

The Minimum Cost Connected Subgraph Problem in Medical Image Analysis

Markus Rempfler¹, Bjoern Andres², and Bjoern H. Menze¹

¹ Institute for Advanced Study & Department of Informatics,
Technical University of Munich, Germany

² Max Planck Institute for Informatics, Saarbrücken, Germany

Abstract. Several important tasks in medical image analysis can be stated in the form of an optimization problem whose feasible solutions are connected subgraphs. Examples include the reconstruction of neural or vascular structures under connectedness constraints.

We discuss the minimum cost connected subgraph (MCCS) problem and its approximations from the perspective of medical applications. We propose a) objective-dependent constraints and b) novel constraint generation schemes to solve this optimization problem exactly by means of a branch-and-cut algorithm. These are shown to improve scalability and allow us to solve instances of two medical benchmark datasets to optimality for the first time. This enables us to perform a quantitative comparison between exact and approximative algorithms, where we identify the geodesic tree algorithm as an excellent alternative to exact inference on the examined datasets.

1 Introduction

The *minimum cost connected subgraph* (MCCS) optimization problem arises in several medical image analysis tasks, most prominently for segmenting neural structures [1] or reconstructing vascular networks [2], where the maximum a posteriori (MAP) subgraph under connectedness constraints is inferred. Variations of this optimization problem have been proposed for anatomical labelling of vasculature [3] or artery-vein separation [4]. Imposing connectedness serves as regularizer, suppressing spurious detections and complementing incomplete observations, and it is often a requirement for further processing steps, e.g. if the reconstructed vasculature shall be used for biophysical simulations.

While [1,2,3,4] successfully solve an MCCS problem on heavily preprocessed, application-specific, sparse graphs, it would also be interesting to enforce connectedness on both very dense or large grid-graphs, for example in low-level segmentation tasks (Fig. 1, left), for 3D/4D reconstruction problems (Fig. 1, middle and right) or when it is not possible to reliably reduce the candidate graphs size. In these cases, however, the computational complexity becomes challenging. In fact, it was shown to be NP-hard in [5]. Nowozin & Lampert [6] propose an exact algorithm that tightens an outer polyhedral relaxation of the connected subgraph polytope by cutting planes. However, without guarantee to terminate

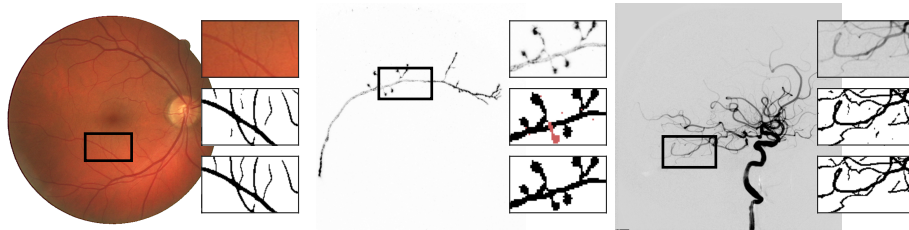


Fig. 1. *Examples for the MCCA on grid graphs.* **Left:** Segmentation of vasculature in retinal images. **Middle:** Reconstruction of a neuron from a 3D stack. Excessive disconnected components are shown in red for better visibility. **Right:** Delineation of vessels in a digital subtraction angiography (DSA) time series. The detail views show: raw image (**top**), without connectedness (**middle**) and with connectedness (**bottom**). Imposing connectedness constraints, i.e. requiring an MCCA, helps to reconnect disconnected terminals and remove spurious detections without penalizing thin tubular structures.

in polynomial time, it was found to be too slow to solve typical instances of medical benchmark datasets to optimality. To this end, two heuristical algorithms were proposed by Chen et al. [7] and Stühmer et al. [8]. They either use an approximative formulation of the connected subgraph polytope by means of a precomputed geodesic shortest path tree [8] or iteratively solve a surrogate problem that is based on altered weights of the original problem [7]. Both approaches are fast enough for medical applications and were reported to yield qualitatively promising results. A quantitative comparison, however, has been prevented by the prohibitively expensive computation of exact solutions to the MCCA problem.

In this paper, we revisit the MCCA in an integer linear programming (ILP) framework for MAP estimation under connectedness constraints. First, we contribute to the exact optimization by proposing a) *objective-dependent constraints* that reduce the size of the polytope and hence, reduce the number of potential solutions to explore, and b) *constraint generation strategies* beyond the standard *nearest* and *minimal* separator strategy, which we show to have a strong impact on the runtime of the ILP. Both propositions together enable us to compute the MCCA on several instances of two medical benchmark datasets – addressing vessel segmentation and neural fiber reconstruction – to optimality. Our second contribution is a first quantitative comparison of the exact algorithm and the two heuristics in terms of runtime, objective function and semantic error metrics.

2 Background

We are interested in the most likely binary labeling $\mathbf{x} \in \{0, 1\}^{|V|}$ of the nodes V in the graph $G = (V, E)$. A node i is active if $x_i = 1$. By imposing connectedness constraints, i.e. $\mathbf{x} \in \Omega$, the MAP estimate becomes a MCCA problem:

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in \{0, 1\}^{|V|}} P(\mathbf{X} = \mathbf{x} | I, \Omega) = \arg \max_{\mathbf{x} \in \Omega} P(\mathbf{X} = \mathbf{x} | I) \quad , \quad (1)$$

where I is the image evidence and Ω denotes the set of \mathbf{x} that are connected subgraphs of G . In this section, we discuss two formulations of Ω , the exact formulation that follows [6] and the geodesic tree formulation of [8].

2.1 Exact Connectedness

Following [6], we can describe Ω with the following set of linear inequality constraints

$$\forall i, j \in V, (i, j) \notin E : \forall \mathcal{S} \in S(i, j) \quad x_i + x_j - 1 \leq \sum_{k \in \mathcal{S}} x_k \quad , \quad (2)$$

where \mathcal{S} is a set of vertices that separate i and j , while $S(i, j)$ is the collection of all vertex separator sets for i and j . In other words, if two nodes i and j are active, then they are not allowed to be separable by any set of inactive nodes. Thus, a path of active nodes has to exist. In practice, this set of constraints is too large to be generated in advance. However, given a labelling \mathbf{x} we can identify at least a subset of the violated connectedness constraints in polynomial time, add them to the ILP and search for a new feasible solution. This approach is known as *lazy constraint generation*. In Sec. 3.2, we detail on identifying and adding these constraints.

Rooted case. In many medical segmentation problems, it is reasonable to assume that a root node can be identified beforehand with an application-specific detector, manually or by a heuristic, such as picking the strongest node in the largest component. If a known root r exists, it suffices to check connectedness to the root node instead of all pairs of active nodes. The constraints in (2) then become

$$\forall i \in V \setminus \{r\}, (r, i) \notin E : \forall \mathcal{S} \in S(i, r) \quad x_i \leq \sum_{k \in \mathcal{S}} x_k \quad . \quad (3)$$

2.2 Geodesic Tree Connectedness

Alternative to the exact description of all connected subgraphs that we discussed in the previous section, we can formulate a connectedness prior as in [8] on a *geodesic shortest path tree* $T(G) = (V, A \subseteq E)$ rooted in r . Here, $T(G)$ is precomputed based on the unary potentials, i.e. with edge weights defined as $f(i, j) = \frac{1}{2} (\max(w_i, 0) + \max(w_j, 0))$. The set of feasible solutions is then given by the inequalities:

$$\forall i \in V \setminus \{r\}, (p, i) \in T(G) \quad x_i \leq x_p \quad , \quad (4)$$

where p is the parent of i in the geodesic tree $T(G)$. With this set of constraints, a node i can only be active if his parent p in the geodesic tree is also active, thus connecting all active nodes to the root r along the branches of $T(G)$. Advantages of this approach are that only $|V| - 1$ constraints are necessary to describe the set of feasible solutions and that the relaxation is tight. On the other hand, the inequalities of (4) describe a strict subset of (3), unless $T(G) = G$. Hence it might discard an optimal solution that is feasible in (3).

3 Methods

Given the probabilistic model $P(\mathbf{X} = \mathbf{x}|I)$ of (1) is a random field over $G = (V, E)$, we can write its MAP estimator $\mathbf{x}^* = \arg \max_{\mathbf{x} \in \{0,1\}^V} P(\mathbf{X} = \mathbf{x}|I, \Omega)$ as an ILP. We will assume for the remaining part that $P(\mathbf{X} = \mathbf{x}|I) = \prod_{i \in V} P(x_i|I)$, leading to the ILP:

$$\text{minimize} \quad \sum_{i \in V} w_i x_i \quad , \quad (5)$$

$$\text{s.t.} \quad \mathbf{x} \in \Omega \quad , \quad (6)$$

$$\mathbf{x} \in \{0, 1\}^{|V|} \quad , \quad (7)$$

where (6) are the connectedness constraints, i.e. either (3) or (4), (7) enforces integrality, and w_i are the weights that can be derived as $w_i = -\log \frac{P(x_i=1|I)}{1-P(x_i=1|I)}$. Higher order terms of the random field can be incorporated by introducing auxiliary binary variables and according constraints as done in [2]. Note, however, that [7] reported problem instances with weak or no pairwise potentials – as we are addressing them here – to be amongst the most difficult.

3.1 Objective-dependent Constraints

Given the problem with unary terms, we observe that, for any connected component $\mathcal{U} \subset V$ composed of *unfavourable* nodes only, i.e. $\forall i \in \mathcal{U}, w_i > 0$, it can only be active in the optimal solution if there are at least two active nodes in its neighbourhood:

$$\forall i \in \mathcal{U} \quad 2x_i \leq \sum_{j \in \cup_{k \in \mathcal{U}} \delta(k) \setminus \mathcal{U}} x_j \quad , \quad (8)$$

where $\delta(k)$ is the set of neighbouring nodes to k . In other words, unfavourable nodes can not form a leaf in the optimal solution (otherwise, removing the unfavourable nodes would give us a better solution without losing connectedness). In the special case of $|\mathcal{U}| = 1$, we can add the constraint from the beginning. This removes feasible solutions from Ω that are a priori known to be suboptimal, hence reducing the search space in the optimization and making it unnecessary to add a large set of separator inequalities.

Higher-order weights. Even though we only define (8) for unary weights, it is possible to adapt the constraint to higher-order models by changing the condition to $w_i + \min_{j \in \delta(i)} w_{ij} > 0$, provided the pairwise weights w_{ij} are only introduced for neighbouring nodes i, j such that $(i, j) \in E$.

3.2 Constraint Generation Strategies

The extensive number of inequalities needed for (3) makes it necessary to identify violated constraints during the optimization and add them to the problem. We note that it suffices to treat individual connected components as one entity,

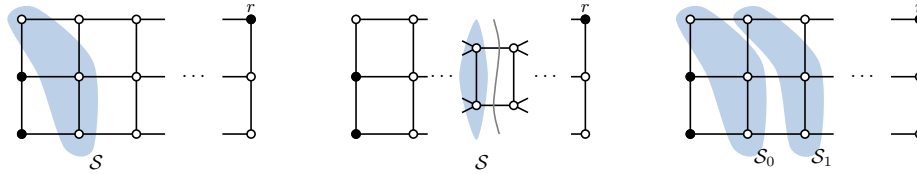


Fig. 2. *Constraint generation strategies.* Illustration of the nearest separator (left), minimal separator (middle) and k -nearest (right) strategies. Active nodes are shown in black, inactive nodes are white and the identified separator sets \mathcal{S} are marked in blue. \mathcal{S} is subsequently used to generate the corresponding constraint in (2) or (3).

since establishing a connection automatically connects all pairs of nodes between them. Identifying violated constraints boils down to finding a vertex separator set \mathcal{S} between two disconnected, active components in the current solution. The constraints corresponding to \mathcal{S} are then generated according to (2) or (3) for all nodes in the given connected component.

At the heart of this technique is the observation that only a subset of inequalities is active at the optimum of a given problem instance. However, depending on the choice of the inequalities that we add in each step, we may explore (and therefore construct) different parts of the polytope Ω , most likely requiring a different number of iterations.

In the following, we first review the two standard strategies, namely the *nearest* and *minimal* separator, and then propose several novel, alternative strategies.

Nearest separator. In this standard approach, the vertex separator set in the immediate neighbourhood of the active component is picked for generating the new constraint. This strategy has been used, for example, in [2]. It is motivated by its simplicity and the fact that it often coincides with the minimal separator strategy for small components.

Minimal separator. A minimal (in terms of $|\mathcal{S}|$) separator set is obtained by solving a max-flow problem between any two disjoint active components at hand and selecting the smaller vertex set on either side of the resulting min-cut. For the max-flow, we set the flow capacity c in edge (i, j) as $c(i, j) = \max(1 - x_i, 1 - x_j)$. The strategy was applied in [6].

Equidistant separator. Alternatively, we can identify the separator set \mathcal{S} that is equidistant to the current active component and all other components by running a breadth-first search (BFS) from either side. Similar to the max-flow of the minimal separator, the distance measure is only accounting for non-active nodes. This strategy originates in the observation that the weakest evidence between two components is often found half-way into the connecting path.

k -Nearest and k -Interleave. We run a BFS from the active component C and collect the k (disjoint) separator sets $\{\mathcal{S}_n\}_{n=0}^{k-1}$ composed of all nodes with identical distance. The search terminates if k equals the number of nodes in C or

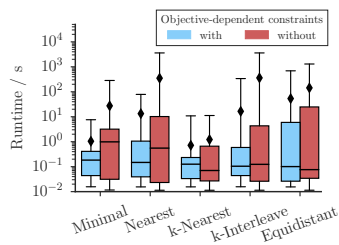


Fig. 3. Runtime with and without the proposed objective-dependent constraints on 64×64 instances. Mean values are depicted by \blacklozenge , whiskers span $[\min, \max]$ values. Unsolved instances are excluded for readability. We find that all strategies benefit from the additional constraints. Additional *per-instance* information can be found in the supplement.

if another active node is reached. For the k -interleave, only separators with even distance are chosen. The intuition behind these strategies is that a wider range of neighbours (and their neighbours) has to be considered for the next solution.

4 Experiments & Results

Datasets & Preprocessing. We conduct experiments on two medical datasets: First, on the DRIVE database of retinal images [9], each being 565×584 px. We use the probability estimates $P(x_i = 1|I)$ for a pixel i being vasculature from the recent state-of-the-art approach of [10] for our unaries. Second, we run experiments on the olfactory projection fibers (OPF) dataset [11], composed of 8 3D confocal microscopy image stacks. We use the stacks prepared in [1], where we estimate $P(x_i = 1|I)$ of voxel i being part of the fiber by a logistic regression on the image intensities. We segment the nerve fiber under the requirement of connectedness on the 3D grid graph of $256 \times 256 \times n$ nodes with $n \in \{30, \dots, 51\}$ depending on the case. The probability $P(x_i = 1|I)$ of voxel i being part of the fiber is estimated by a logistic regression on the image intensities. Both datasets are illustrated in Fig. 1.

Optimization. We solve the ILP (5) by the branch-and-cut algorithm of the solver Gurobi [12] with a default relative gap of 10^{-4} . Objective-dependent constraints for single nodes (Sec. 3.1) are added from the beginning. For the exact connectedness (Sec. 2.1), the strategies described in Sect. 3.2 are implemented as a callback: Whenever the solver arrives at an integral solution \mathbf{x}' , violated constraints are identified and added to the model. If no such violation is found, i.e. \mathbf{x}' is already connected, then it is accepted as new current solution \mathbf{x}^* . For the geodesic tree connectedness (Sec. 2.2), all constraints are added at once. In order to arrive at a fair comparison, we define the root node for both approaches.

Experiment: Objective-dependent constraints. To examine the impact of the objective-dependent constraints, we subsample 25 subimages of 64×64 px from the DRIVE instances and run the ILP once with and once without the additional first order constraints of (8). As shown in Fig. 3, we find that all strategies benefit from the additional constraints.

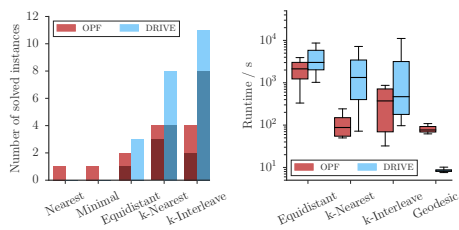


Fig. 4. Left: Number of solved instances per strategy. The darker bar indicates how often a strategy was the *fastest* to solve an instance. **Right:** Runtime on solved instances. Strategies with too few solved instances are not included. k -Nearest and k -Interleave are found to be the most successful exact strategies.

Table 1. Segmentation scores in terms of **F1**-score, (**P**recision, **R**ecall) in % on the solved instances. All approaches outperform the baseline (MaxComp), while no significant difference can be found between them.

	OPF		DRIVE	
	F1	(P R)	F1	(P R)
Maxcomp	68.5	(67.7, 71.9)	78.7	(87.2, 72.1)
Geodesic	76.2	(69.1, 85.4)	80.1	(86.2, 75.2)
Topocut	-	-	-	80.1 (86.4, 74.9)
Exact	76.2	(69.1, 85.4)	80.1	(86.2, 75.2)

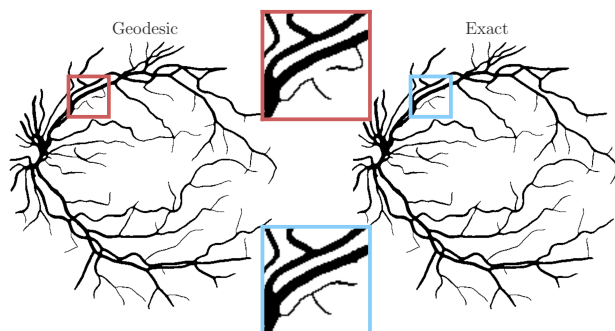


Fig. 5. Comparison of exact and approximative connectedness: Major differences as the one indicated are encountered mainly if solutions are competing under the model $P(\mathbf{X} = \mathbf{x}|I)$ and thus almost equivalent w.r.t. objective value.

Experiment: Comparing exact and approximative algorithms. We compare exact and geodesic tree MCCS on both datasets. On 2D images, we additionally compare to the method by [7] called Topocut. As a baseline, we compute the maximum connected component in the non-constrained solution (Maxcomp). The results are presented in Fig. 4 and Table 4 (additional information *per instance* is provided in the supplement). We observe that 6/8 and 12/20 instances were solved to optimality with our propositions, while standard strategies solved ≤ 1 . k -Nearest and k -interleave are the two most successful exact strategies in terms of solved instances and speed. In terms of segmentation scores, the two heuristics are on par with the exact algorithm, while all of them outperform the baseline. We find the geodesic approach to match the exact solution with respect to objective values in all instances (within a relative difference of 10^{-4}), whereas Topocut often obtains slightly lower objective values than the geodesic approach. A qualitative comparison between an exact and geodesic solution is presented in Fig. 5.

5 Conclusions

We have shown that exact optimization of the MCCS, as it is typical for neural and vascular structure reconstruction tasks, strongly benefits from the proposed objective-dependent constraints and the constraint generation strategies. In a first quantitative comparison between exact and approximative approaches on two datasets, we found that the geodesic tree formulation is a fast, yet highly competitive alternative to exact optimization.

While we focussed on large grid-graphs that are most important for low-level segmentation and reconstruction, we expect that our findings transfer to MCCS problems and related ILP-based formulations on sparse graphs, e.g. those discussed in [1,2,3,4], and thus consider this a promising direction for future work. Besides, it will be interesting to investigate the effect of our propositions in the presence of higher-order terms.

Acknowledgements. With the support of the Technische Universität München – Institute for Advanced Study, funded by the German Excellence Initiative (and the European Union Seventh Framework Programme under grant agreement n 291763).

References

1. Türetken, E., Benmansour, F., Andres, B., et al.: Reconstructing curvilinear networks using path classifiers and integer programming. *IEEE TPAMI*, preprint (2016)
2. Rempfler, M., Schneider, M., Ielacqua, G.D., et al.: Reconstructing cerebrovascular networks under local physiological constraints by integer programming. *Medical Image Analysis* 25(1), 86–94 (2015)
3. Robben, D., Türetken, E., Sunaert, S., et al.: Simultaneous segmentation and anatomical labeling of the cerebral vasculature. *Medical Image Analysis* 32, 201–215 (2016)
4. Payer, C., et al.: Automated integer programming based separation of arteries and veins from thoracic CT images. *Medical Image Analysis*, preprint (2016)
5. Vicente, S., Kolmogorov, V., Rother, C.: Graph cut based image segmentation with connectivity priors. In: *Proc. CVPR*, pp. 1–8 (2008)
6. Nowozin, S., Lampert, C.H.: Global Connectivity Potentials for Random Field Models. In: *Proc. CVPR*, pp. 818–825 (2009)
7. Chen, C., Freedman, D., Lampert, C.H.: Enforcing topological constraints in random field image segmentation. In: *Proc. CVPR*, pp. 2089–2096 (2011)
8. Stühmer, J., Schroder, P., Cremers, D.: Tree shape priors with connectivity constraints using convex relaxation on general graphs. In: *Proc. ICCV*, pp. 2336–2343 (2013)
9. Staal, J.J., Abramoff, M.D., Niemeijer, M., et al.: Ridge based vessel segmentation in color images of the retina. *IEEE TMI* 23(4), 501–509 (2004)
10. Ganin, Y., Lempitsky, V.: N4-Fields: Neural network nearest neighbor fields for image transforms. In: Cremers, D., Reid, I., Saito, H., Yang, M.H., (eds.): *ACCV 2014. LNCS*, vol. 9004, pp. 536–551 Springer International, Cham (2015)
11. Brown, K.M., Barrionuevo, G., Canty, A.J., et al.: The DIADEM data sets. *Neuroinformatics* 9(2), 143–157 (2011)
12. Gurobi Optimization, I.: *Gurobi Optimizer Reference Manual* (2015)