Bergische Universität Wuppertal

Fakultät für Mathematik und Naturwissenschaften

Institute of Mathematical Modelling, Analysis and Computational Mathematics (IMACM)

Preprint BUW-IMACM 19/34

Michael Stiglmayr

Layout problems with reachability constraint

December 3, 2019

http://www.math.uni-wuppertal.de

Layout problems with reachability constraint

Michael Stiglmayr

University of Wuppertal, Wuppertal, Germany
stiglmayr@math.uni-wuppertal.de,
WWW home page: http://www.uni-w.de/u9

Abstract. Many design/layout processes of warehouses, depots or parking lots are subject to reachability constraints, i.e., each individual storage/parking space must be directly reachable without moving any other item/car. Since every storage/parking space must be adjacent to a corridor/street one can alternatively consider this type of layout problem as a network design problem of the corridors/streets.
More specifically, we consider the problem of placing quadratic parking spaces on a rectangular shaped parking lot such that each of it is connected to the exit by a street. We investigate the optimal design of parking lot as a combinatorial puzzle, which has—as it turns out—many relations to classical combinatorial optimization problems.

Keywords: combinatorial optimization, network design problem, maximum leaf spanning tree, connected dominating set

1 Introduction

In contrast to articles [1, 2] in the area of civil engineering and architecture investigating the (optimal) design of a parking lot, we focus on the topological layout of the parking lot rather than on issues like the optimal width of parking spaces and streets, one- or two-way traffic, traffic flow, angle of parking spaces to the streets, or irregular shaped parking lots. These modeling assumptions and restrictions allow to formulate a complex practical problem as a combinatorial optimization problem. This model was proposed in the didactic textbook [3] as a combinatorial puzzle. In the Bachelor thesis [4] integer programming formulations were presented and solved using a constructive heuristic.
We search for the optimal layout of a parking lot on a given a rectangular shaped area maximizing the number of individual parking spaces. The rectangular area is subdivided by a square grid such that every cell of the grid represents either an individual parking space or a part of a street. For the reachability constraint we assume that cars can move on that grid only vertically or horizontally. This implies that every individual parking space must be connected by a street to the exit, where "connected" is defined based on the 4-neighborhood N_4 of a grid cell (see Figure 1 for an example). One of the major modeling aspects is thereby the connectivity of street fields to the exit, for which we will present two different formulations. In general there may be cells not neighboring a street, which can

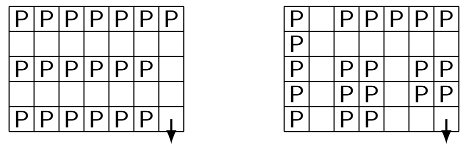

Fig. 1. Sketch of a parking lot layout, individual parking spaces are marked with P. Feasible solution with 19 parking spaces (on the left) and optimal solution with 20 parking spaces (right).

not be used as parking space and are not streets fields connected to the exit. However, such *blocked* cells can be neglected since there is always a solution without blocked cells having the same number of individual parking spaces.

2 Model Formulations

2.1 Formulation Based on the Distance to the Exit

Let (i, j) denote a cell in the grid with $i \in \mathcal{I} = \{1, \dots, m\}$ being its row index, $j \in \mathcal{J} = \{1, \dots, n\}$ its column index. Then, we introduce a binary variable x_{ij} which is equal to one if (i, j) serves as a parking space and zero if (i, j) is part of a street. A method to model the reachability constraint based on the connectivity of street fields is to measure the discrete distance to the exit by a binary variable z_{ij}^k, with $k \in \mathcal{K} = \{1, \dots, n \cdot m\}$ denoting the number of street cells to the exit. Thereby $z_{ij}^k = 1$, if (ij) represents a street field, which is k steps away from the exit, and zero otherwise. Note that many of these variables can be set to zero in advance, if k is smaller than the shortest path to the exit.
Then, the parking lot problem can be written as:

$$\max \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} \tag{1}$$

$$\begin{aligned}
\text{s.t.} \quad z_{ij}^k + x_{ij} &\leq 1 && \forall i, j, k && \text{(1a)}\\
z_{ij}^k &\leq z_{i-1,j}^{k-1} + z_{i,j-1}^{k-1} + z_{i+1,j}^{k-1} + z_{i,j+1}^{k-1} && \forall i, j, k && \text{(1b)}\\
x_{ij} &\leq \sum_{k=1}^{mn} \left(z_{i-1,j}^k + z_{i,j-1}^k + z_{i+1,j}^k + z_{i,j+1}^k \right) && \forall i, j && \text{(1c)}\\
\sum_k z_{ij}^k &\leq 1 && \forall i, j && \text{(1d)}\\
z_{m,n}^0 &= 1 && && \text{(1e)}\\
x_{ij}, z_{ij}^k &\in \{0, 1\} && \forall i, j, k && \text{(1f)}
\end{aligned}$$

The constraint (1a) ensures that the distance values $z_{ij}^k = 0$ for all k if the cell i, j is a parking space, i.e., the distance to the exit is only measured on and along the streets. A street cell can only have a distance of k to the exit if one of its neighboring cells has a distance of $k-1$ to the exit (constraint (1b)) and one cell can not have two different distances to the exit (constraint (1d)). Note that (1a) and (1d) can be merged to $\sum_k z_{ij}^k + x_{ij} \leq 1 \; \forall i,j$. Constraint (1c) states that any parking space requires one neighboring street field, i.e., a cell with a distance to the exit. The large number of binary variables to formulate the reachability constraint, $\mathcal{O}(n^2m^2)$, makes this problem difficult for branch and bound solvers.

2.2 Network Flow Based Formulation

Consider the movement of the cars to leave the parking lot as a network flow. To model this approach we identify each cell of the grid by one node and connect a node (i,j) with a node (rs) if (rs) is in the 4-neighborhood of (i,j), i.e., $(rs) \in N_4(ij) := \{(i-1,j-1),(i-1,j+1),(i+1,j-1),(i+1,j+1)\}$. Since every cell is the potential location of a parking space, we set the supply of every node to one unit of flow, and associate with the exit node (in our instances node (mn)) a demand of $m\,n-1$. Then, nodes without inflow represent parking spaces, all other nodes represent street fields. For this network flow based formulation we need two types of variables: continuous flow variables and binary decision variables.

$$x_{(ij),(rs)} \in \mathbb{R}_+ \quad \text{flow between node } (ij) \text{ and node } (rs)$$

$$z_{(ij)} = \begin{cases} 1 & \text{if node } (ij) \;\text{ has not zero inflow} \\ 0 & \text{otherwise} \end{cases}$$

$$\min \sum_{i=1}^{n}\sum_{j=1}^{m} z_{(ij)} \tag{2}$$

$$\text{s.t.} \quad \sum_{(rs)\in N_4(ij)} x_{(ij),(rs)} - \sum_{(rs)\in N_4(ij)} x_{(rs),(ij)} = 1 \qquad \forall (ij) \setminus (mn) \tag{2a}$$

$$\sum_{(rs)\in N_4(mn)} x_{(mn),(rs)} - \sum_{(rs)\in N_4(mn)} x_{(rs),(mn)} = -m\,n+1 \tag{2b}$$

$$\sum_{(rs)\in N_4(ij)} x_{(rs),(ij)} \leq M \cdot z_{(ij)} \quad \forall (ij) \tag{2c}$$

$$x_{(ij),(rs)} \in \mathbb{R}_+ \qquad (rs) \in N_4(ij) \tag{2d}$$

$$z_{(ij)} \in \{0,1\} \tag{2e}$$

Equations (2a)–(2b) are classical flow conservation constraints. The big-M constraint (2c) couples the inflow to the decision variable z_{ij}: If the sum of incoming flow of a node i,j is zero, $z_{(ij)}$ can be set to zero. Otherwise, if there is a non zero inflow, $z_{(ij)} = 1$. Setting $M := m \cdot n - 1$ does not restrict the amount of inflow if $z_{(ij)} = 1$.

3 Properties of the Parking Lot Problem

3.1 Upper Bound and Geometrically Implied Constraints

We will in the following investigate theoretical properties of the parking lot problem, which hold for both problem formulations (1) and (2). Based on the 4-neighborhood obviously every street field serves directly at most three parking spaces, if it is end point of a street, two parking spaces, if it is in the middle of a straight street and one parking space, if it is a T-junction. Since every additional end point of a street is associated with one T-junction we obtain the following upper bound on the number of parking spaces:

$$\#\text{ parking spaces} \leq \frac{2}{3}\, n\, m$$

Besides this global upper bound on the number of parking spaces, the grid structure of the problem allows to state several geometrically implied constraints which are all satisfied by at least one optimal solution.

no streets along the border A street along the border of the parking lot can be moved in parallel one step to the interior of the parking lot, where each street field can serve directly more than one parking space.

street field next to the exit The exit field has only two neighboring fields, only one of which can be a street field in a optimal solution.

street field on the border of every rectangle In the border of every rectangular shaped region of size larger than 3×3 is at least one street field.

street field in every row/column cut Every row $1 < i \leq m$ and every column $1 < j \leq n$ contains at least one street field.

3.2 Tree Structure

Theorem 1. *For any instance of* (2) *there is an optimal solution in which the edges with positive flow value form a spanning tree.*

Proof. Consider a basic feasible solution of the minimum cost flow problem given by the supply and demand values of (2). The edges with strictly positive flow form a tree of the grid graph, since there are no capacity constraints and consequently all non-basic edges have a flow value of zero. Furthermore, the tree is spanning (every basic edge has positive flow), since every node has a supply of one flow unit.

Definition 1 (see e.g., [5]). *Let $G = (V, E)$ be a graph. A* connected dominating set *in G is a subset $U \subseteq V$ for which the following two properties hold*

- *connected: $\forall u_1, u_2 \in U$ there is a path $P = (u_1, \ldots, u_2) \subset U$ from u_1 to u_2*
- *dominating: $\forall v \in V \setminus U\ \exists u \in U$ such that $(u, v) \in E$*

Theorem 2 ([6]). *Let $n := |V|$ and $d := |U|$ be the cardinality of a minimum connected dominating set U, then $\ell = n - d$ is the maximal number of leafs of a spanning tree in $G = (V, E)$.*

Proof. Let U be a minimum connected dominating set. Then there exists a tree T in U, and all nodes in $V \setminus U$ can be connected as leafs to T, consequently $\ell \geq n - d$. Contrary, let $T = (V, E')$ be a spanning tree in $G = (V, E)$ and L the set of leafs of T. Then $V \setminus L$ is a connected dominating set. Thus, $\ell = n - d$

Identifying the leafs with the individual parking spaces and the street fields with a connected dominating set, the maximum leaf spanning tree problem maximizes the number of individual parking spaces, while the minimum connected dominating set minimizes the number of street fields. Independently, Reis et al. [7] proposed a flow based formulation of the maximum leaf spanning tree problem which is equivalent to (2). Alternative formulation of the maximum leaf spanning tree problems are presented in [8].

Theorem 3 ([9]). *The maximum leaf spanning tree problem is $\mathcal{NP}$-complete even for planar graphs with maximum node degree 4.*

Proof. By reduction from *dominating set*, which can be reduced form *vertex cover*.

The parking lot problem is, thus, a special case of an $\mathcal{NP}$-complete optimization problem. In contrast to general maximum leaf spanning tree problems the parking lot problem has a very regular structure, such that this complexity result does not directly transfer.

4 Heuristic Solution Approach

In [4] a constructive heuristic is proposed, which is based on the use of building blocks of three row or columns, respectively (see Figure 2). The parking lot is

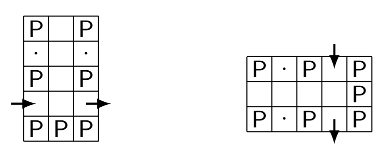

Fig. 2. Illustration of column- and row-wise building blocks.

filled row- or column-wise with blocks of three rows/columns, where the last block of rows/columns has one additional street field at the exit. If the number of rows n is not a multiple of three, one or two rows remain, which can be used for one or $m - 1$ additional parking spaces. Analogously, in the case of column building blocks. Based on the number of rows and columns the performance of the row- and column-wise building blocks differs. This constructive heuristic works best if the number of rows/columns is a multiple of three, since the building blocks achieve the theoretical upper bound of $\frac{2}{3}$. See Figure 3 for a suboptimal heuristic solution in comparison to the optimal solution.

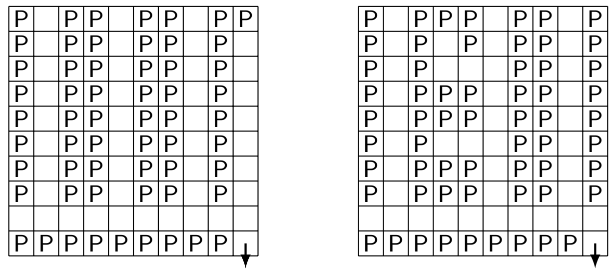

Fig. 3. Example of a suboptimal heuristic solution consisting of column building blocks for 10×10 with 58 parking spaces (left), optimal solution with 60 parking spaces (right).

5 Conclusion

We presented two integer programming formulations of the parking lot problem and focused thereby in particular on the reachability constraint. The first model (1) based on a distance along the streets to the exit is intuitive but requires many binary variables. However, this formulation allows to limit the distance to the exit which could be relevant, e.g., in evacuation scenarios. In the second model (2), which is based on network flows, the distances to the exit are not encoded. Possible extensions of it could, e.g., balance the flow on streets and thus avoid congestions.

References

1. R. Bingle, D. Meindertsma, W. Oostendorp, and G. Klaasen. Designing the optimal placement of spaces in a parking lot. *Mathematical Modelling*, 9(10):765 – 776, 1987.
2. Akmal S. Abdelfatah and Mahmoud A. Taha. Parking capacity optimization using linear programming. *Journal of Traffic and Logistics Engineering*, 2(3), 2014.
3. Ferdinand Verhulst and Sebastian Walcher, editors. *Das Zebra-Buch zur Geometrie*. Springer, 2010.
4. Julia Kleinhans. Ganzzahlige Optimierung zur Bestimmung optimaler Parkplatz-Layouts. Bachelor thesis, Bergische Universität Wuppertal, 5 2013.
5. Ding-Zhu Du and Peng-Jun Wan. *Connected Dominating Set: Theory and Applications*, volume 77 of *Springer Optimization and Its Applications*. Springer, 2013.
6. Robert James Douglas. NP-completeness and degree restricted spanning trees. *Discrete Mathematics*, 105(1):41 – 47, 1992.
7. M. F. Reis, O. Lee, and F. L. Usberti. Flow-based formulation for the maximum leaf spanning tree problem. *Electronic Notes in Discrete Mathematics*, 50:205 – 210, 2015. LAGOS'15 – VIII Latin-American Algorithms, Graphs and Optimization Symposium.
8. Tetsuya Fujie. The maximum-leaf spanning tree problem: Formulations and facets. *Networks*, 43(4):212–223, 2004.
9. M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.