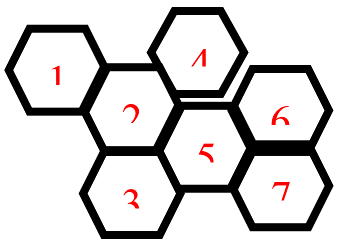
CEE 5290/COM S 5722/ORIE 5430 Heuristic Methods for Optimization
Fall 2012
Channel Allocation for Cellular Networks

Congratulations! You have been appointed the design engineer for Big Red Wireless, Inc. This start-up company, which seeks to provide cellular service in Ithaca and surrounding areas, has just been allocated a total of 50 user channels by the FCC, each occupying a bandwidth of 30 kHz contiguously from 1830.0 MHz to 1831.5 MHz. Your job description requires you to decide the optimal allocation of these channels to each cell.



Based on surveys, Big Red Wireless has decided to divide the service area into 7 cells, located as shown on the left. You have determined that an average of 166 users (out of a total customer base of 2880) are to be expected to be talking on the cell phone at any given time. This average traffic is distributed across the 7 cells as described by the vector **T** = [32 26 14 32 18 20 24].

Based on the power level settings of the base stations and the arrangement of the cells, you have come up with the following simple 7x7 interference matrix **I** to model the co-channel and co-cell constraints:

$$\mathbf{I} = \begin{pmatrix} 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 2 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 \end{pmatrix}$$

The diagonal elements I(j,j) model the co-cell constraints. Say the channels are numbered sequentially from 1 to 50. If channel c is occupied in cell j, then the next channel that can be used in the same cell must be at least distance I(j,j) away in the frequency spectrum. Thus for the given interference matrix, this next channel must be at least channel number c+2. The non-diagonal elements I(j,k), $j \neq k$, model the co-channel constraint. If I(j,k) = 0, the same channel can be used in both cells j and k. If I(j,k) = 1, the two cells must use different frequency channels.

You have been asked to come up with an appropriate channel allocation that minimizes the number of unsatisfied customers. The number of unsatisfied customers is, of course, related to the number of constraint violations. Being a Cornell graduate, and having taken the Heuristics Optimization course, you know this problem is NP-complete and have decided to use heuristic search methods like Genetic Algorithms, Simulated Annealing, and Tabu Search for it. The following information will prove useful:

Decision Variables: You need to search over the space of possible channel allocations. This will be represented by the 7x50 binary matrix **A**, which represents a Channel Allocation/Occupancy Table. The element A(j,k) = 1 if channel k is allocated to cell j, and 0 otherwise. It is possible to restrict the search space by only considering those allocations where the j^{th} row of **A** sums to **T(j)**.

In other words, allocate exactly as many channels to cell j as the average traffic for that cell. The size of the search space is therefore

$$\binom{350}{166} = \frac{350!}{166!\,(350-160)!}$$

To simplify the representation for use with heuristic search algorithms, it is advisable to concatenate all the rows of **A** together into a row vector of length 350 as discussed in class.

Neighborhood Definition: For Simulated Annealing and Tabu Search you may test out neighborhood schemes and tabu lists like the ones you have encountered previously in the course. However, there is one important thing to keep in mind -- that the neighbors generated preserve the number of ones in each row of the **A** matrix. If you invert one of the zero bits in generating a neighbor, you must invert a one somewhere else in that row so that the number of ones is still the same (and apply a similar procedure when inverting the one bits). For Genetic Algorithms the standard mutation operator modified slightly to incorporate this row-cardinality preservation should work just fine, but the crossover operation needs a little thought. One clean way of doing this is to treat each row of **A** as a single gene, and perform crossover only at the end-points of each such row. In other words the crossover operator recombines whole genes without disruption.

Cost Function: Say a1 = A(j1,k1) and a2 = A(j2,k2). We say a1 and a2 are conflicting assignments if a1 and a2 are 1, and one of the following holds:

- if (j1 = j2) and |k2 - k1| < I(j1,j1) (Co-cell constraint)
- if (j1 ≠ j2) and |k2 - k1| < I(j1,j2) (Co-channel constraint)

The cost for a given assignment **A** is the total number of such conflicting assignments. The total number of such pairs a1, a2 for this problem is equal to $\binom{166}{2}$=13695 . Hence the cost values could potentially range from 0 to 13695. To determine the cost for a given **A** matrix, you will need to test out all such pairs and add the number of conflicts. The goal of the optimization will be to minimize the number of conflicts.

For the best solution **A*** you find, give the total number of conflicts, and say how many users from the average of 166 can actually be supported by the system. To determine this, you will need to revoke one of the two assignments for each conflict.