P}.$$

Any pixel $p \in \mathcal{P}$ satisfying the second condition of (14) in $\mathcal{G}$, satisfies the first condition of (14) in $\mathcal{G}'$ and there exists a max-flow f' in $\mathcal{G}'$ such that

$$\forall q \in \mathcal{N}_{\mathcal{E}'}(p), \qquad f'_{p,q} = f'_{q,p} = 0.$$

It is not difficult to check that the mapping $f : \mathcal{V} \times \mathcal{V} \longrightarrow \mathbb{R}$ defined by

$$f_{q',q} = f'_{q,q'} \quad , \forall (q, q') \in \mathcal{P} \times \mathcal{P}, \text{ and } f_{q,t} = f'_{s,q} \text{ and } f_{s,q} = f'_{q,t} \quad , \forall q \in \mathcal{P},$$

is a max-flow in $\mathcal{G}$ and satisfies (15).

In the current paper, we propose to use the Theorem 1 to build a simple algorithm which permits to avoid the construction of useless nodes. The test (14) is computed, with a preset B, on the fly during the graph construction. Although the test is different the rest of the algorithm is similar to the one described in [12] and the reader can find algorithmic details in the latter paper. The above theorem guarantees indeed that, if the node satisfies (14), it is not useful to the max-flow evaluation and can be removed without alteration of the max-flow value. We can compute the max-flow in the subgraph of $\mathcal{G}$ reduced to $\mathcal{V} \setminus \{p\}$ and extend the flow with zeros, using (15), to obtain a max-flow f in $\mathcal{G}$. This permits to deduce the min-cut in the non-reduced graph from the min-cut in the reduced graph.

In practice, if the graph $\mathcal{G}$ is obtained using the construction in [8], we deduce the minimizer $u^* \in \{0,1\}^{\mathcal{P}}$ of (9), from the min-cut

$$\Big(S \cup \{s\}\ ,\ \big((\mathcal{P} \setminus \{p\}) \setminus S\big) \cup \{t\}\Big),$$

of the reduced graph (i.e. $S \subset (\mathcal{P} \setminus \{p\})$ has been computed in the reduced graph) using:

$$u_r^* = \begin{cases} 1 & \text{, if } r = p \text{ and } \forall q \in B_p, c_q \geq \sum_{\substack{q' \in \mathcal{N}_{\mathcal{E}}(q) \\ q' \notin B_p}} c_{q,q'}, \\ 0 & \text{, if } r = p \text{ and } \forall q \in B_p, c_q \leq -\sum_{\substack{q' \in \mathcal{N}_{\mathcal{E}}(q) \\ q' \notin B_p}} c_{q',q}, \\ 1 & \text{, if } r \neq p \text{ and } r \in S, \\ 0 & \text{, if } r \neq p \text{ and } r \notin S. \end{cases} \tag{16}$$

Finally, many different algorithmic strategies exploiting Theorem 1 could be proposed. The one proposed in this paper uses a fixed set B and might be improved in many regards. In particular, Theorem 1 allows to search for large sets B_p satisfying (14). This should lead to better reduction performances. In this regards, it is interesting to note that when (14) holds for two sets B_p and B'_q it necessarily holds for the union of these sets $B_p \cup B'_q$.

4 Avoiding useless traversing flow

Before going ahead, we give few preliminaries.

First, notice that, (2) and (5) guarantee that

$$f_{p,q} = 0, \text{ for any } (p,q) \notin \mathcal{E}. \tag{17}$$

Moreover, as is well known (and can easily be shown by induction on the cardinality of S), for any flow f and any $S \subset \mathcal{V}$ the flow entering S is equal to the flow exiting S:

$$\sum_{\substack{p \in S \\ q \notin S}} f_{q,p} = \sum_{\substack{p \in S \\ q \notin S}} f_{p,q}. \tag{18}$$

Considering (3), (5) and (8), we can rewrite (18) and obtain that

$$\text{for any } S \subset \mathcal{P}, \qquad \sum_{p \in S} f_p + \sum_{\substack{p \in S \\ q \in \mathcal{V} \setminus S}} (f_{q,p} - f_{p,q}) = 0. \tag{19}$$

Also, we denote a walk of positive length $l \in \mathbb{N}$ by $p_0 - p_1 - \ldots - p_l$, where $p_i \in \mathcal{V}$, for all $i \in \{0, \ldots, l\}$, and $(p_i, p_{i+1}) \in \mathcal{E}$, for all $i \in \{0, \ldots, l-1\}$. We also remind that a closed walk is such that $p_0 = p_l$. Moreover, a simple walk is such that $p_i \neq p_j$, for all $(i,j) \in \{0, \ldots, l-1\}^2$ satisfying $i \neq j$ and $p_i \neq p_l$, for all $i \in \{1, \ldots, l-1\}$. We denote by $W(p,q)$ the set containing all the walks starting at $p \in \mathcal{V}$ and ending at $q \in \mathcal{V}$.

Finally, we remind a known result (see [17]). It says that there is a max-flow without circulations.

Proposition 1 *Let $\mathcal{G}$ be the graph defined in Section 2. There exists a max-flow f in $\mathcal{G}$ satisfying*

$$\left\{ \begin{array}{l} \textit{for any positive } l \in \mathbb{N} \textit{ and any closed walk } p_0 - p_1 - \ldots - p_l \\ \textit{of length } l \textit{ in } \mathcal{G}, \textit{there exists } i \in \{0, \ldots, l-1\} \textit{ such that} \\ f_{p_i, p_{i+1}} \leq f_{p_{i+1}, p_i}. \end{array} \right. \tag{20}$$

Throughout the remaining of this section, we consider a graph $\mathcal{G}$ as constructed in Section 2 and a max-flow f in $\mathcal{G}$ satisfying (20). We also consider $p \in \mathcal{P}$ such that

$$\forall q \in B_p, f_q \geq 0,$$

where B_p is defined in (12).

The purpose of the remaining of this section is to establish a sufficient condition so-that f can be modified in such a way that

$$f_{p,q} \geq f_{q,p}, \text{ for all } q \in \mathcal{N}_{\mathcal{E}}(p).$$

In words, the node p sends more flow to its neighbors than it can receive from them.

In order to do so, our strategy consists in modifying the max-flow f in $\mathcal{G}$ in such a way that it satisfies this property. The modification of f is done by constructing a graph $\mathcal{G}'$ and combining f with a max-flow f' in $\mathcal{G}'$. Heuristically, the construction of $\mathcal{G}'$ permits to redirect (and avoid) the flow traversing p in the grid $\mathcal{E} \cap (\mathcal{P} \times \mathcal{P})$ (see Figure 1).

In order to construct $\mathcal{G}'$, we first consider

$$\Sigma_i(p) = \{q \in \mathcal{P}, \exists p_0 - \ldots - p_l \in W(q,p) \text{ such that } \\ \forall i \in \{0, \ldots, l-1\}, f_{p_i, p_{i+1}} > f_{p_{i+1}, p_i}\},$$

and

$$\Sigma_o(p) = \{q \in B_p, \exists p_0 - \ldots - p_l \in W(p,q) \text{ such that } \\ \forall i \in \{0, \ldots, l-1\}, f_{p_i, p_{i+1}} > f_{p_{i+1}, p_i}\}.$$

Let us first notice that, since f satisfies (20),

$$\forall q \in (\Sigma_i(p) \cap \mathcal{N}_{\mathcal{E}}(p)), \quad f_{q,p} \geq f_{p,q} \tag{21}$$

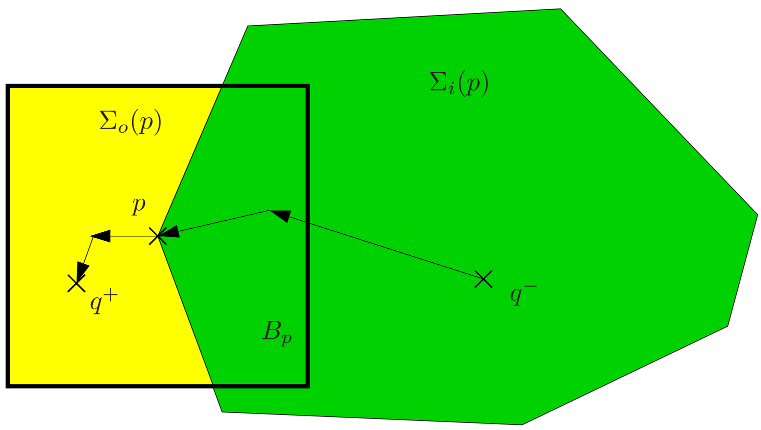


Figure 1: Example of configuration for $\Sigma_o(p)$ (in yellow), $\Sigma_i(p)$ (in green), B_p (the square). The arrows represent the direction of the flow on two walks in $W(q^-, p)$ and $W(p, q^+)$. We know that some flow goes from $q^- \in \Sigma_i(p)$ to $q^+ \in \Sigma_o(p)$ and traverses p. This is the flow we are redirecting in this section. A (too) simplistic solution for avoiding the traversing flow would be to increase f_{q^+} (i.e. f_{s,q^+}), and decrease f_{q^-} (i.e. f_{s,q^-}) in order to reduce the flow on the walk described by the arrows.

and

$$\forall q \in (\Sigma_o(p) \cap \mathcal{N}_{\mathcal{E}}(p)), \quad f_{p,q} \geq f_{q,p}. \tag{22}$$

Similarly, since f satisfies (20), we have

$$\Sigma_i(p) \cap \Sigma_o(p) = \emptyset. \tag{23}$$

Moreover,

$$p \notin \Sigma_i(p) \text{ and } p \notin \Sigma_o(p).$$

For simplicity, we denote

$$\Sigma^- = \Sigma_i(p) \text{ and } \Sigma^+ = \Sigma_o(p) \cup \{p\}.$$

Examples of sets Σ^+ and Σ^- are drawn on Figure 1.

Also, since f satisfies (20), we have

$$\forall q \in \Sigma^-, \forall q' \in \Sigma^+, \qquad f_{q,q'} \geq f_{q',q}. \tag{24}$$

Otherwise, we could easily build a closed walk contradicting (20).

We also denote

$$\mathcal{P}' = \Sigma^- \cup \Sigma^+ \quad , \quad \mathcal{V}' = \mathcal{P}' \cup \{s,t\} \tag{25}$$

and construct the graph

$$\mathcal{G}' = (\mathcal{V}', \mathcal{E}', c'),$$

where $\mathcal{E}'$ and c' are defined below. We set

$$\mathcal{E}' = \mathcal{E}'_t \cup (\mathcal{E}'_n \cap \mathcal{E}^T), \tag{26}$$

where $\mathcal{E}^T = \{(q, q'), \text{ such that } (q', q) \in \mathcal{E}\}$ and with

$$\mathcal{E}'_t = \{(q, t), \text{ with } q \in \Sigma^- \text{ such that } f_q \geq 0\} \bigcup \left(\{s\} \times \Sigma^+\right) \tag{27}$$

and

$$\mathcal{E}'_n = \left(\Sigma^+ \times \Sigma^+\right) \bigcup \left((\Sigma^- \cup \{p\}) \times (\Sigma^- \cup \{p\})\right). \tag{28}$$

The capacities c', on the t-links, are defined by

$$c'_{q,t} = f_q \qquad , \text{ for } q \in \Sigma^- \text{ such that } f_q \geq 0, \tag{29}$$

$$c'_{s,q} = c_q - f_q \qquad , \text{ for } q \in \Sigma^+, \tag{30}$$

and those on the n-links are defined by

$$c'_{q,q'} = \begin{cases} f_{q',q} - f_{q,q'} & , \text{ if } f_{q',q} > f_{q,q'} \\ 0 & , \text{ otherwise} \end{cases} \quad , \text{ for } (q, q') \in (\mathcal{E}'_n \cap \mathcal{E}^T). \tag{31}$$

Notice that there exist some nodes in Σ^- which are linked to no terminals. As in Section 2, we artificially extend all the capacities c' and set

$$c'_{q,q'} = 0, \text{ for all } (q, q') \in ((\mathcal{V}' \times \mathcal{V}') \setminus \mathcal{E}').$$

Notice that, in the graph $\mathcal{G}'$ all the flow sent by s goes in Σ^+ and all the flow arriving at t comes from Σ^-. Moreover, all the edges between Σ^+ and Σ^- contain p. The rest of the section will permit to establish that when p satisfies the conditions of Theorem 1, Σ^+ is a min-cut in $\mathcal{G}'$. We will then be able to use a max-flow in $\mathcal{G}'$ to modify f in such a way that

$$\forall q \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(p), f_{p,q} \geq f_{q,p} = 0.$$

In order to do so, we denote, for any $S \subset \mathcal{P}'$, the value of the s-t cut $(S \cup \{s\}, (\mathcal{P}' \setminus S) \cup \{t\})$ in $\mathcal{G}'$ by

$$\mathrm{val}_{\mathcal{G}'}(S) = \sum_{\substack{q \in (S \cup \{s\}) \\ q' \notin (S \cup \{s\})}} c'_{q,q'}.$$

Using (29), (30) and (31), we find

$$\mathrm{val}_{\mathcal{G}'}(S) = E_1 + E_2 + E_3,$$

where we write

$$E_1 = \sum_{q \in (\Sigma^+ \setminus S)} c'_{s,q} \quad , \quad E_2 = \sum_{q \in (\Sigma^- \cap S)} c'_{q,t} \quad \text{and} \quad E_3 = \sum_{\substack{q \in S \\ q' \in (\mathcal{P}' \setminus S)}} c'_{q,q'}. \tag{32}$$

In particular, using (23) and (25), we have

$$\mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) = \sum_{\substack{q\in\Sigma^+\\ q'\in\Sigma^-}} c'_{q,q'},$$

which, using (28), (23) becomes

$$\mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) = \sum_{q\in\Sigma^-} c'_{p,q}.$$

Finally, we obtain using (31) and (21)

$$\mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) = \sum_{q\in\Sigma^-} (f_{q,p} - f_{p,q}). \tag{33}$$

The following proposition has the most technical proof of the paper but will later provide a condition implying that Σ^+ is a min-cut in $\mathcal{G}'$.

Proposition 2 *Let $\mathcal{G}'$ be the graph constructed in Section 4. For any $S \subset \mathcal{P}'$,*

$$\mathrm{val}_{\mathcal{G}'}(S) \geq \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) + \sum_{q\in\Sigma^+\setminus(S\cup\{p\})} \left[c_q + \sum_{q'\in\mathcal{P}\setminus\Sigma^+} (f_{q',q} - f_{q,q'}) \right]. \tag{34}$$

Proof. Let us first decompose E_3 according to

$$E_3 = E'_1 + E'_2 + E'_3 + E'_4,$$

with

$$E'_1 = \sum_{\substack{q\in(S\cap\Sigma^+)\\ q'\in(\Sigma^+\setminus S)}} c'_{q,q'} \quad , E'_2 = \sum_{\substack{q\in(S\cap\Sigma^+)\\ q'\in(\Sigma^-\setminus S)}} c'_{q,q'}$$

$$E'_3 = \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in(\Sigma^+\setminus S)}} c'_{q,q'} \quad , E'_4 = \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in(\Sigma^-\setminus S)}} c'_{q,q'}$$

We rewrite, using (31),

$$E'_1 = \sum_{\substack{q\in(S\cap\Sigma^+)\\ q'\in(\Sigma^+\setminus S)\\ f_{q',q}>f_{q,q'}}} (f_{q',q} - f_{q,q'}), E'_2 = \sum_{\substack{q\in(S\cap\Sigma^+)\\ q'\in(\Sigma^-\setminus S)\\ (q,q')\in\mathcal{E}', f_{q',q}>f_{q,q'}}} (f_{q',q} - f_{q,q'}) \tag{35}$$

$$E'_3 = \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in(\Sigma^+\setminus S)\\ (q,q')\in\mathcal{E}', f_{q',q}>f_{q,q'}}} c'_{q,q'}, E'_4 = \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in(\Sigma^-\setminus S)\\ f_{q',q}>f_{q,q'}}} (f_{q',q} - f_{q,q'}) \tag{36}$$

Using (28) and (23), then (31) and (21), we immediately find that

$$E'_2 = \begin{cases} \sum_{q\in(\Sigma^-\setminus S)}(f_{q,p} - f_{p,q}) & \text{, if } p \in S \\ 0 & \text{, otherwise,} \end{cases} \qquad \text{and} \qquad E'_3 = 0. \tag{37}$$

Moreover, since the total amount of flow entering and exiting $(S \cap \Sigma^-)$ are equal, we have (see (19))

$$\sum_{\substack{q\in(S\cap\Sigma^-)\\ f_q\geq 0}} f_q + \sum_{\substack{q\in(S\cap\Sigma^-)\\ f_q<0}} f_q + \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\notin(S\cap\Sigma^-)}} (f_{q',q} - f_{q,q'}) = 0$$

Moreover, if we decompose the last term and reorganize the equation we obtain

$$\sum_{\substack{q\in(S\cap\Sigma^-)\\ f_q\geq 0}} f_q + \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in(\Sigma^-\setminus S)}} (f_{q',q} - f_{q,q'}) = -\sum_{\substack{q\in(S\cap\Sigma^-)\\ f_q<0}} f_q - \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in\Sigma^+}} (f_{q',q} - f_{q,q'})$$

Together with the definition of E_2 in (32), (29) and the definition of E'_4 in (36) this leads to

$$\begin{aligned} E_2 + E'_4 &\geq \sum_{\substack{q\in(S\cap\Sigma^-)\\ f_q\geq 0}} f_q + \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in(\Sigma^-\setminus S)}} (f_{q',q} - f_{q,q'}) \\ &= -\sum_{\substack{q\in(S\cap\Sigma^-)\\ f_q<0}} f_q - \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in\Sigma^+}} (f_{q',q} - f_{q,q'}) \\ &\geq \sum_{q\in(S\cap\Sigma^-)} (f_{q,p} - f_{p,q}) + \sum_{\substack{q\in(S\cap\Sigma^-)\\ q'\in(\Sigma^+\setminus\{p\})}} (f_{q,q'} - f_{q',q}). \end{aligned}$$

Then, using (24), we immediately obtain

$$E_2 + E'_4 \geq \sum_{q\in(S\cap\Sigma^-)} (f_{q,p} - f_{p,q}).$$

Together with (37) and (33), this leads to the following intermediate result:

$$E_2 + E'_2 + E'_3 + E'_4 \geq \begin{cases} \mathrm{val}_{\mathcal{G}'}(\Sigma^+) & \text{, if } p \in S \\ \sum_{q\in(S\cap\Sigma^-)}(f_{q,p} - f_{p,q}) & \text{, otherwise.} \end{cases} \tag{38}$$

In order to finish the proof, let us first notice that using the definition of E_1 in (32), (30) and the definition of E'_1 in (35)

$$E_1 + E'_1 \geq \sum_{q\in(\Sigma^+\setminus S)} (c_q - f_q) + \sum_{\substack{q\in(S\cap\Sigma^+)\\ q'\in(\Sigma^+\setminus S)}} (f_{q',q} - f_{q,q'}) \tag{39}$$

Expressing that the total amount of flow entering and exiting $(\Sigma^+ \setminus S)$ are equal, we have (see (19))

$$\sum_{q\in(\Sigma^+\setminus S)} f_q + \sum_{\substack{q\in(\Sigma^+\setminus S)\\ q'\in(\Sigma^+\cap S)}} (f_{q',q} - f_{q,q'}) + \sum_{\substack{q\in(\Sigma^+\setminus S)\\ q'\in\mathcal{P}\setminus\Sigma^+}} (f_{q',q} - f_{q,q'}) = 0.$$

Together with (39), this guarantees that

$$\begin{aligned} E_1 + E_1' &\geq \sum_{q\in(\Sigma^+\setminus S)} c_q + \sum_{\substack{q\in(\Sigma^+\setminus S)\\ q'\in\mathcal{P}\setminus\Sigma^+}} (f_{q',q} - f_{q,q'}), \\ &\geq \sum_{q\in(\Sigma^+\setminus S)} \left[c_q + \sum_{q'\in\mathcal{P}\setminus\Sigma^+} (f_{q',q} - f_{q,q'}) \right] \end{aligned} \tag{40}$$

When $p \in S$, by combining the latter result with (38), we immediately get (34). If $p \notin S$, (40) can be rewritten using (33)

$$E_1 + E_1' \geq \sum_{q\in(\Sigma^+\setminus(S\cup\{p\}))} \left[c_q + \sum_{q'\in\mathcal{P}\setminus\Sigma^+} (f_{q',q} - f_{q,q'}) \right] + c_p + \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right).$$

Since $c_p \geq 0$, and (38) and (21) guarantee that $E_2 + E_2' + E_3' + E_4' \geq 0$, this ensures that (34) holds even when $p \notin S$ and concludes the proof. $\square$

All along the remaining of this Section, we consider a max-flow f' in $\mathcal{G}'$. Notice also that $\mathcal{G}'$ satisfies analogues of (1) and (3). Therefore, as in Section 2, we denote

$$f_q' = f_{s,q}' - f_{q,t}',$$

for all $q \in \mathcal{P}'$. We also artificially extend the flow f' and set

$$f_q' = 0, \text{ for all } q \notin \mathcal{P}'$$

and

$$f_{q,q'}' = 0, \text{ for all } (q,q') \in ((\mathcal{V}' \times \mathcal{V}') \setminus \mathcal{E}').$$

We are now going to combine f and f' in order to build a mapping $f'' : \mathcal{E} \to \mathbb{R}$ which will turn out to be a max-flow in $\mathcal{G}$ such that

$$f_{p,q}'' \geq f_{q,p}'' = 0 \quad ,\forall q \in \mathcal{N}_{\mathcal{E}}(p).$$

Let us begin with the definition of f''. We distinguish in the definition the different possible configuration for edges of $\mathcal{E}$.

$$f_{q,q'}'' = \begin{cases} f_{q,q'} - f_{q',q} & \text{, if } f_{q,q'} \geq f_{q',q} \\ 0 & \text{, otherwise,} \end{cases} \quad \text{, for } (q,q') \notin \mathcal{E}' \tag{41}$$

$$\begin{cases} f''_{s,q} = 0 & \text{and } f''_{q,t} = -f_q & \text{, for } q \in \mathcal{P}' \text{ such that } f_q < 0 \\ f''_{s,q} = f_q + f'_q & \text{and } f''_{q,t} = 0 & \text{, for } q \in \mathcal{P}' \text{ such that } f_q \geq 0 \end{cases} \tag{42}$$

$$f''_{q',q} = \begin{cases} f_{q',q} - f_{q,q'} - f'_{q,q'} & \text{, if } f_{q',q} > f_{q,q'}, \\ 0 & \text{, otherwise} \end{cases} \text{, for } (q',q) \in \mathcal{P}'^2 \cap \mathcal{E}^T. \tag{43}$$

The equations (41), (42) and (43) permit to define $f''_{q,q'}$ for all $(q,q') \in \mathcal{E}$ and (again) we extend f'' outside $\mathcal{E}$ and set

$$f''_{q,q'} = 0, \text{ for all } (q,q') \in ((\mathcal{V} \times \mathcal{V}) \setminus \mathcal{E}).$$

We again summarizes the capacities of the t-links using

$$f''_q = f''_{s,q} - f''_{q,t} \quad , \forall q \in \mathcal{P}.$$

Notice that, since $f'_q = 0$ for all $q \notin \mathcal{P}'$ as well as for $q \in \mathcal{P}'$ such that $f_q < 0$ (see (29) and (30)), we always have, according to (41) and (42),

$$f''_q = f_q + f'_q \quad , \forall q \in \mathcal{P}. \tag{44}$$

Proposition 3 *The mapping $f'' : (\mathcal{V} \times \mathcal{V}) \to \mathbb{R}$ is a max-flow in $\mathcal{G}$.*

Proof. Let us first show that f'' satisfies the capacity constraints. Let $(q',q) \in \mathcal{E}$. We distinguish below the different configuration for (q',q).

- If $(q',q) \notin \mathcal{E}'$ and using (41) we either have

$$0 \leq f''_{q',q} = f_{q',q} - f_{q,q'} \leq c_{q',q},$$

  or

$$0 \leq f''_{q',q} = 0 \leq c_{q',q}.$$

- If $q' \in \Sigma^-$ and $q = s$ or t:
  - If moreover $f_{q'} < 0$, then using (42), $0 \leq f''_{s,q'} = 0 \leq c_{s,q'}$ and $0 \leq f''_{q',t} = f_{q',t} \leq c_{q',t}$.
  - If $f_{q'} \geq 0$, then using (42) and (29), we find that $0 \leq f''_{s,q'} = f_{s,q'} - f'_{q',t} \leq c_{s,q'}$ and $0 \leq f''_{q',t} = 0 \leq c_{q',t}$.
- If $q' \in \Sigma^+$ and $q = s$ or t: since $q' \in B_p$, we necessarily have $f_{q'} \geq 0$, then using (42) and (30), we have $0 \leq f''_{s,q'} = f_{s,q'} + f'_{s,q'} \leq c_{s,q'}$ and $0 \leq f''_{q',t} = 0 \leq c_{q',t}$.
- If $(q',q) \in (\mathcal{P}' \times \mathcal{P}')$:
  - If moreover $f_{q',q} \leq f_{q,q'}$, then (43) guarantees $0 \leq f''_{q',q} = 0 \leq c_{q',q}$.
  - If $f_{q',q} > f_{q,q'}$, using (31), we have

$$0 \leq f'_{q,q'} \leq c'_{q,q'} = f_{q',q} - f_{q,q'},$$

    and finally (43) guarantees that

$$0 \leq f''_{q',q} = f_{q',q} - f_{q,q'} - f'_{q,q'} \leq c_{q',q}.$$

Let us now prove the flow conservation. Let $q \in \mathcal{P}$. We distinguish below the different possible position for q.

- If $q \notin \mathcal{P}'$, then for any $q' \in \mathcal{N}_{\mathcal{E}}(q)$ the definition of $\mathcal{E}'$ given in (26), (27) and (28) guarantees that both (q,q') and $(q',q) \notin \mathcal{E}'$. Using (41), we obtain $f''_{q,q'} - f''_{q',q} = f_{q,q'} - f_{q',q}$, for all $q' \in \mathcal{N}_{\mathcal{E}}(q)$, and therefore

$$\sum_{q' \in \mathcal{N}_{\mathcal{E}}(q)} \left(f''_{q',q} - f''_{q,q'}\right) = \sum_{q' \in \mathcal{N}_{\mathcal{E}}(q)} \left(f_{q',q} - f_{q,q'}\right) = 0.$$

- If $q \in \mathcal{P}'$, the flow conservation constraint given by (19) for f and f' at q can be decomposed to provide

$$f_q + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q) \\ q' \notin \mathcal{N}_{\mathcal{E}'}(q)}} \left(f_{q',q} - f_{q,q'}\right) + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}'}(q) \\ f_{q',q} > f_{q,q'}}} \left(f_{q',q} - f_{q,q'}\right) + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}'}(q) \\ f_{q',q} \leq f_{q,q'}}} \left(f_{q',q} - f_{q,q'}\right) = 0$$

  and

$$f'_q + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}'}(q) \\ f_{q',q} > f_{q,q'}}} \left(0 - f'_{q,q'}\right) + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}'}(q) \\ f_{q',q} \leq f_{q,q'}}} \left(f'_{q',q} - 0\right) = 0.$$

  Summing these equalities and using (44), (41) and (43), we obtain

$$f''_q + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q) \\ q' \notin \mathcal{N}_{\mathcal{E}'}(q)}} \left(f''_{q',q} - f''_{q,q'}\right) + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}'}(q) \\ f_{q',q} > f_{q,q'}}} \left(f''_{q',q} - f''_{q,q'}\right) + \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}'}(q) \\ f_{q',q} \leq f_{q,q'}}} \left(f''_{q',q} - f''_{q,q'}\right) = 0.$$

  The latter corresponds to flow conservation constraint (19) at the node q for f''.

Altogether, we now know that f'' is a flow. We still need to show that it is a max-flow. The latter property is in fact trivially obtained since (42) and (41) guarantee that $f''_{q,t} = f_{q,t}$, for all $q \in \mathcal{P}$. Therefore, the value of f'' is equal to the value of f. Since f is a max-flow, this value is maximal and f'' is a max-flow. □

Proposition 4 *If Σ^+ is a min-cut in the graph $\mathcal{G}'$ defined in Section 4, then the max-flow f'' is such that*

$$\forall q \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(p), \qquad f''_{q,p} = 0.$$

As a consequence,

$$\forall q \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(p), \qquad f''_{p,q} \geq f''_{q,p}.$$

Proof. Since f' is a max-flow in $\mathcal{G}'$ and Σ^+ is a min-cut in $\mathcal{G}'$, Ford-Fulkerson theorem guarantees that they have the same value. We therefore have

$$\begin{aligned} \mathrm{val}_{\mathcal{G}'}\left(f'\right) = \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) &= \sum_{\substack{q' \in \Sigma^+ \\ q \notin \Sigma^+ \\ (q',q) \in \mathcal{E}'}} c'_{q',q} \\ &= \sum_{q \in \Sigma^-} c'_{p,q} \end{aligned} \tag{45}$$

Moreover, since f' is a flow, the total amount of flow entering and exiting Σ^+ are equal. Therefore, we have (see (19))

$$\sum_{q \in \Sigma^+} f'_q + \sum_{\substack{q' \in \Sigma^+ \\ q \notin \Sigma^+ \\ q \in \mathcal{N}_{\mathcal{E}'}(q')}} (f'_{q,q'} - f'_{q',q}) = 0.$$

Together with (7), (27) and (28), this guarantees that

$$\mathrm{val}_{\mathcal{G}'}\left(f'\right) = \sum_{\substack{q \in \Sigma^- \\ f_q \geq 0}} f'_q = \sum_{q \in \Sigma^-} (f'_{p,q} - f'_{q,p}),$$

Since the amounts of flow entering and exiting Σ^- are equal. Combined with (45), this provides

$$\sum_{q \in \Sigma^-} c'_{p,q} = \sum_{q \in \Sigma^-} f'_{p,q} - \sum_{q \in \Sigma^-} f'_{q,p}. \tag{46}$$

As a consequence,

$$\sum_{q \in \Sigma^-} f'_{q,p} = \sum_{q \in \Sigma^-} (f'_{p,q} - c'_{p,q}) \leq 0.$$

However, since for all $q \in \Sigma^-$, $f'_{q,p} \geq 0$, we finally obtain that

$$\forall q \in \Sigma^-, f'_{q,p} = 0.$$

Using (46) again, (21) and (31), this provides

$$\forall q \in \Sigma^-, f'_{p,q} = c'_{p,q} = f_{q,p} - f_{p,q}.$$

Therefore, using (21) and (43),

$$\forall q \in \Sigma^-, f''_{q,p} = 0. \tag{47}$$

Moreover, using (22) and (43), we also have

$$\forall q \in (\Sigma^+ \cap \mathcal{N}_{\mathcal{E}}(p)), f''_{q,p} = 0. \tag{48}$$

Combining this result with (47), we obtain

$$\forall q \in \left(\Sigma^+ \cup \Sigma^-\right) \cap \mathcal{N}_{\mathcal{E}}(p), \quad f''_{q,p} = 0. \tag{49}$$

Now, if $q \in \mathcal{N}_{\mathcal{E}}(p) \setminus (\Sigma^+ \cup \Sigma^-)$, the definitions of Σ^+ and Σ^- imply that necessarily $f_{p,q} = f_{q,p}$. The definition of $\mathcal{E}'$ also guarantees that (p,q) and $(q,p) \notin \mathcal{E}'$. Together, with (41), we finally obtain that

$$\forall q \in \mathcal{N}_{\mathcal{E}}(p) \setminus \left(\Sigma^+ \cup \Sigma^-\right), \quad f''_{p,q} = f''_{q,p} = 0.$$

Together with (49), this concludes the proof. □

Proposition 5 *Let $\mathcal{G}$ be the graph defined in Section 2, let B satisfy* (13) *and let us assume that $p \in \mathcal{P}$ is such that*

$$\forall q \in B_p, \quad c_q \geq \sum_{\substack{q' \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q) \\ q' \in \mathcal{P} \setminus B_p}} c_{q,q'}, \tag{50}$$

then, there exists a max-flow f in $\mathcal{G}$ such that

$$\forall q \in \mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(p), f_{p,q} \geq f_{q,p} = 0. \tag{51}$$

Proof. This is a straightforward consequence of Proposition 3, Proposition 2 and Proposition 4.

Indeed, if (50) holds, we know that for any max-flow f in $\mathcal{G}$ as in Proposition 1 and any $S \subset \mathcal{P}'$

$$\sum_{q \in \Sigma^+ \setminus (S \cup \{p\})} \left[c_q + \sum_{q' \in \mathcal{P} \setminus \Sigma^+} (f_{q',q} - f_{q,q'}) \right] \geq$$
$$\sum_{q \in \Sigma^+ \setminus (S \cup \{p\})} \left[c_q + \sum_{q' \in \mathcal{P} \setminus B_p} (f_{q',q} - f_{q,q'}) \right] \geq 0,$$

since for all $q' \in B_p \setminus \Sigma^+$, $f_{q',q} - f_{q,q'} \geq 0$. Therefore, Proposition 2 guarantees that Σ^+ is a min-cut in $\mathcal{G}'$. Then, Proposition 3 guarantees that f'' is a max-flow in $\mathcal{G}$ and Proposition 4 guarantees that f'' satisfies (51). □

5 A useless nodes

Throughout this section, we consider a graph $\mathcal{G}$ as constructed in Section 2, a set B satisfying (13), a pixel $p \in \mathcal{P}$ satisfying (50) and a max-flow f in $\mathcal{G}$ satisfying (51).

The purpose of this section is to modify f so-that it remains a max-flow in $\mathcal{G}$ and satisfies

$$\forall q \in \mathcal{N}_{\mathcal{E}}(p), \quad f_{p,q} = f_{q,p} = 0.$$

The latter obviously implies that the node p is useless when computing the max-flow in $\mathcal{G}$.

Notice that, since the flow f satisfies (51), the only flow entering p comes from the source s. Therefore, in this section, we want to decrease f_s (i.e. $f_{s,p}$). However, since the flow f_s entering p contributes to the flow exiting B_p, we need to compensate f_s by increasing f_q (i.e. $f_{s,q}$), for $q \in B_p$. Similarly to Section 4, this intuitive (and too simplistic) strategy is strengthened by considering a max-flow f' is an appropriate graph $\mathcal{G}'$.

Since the method for modifying f is analogous to the one used in Section 4, we chose to use the same notations for the objects playing the same role. Beware not to confuse their definition.

First, we denote

$$\mathcal{P}' = B_p \quad , \Sigma^+ = B_p \setminus \{p\} \quad \text{and} \quad \Sigma^- = \{p\}. \tag{52}$$

In order to modify f, we build a graph $\mathcal{G}' = (\mathcal{P}', \mathcal{E}', c')$ where $\mathcal{E}'$and c' are defined below. We consider

$$\begin{cases} & \mathcal{E}' = \mathcal{E}'_t \cup \mathcal{E}'_n \\ \text{where} & \mathcal{E}'_t = (\{s\} \times \Sigma^+) \quad \bigcup \quad \{(p,t)\} \\ \text{and} & \mathcal{E}'_n = (\mathcal{E} \cap (\Sigma^+ \times \Sigma^+)) \quad \bigcup \quad ((\mathcal{N}_{\mathcal{E}}(p) \cap \Sigma^+) \times \Sigma^-)\,. \end{cases} \tag{53}$$

We define the capacities c' by

$$c'_{q,q'} = c_{q,q'} - f_{q,q'} + f_{q',q}, \forall (q,q') \in \left(\mathcal{E} \cap \left(\Sigma^+ \times \Sigma^+\right)\right) \tag{54}$$

$$c'_{q,p} = f_{p,q} \qquad , \forall q \in (\mathcal{N}_{\mathcal{E}}(p) \cap \Sigma^+) \tag{55}$$

$$c'_{s,q} = c_q - f_q \qquad , \forall q \in \Sigma^+ \tag{56}$$

$$c'_{p,t} = f_p \tag{57}$$

As usual, in order to simplify the notations, we artificially set

$$c'_{q,q'} = 0 \quad , \forall (q,q') \in (\mathcal{P}' \times \mathcal{P}') \setminus \mathcal{E}', \tag{58}$$

and we write

$$c'_q = c'_{s,q} - c'_{q,t} \quad , \forall q \in \mathcal{P}'. \tag{59}$$

Notice first that, for any $S \subset \mathcal{P}'$, the value of the s-t cut $((S \cup \{s\}), (\mathcal{P}' \setminus S) \cup \{t\})$ in $\mathcal{G}'$ depends on whether $p \in S$ or $p \notin S$. If $p \in S$, we have

$$\mathrm{val}_{\mathcal{G}'}(S) = c'_{p,t} + \sum_{q \in (\Sigma^+ \setminus S)} c'_q + \sum_{\substack{q \in S \\ q' \in (\mathcal{P}' \setminus S)}} c'_{q,q'}.$$

Therefore, we trivially have using (54)-(59)

$$\mathrm{val}_{\mathcal{G}'}(S) \geq c'_{p,t} = f_p \quad , \text{if } p \in S. \tag{60}$$

Moreover, for any $S \subset \mathcal{P}'$, the value of the s-t cut $((S \cup \{s\}), (\mathcal{P}' \setminus S) \cup \{t\})$ in $\mathcal{G}'$ is given by

$$\mathrm{val}_{\mathcal{G}'}(S) = \sum_{q \in (\Sigma^+ \setminus S)} c'_q + \sum_{\substack{q \in S \\ q' \in (\mathcal{P}' \setminus S)}} c'_{q,q'} \quad , \text{ if } p \notin S. \tag{61}$$

In particular, if $S = \Sigma^+$, we obtain using (55), the conservation of the flow f at p and (51) that

$$\begin{aligned} \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) &= \sum_{q \in (\Sigma^+ \cap \mathcal{N}_{\mathcal{E}'}(p))} c'_{q,p}, \\ &= \sum_{q \in (\Sigma^+ \cap \mathcal{N}_{\mathcal{E}}(p))} f_{p,q}, \\ &= f_p \ . \end{aligned} \tag{62}$$

The following proposition will later give a sufficient condition for Σ^+ to be a min-cut in $\mathcal{G}'$.

Proposition 6 *Let $\mathcal{G}'$ be the graph constructed in Section 5. For any $S \subset \mathcal{P}'$,*

- *if $p \notin S$*

$$\mathrm{val}_{\mathcal{G}'}(S) = \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) + \sum_{\substack{q \in S \\ q' \in (\Sigma^+ \setminus S)}} c_{q,q'} + \sum_{q \in (\Sigma^+ \setminus S)} \left[c_q + \sum_{q' \in \mathcal{P} \setminus \mathcal{P}'} (f_{q',q} - f_{q,q'}) \right], \tag{63}$$

- *if $p \in S$*

$$\mathrm{val}_{\mathcal{G}'}(S) \geq \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right). \tag{64}$$

Proof. Notice first that, if $p \in S$, (64) is a straightforward consequence of (60) and (62). Let us assume from now on that $p \notin S$.

Since f is a flow, the total amount of flow entering and exiting $(\mathcal{P}' \setminus S)$ are equal (see (19)) and therefore, using (52)

$$f_p + \sum_{q \in (\Sigma^+ \setminus S)} f_q + \sum_{\substack{q \in (\mathcal{P}' \setminus S) \\ q' \in \mathcal{P} \setminus (\mathcal{P}' \setminus S)}} (f_{q',q} - f_{q,q'}) = 0.$$

Using (62), (56) and (59), we obtain

$$\mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) + \sum_{q \in (\Sigma^+ \setminus S)} (c_q - c'_q) + \sum_{\substack{q \in (\mathcal{P}' \setminus S) \\ q' \in \mathcal{P} \setminus (\mathcal{P}' \setminus S)}} (f_{q',q} - f_{q,q'}) = 0.$$

Combined with (61), this becomes

$$\mathrm{val}_{\mathcal{G}'}(S) = \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) + \sum_{q \in (\Sigma^+ \setminus S)} c_q + \sum_{\substack{q \in (\mathcal{P}' \setminus S) \\ q' \in \mathcal{P} \setminus (\mathcal{P}' \setminus S)}} (f_{q',q} - f_{q,q'}) + \sum_{\substack{q \in S \\ q' \in (\mathcal{P}' \setminus S)}} c'_{q,q'}. \tag{65}$$

We now decompose the last term of the above equation using (54), (55) and (51) and write

$$
\begin{aligned}
\sum_{\substack{q\in S\\ q'\in(\mathcal{P}'\setminus S)}} c'_{q,q'} &= \sum_{\substack{q\in S\\ q'\in(\Sigma^+\setminus S)}} (c_{q,q'} - f_{q,q'} + f_{q',q}) + \sum_{q\in S} f_{p,q},\\
&= \sum_{\substack{q\in S\\ q'\in(\Sigma^+\setminus S)}} c_{q,q'} - \sum_{\substack{q'\in S\\ q\in(\Sigma^+\setminus S)}} (f_{q',q} - f_{q,q'}) + \sum_{q'\in S} (f_{p,q'} - f_{q',p}),\\
&= \sum_{\substack{q\in S\\ q'\in(\Sigma^+\setminus S)}} c_{q,q'} - \sum_{\substack{q\in(\mathcal{P}'\setminus S)\\ q'\in S}} (f_{q',q} - f_{q,q'}).
\end{aligned}
$$

Combining the latter with (65), we finally obtain

$$
\mathrm{val}_{\mathcal{G}'}(S) = \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) + \sum_{q\in(\Sigma^+\setminus S)} c_q + \sum_{\substack{q\in S\\ q'\in(\Sigma^+\setminus S)}} c_{q,q'} + \sum_{\substack{q\in(\mathcal{P}'\setminus S)\\ q'\in\mathcal{P}\setminus\mathcal{P}'}} (f_{q',q} - f_{q,q'}).
$$

Using (13), we remark that for any $q' \notin \mathcal{P}\setminus\mathcal{P}'$, we have $q' \notin \mathcal{N}_{\mathcal{E}}(p)$ and we can finally deduce that (63) holds for all $S \subset \mathcal{P}'$ such that $p \notin S$. $\square$

As in Section 4, we will from now on consider a max-flow f' in the graph $\mathcal{G}'$ built in the current section. We also artificially extend the flow f' and set

$$
f'_{q,q'} = 0, \text{ for all } (q,q') \in \left((\mathcal{V}' \times \mathcal{V}') \setminus \mathcal{E}'\right). \tag{66}
$$

Once again, the graph $\mathcal{G}'$ satisfies analogues of (1) and (3), therefore, as usual, we denote for simplicity

$$
f'_q = f'_{s,q} - f'_{q,t} \quad ,\forall q \in \mathcal{P}'. \tag{67}
$$

We are now going to combine f and f' in order to build a mapping $f'' : \mathcal{E} \to \mathbb{R}$ which will turn out to be a max-flow in $\mathcal{G}$ such that

$$
f''_{p,q} = f''_{q,p} = 0 \quad ,\forall q \in \mathcal{N}_{\mathcal{E}}(p).
$$

As for $\mathcal{G}'$ and f', beware that the mapping f'' is different in Section 4 and in the current section.

Let us begin with the definition of f''. We distinguish below the different possible configurations for the elements of $\mathcal{E}$.

$$
\begin{aligned}
f''_q &= f_q && \forall q \notin \mathcal{P}', && (68)\\
f''_{q,q'} &= f_{q,q'} && \forall (q,q') \in \mathcal{E},\ \text{with } q \notin \mathcal{P}' \text{ or } q' \notin \mathcal{P}' && (69)\\
f''_q &= f_q + f'_q && \forall q \in \mathcal{P}', && (70)
\end{aligned}
$$

and for $(q,q') \in \left(\mathcal{E} \cap (\Sigma^+)^2\right)$

$$f''_{q,q'} = \begin{cases} (f_{q,q'} + f'_{q,q'}) - (f_{q',q} + f'_{q',q}) & \text{if } f_{q,q'} + f'_{q,q'} \geq f_{q',q} + f'_{q',q} \\ 0 & \text{if } f_{q,q'} + f'_{q,q'} < f_{q',q} + f'_{q',q} \end{cases} \tag{71}$$

and

$$f''_{p,q} = f_{p,q} - f'_{q,p} \qquad \forall q \in (\mathcal{P}' \cap \mathcal{N}_{\mathcal{E}}(p)) \tag{72}$$

$$f''_{q,p} = 0 \qquad \forall q \in (\mathcal{P}' \cap \mathcal{N}_{\mathcal{E}}(p)) \tag{73}$$

We also define

$$f''_{s,q} = \max(f''_q, 0) \quad \text{and} \quad f''_{q,t} = \max(-f''_q, 0) \quad , \forall q \in \mathcal{P}. \tag{74}$$

Notice that the equation (68)-(74) permit to define $f''_{q,q'}$ for all $(q,q') \in \mathcal{E}$. Once again, we extend f'' outside $\mathcal{E}$ and set

$$f''_{q,q'} = 0, \text{ for all } (q,q') \in ((\mathcal{V} \times \mathcal{V}) \setminus \mathcal{E}).$$

The following proposition holds.

Proposition 7 *The mapping $f'' : (\mathcal{V} \times \mathcal{V}) \to \mathbb{R}$ is max-flow in $\mathcal{G}$.*

Proof. Notice first that, if f'' is a flow in $\mathcal{G}$ it is necessarily a max-flow since, according to (50), $(\mathcal{N}_{\mathcal{E}}(t) \cap \mathcal{P}') = \emptyset$ and therefore, using (68), we always have $f''_{q,t} = f_{q,t}$, for all $q \in \mathcal{N}_{\mathcal{E}}(t)$. Therefore, we have $\mathrm{val}_{\mathcal{G}}(f'') = \mathrm{val}_{\mathcal{G}}(f)$ and, if f'' is a flow in $\mathcal{G}$, f'' is necessarily a max-flow in $\mathcal{G}$.

In order to show that f'' is a flow we first show that it satisfies the capacity constraints. Let $(q,q') \in \mathcal{E}$. We distinguish below the different possible configurations for the elements of (q,q').

- If $q = s$ and $q' \notin B_p$ or if $q \notin B_p$ and $q' = t$, using (68) and (74), we know that
$$0 \leq f''_{q,q'} = f_{q,q'} \leq c_{q,q'} \quad \text{and} \quad 0 \leq f''_{q',q} = f_{q',q} \leq c_{q',q}.$$
- If $q \notin B_p$ or $q' \notin B_p$, using (69), we obtain again
$$0 \leq f''_{q,q'} = f_{q,q'} \leq c_{q,q'}.$$
- If $q = s$ and $q' \in \Sigma^+$, using (70) and (56), we get
$$0 \leq f''_{q,q'} = f_{s,q'} + f'_{s,q'} \leq c_{q,q'}.$$
- If $q = s$ and $q' = p$, using (70) and (57), we get
$$0 \leq f''_{q,q'} = f_{s,p} - f'_{p,t} \leq c_{q,q'}.$$

- If $(q,q') \in (\Sigma^+)^2$ and $f_{q,q'} + f'_{q,q'} \geq f_{q',q} + f'_{q',q}$, using (71) and (54), we obtain

$$0 \leq f''_{q,q'} = f_{q,q'} + f'_{q,q'} - f_{q',q} - f'_{q',q} \leq c_{q,q'} - f'_{q',q} \leq c_{q,q'}.$$

- If $(q,q') \in (\Sigma^+)^2$ and $f_{q,q'} + f'_{q,q'} < f_{q',q} + f'_{q',q}$, using (71), we trivially have

$$0 \leq f''_{q,q'} = 0 \leq c_{q,q'}.$$

- If $q = p$ and $q' \in (B_p \cap \mathcal{N}_{\mathcal{E}}(p))$, using (72) and (55), we get

$$0 \leq f''_{q,q'} = f_{p,q'} - f'_{q',p} \leq c_{q,q'}.$$

- If $q \in (B_p \cap \mathcal{N}_{\mathcal{E}}(p))$ and $q' = p$, then (73) trivially guarantees that

$$0 \leq f''_{q,q'} = 0 \leq c_{q,q'}.$$

In order to show the flow conservation constraints, we consider, from now on, $q \in \mathcal{P}$. We distinguish below the different possible position for q.

- If $q \in \mathcal{P} \setminus \mathcal{P}'$, we have, using (68) and (69), we have $f''_{q,q'} = f_{q,q'}$ and $f''_{q',q} = f_{q',q}$, for all $q' \in \mathcal{N}_{\mathcal{E}}(q)$. Therefore,

$$\sum_{q' \in \mathcal{N}_{\mathcal{E}}(q)} f''_{q',q} = \sum_{q' \in \mathcal{N}_{\mathcal{E}}(q)} f_{q',q} = \sum_{q' \in \mathcal{N}_{\mathcal{E}}(q)} f_{q,q'} = \sum_{q' \in \mathcal{N}_{\mathcal{E}}(q)} f''_{q,q'}.$$

- If $q \in \Sigma^+$, expressing that the two flows f and f' are conserved at q, we obtain using (17) and (51)

$$f_q + \sum_{\substack{q' \in (\mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q)) \\ q' \in \mathcal{P} \setminus \mathcal{P}'}} (f_{q',q} - f_{q,q'}) + \sum_{\substack{q' \in (\mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q)) \\ q' \in \Sigma^+}} (f_{q',q} - f_{q,q'}) + f_{p,q} = 0$$

  and

$$f'_q + \sum_{\substack{q' \in (\mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q)) \\ q' \in \Sigma^+}} (f'_{q',q} - f'_{q,q'}) \quad - f'_{q,p} = 0.$$

  Summing those inequalities and using (69)-(73), we obtain

$$f''_q + \sum_{\substack{q' \in (\mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q)) \\ q' \in \mathcal{P} \setminus \mathcal{P}'}} (f''_{q',q} - f''_{q,q'}) + \sum_{\substack{q' \in (\mathcal{P} \cap \mathcal{N}_{\mathcal{E}}(q)) \\ q' \in \Sigma^+}} (f''_{q',q} - f''_{q,q'}) + (f''_{p,q} - f''_{q,p}) = 0.$$

  The latter expresses that f'' is conserved at the node q.

- If $q = p$, then using (70), (72) and (73) as well as (13) and (57), we obtain

$$\sum_{q' \in \mathcal{N}_{\mathcal{E}}(p)} (f''_{q',p} - f''_{p,q'}) = f_{s,p} - f'_{p,t} - \sum_{q' \in (\mathcal{P}' \cap \mathcal{N}_{\mathcal{E}}(p))} (f_{p,q'} - f'_{q',p}).$$

Using that $f_{p,t} = 0$ (see (50), (4) and (3)), $f'_{s,p} = 0$ (see (53) and (58)), $f_{q',p} = 0$ (see (51)) and $f'_{p,q'} = 0$ (see (53) and (58)), we obtain

$$\sum_{q' \in \mathcal{N}_{\mathcal{E}}(p)} (f''_{q',p} - f''_{p,q'}) = (f_{s,p} - f_{p,t}) + (f'_{s,p} - f'_{p,t}) \\ - \sum_{q' \in (\mathcal{P}' \cap \mathcal{N}_{\mathcal{E}}(p))} \left[(f_{p,q'} - f_{q',p}) + (f'_{p,q'} - f'_{q',p}) \right]. \quad (75)$$

Simplifying, we finally obtain

$$\sum_{q' \in \mathcal{N}_{\mathcal{E}}(p)} (f''_{q',p} - f''_{p,q'}) = \sum_{q' \in \mathcal{N}_{\mathcal{E}}(p)} (f_{q',p} - f_{p,q'}) + \sum_{q' \in \mathcal{N}_{\mathcal{E}}(p)} (f'_{q',p} - f'_{p,q'}), \\ = 0,$$

since the two flows f and f' are conserved at p.

This concludes the proof. $\square$

Proposition 8 *If Σ^+ is a min-cut in the graph $\mathcal{G}'$ defined in Section 5, then the max-flow f'' is such that*

$$\forall q \in \mathcal{N}_{\mathcal{E}}(p), \qquad f''_{q,p} = f''_{p,q} = 0.$$

As a consequence, removing the node p from the graph $\mathcal{G}$ does not modify its maximal flow value.

Proof. If Σ^+ is a min-cut in the graph $\mathcal{G}'$ defined in Section 5, then Ford-Fulkerson theorem, (62) and (53) guarantee that

$$f_p = \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right) = \mathrm{val}_{\mathcal{G}'}\left(f'\right) = f'_{p,t}.$$

Using (70), (67) and (66) this yields

$$f''_p = f_p - f'_{p,t} = 0,$$

which, using (74), provides

$$f''_{s,p} = f''_{p,t} = 0.$$

Together with (73), this guarantees that

$$\text{for all } q \in \mathcal{N}_{\mathcal{E}}(p), \quad f''_{q,p} = 0. \quad (76)$$

Expressing the flow conservation constraint at p for f'', we deduce from (76) that

$$\sum_{q\in\mathcal{N}_{\mathcal{E}}(p)} f''_{p,q} = \sum_{q\in\mathcal{N}_{\mathcal{E}}(p)} f''_{q,p} = 0,$$

which guarantees that

$$\text{for all } q \in \mathcal{N}_{\mathcal{E}}(p), \quad f''_{p,q} = 0,$$

since $f''_{p,q} \geq 0$, for all $q \in \mathcal{N}_{\mathcal{E}}(p)$.

Together with (76), this concludes the proof. □

We can now conclude with the following proposition.

Proposition 9 *Let $\mathcal{G}$ be the graph defined in Section 2, let $B \subset \mathcal{P}$ and $p \in \mathcal{P}$ satisfy* (13) *and* (50)*. Then, there exists a max-flow f in $\mathcal{G}$ such that*

$$\forall q \in \mathcal{N}_{\mathcal{E}}(p), \qquad f_{p,q} = f_{q,p} = 0. \tag{77}$$

As a consequence, removing the node p from the graph $\mathcal{G}$ does not modify its maximal flow value.

Proof. This is a straightforward consequence of Proposition 5, Proposition 6, Proposition 7 and Proposition 8.

Indeed, if (50) holds, we know that there is max-flow f in $\mathcal{G}$ satisfying (51). Therefore, using the notations of Section 5, we know that for any $S \subset \mathcal{P}'$ such that $p \notin S$

$$\sum_{q\in\Sigma^+\setminus S} \left[c_q + \sum_{q'\in\mathcal{P}\setminus\mathcal{P}'} (f_{q',q} - f_{q,q'}) \right] \geq 0.$$

Therefore, for $\mathcal{G}'$ as defined in Section 5, Proposition 6 guarantees that for any $S \subset \mathcal{P}'$

$$\mathrm{val}_{\mathcal{G}'}(S) \geq \mathrm{val}_{\mathcal{G}'}\left(\Sigma^+\right),$$

and therefore Σ^+ is a min-cut in $\mathcal{G}'$. Then, Proposition 7 guarantees that f'' is a max-flow in $\mathcal{G}$ and Proposition 8 guarantees that f'' satisfies (77). □

6 Numerical experiments

6.1 Experimental Framework

In this section, we consider a simple algorithm exploiting Theorem 1. For a fixed neighborhood $B = \{-1, 0, 1\}^2$, it consists in testing during the graph construction whether a node satisfies (14) or not. If the node satisfies the test, the result at the corresponding pixel is assigned according to (16); if the node does not satisfy the test, it is constructed and added to the graph that will be sent to a max-flow algorithm. Of course, more sophisticated algorithms could be investigated. The aim of the proposed experiments is to illustrate when such a simple application of Theorem 1 actually permits to reduce the graph size.

We will also compare the ability of this simple algorithm to a similar algorithm which uses the test proposed in [12] (see below). More precisely, we compare the ability of these algorithms to yield to small reduced graph. The size of the reduced graph is measured by the relative reduced graph size, as defined by

$$\rho = \frac{\sharp\mathcal{V}^*}{\sharp\mathcal{V}} \times 100, \tag{78}$$

where $\sharp$ stand for the cardinality and $\mathcal{V}^*$ denote the set of useful nodes actually present in the reduced graph. We also provide the difference between the relative reduced graph size obtained with the test (14) and the test proposed in [12]. In words, if the test (14) yield to larger reduced graph than the test proposed in [12] when $\Delta\rho > 0$ and conversely.

Let us remind that the test proposed in [12] takes the form:

$$\begin{cases} \text{either} & \left(\forall q \in B_p, \quad c_q \geq \frac{Per(B_p)}{Area(B_p)-1}\right), \\ \text{or} & \left(\forall q \in B_p, \quad c_q \leq -\frac{Per(B_p)}{Area(B_p)-1}\right), \end{cases} \tag{79}$$

where

$$Per(B_p) = \max\Big(\sharp\{(q,q') \in \mathcal{E}, q \in B_p, q' \in \mathcal{P} \setminus B_p\}, \\ \sharp\{(q',q) \in \mathcal{E}, q \in B_p, q' \in \mathcal{P} \setminus B_p\}\Big),$$

is the perimeter of B_p and

$$Area(B_p) = \sum_{q \in B_p} |c_q|,$$

is its area. Moreover, when using this test, we consider several neighborhoods $B = \{-r, \ldots, 0, \ldots, r\}^2$, with $r \in \{1, \ldots, 5\}$ and take the value r providing the smallest reduced graph. This is a significant advantage in favor of this method.

Notice that we do not investigate computational speed since it strongly depends on the quality of the algorithm exploiting Theorem 1 and its implementation. For the algorithm investigated in the current experiments, we remark that the test requires few computations. Therefore, the computation time may increase when the reduction strategy fails. However, the computations required by the test do not depend on the pixel location and the overall complexity for performing all the tests is linear with respect to the size of $\mathcal{V}$. This grows less rapidly than the resolution of the max-flow. Its worst-case complexity is indeed bounded from below by $O(\sharp\mathcal{V}.\sharp\mathcal{E})$. Therefore, we expect the algorithm to require smaller computational time than the straightforward max-flow computation when the reduction is significant. This phenomenon has already been observed in [12].

The graph-cut problems used for the experiments aim at solving the interactive image segmentation model described in [1]. More precisely, we consider an input image $(I_p)_{p\in\mathcal{P}}$ mapping each pixel $p \in \mathcal{P}$ to a color vector $I_p \in [0,1]^c$, where $c = 1$ for grayscale images and $c = 3$ for color images. The segmentation method finds, among $u \in \{0,1\}^{\mathcal{P}}$, a minimizer of the MRF defined by

$$E(u) = \beta \sum_{p\in\mathcal{P}} E_p(u_p) + \sum_{(p,q)\in(\mathcal{E})} E_{p,q}(u_p, u_q),$$

where $\mathcal{P}$ denotes the pixels, $\mathcal{E}$ corresponds to the 8-connectivity for 2D images and the 26-connectivity for 3D images. The unary terms $E_p(.)$ is defined at any pixel $p \in \mathcal{P}$ by

$$\begin{cases} E_p(1) = -log\, \mathbb{P}(I_p \mid p \in \mathcal{O}), \\ E_p(0) = -log\, \mathbb{P}(I_p \mid p \in \mathcal{B}). \end{cases} \tag{80}$$

where $\mathcal{O}$ and $\mathcal{B}$ denote respectively the object and background and $\mathbb{P}()$ is the density of the distribution of the colors for the object or the background. These densities are estimated, using normalized histograms, from objects and background seeds provided by the user.

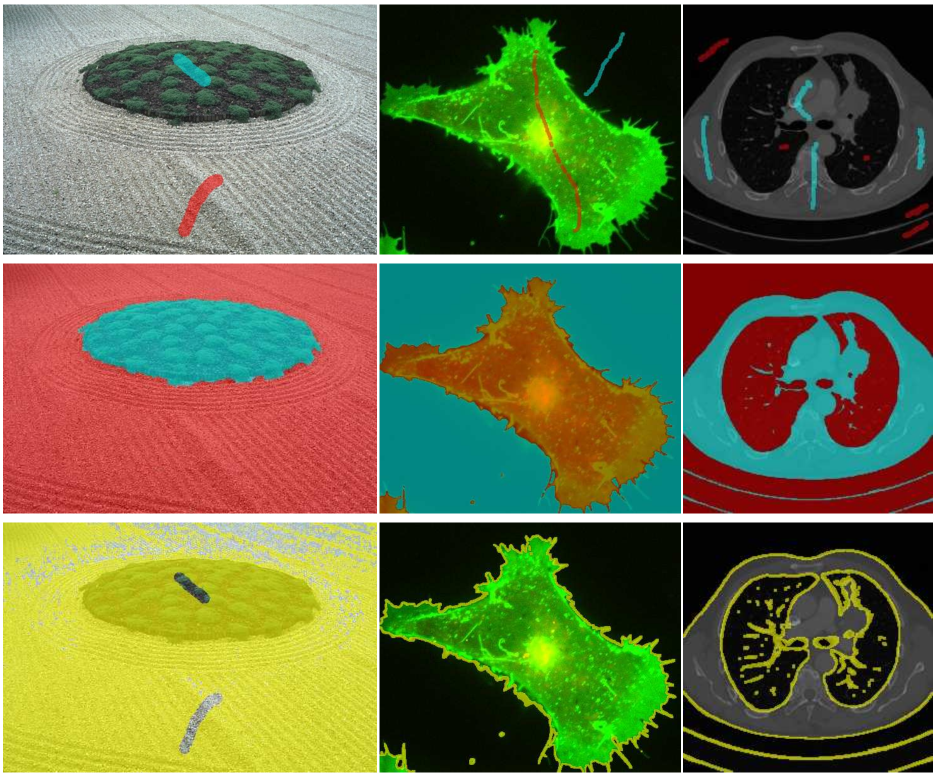
Moreover, for any pixel pair $(p,q) \in \mathcal{E}$, the interaction term $E_{p,q}(.)$ corresponds to a contrast-sensitive Ising model

$$E_{p,q}(u_p, u_q) = \begin{cases} 0 & \text{if } u_p = u_q, \\ \frac{1}{\|p-q\|_2} exp\Big(- \frac{\|I_p - I_q\|_2^2}{2\sigma^2} \Big) & \text{otherwise,} \end{cases} \tag{81}$$

where $\|.\|_2$ denotes the Euclidean norm (either in $\mathbb{R}^d$ or $\mathbb{R}^c$) and σ is a parameter. As an illustration, when the intensities I_p and I_q in (81) appear in the same range, we have $\|I_p - I_q\|_2 \ll \sigma$. This leads to a large cost and therefore discourages any cut between p and q. The opposite situation is also valid when the nodes p and q are located on both sides of a contour of the input image I. In such a case, the cost of a cut separating p and q is almost equal to 0.

The images used for the experiments are those already used in [12]. For each image, the seeds and the model parameters are tuned to obtain a good segmentation of the object (see Figure 2, where some images, seeds and segmentations are displayed). Using these seeds and these parameters, a reference segmentation is computed with standard graph-cut, without reduction (SGC). Afterwards, a second segmentation is computed, for the same seeds and parameters, when the graph is reduced with the test (14). A third segmentation is computed for the test (79), for the best value of r. The values ρ and $\Delta\rho$ (see above) are then computed. We also provide some values for the memory consumption. They are simply obtained by a straightforward closed-form expression involving the number of nodes and edges and the sizes required for representing these structures.

Finally, the experiments are performed on a computer with 2Gb of RAM. The max-flow algorithm used for the experiments is the one described in [2].



"zen-garden-c" (90.24%) $\beta = 4 \times 10^{-6}$ | "fluo-cell-c" (5.88%) $\beta = 2$ | "ct-thorax" (17.30%) $\beta = 20$

Figure 2: Seeds (top row), segmentations (middle row) and reduced graphs (bottom row) for a subset of images of Table 1. Reduced graphs are superimposed in yellow to the original image. Relative reduced graph sizes as well as β values are indicated below each image.

6.2 Experimental results

The results of these experiments are summarized in Table 1 and illustrated in Figure 2.

In Table 1 we see that, as expected, both tests (79) and (14) permit to use less memory than SGC. Moreover, when SGC fails to segment large 2D+t and 3D volumes, the algorithm using test (14) requires less memory and still succeeds to segment them. In Table 1, we also observe that the reduction strategy based on the test (14) is often as efficient as the one based on the test (79). It is however sometimes significantly less efficient than the reduction strategy based on the test (79). Taking a closer look at our experiments, this happens when the amount of regularization is relatively large (i.e. β is small). Indeed, in such a situation, the test (79) might remain true for small values of β, when the size of

Image	**Size**	**SGC**	**Test** (14)	ρ (%)	$\Delta\rho$ (%)
zen-garden-c	481 ×321	22.90MB	22.90MB	90.24	-0.51
red-flowers-c	481 ×321	22.90MB	10.40MB	46.26	22.96
book	3012 ×2048	917.26MB	78.95MB	7.91	-0.27
cells-z	512 ×512	38.91MB	23.39MB	61.65	12.74
interview-man2-c	426 ×240 ×180	7.55GB (*)	228.44MB	3.21	0.0
plane-take-off-c	492 ×276 ×180	10.03GB (*)	532MB	6.09	-0.11
fluo-cell-c	478 ×396 ×121	9.39GB (*)	514MB	5.88	0.0
ct-thorax	245 ×245 ×151	3.71GB (*)	771MB	17.30	0.0
cells	230 ×230 ×57	1.23GB	771MB	59.38	8.0
brain	181 ×217 ×181	2.91GB (*)	771MB	24.22	-0.16

Table 1: The memory required by SGC and the graph-cut using reduced graph with the test (14) are compared when segmenting 2D (top rows), 2D+t (middle rows) and 3D (bottom rows) images. Color images names are suffixed by "c". (*) means that SGC fail to segment the image due to a too large memory usage.

B increases. This does not occur with the test (14). However, when the amount of regularization is of moderate level, the memory gains are almost identical.

In a similar manner to the test (79), one can indeed observe in Figure 2 that, with the test (14), the reduced graphs have a larger size when β is small. This corresponds to a MRF in which the interaction term dominates. This is not surprising since the capacities c_q of the t-links are proportional to β. The test (14) is more often satisfied when β is large and conversely. The situation where β need to be small are typical of noisy images and segmentation problems involving object and background sharing similar colors (see Figure 3). In such a situation, many nodes inside B_p are connected to different terminals. The ideal situation in which the test permits to reduce the memory requirement therefore consists of graphs with large area of nodes linked to the same terminal and separated by rough borders (see images "fluo-cell-c" and "ct-thorax" in Figure 2).

Finally, as anticipated by Theorem 1, the value of the min-cut of all the reduced graphs, when using the test (14), are identical to the one of the entire graph. In fact, the cut/segmentation are even identical. More surprisingly, although there does not exist any theorem supporting this property, this is also the case for the results obtained with the test (79).

References

[1] Y. Boykov and M-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *International Conference on Computer Vision*, volume 1, pages 105–112, 2001.

[2] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis And Machine Intelligence*, 26(9):1124–1137, 2004.

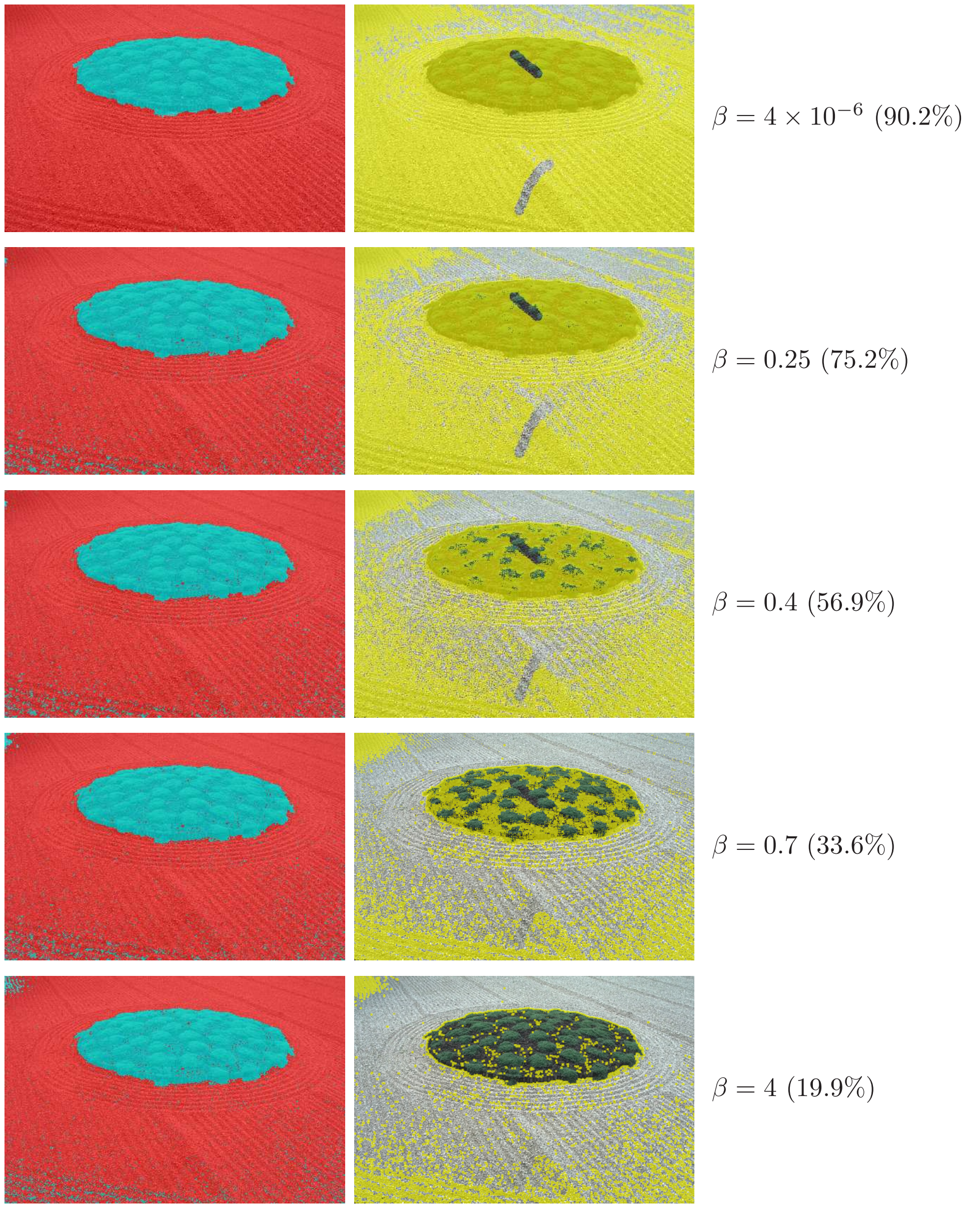


Figure 3: Influence of the parameter β on the relative reduced graph size ρ for the image "zen-garden-c". The segmentations (left column) and reduced graphs (in yellow, right column) are superimposed on the original image. For each value of β, the relative reduced graph size is indicated between parenthesis. In these experiments, the same seeds are used as in Table 1. We remind that smaller values of ρ means smaller memory consumption.

[3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. In *International Conference on Computer Vision*, volume 1, pages 377–384, 1999.

[4] G.B. Dantzig and D.R. Fulkerson. On the max-flow min-cut theorems of networks. *Annals of Mathematics Study*, 38:215–221, 1956.

[5] A. Delong and Y. Boykov. A scalable graph-cut algorithm for N-D grids. In *Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[6] K. Hogstedt, D. Kimelman, V.T. Rajan, T. Roth, and M. Wegman. Graph cutting algorithms for distributed applications partitioning. *ACM SIGMETRICS Performance Evaluation Review*, 28(4):27–29, March 2001.

[7] P. Kohli, V. Lempitsky, and C. Rother. Uncertainty driven multi-scale energy optimization. In *Symposium of the German Association for Pattern Recognition (DAGM)*, pages 242–251, 2010.

[8] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis And Machine Intelligence*, 26(2):147–159, 2004.

[9] V. Lempitsky and Y. Boykov. Global optimization for shape fitting. In *Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[10] N. Lermé and F. Malgouyres. Simultaneous segmentation and filtering via reduced graph cuts. In *Advanced Concepts for Intelligent Vision Systems*, pages 201–212, 2012.

[11] N. Lermé, F. Malgouyres, and L. Létocart. Reducing graphs in graph cut segmentation. In *International Conference on Image Processing*, pages 3045–3048, September 2010.

[12] N. Lermé, F. Malgouyres, and L. Létocart. A reduction method for graph cut optimization. *Pattern Analysis and Applications*, To appear with full reference (published online), 2013.

[13] N. Lermé, F. Malgouyres, and J.-M. Rocchisani. Fast and memory efficient segmentation of lung tumors using graph cuts. In *MICCAI, Third International Workshop on Pulmonary Image Analysis*, pages 9–20, September 2010.

[14] Y. Li, J. Sun, CK. Tang, and HY. Shum. Lazy Snapping. *ACM Transactions on Graphics*, 23(3):303–308, 2004.

[15] H. Lombaert, Y.Y. Sun, L. Grady, and C.Y. Xu. A multilevel banded graph cuts method for fast image segmentation. In *International Conference on Computer Vision*, volume 1, pages 259–265, 2005.

[16] F. Malgouyres and N. Lermé. Non-heuristic reduction of the graph in graph-cut optimization. In *NCMIP, IOP science, Journal of physics: Conference Series*, volume 386, page 012002, 2012.

[17] A.K. Ravindra, T.L. Magnanti, and J.B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.

[18] B. Scheuermann and B. Rosenhahn. Slimcuts: Graph cuts for high resolution images using graph reduction. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 6819 of *Lecture Notes in Computer Science*, pages 219–232. Springer, 2011.

[19] A.K. Sinop and L. Grady. Accurate banded graph cut segmentation of thin structures using laplacian pyramids. In *Medical Image Computing and Computer Assisted Intervention*, volume 2, pages 896–903, 2006.

[20] P. Strandmark and F. Kahl. Parallel and distributed graph cuts by dual decomposition. In *Conference on Computer Vision and Pattern Recognition*, pages 2085–2092, 2010.