1. Magic Square Problem

- Given an integer n, find an n x n square such that
  - the sum of all rows, all columns, and both main diagonals are the same 'magic number' M
  - all values {1,2,…, n^2} are used
- The magic number is $M = \frac{n(n^2+1)}{2}$.
- Formulate a CP model and implement it in AIMMS
  - Can you find a 10 x 10 Magic Square?

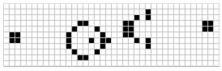
2. Maximum Density Still Life

Conway's Game of Life

- Given infinite orthogonal grid
- Each cell has 8 neighbors and can be either dead or alive

Rules:

- Any live cell with fewer than two live neighbors dies, as if caused by under-population
- Any live cell with two or three live neighbors lives on to the next generation
- Any live cell with more than three live neighbors dies, as if by overcrowding
- Any dead cell with exactly three live neighbors becomes a live cell, as if by reproduction



Maximum Density Still Life (cont'd)

'Still Life':

- Cell configuration that is not modified by the transition rules

Maximum density still life

- Given grid size n by n
- What are the maximum number of live cells in the grid that form a still life?
- Note that the boundary cells must remain dead

maximum density 3 by 3 still life

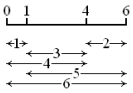
Formulate a CP model and implement it in AIMMS

- Can you find an optimal 6 by 6 still life?

3. Golomb Ruler Problem

- Given an integer m, we define a ruler of m marks with respective positions $x_1, x_2, \ldots, x_m$
  - wlog assume that $x_1 = 0$ and $x_i < x_j$
  - (largest known optimal solution for m = 26)
- A Golomb ruler has distinct pairwise distances between the marks, i.e.,

  $\{ x_j - x_i \mid 1 \le i < j \le m \}$ are all distinct



- Golomb ruler problem: given m, find a Golomb ruler with minimum length

Formulate a CP model and implement it in AIMMS

- Can you find an optimal Golomb ruler for m=9?

4. Spatially Balanced Latin Squares

- Given integer n, a *Latin square* of order n is an n x n matrix such that each number in {1, 2,…, n} appears exactly once in each row and each column
- For each distinct pair of numbers i, j in {1, 2,…, n} and a given row r, define the distance in that row as

  dist(i,j,r) = | col(i,r) – col(j,r) |

  where col(i,r) represents the column index of element i in row r
- The total distance of (i,j) is the sum of distances over all rows
- A (row-)*spatially balanced Latin square* is a Latin square in which all pairs (i,j) have the same total distance

	SBLS of order 5				
	1	2	3	4	5
1	3	1	4	2	5
2	4	5	1	3	2
3	2	4	3	5	1
4	1	2	5	4	3
5	5	3	2	1	4

	Dist(1,2)
1	2
2	2
3	4
4	1
5	1

Spatially Balanced Latin Squares (cont'd)

Some facts:

- Spatially balanced Latin squares do not exist for $n \mid n\%3 = 1$
- Otherwise, the total distance for each pair is $n(n+1)/3$
- If 2n+1 is prime, polynomial construction exists [Le Bras et al., 2012]
- 'Streamlining' constraints (e.g., Cell(i,j) = Cell(j,i)) may be useful

Formulate a CP model and implement it in AIMMS

- Can you find a SBLS for n=9?

	SBLS of order 5				
	1	2	3	4	5
1	3	1	4	2	5
2	4	5	1	3	2
3	2	4	3	5	1
4	1	2	5	4	3
5	5	3	2	1	4

	Dist(1,2)
1	2
2	2
3	4
4	1
5	1

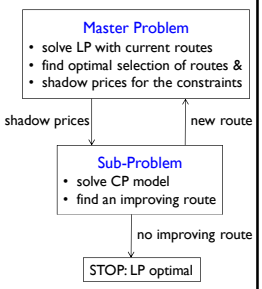
5. CP-Based Column Generation

See slides on "Operations Research in CP" for a description

- Vehicle routing application
- Methodology

AIMMS components:

- Restricted master problem to select routes (LP)
- Subproblem to generate new routes (CP)
- Column generation procedure



Assumptions and suggestions

For simplicity of this exercise, we assume that

- the loading times are 0 for each client
- there are no time windows or precedence constraints given

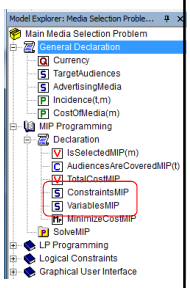
This allows the 'makespan' of the schedule to accurately reflect the length of the route

Start with the AIMMS project in routing.aimmspack on http://www.andrew.cmu.edu/user/vanhoeve/summerschool

(this project also contains a full CP model to compare with)

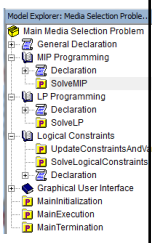
Implementation Guidelines

- Each model declares its own variables, constraints, …
  - We need to replace 'AllConstraints' and 'AllVariables' in Math Program
  - Variables and constraints are grouped together in *sets* for each section
    - make these 'subset of AllVariables' and 'subset of AllConstraints'
  - Variables or constraints may appear in more than one set (i.e., they may be shared among sections)



Implementation Guidelines (cont'd)

- When combining multiple models, there is no longer one single 'MainExecution' to solve the math program
  - Instead, each model is solved by its own solving procedure
  - The 'MainExecution' can still be used, for example to combine the individual models



Using LP relaxation from IP model

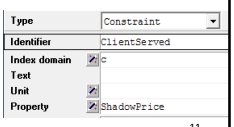
- If the master problem is an IP model (here with binary variables), then we can solve the LP relaxation as

```
solve MasterProblem where type = 'rmip';
```

here, "rmip" stands for "relaxed MIP"

- To obtain the shadow price from a constraint:
  - 'release' the shadow price via 'property' of the constraint
  - use elsewhere in model as

```
ClientServed.ShadowPrice(c)
```



Structure of Procedure in AIMMS

```
generate initial routes;
repeat
  solve Master Problem;
  solve Subproblem;
  break when ReducedCost > tolerance;          ! cannot find improving route
  ! add new route to the Master Problem
  Routes += Card(Routes) + 1;                  ! Integer set of route indices
  for c do                                     ! clients c
    clientInRoute( c, Card(Routes) ) := isSelected(c);   ! isSelected is CP variable
  endfor;
PageRefreshAll;                                ! for visualization
endrepeat;
```