**Resolution 1 Flow Model**

**Sets**

the set of grid cells on the border

the set of grid cells not on the border

the entire grid

**Data**

the column of the entrance (note that the problem is symmetric with rotation so we can restrict the entrance to the “top” of the grid without loss of generality)

if is blocked

a big integer

**Variables**

if is a parking field

if is a driving field

horizontal flow from driving field to the right if positive, left if negative

vertical flow from driving field below if positive, up if negative

**Objective**

Maximize the number of parking spaces

**Constraints**

The entrance is accessible by a driving field:

On the edge exactly one grid cell is on:

Parking fields are accessible by a driving field:

where is the set of grid cells adjacent to

Each square serves at most one purpose

Connectivity of street fields by network flow:

notice we are restricting the total net flow to any driving field to be at least one, which is guided to the entrance.

**Resolution 1 Lazy**

**Sets**

the set of grid cells

**Data**

the column of the entrance (note that the problem is symmetric with rotation so we can restrict the entrance to the “top” of the grid without loss of generality)

if is blocked

**Variables**

if is a parking field

if is a driving field

**Objective**

Maximize the number of parking spaces

**Constraints**

The entrance is accessible by a driving field:

Parking fields are accessible by a driving field:

where is the set of grid cells adjacent to

Each square serves at most one purpose

Each driving field is adjacent to at least one other driving field

note that while this constraint is technically redundant with the lazy constraint, it reduces the solution space at solve time so we leave it in.

*Lazy Constraints*

With the constraints as they are, it is possible to get a solution with a contiguous region of driving fields that isn’t accessible by the entrance. We would like to restrict the solution to include only a single contiguous driving region. Given an optimal solution with a set of disconnected regions , add the following constraint

*Sketch Proof*

**Resolution 2 Lazy**

**Sets**

the set of grid cells

set of possible parking squares – pairs of grid cells *a priori column generation*

set of possible driving squares – come in fours *a priori column generation*

**Data**

1 if square is on in parking square

1 if square is on in parking square

1 if square is blocked

the driving field corresponding to the entrance

**Variables**

1 if parking square is enabled

1 if driving field is enabled

**Constraints**

Entrance is accessible by a driving field

Parking fields do not overlap

Parking fields are accessible by a driving field

Each square serves at most one purpose

Each driving field is adjacent to at least one other driving field

note that while this constraint is technically redundant with the lazy constraint, it reduces the solution space at solve time so we leave it in.

**Objective**

*Lazy Constraints*

Given an optimal solution with a set of disconnected regions , add the following constraint