Optimal Embedding of Heterogeneous Graph Data with Edge Crossing Constraints [*][†]

A. Shabbeer, C. Ozcaglar, M. Gonzalez, K. P. Bennett
Rensselaer Polytechnic Institute, Troy, NY
shabba@cs.rpi.edu, ozcagc2@cs.rpi.edu, gonzam3@rpi.edu, bennek@rpi.edu

November, 2010

Abstract

We propose a novel approach to visualization of heterogeneous data characterized by both a relationship graph structure and intrinsic features. Each data point is a node in a graph with a given structure. Each data point is also associated with a set of features that have a corresponding distance or similarity measure. A successful visualization accurately captures the desired proximity structure as measured by some embedding objective while simultaneously optimizing an aesthetic criterion, no edge crossings. The edge-crossing constraint is expressed as a nonlinear constraint which has an intuitive geometric interpretation closely related to support vector machine classification. The approach can be generalized to remove intersections of general convex polygons including node-edge and node-node intersections. We demonstrate the approach on multi-dimensional scaling or equivalently Kamada-Kawai force-directed graph layout, by modifying the stress majorization algorithm to include penalized edge crossings. The resulting Expectation-Maximization-like algorithm can be readily adapted to other supervised and unsupervised optimization-based embedding or dimensionality reduction methods. The method is demonstrated on a problem in tuberculosis molecular epidemiology – creating *spoligoforests* for visualizing genetic relatedness between strains of the *Mycobacterium tuberculosis* complex characterized by a phylogenetic forest, and multiple biomarkers with a corresponding non-metric genetic distance.

1 Introduction

Graphs can be used to model relationships between elements, where the elements are represented as nodes and the relations by edges. Graph visualization can be used to better understand these underlying relationships in a dataset. Frequently, the nodes of a graph represent objects that have their own intrinsic properties with associated distances or similarity measures. The motivating heterogenous data/graph application for this work is visualization of phylogenetic forests of bacteria (here spoligoforests). Each node represents a genetic strain of *Mycobacterium tuberculosis* complex (MTBC) and each edge represents a putative evolutionary change. Each node or strain has a genetic fingerprint and a natural non-metric distance that can be defined to every other strain even if they are not connected in the underlying graph. Similarly, in a Web hyperlinks graph, each node may be a web page and the edge may represent a hyperlink between the pages. Each webpage is a document with intrinsic properties, so there is an associated distance or similarity measure between nodes even if no link exists between them.

The quality of a visualization can be gauged on the basis of how easily it can be understood and interpreted. Certain criteria have been identified that characterize a good visualization. For graphs, it is desirable to minimize edge crossings. For general data embedding, the desired quality is frequently expressed as a function of the embedding and then optimized. For example in Multidimensional Scaling (MDS) (or equivalently

[*] This work was supported by NIH R01LM009731.

[†] Presented at NIPS Workshop on Challenges of Data Visualization, 2010

the Kamada-Kawai model in force-directed graph placement (FDP)), the goal is to produce an embedding that minimizes the difference between the actual and embedded distances between all nodes. If a graph is planar, we would like to produce a planar embedding of the graph that minimizes the MDS or other embedding objective. Thus, a natural question for such heterogeneous data that comprises of data points characterized by features and by an underlying graph structure [**?**, **?**] is how to optimize the embedding criteria while minimizing the number of edge crossings in the embedded graph.

Figure 2 shows the visualization of planar spoligoforest for the LAM subfamilies of MTBC created by the proposed approach and three widely-used graph drawing methods: (a) the proposed approach; (b) Graphviz Neato - which is an FDP algorithm equivalent to MDS; and (c) Graphviz Twopi - a planar radial graph algorithm (http://www.graphviz.org/). In (c), the radial graph is visually appealing but inaccurate because genetically similar strains (represented by the same color sublineages) are placed far apart especially when the graphs are disconnected. In (b), the distances between strains match the sublineage structure but there are many edge crossings. The proposed approach in (a) represents distance correctly without any edge crossings in the layout by optimizing the MDS embedding or dimensionality reduction objective with additional edge cross penalties.

The key insight of the paper is that the condition that two edges do **not** cross is equivalent to the feasibility of a system of nonlinear inequalities. In Section 2, we prove this using a theorem of the alternative: Farkas' Theorem. The transformed system ensures that the two edges are separated by a linear hyperplane. Thus the edge-crossing constraint reduces to a classification problem which is very closely related to support vector machines (SVM). The system of inequalities can then be relaxed to create a natural penalty function for each possible edge crossing. This non-negative function goes to zero if no edge crossings occur. This general approach is applicable to the intersection of groups of convex polyhedrons including nodes represented as boxes and edges represented as bars.

In Section 3, we explore how edge-crossing constraints can be added to stress majorization algorithms for MDS/FDP. We develop an algorithm which simultaneously minimizes stress while eliminating or reducing edge crossings using penalized stress majorization. The method solves a series of unconstrained nonlinear programs in Matlab. We demonstrate the approach on a compelling problem in tuberculosis molecular epidemiology. The graphical results are shown for spoligoforests drawn using two different types of biomarkers. Animations of the algorithm illustrating how the edge crossing penalty progressively transforms the graphs are provided at www.cs.rpi.edu/ bennek/tbinsight/FinalCuts/.

2 Continuous Edge-Crossing Constraints

We show how edge-crossing constraints can be expressed as a system of nonlinear inequalities. Each point on an edge can be represented as the convex combination of the extreme points of the edge. Consider edge $\mathcal{A}$ with end points $a = [a_x \;\; a_y]$ and $c = [c_x \;\; c_y]$ and edge $\mathcal{B}$ with end points $b = [b_x \;\; b_y]$ and $d = [d_x \;\; d_y]$. The matrices A and B contain the end or extreme points of the edges $\mathcal{A}$ and $\mathcal{B}$ respectively. Any point in the intersection of edge $\mathcal{A}$ and $\mathcal{B}$ can be written as a convex combination of the extreme points of $\mathcal{A}$ and convex combination of the extreme points of $\mathcal{B}$. Therefore, two edges do not intersect if and only if the following system of equations has **no solution**:

$$\text{there exists no } \delta_A \text{ and } \delta_B \text{ such that } \quad A'\delta_A = B'\delta_B \quad e'\delta_A = 1 \quad e'\delta_B = 1 \quad \delta_A \geq 0 \quad \delta_B \geq 0 \tag{1}$$

where e is a vector of ones and $A = \begin{bmatrix} a_x & a_y \\ c_x & c_y \end{bmatrix}$ and $B = \begin{bmatrix} b_x & b_y \\ d_x & d_y \end{bmatrix}$. The conditions that two given edges do *not* cross, i.e. that (1) has no solution, are precisely characterized by using Farkas' Theorem.

Theorem 1 (Conditions for no edge crossing)**.** *The edges $\mathcal{A}$ and $\mathcal{B}$ do not cross if and only if there exists u, α and β,*

$$\textit{such that } Au \geq \alpha e \qquad Bu \leq \beta e \qquad \alpha - \beta > 0. \tag{2}$$

Allowing $\mathcal{A}$ and $\mathcal{B}$ to be of an arbitrary number of extreme points, the following corollary can be easily proven to apply to intersections between convex polyhedrons expressed as a convex combination of their extreme points.

Corollary 1 (Conditions for no intersection of two polyhedrons)**.** *Consider the polyhedrons* $\mathcal{A} = \{x|x = A'\delta_A, e'\delta_A = 1, \delta_A \geq 0\}$ *and* $\mathcal{B} = \{x|x = B'\delta_B, e'\delta_B = 1, \delta_B \geq 0\}$*. The polyhedrons do not intersect,* $\mathcal{A} \bigcap \mathcal{B} = \emptyset$*, if and only if*

$$\textit{there exists } u \textit{ and } v \textit{ such that } Au + \gamma e \geq 0 \qquad Bu + \gamma e \leq -e \tag{3}$$

Therefore, two edges (or more generally two polyhedrons) **do not** intersect if and only if

$$0 = \min_{u,v} ||(-Au - e\gamma)_+||_q^q + ||(Bu + (1+\gamma)e)_+||_q^q \text{ where } (z)_+ = max(0, z) \text{ for } q = 1 \text{ or } \ q = 2. \tag{4}$$

Much like soft-margin SVM classification, two edges (or more generally two polyhedrons) do not intersect if and only if there exists a hyperplane that strictly separates the extreme points of $\mathcal{A}$ and $\mathcal{B}$. If the edges do not cross, then the optimal objective of (4) will be 0; while it will be strictly greater than 0 if the edges do cross. As in SVM, (4) can be converted into a linear or quadratic program depending on the choice of $q = 1$ or $q = 2$ respectively. Figure 2 illustrates that the no-edge-crossing constraint corresponds to introducing a separating hyperplane and requiring each edge to lie in opposite half spaces.

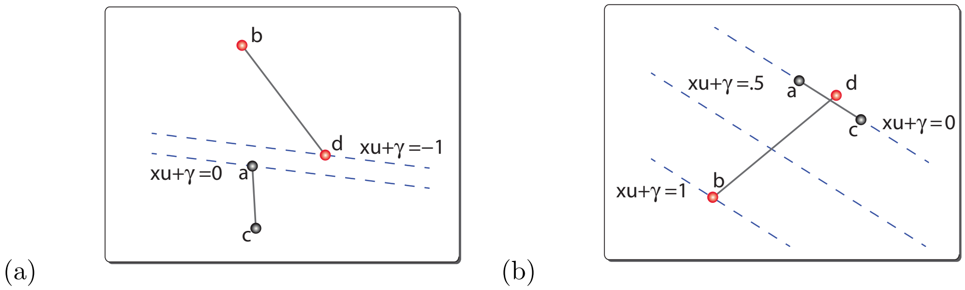


Figure 1: In (a) Edge $\mathcal{A}$ from a to c and edge $\mathcal{B}$ from b to d do not cross. Any line between $xu + \gamma = 0$ and $xu + \gamma = -1$ strictly separates the edges. Using a soft margin, the plane in (b) $xu + \gamma = 0.5$ separates the plane into half spaces that should contain each edge.

Using a penalty approach, edge crossing minimization can be incorporated into any optimization-based embedding or graph drawing formulation. In this paper, we use the Kamada-Kawai stress, a weighted version of MDS, with the addition of 1-norm edge crossing penalties, thus producing

$$\min_{X,u,\gamma} stress(X) + \sum_{i=1}^{m} \frac{\rho_i}{2}[||(-A^i(X)u^i - \gamma^i)_+||_1 + ||(B^i(X)u^i + (1+\gamma^i)e)_+||_1] \tag{5}$$

where the penalties $\rho \geq 0$, X_i is the position of the node i in the embedding and d_{ij} represents the distance between nodes i and j, and $stress(X) = \sum_{i<j} w_{ij}(||X_i - X_j|| - d_{ij})^2$. The normalization constant $w_{ij} = d_{ij}^{-\alpha}$, $\alpha = 3$ is used. The penalty approach provides an efficient mechanism for dealing with the large number of potential edge crossings ($\frac{\ell(\ell-1)}{2}$ for ℓ edges).

For each fixed value of ρ, Problem 5 is solved using an EM-like algorithm that alternates between minimizing with respect to X and u. For the X phase, a modified version of "Stress Majorization" [**?**] is used to optimize (5). The Matlab BFGS optimization algorithm "fminunc" is used to optimize a quadratic upper bound on the stress plus the edge crossing penalties for a fixed u. In the u phase, the soft margin separating plane (u) for each edge pair as defined by X is determined either by solving (4) or by an inexpensive heuristic (boxes enclosing edges do not intersect). The penalties for crossed edges are driven higher until no edge crossings exist or the problem converges; thus, most edge pairs have penalty parameter $\rho_i = 0$ since they never cross. The initial solution X^0 is calculated using classical multidimensional scaling via the *cmdscale* command. This algorithm represents a preliminary effort to demonstrate the potential of the approach. Many improvements are possible.

3 Results

To demonstrate the performance of the approach, we return to the motivating application: visualization of spoligoforests [?] created from DNA fingerprints of MTBC. We examine the visualization of spoligoforests with distance matrices defined using spoligotype and MIRU for four problems as summarized in Table 1. For each problem, we present three visualizations of the spoligoforest drawn using: a) the proposed approach that minimizes stress with edge-crossing penalties, (b) Graphviz Neato or stress majorization with distances specified between all pairs of nodes or equivalently MDS [?], and (c) Graphviz Twopi that produces radial layouts. In every case, the proposed method can dramatically reduce the edge crossings (to zero in three of four cases), while making only minor changes in the total stress. The pictures produced are more informative and accurate than those produced by all existing spoligoforest visualization software that use Graphviz algorithms, including Twopi, that disregard genetic distances available in the heterogeneous data (www.emi.unsw.edu.au/spolTools and tbinsight.cs.rpi.edu). The results reported were performed on a Lenovo Thinkpad W500 laptop with 4GB RAM. The proposed approach can be used to dynamically remove edge crossings in an existing graph. An animation of the proposed algorithm altering the initial MDS solution can be viewed at www.cs.rpi.edu/ bennek/tbinsight/FinalCuts/.

Data	Number of Nodes	Number of Edges	Neato		EdgeCrossMin				
			Stress	# Edge cross	Init. MDS stress	Final stress	Init. crossings	Final crossings	# iterations
LAMs	68	66	204.29	30	180.23	217.54	26	0	41
M. africanum	45	29	2	2	1.71	1.79	9	0	11
H, X, LAM	97	89	276.14	22	255.92	261.93	17	0	101
MIRUVNTRplus	197	124	1320	243	3525.10	1346.52	219	17	29

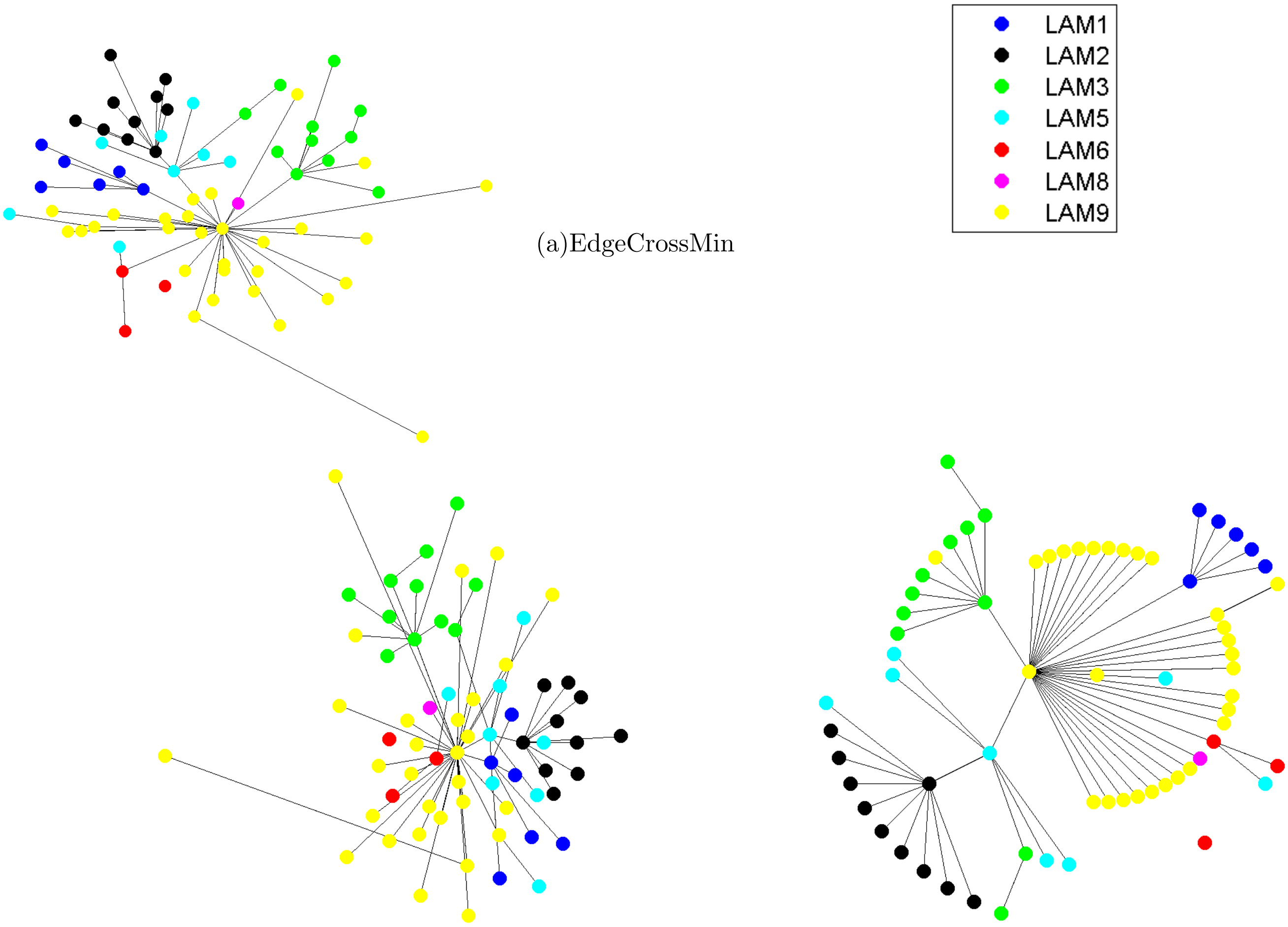
Table 1: Results of the proposed approach versus MDS/FDP using Neato on four MTBC spoligoforest datasets.

4 Discussion

We developed a novel approach to simultaneously optimizing preservation of proximity relations and aesthetic criteria for heterogeneous graph data by introducing a fundamentally new paradigm for elimination of edge crossings in graph visualizations. This work demonstrates how edge-crossing constraints can be formulated as a system of nonconvex constraints. Edges do not cross if and only if they can be strictly separated by a hyperplane. If the edges cross, then the hyperplane defines the desired half-spaces that the edges should lie within. The edge-crossing constraints can be transformed into a continuous edge-crossing penalty function in either 1-norm or least-squares form. We developed a stress majorization algorithm with edge-crossing penalties. Computational results demonstrate that this approach is quite practical and tractable. Continuous optimization methods can be used to effectively find local solutions. Successful results were illustrated on problems of the epidemiology of Tuberculosis that were not adequately addressed using existing graphing approaches since they give undesirable results on disconnected graphs. Edge crossings can be eliminated by making only small changes in the stress.

This work opens up many avenues for future research at the intersection of machine learning and data visualization. Here we focused on elimination of edge crossings and FDP/MDS stress optimization. The general approach is applicable to any optimization-based graph drawing, dimensionality reduction or embedding methods [?, ?] used for data visualization in both supervised and unsupervised learning. Also, the theorems and algorithms are directly applicable to the intersection of convex polygons in general within embeddings of arbitrary dimensions. Thus, the method can also be used to eliminate node-node overlaps and node-edge crossings. Our preliminary work was limited to planar graphs, but the penalty approach could be used to reduce crossing in more general graphs. Since the edge-crossing constraints are very closely related to linear SVM, all the different classification and regularization loss functions could be used to produce crossing-penalty functions with different aesthetic effects (e.g. minimum margin separation) and algorithmic ramifications. Our preliminary work used the Matlab function “fminunc” as its primary workhorse – which inherently limits the problem size. In reality, there is a great potential for making highly scalable special purpose algorithms for edge crossing constrained graph embeddings. The state-of-the-art linear SVM algorithms which are massively scalable can potentially adapted to this problem as well. While the method was motivated by heterogeneous graph data, it can be used to eliminate edge-crossing in any optimization-based graph drawing algorithm. We leave these promising research directions as future work.

5 Visualizations

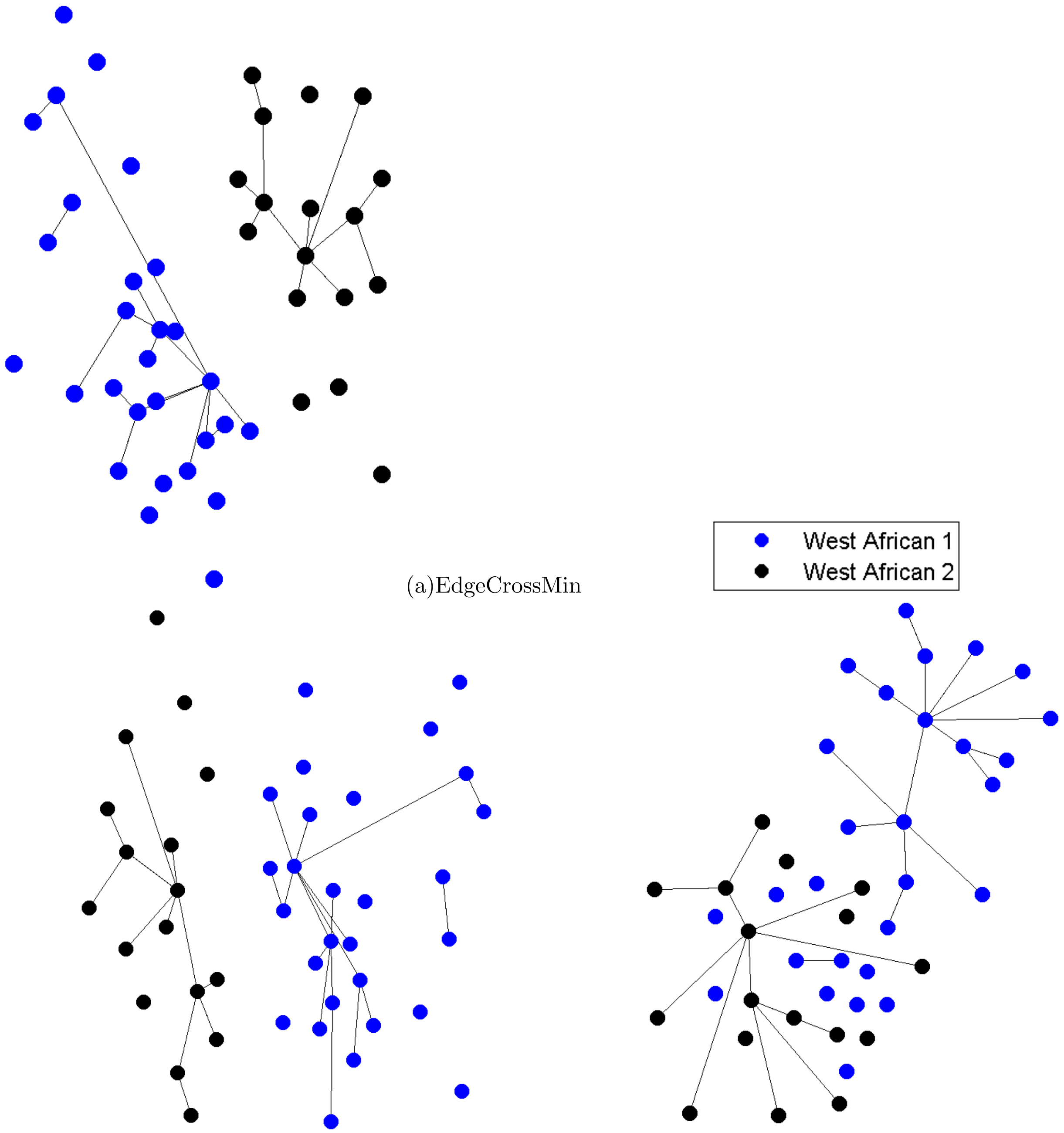


(a)EdgeCrossMin

(b)Neato

(c)Twopi

Figure 2: Visualization of LAM (Latin-American-Mediterranean) sublineages. In graph (c) drawn using Twopi, the radial layout is visually appealing, but genetic distances between strains are not faithfully reflected. Graph (b), drawn using Neato preserves proximity relations but has edge-crossings. In graph (a), the proposed approach eliminates all edge crossings with little change in the overall stress. Note how in graph (a), the radial structure emerges when both distances and the graph structure are considered.

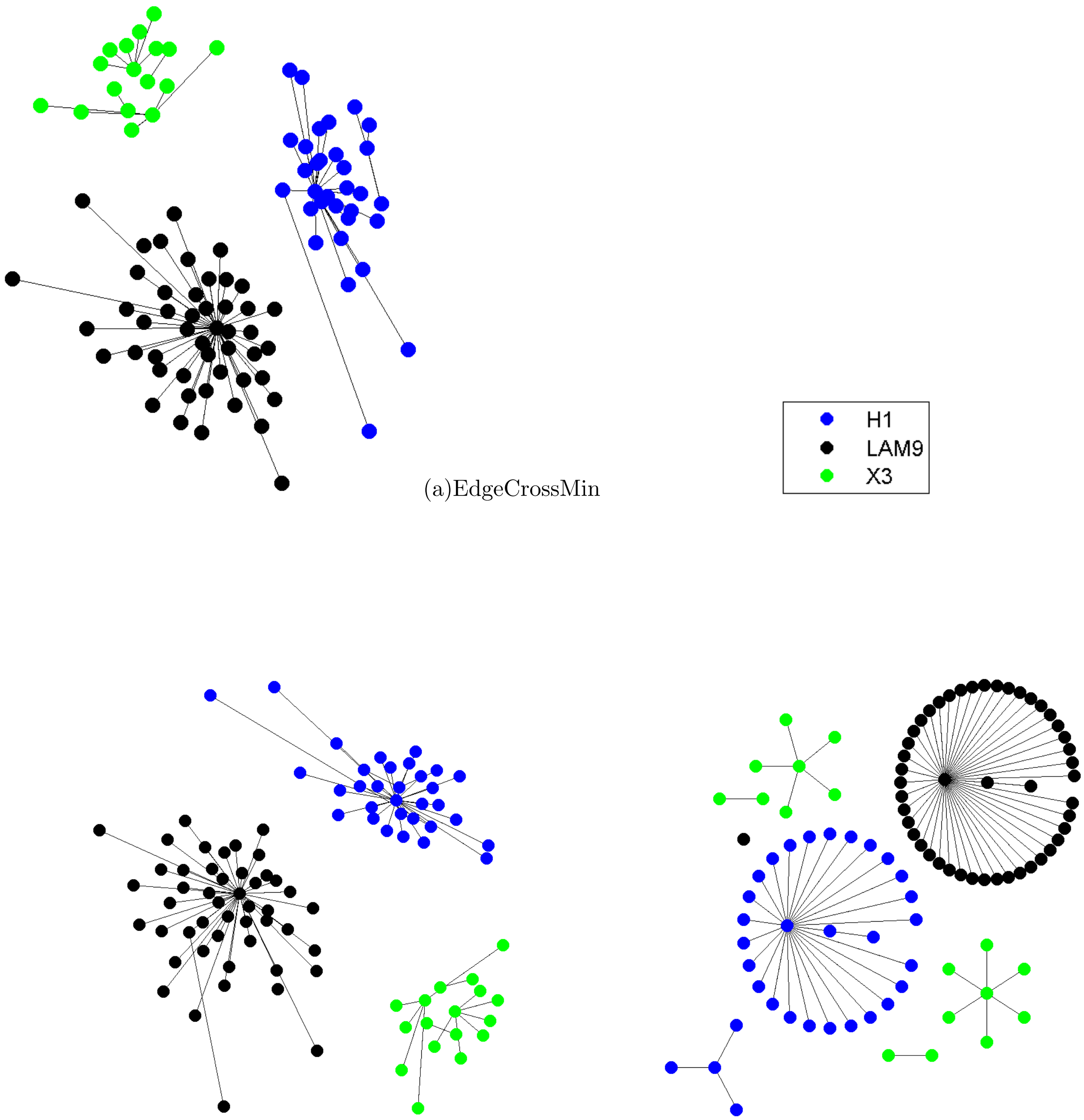


(a)EdgeCrossMin

(b)Neato

(c)Twopi

Figure 3: The *M. africanum* lineage is divided into two distinct sublineages. However, the distinction between the two sublineage is not visible in the graph (c) produced using the radial graph drawing algorithm Twopi. Graph (b), drawn using GraphViz Neato (stress majorization), clearly shows the separation but is difficult to understand because of edge crossings. Graph (a) drawn using the proposed approach eliminates all edge crossings with little change in the overall stress.

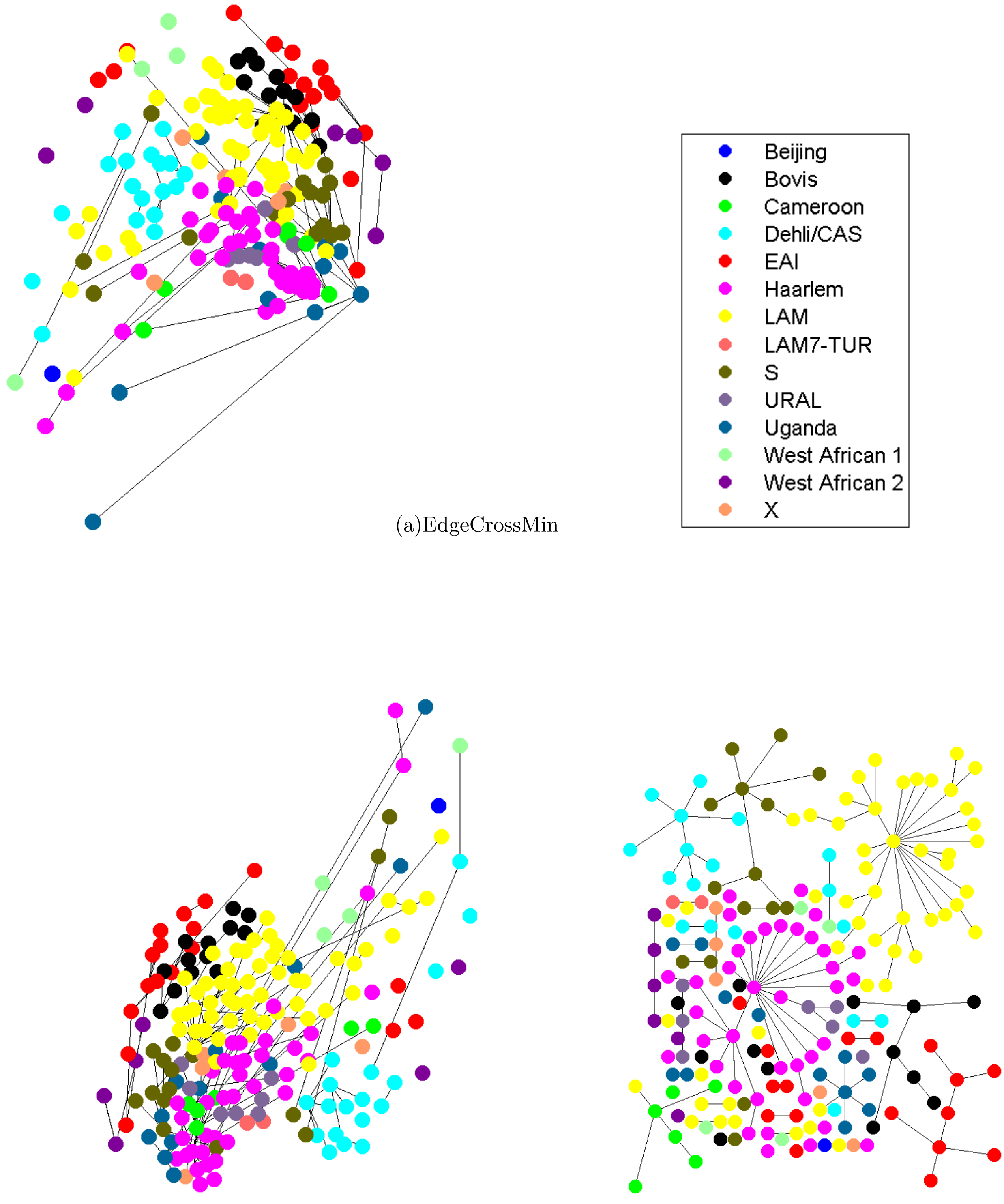


(a)EdgeCrossMin

(b)Neato

(c)Twopi

Figure 4: Graphs showing the Haarlem, X, and LAM sublineages of MTBC drawn with (a)EdgeCrossMin, (b)Neato and (c)Twopi. The proposed method-EdgeCrossMin eliminates all edge crossings and shows the most genetically relevant arrangements within the sublineages.



(a)EdgeCrossMin

(b)Neato

(c)Twopi

Figure 5: Visualization of strains from the MIRUVNTRplus database [?]. Graph (c), which does not exploit the distances between all nodes, does not display the data in a genetically sensical manner for disconnected components. The Neato graph (b) clearly shows the cohesiveness of the lineages, but the spoligotype structure as represented by the graphs can be hard to follow. In (a), the graph with penalized edge crossing, the spoligoforest is nicely displayed. Almost all edge crossings are eliminated. The few that remain would cause great perturbations in the stress. The penalized edge-crossing function allows one to trade off the degree of proximity preservation and edge crossing removal.