Minimum Linear Arrangements

Rafael Andrade, Tibérius Bonates, Manoel Câmpelo, Mardson Ferreira

ParGO - Research team in Parallel computing, Graph theory and Optimization
Department of Statistics and Applied Mathematics
Master and Doctoral Program in Computer Science
Federal University of Ceará

IX LAGOS, Marseille, 11-15 September, 2017

Summary

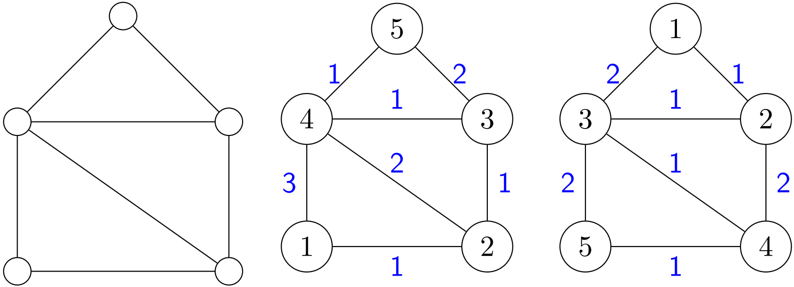
Problem definition

- Let $G = (V, E)$ be a simple and undirected graph with set of vertices V and set of edges E.
- The MinLA problem consists in assigning a permutation $\{\pi_1, \pi_2, \ldots, \pi_{|V|}\}$ of $\{1, 2, \ldots, |V|\}$ to the nodes of G, with a one-to-one correspondence, such that the following sum is minimized

$$\sum_{uv \in E} |\pi_u - \pi_v| \,.$$

- MinLA is NP-Hard [Garey et al., 1976].

Example



(a) Connected graph G. (b) A feasible layout. (c) An optimal layout.

Figure: Example of a graph G and layouts of its vertices.

Related works

- [Díaz et al., 2002] analyzed different combinatorial techniques to obtain lower and upper bounds for the problem.
- [Amaral, 2009] presents a model able to find optimal solutions of the problem for dense graphs with up to 23 vertices.
- [Seitz, 2010] proposed a model based on binary distance and from it presents an interesting polyhedral study of the problem.
- [Moeini et al., 2014] propose a model whose interest is to find relaxed solutions of good quality.

Mathematical programming model

Consider:

- a directed complete graph $D = (V, A)$ obtained from $G = (V, E)$, with $A = \{uv \,|\, u, v \in V,\, u \neq v\}$.
- $A(E) \subset A$ the subset of arcs of D where uv, $vu \in A(E) \leftrightarrow uv \in E$.
- π a permutation of $\{1, \cdots, |V|\}$ to be assigned to the vertices of D.

Let x_{uv}, for all $uv \in A$, be a binary variable, where

$$x_{uv} = \begin{cases} 1 & \text{if } \pi_v > \pi_u, \\ 0, & \text{otherwise,} \end{cases}$$

and w_{uv}, for all $uv \in A$, be continuous variables representing the weight of each arc $uv \in A$.

Mathematical quadratic programming model

$$
\begin{aligned}
(P) \quad & \min_{\pi,x,w} \sum_{uv \in A(E)} w_{uv} x_{uv} && && (1)\\
s.t. \quad & x_{uv} + x_{vu} = 1, && \forall\, uv \in A, && (2)\\
& \pi_v - \pi_u \leq w_{uv} + |V|(1 - x_{uv}), && \forall\, uv \in A, && (3)\\
& \pi_v - \pi_u \geq w_{uv} - |V|(1 - x_{uv}), && \forall\, uv \in A, && (4)\\
& 1 \leq \pi_v \leq |V|, && \forall\, v \in V, && (5)\\
& 1 \leq w_{uv} \leq |V| - 1, && \forall\, uv \in A. && (6)\\
& x_{uv} \in \{0, 1\}, && \forall\, uv \in A. && (7)
\end{aligned}
$$

Constraints (2) ensure that exactly one of the orientations of each pair of vertices is selected. Constraints (3) and (4) impose that if arc uv is in the solution, i.e. $x_{uv} = 1$, then $w_{uv} = \pi_v - \pi_u$; otherwise both constraints become redundant for this arc. The remaining constraints bound the variables.

Mathematical linear programming model

We can linearize the (P) model by dropping the x variables in (1):

$$(Q) \quad \min_{\pi,x,w} \sum_{uv \in A(E)} w_{uv} \tag{8}$$

$$s.t. \text{ (2)} - \text{(7)}.$$

Due to the sense of optimization, whenever $x_{uv} = 0$ we have that the corresponding variable w_{uv} is fixed at its lower bound, that is, at 1.

Strengthening the model

Note that both models (P) and (Q) can be strengthened if we replace constraints (3) by

$$\pi_v - \pi_u \leq w_{uv}, \qquad \forall\, uv \in A, \tag{9}$$

and constraints (6) by

$$1 \leq w_{uv} \leq 1 + (|V| - 2)x_{uv}, \qquad \forall\, uv \in A. \tag{10}$$

Valid inequalities I

Proposition 1. *Let* p *be the largest natural number such that* $\sum_{i=1}^{p} i \leq |E|$ *and let* $\bar{p} = |E| - \sum_{i=1}^{p}(|V| - i)$*. A lower bound on the optimal solution value* z *of* (Q) *is*

$$z \geq |E| + \sum_{i=1}^{p}(|V| - i)i + \bar{p}(p + 1). \tag{11}$$

Proposition 2. *For every arc* $uv \in A$ *and for every node* $k \in V \setminus \{u, v\}$*, the following triangle inequality is valid for* (Q)

$$x_{uv} + x_{vk} + x_{ku} \leq 2. \tag{12}$$

Valid inequalities II

Proposition 3. *In any optimal solution to* (Q) *we have*

$$\sum_{uv \in A} w_{uv} = \frac{|V|(|V|-1)(|V|+4)}{6}. \tag{13}$$

Proposition 4. *A valid constraint for* (Q) *is*

$$\sum_{v \in V} \pi_v = |V|(|V|+1)/2. \tag{14}$$

Valid inequalities III

Proposition 5. *Let* π *represent a permutation of* $\{1, \cdots, |V|\}$ *assigned to the vertices of the digraph* D_π*, where arc* uv *is in* D_π *if and only if* $x_{uv} = 1$ *and* $\pi_v > \pi_u$*. The following equality is valid for* (Q)

$$\pi_u + \sum_{uv \in A} x_{uv} = |V|, \quad \forall u \in V. \tag{15}$$

Corollary 1. *If we replace* x_{uv} *by* $1 - x_{vu}$ *in the expression of Proposition 5, we have*

$$\pi_u - \sum_{vu \in A} x_{vu} = 1, \quad \forall u \in V. \tag{16}$$

Valid inequalities IV

Proposition 6. *For every* $uv \in A$ *we must have*

$$w_{uv} + \sum_{tu \in A} x_{tu} \leq |V|. \tag{17}$$

Proposition 7. *For every arc* $uv \in A$ *and for every node* $k \in V \setminus \{u, v\}$*, the following triangle inequality on the arc weights is valid for* (Q)

$$w_{uv} + w_{vu} \leq w_{vk} + w_{kv} + w_{ku} + w_{uk} - 1. \tag{18}$$

Final expanded model

Therefore, the following mixed integer linear programming model is a valid formulation to MinLA:

$$(Q) : \min\{(8) : (2), (4), (5), (7), (9) - (18)\}.$$

Computational results

- We report a summary of numerical results performed on 6 challenging benchmark instances [Amaral, 2009] and 14 new randomly generated instances.
- We compare results for 4 models of MinLA. We implemented all models, including the ones in [Amaral, 2009] and [Moeini et al., 2015] in C++ with IBM CPLEX 12.6.1 Concert Technology in a PC Intel Core i7, 3.40 GHz of 16GB RAM DDR3 - 1333 MHz running Linux 14.04 LTS/64 bits.

Computational results

Instance	$\|V\|$	$\|E\|$	Edge density	$Optimal$
Benchmark instances [Amaral, 2009]				
GraphNug-n-12-t5	12	61	0,92	241
GraphNug-n-15-t5	15	97	0,92	474
GraphNug-n-16-t6	16	116	0,96	629
GraphNug-n-17-t6	17	131	0,96	748
GraphNug-n-20-t5	20	170	0,89	1.076
GraphNug-n-23-t5	23	221	0,87	1.581
New randomly generated instances				
minla-n10-t0.200-s1	10	34	0,75	100
minla-n10-t0.200-s2	10	36	0,80	108
minla-n10-t0.300-s1	10	33	0,73	100
minla-n10-t0.300-s2	10	34	0,75	98
minla-n10-t0.400-s1	10	25	0,55	64
minla-n10-t0.400-s2	10	31	0,68	84
minla-n10-t0.500-s1	10	25	0,55	64
minla-n10-t0.500-s2	10	22	0,48	54
minla-n10-t0.600-s1	10	15	0,33	30
minla-n10-t0.600-s2	10	20	0,44	43
minla-n10-t0.700-s1	10	14	0,31	27
minla-n10-t0.700-s2	10	14	0,31	29
minla-n10-t0.800-s1	10	10	0,22	17
minla-n10-t0.800-s2	10	12	0,26	21

Table: Instances characteristics and their optimal solution value.

Computational results

Instance	Quadratic (P) B&B Nodes	Quadratic (P) Iterations	Quadratic (P) CPU (s)	MILP (Q) B&B Nodes	MILP (Q) Iterations	MILP (Q) CPU (s)	[Amaral, 2009] B&B Nodes	[Amaral, 2009] Iterations	[Amaral, 2009] CPU (s)	[Moeini et al., 2015] B&B Nodes	[Moeini et al., 2015] Iterations	[Moeini et al., 2015] CPU (s)
GraphNug-n-12-t5	781	34.342	2	0	707	0	0	8	4	57	4.240	1
GraphNug-n-15-t5	1.406	106.371	6	582	25.361	3	0	9	15	1.619	197.261	18
GraphNug-n-16-t6	672	243.524	14	1.327	183.700	10	6	366.065	937	150	5.554	2
GraphNug-n-17-t6	74	51.752	9	718	318.165	17	5	321.741	1.300	262	16.881	5
GraphNug-n-20-t5	12.185	2.231.843	273	7.907	770.576	105	8	4.733.536	84.120	107.190	10.825.888	5.826
GraphNug-n-23-t5	6.560	2.684.208	934	6.378	2.965.477	646	3	452.931	8.929	*	*	*

Table: Numerical results for the benchmark instances [Amaral, 2009].

Computational results

Instance	Quadratic (P)			MILP (Q)			[Amaral, 2009]			[Moeini et al., 2015]		
	B&B Nodes	Iterations	CPU (s)	B&B Nodes	Iterations	CPU (s)	B&B Nodes	Iterations	CPU (s)	B&B Nodes	Iterations	CPU (s)
minla-n10-t0.200-s1	555	20.360	1	0	420	1	0	13.911	7	6.564	254.884	8
minla-n10-t0.200-s2	0	736	0	0	403	1	0	10.066	3	1.485	48.310	2
minla-n10-t0.300-s1	2.893	94.248	2	694	19.625	1	0	24.632	9	11.428	430.498	11
minla-n10-t0.300-s2	0	995	0	0	411	0	0	1.137	2	3.039	96.913	3
minla-n10-t0.400-s1	617	27.720	0	436	18.033	1	29	107.932	38	12.782	591.228	13
minla-n10-t0.400-s2	236	7.148	0	0	420	1	0	16.792	9	1.621	59.449	3
minla-n10-t0.500-s1	652	24.852	1	383	14.101	1	13	91.482	24	16.543	771.751	16
minla-n10-t0.500-s2	615	26.315	0	646	24.364	0	21	100.728	28	13.420	694.767	14
minla-n10-t0.600-s1	437	16.900	0	404	13.857	0	24	155.420	40	10.507	427.863	10
minla-n10-t0.600-s2	238	8.875	1	301	9.374	0	0	43.585	15	10.986	505.198	12
minla-n10-t0.700-s1	909	31.399	1	1.073	30.064	1	23	83.820	23	7.866	420.430	8
minla-n10-t0.700-s2	2.045	84.692	2	5.067	150.993	1	23	137.612	36	15.841	1.032.202	18
minla-n10-t0.800-s1	574	21.803	1	1.286	39.166	1	0	51.985	20	1.816	92.542	2
minla-n10-t0.800-s2	355	17.488	0	915	28.758	1	9	23.718	8	2.217	124.721	3

Table: Numerical results for the new instances.

Summary of computational results

Models	Quadratic (P)	MILP (Q)	[Amaral, 2009]	[Moeini et al., 2014]
CPU Time(s)	60,80	27,00	17.275,20	1.170,40
B&B Nodes	3.023,60	2.106,80	3,80	21.855,60
Iterations	533.566,40	259.701,80	1.084.271,80	2.209.964,80

Table: Summary of average results for benchmark instances [Amaral, 2009].

Models	Quadratic (P)	MILP (Q)	[Amaral, 2009]	[Moeini et al., 2014]
CPU Time(s)	0,64	0,71	18,71	8,78
B&B Nodes	723,28	800,35	10,14	8.293,92
Iterations	27.395,07	24.999,21	61.630,00	396.482,57

Table: Summary of average results for new instances.

Conclusion

- Novel compact quadratic and MILP models for the minimum linear arrangement problem.
- The MILP model has a smaller number of variables and constraints than existing models for the problem.
- We propose new valid inequalities that proved to be very useful for solving benchmark MinLA instances.
- Both quadratic and MILP models outperform existing mathematical formulations for this problem.

Future works

- Explore the geometric structure of the permutahedron to strengthen the proposed models.
- Study new valid inequalities to improve the linear relaxation bound.
- Develop a specialized branch and bound algorithm for model (Q).

References I

Amaral, A. (2009).
A mixed 0-1 linear programming formulation for the exact solution of the minimum linear arrangement problem.
Optimization Letters, 3(4):513–520.

Díaz, J., Petit, J., and Serna, M. (2002).
A survey of graph layout problems.
ACM Computing Surveys (CSUR), 34(3):313–356.

Garey, M. R., Johnson, D. S., and Stockmeyer, L. (1976).
Some simplified np-complete graph problems.
Theoretical computer science, 1(3):237–267.

Moeini, M., Gueye, S., and Michel, S. (2014).
A new mathematical model for the minimum linear arrangement problem.
In *ICORES - Angers - France*, pages 57–62.

Moeini, M., Gueye, S., and Michel, S. (2015).
Adjacency variables formulation for the minimum linear arrangement problem.
In *Communications in Computer and Information Science: Operations Research and Enterprise Systems*, volume 509, pages 95–107. Springer International Publishing.

Seitz, H. (2010).
Contributions to the Minimum Linear Arrangement Problem.
PhD thesis, Heidelberg University, Natural Sciences and Mathematics Faculty, Heidelberg, Germany.