CS124	Games

We can represent various situations of conflict in life in terms of *matrix games*. For example, the game shown below is the *rock-paper-scissors* game. The Row player chooses a row strategy, the Column player chooses a column strategy, and then Column pays to Row the value at the intersection (if it is negative, Row ends up paying Column).

$$\begin{array}{c} \\ r \\ p \\ s \end{array} \begin{array}{c} \begin{array}{ccc} r & p & s \end{array} \\ \begin{pmatrix} 0 & -1 & 1 \\ 1 & 0 & -1 \\ -1 & 1 & 0 \end{pmatrix} \end{array}$$

Games do not necessarily have to be symmetric (that is, Row and Column have the same strategies, or, in terms of matrices, $A = -A^T$). For example, in the following fictitious *Clinton-Dole* game the strategies may be the issues on which a candidate for office may focus (the initials stand for "economy," "society," "morality," and "tax-cut") and the entries are the number of voters lost by Column.

$$\begin{array}{c} \\ e \\ s \end{array} \begin{array}{c} \begin{array}{cc} m & t \end{array} \\ \begin{pmatrix} 3 & -1 \\ -2 & 1 \end{pmatrix} \end{array}$$

We want to explore how the two players may play "optimally" these games. It is not clear what this means. For example, in the first game there is no such thing as an optimal "pure" strategy (it very much depends on what your opponent does; similarly in the second game). But suppose that you play this game repeatedly. Then it makes sense to *randomize*. That is, consider a game given by an $m \times n$ matrix G_{ij}; define a *mixed strategy* for the row player to be a vector $(x_1, \ldots, x_m)$, such that $x_i \geq 0$, and $\sum_{i=1}^m x_i = 1$. Intuitively, x_i is the probability with which Row plays strategy i. Similarly, a mixed strategy for Column is a vector $(y_1, \ldots, y_n)$, such that $y_j \geq 0$, and $\sum_{j=1}^n y_j = 1$.

Suppose that, in the Clinton-Dole game, Row decides to play the mixed strategy $(.5, .5)$. What should Column do? The answer is easy: If the x_i's are given, there is a *pure strategy* (that is, a mixed strategy with all y_j's zero except for one) that is optimal. It is found by comparing the n numbers $\sum_{i=1}^m G_{ij} x_i$, for $j = 1, \ldots, n$ (in the Clinton-Dole game, Column would compare .5 with 0, and of course choose the smallest —remember, the entries denote what Column pays). That is, *if* Column knew Row's mixed strategy, s/he would end up paying the smallest among the n outcomes $\sum_{i=1}^m G_{ij} x_i$, for $j = 1, \ldots, n$. On the other hand, Row will seek the mixed strategy that *maximizes this minimum;* that is,

$$\max_x \min_j \sum_{i=1}^m G_{ij} x_i.$$

This maximum would be the best possible *guarantee* about an expected outcome that Row can have by choosing a mixed strategy. Let us call this guarantee z; what Row is trying to do is solve the following LP:

$$\begin{array}{llll} \max z & & & \\ z & -3x_1 & +2x_2 & \leq 0 \\ z & +x_1 & -x_2 & \leq 0 \\ & x_1 & +x_2 & = 1 \end{array}$$

Symmetrically, it is easy to see that Column would solve the following LP:

$$\begin{array}{llll} \min w & & & \\ w & -3y_1 & +y_2 & \geq 0 \\ w & +2y_1 & -y_2 & \geq 0 \\ & y_1 & +y_2 & = 1 \end{array}$$

The crucial observation now is that *these LP's are dual to each other*, and hence have the same optimum, call it V.

Let us summarize: By solving an LP, Row can guarantee an expected income of at least V, and by solving the dual LP, Column can guarantee an expected loss of at most the same value. It follows that this is the uniquely defined optimal play (it was not *a priori* certain that such a play exists). V is called *the value of the game*. In this case, the optimum mixed strategy for Row is $(3/7, 4/7)$, and for Column $(2/7, 5/7)$, with a value of $1/7$ for the Row player.

The existence of mixed strategies that are optimal for both players and achieve the same value is a fundamental result in Game Theory called *the min-max theorem*. It can be written in equations as follows:

$$\max_x \min_y \sum x_i y_j G_{ij} = \min_y \max_x \sum x_i y_j G_{ij}.$$

It is surprising, because the left-hand side, in which Column optimizes last, and therefore has presumably an advantage, should be intuitively smaller than the right-hand side, in which Column decides first. Duality equalizes the two, as it does in max-flow min-cut.