Maximal Parking Lot Utilisation

Billy Jeremijenko, Yiwen Jiang, Joshua Morris

October 2021

Table of Contents

Table of Contents

The problem

Given a parking lot, we represent the parking lot as an $N \times M$ grid where each grid cell may have an obstacle.

We require a model to find a layout, an allocation of parking and driving fields, which maximises the number of available parking spaces in the parking lot.

The problem

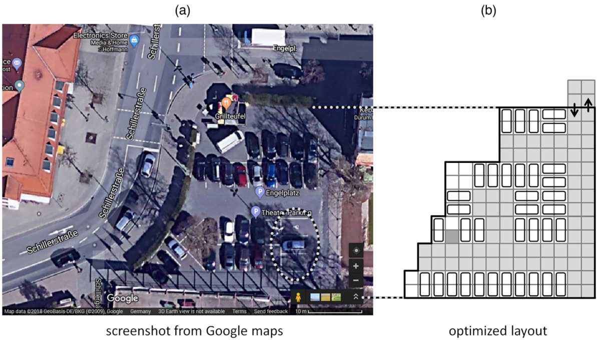


Figure: An example of a parking lot begin represented by a grid [SWB21]

The paper

The paper [SWB21] that we have built upon considers grid cells of varying resolutions. The $5m \times 5m$ grid cells allow for a simple formulation as each parking field and driving field is contained within one grid cell.

However, at finer resolutions such as $2.5m \times 2.5m$ a parking space corresponds to two grid cells, which requires a more complex formulation, but allows for a closer to optimal solution.

The flow model at a finer resolution has a large solution space due to directional variables and solved on an example $35m^2$ data set in 6 hours. Our formulation solved on the same data in only 2 minutes, and allows for arbitrarily fine grid cells.

Table of Contents

Sets and Data

Sets

S_1 the set of grid cells on the border of the grid

S_2 the set of grid cells not on the border of the grid

$S = S_1 \cup S_2 = \{(i, j) : i \in N, j \in M\}$ the set of grid cells

Data

e the column of the entrance

$b_{ij} = 1$ if grid cell $(i, j) \in S$ is blocked

M a big integer

Variables

Variables

$x_{ij} = 1$ if $(i, j) \in S$ is a parking field

$y_{ij} = 1$ if $(i, j) \in S$ is a driving field

$f_{ij}^{\rightarrow} \in \mathbb{R}$ horizontal flow from field (i, j) to $(i, j+1)$ if $f_{ij}^{\rightarrow} > 0$, or from $(i, j+1)$ to (i, j) otherwise

$f_{ij}^{\downarrow} \in \mathbb{R}$ vertical flow from field (i, j) to $(i+1, j)$ if $f_{ij}^{\downarrow} > 0$, or from $(i+1, j)$ to (i, j) otherwise

Constraints

The entrance is accessible by a driving field

$$y_{0e} + y_{1e} = 2$$

In S_1 exactly one grid cell is enabled (the entrance, see previous constraint)

$$1 = \sum_{(i,j) \in S} (x_{ij} + y_{ij}) - \sum_{(i,j) \in S_2} (x_{ij} + y_{ij}) = 1$$

Parking fields are accessible by driving fields

$$x_{ij} \leq \sum_{(i',j') \in N(i,j)} y_{i'j'}, \forall (i,j) \in S$$

Each square serves at most a single purpose

$$x_{ij} + y_{ij} + b_{ij} \leq 1, \forall (i,j) \in S$$

Flow Constraints

Only driving fields send or receive flow

$$|f_{ij}^{\rightarrow}| \leq M \cdot y_{ij},\ \forall (i,j) \in S$$

$$|f_{ij}^{\rightarrow}| \leq M \cdot y_{i(j+1)},\ \forall (i,j) \in S$$

$$|f_{ij}^{\downarrow}| \leq M \cdot y_{ij},\ \forall (i,j) \in S$$

$$|f_{ij}^{\downarrow}| \leq M \cdot y_{(i+1)j},\ \forall (i,j) \in S$$

Every driving field has a net flow of at least one

$$y_{ij} \leq f_{ij}^{\rightarrow} + f_{ij}^{\downarrow} - f_{i(j+1)}^{\rightarrow} - f_{(i+1)j}^{\downarrow},\ \forall (i,j) \in S_2$$

Net flow of driving fields does not exceed one

$$-f_{0e}^{\downarrow} \leq \sum_{(i,j)\in S_2} y_{ij}$$

Objective

Objective

Maximise the number of parking spaces, each parking field contains two

$$\text{maximize} \sum_{(i,j)\in S} 2 \cdot x_{ij}$$

Network flow formulation

We omit the formulation at a finer resolution, which utilises multiple variables to represent the direction in which a parking lot is facing, due to time constraints.

Furthermore, the model is slow to solve and made redundant by our own model which solves significantly faster.

We have implemented the flow model at this finer resolution and the outcome is included it in our results.

We now present our own solution focusing on the finer resolution.

Table of Contents

Our Solution

The network flow model in the referenced paper is slow with larger grid sizes and finer resolutions.

It is also difficult to generalise to finer resolutions, requiring the introduction of multiple variables to indicate direction.

We develop a formulation based on a priori column generation to allow for various resolutions. As well as utilising lazy constraints, instead of flow constraints, to enforce the connectivity of driving fields.

Sets and Data

Sets

$S = \{(i, j) : i \in N, j \in M\}$ the set of grid cells

P the set of possible parking fields

D the set of possible driving fields

P and D are generated a priori and consist of groups of 2×1 and 2×2 grid cells, respectively

Data

$p_{ijp} = 1$ if square $(i, j) \in S$ is contained in parking field $p \in P$

$d_{ijd} = 1$ if square $(i, j) \in S$ is contained in driving field $d \in D$

$b_{ij} = 1$ if square $(i, j) \in S$ is blocked

e the driving field $d \in D$ corresponding to the entrance and exit

Variables

$x_p = 1$ if parking field $p \in P$ is enabled

$y_d = 1$ if driving field $d \in D$ is enabled

Constraints

The entrance and exit are accessible by a driving field

$$y_e = 1$$

Parking fields do not share grid cells

$$\sum_{p \in P} p_{ijp} \cdot x_p \leq 1, \, \forall (i,j) \in S$$

Parking fields are adjacent to a driving field

$$x_p \leq \sum_{d \in N(p)} y_d, \, \forall p \in P$$

Constraints

Each grid cell serves at most a single purpose

$$\sum_{p \in P} p_{ijp} \cdot x_p + \sum_{d \in D} \frac{1}{4} \cdot d_{ijd} \cdot y_d + b_{ij} \leq 1, \forall (i,j) \in S$$

Each driving field is adjacent to a driving field

$$y_d \leq \sum_{d' \in N(d)} y_{d'}, \forall d \in D$$

Objective

Maximise the number of parking fields, that is

$$\text{maximise} \sum_{p \in P} x_p$$

Lazy Constraints

This formulation allows for a solution which has multiple contiguous regions of driving fields which are disconnected from each other, resulting in a solution in which parking fields are not accessible from the entrance.

Given an optimal solution y_i to the master problem with disconnected regions $\mathcal{C} = C_1 \cup C_2 \cup \cdots \cup C_k$ add the following constraint

$$y_d \leq \sum_{d' \in N(C)} y_d, \forall C \in \mathcal{C}, \forall d \in D$$

Table of Contents

Results

We run our models on the same data as used for demonstration purposes in the paper, which was based on real world data. For the finer grid cells, our model solves in 1.87 minutes which is faster than the 5.96 hour solve time of the paper.

Model	Time (s)	Iterations
R1 Flow	4.81	22
R1 Lazy	0.32	13
R2 Flow	21458.08	2672
R2 Lazy	112.97	58

Table: Results for Models from Paper and Lazy Constraints

Resolution One Optimal Solution

The optimal solution for resolution one fits 38 parking spaces

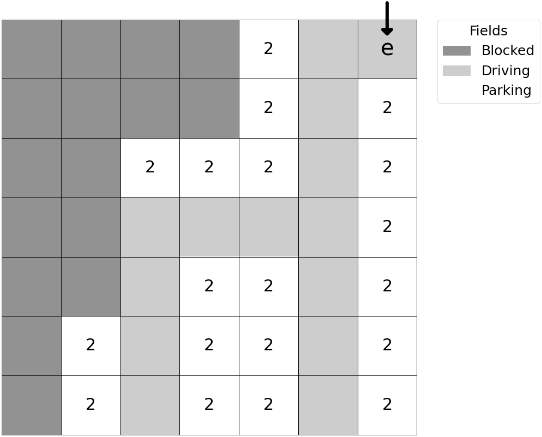


Figure: Optimal layout for resolution one

Resolution Two Optimal Solution

The optimal solution for resolution two fits 40 parking spaces

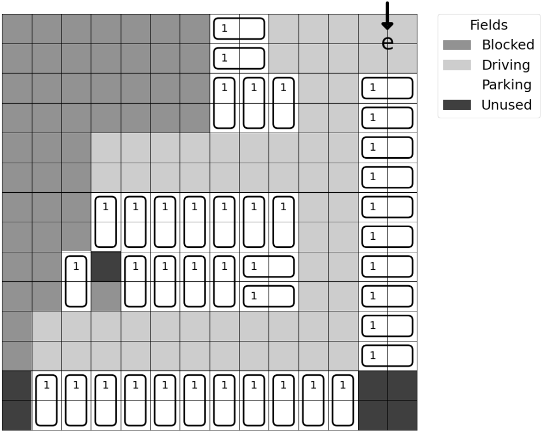
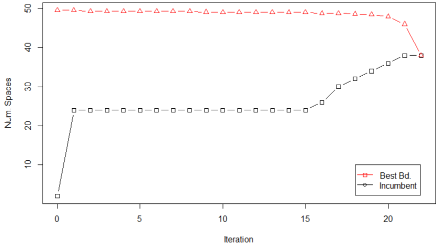
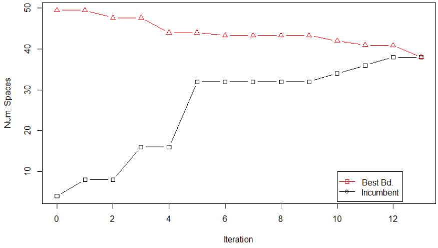


Figure: Optimal layout for resolution two

Coarse Resolution Evolution



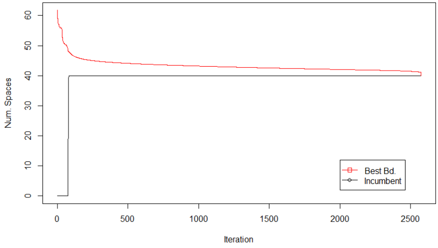
(a) Network Flow



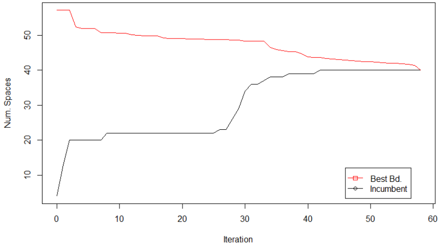
(b) Lazy Constraints

Figure: Evolution of resolution one models

Fine Resolution Evolution



(a) Network Flow



(b) Lazy Constraints

Figure: Evolution of resolution two models

Bibliography

Konrad Stephan, Felix Weidinger, and Nils Boysen, *Layout design of parking lots with mathematical programming*, Transportation Science **55** (2021), no. 4, 930–945.