Background: Linear Programming

Goal: Solve a system of linear inequalities
and in real-valued variables while maximizing
subjecting some linear objective function.

Example: | Cutting | Assembly | Profit/Unit |
-----------|----------|------------|
| x          |          |            |
| y          |          |            |

| Sort 1 | 25      | 60 | 68 | 30 ≤ x |
| Sort 2 | 75      | 60 | 34 | 40 ≤ x |
|        | ≤ 450   | ≤ 480 | ≤ 476 | max \( 290 \) |

Goal: Find number of pieces/tons of sorts 1 \((x)\) and 2 \((y)\) produced per day and that the resource constraints are met and the profit is maximized.

Formulation: \( x \geq 0, \ y \geq 0 \)

\[ 25 \cdot x + 75 \cdot y \leq 450 \]
\[ 60 \cdot x + 60 \cdot y \leq 480 \]
\[ 68 \cdot x + 34 \cdot y \leq 476 \]
\[ \text{maximize} \ 30 \cdot x + 40 \cdot y \]
\((1)-(4): \text{Feasible solutions}\)
\((5): \text{Objective function}\)

Def: A linear program (LP) is standard
form consists of

- \( n \) real-valued variables \( x_i \)
- \( m \) coefficients \( b_i \)
- \( m \) constants \( c_j \)
- \( m \) coefficients \( a_{ij} \)
- \( m \) inequalities \( c_j \leq \sum_{i=1}^{n} a_{ij} x_i \)

\((\text{for } j = 1, \ldots, m)\)
- an objective function \( \sum_{i=1}^{n} b_i x_i \)
  to be minimized.
Remark: Maximize instead of minimize:

- Drop the signs of all $b_j$'s.
- Equality: $x + y \leq c$ if $c - x > 0$.
- $x + y + z = c$.

LP solving algorithms: Usually, one uses the
Simplex algorithm (cost-an expander);
Simplex algorithm is still often preferred in practice
over existing polynomial algorithms.

Encoding of finite 2SG MSNE as LP
Let $G = \langle N, (\mathcal{A}_i), (u_i) \rangle$ onde
- $N = \{1, 2\}$
- $\mathcal{A}_1, \mathcal{A}_2$ are finite
- $u_i(x, \beta) = -u_i(x, \beta)$ for $x \in \Delta(\mathcal{A}_i)$,
  $\beta \in \Delta(\mathcal{A}_j)$.

Maximin Theorem: $NE \Rightarrow$ pair HH.

- Pair of HH $\Rightarrow$ NE
- Mixed strategies $\Rightarrow$ some NE ex.

Hence, to find a MSNE, look for pairs of
(mixed-strategy) HH.
Assume that player 1 seeks a kHH $x_i$.

For each $x_i \in \Delta(\mathcal{A}_1)$ of player 1:
- Determine utility under player 2's best response
  sufficient to ensure pair response.

Maximize over these choices.

LP constraints:

- $x_i(a_i) \geq 0$ for $a_i \in \mathcal{A}_i$
- $\sum_{a_i \in \mathcal{A}_i} x_i(a_i) = 1$ for $i = 1, 2$.

For $u_i(x_i, b)$
- $\sum_{a_i \in \mathcal{A}_i} x_i(a_i) \cdot u_i(a_i, b) 
  \text{const. in } \mathbb{R}$

Maximize $\sum_{a_i \in \mathcal{A}_i} x_i(a_i) \cdot u_i(a_i, b)$.
• A solution to this LP is a PPF for player 1.
• A solution to a similar LP for player 2 is a PPF for player 2.

Example: Matching problem

<table>
<thead>
<tr>
<th></th>
<th>H</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

LP for player 1:

\[
\alpha_s(H) \geq 0, \quad \alpha_s(T) \geq 0 \\
\alpha_s(H) + \alpha_s(T) = 1 \\
1 \cdot \alpha_s(H) - 1 \cdot \alpha_s(T) \geq u \\
-1 \cdot \alpha_s(H) + 1 \cdot \alpha_s(T) \geq U - u
\]

Maximize in subject to the four constraints.

Solve \( \alpha_s(H) = \alpha_s(T) = \frac{1}{2} \)

Remark: Alternating (but weaker) encoding using minimization instead of maximization possible. LP with regularities:

\[ U_s(\alpha_s, \beta_s) \leq u \text{ for all } \alpha_s, \beta_s \in \Theta_s \] and "minimize \( U \)" as objective function.

Next step: Do the same thing for non-two-person games.

Instead of an LP, we use Linear Complementarity Problems (LCP):

• In LCP, there is no objective function.
• In LCP, one has so-called complementarity constraints: for two vectors of variables, \((x_1, \ldots, x_n)\) and \((y_1, \ldots, y_n)\), there are constraints \( x_i \cdot y_i = 0 \) (for \( i = 1, \ldots, n \)).
Proposition: A mixed strategy profile \((\alpha, \beta)\) with payoff profile \((u, v)\) is a MSNE in \(G\) iff it exists a solution to the above LCP with variables \(u, v, \alpha(a_i), \ldots, \alpha(a_n), \beta(b_j), \ldots, \beta(b_L)\).

Proof: \(\Rightarrow\): Let \((\alpha, \beta)\) be a MSNE with payoff \((u, v)\). By the support lemma, for each player and for each strategy in the support of his mixed strategy, this is a best response to the other player’s mixed strategy. Therefore, constraints \(\otimes\) are satisfied.

Constraints \(\otimes\) (in bold) are totally satisfied by MSNE strategies.

\(\Leftarrow\): Assume, we have a solution to the LCP. Because of \(\otimes\), \(\alpha\) and \(\beta\) will be mixed strategies. For all \(a_i \in A_n\), (either) \(a_i \notin \text{supp}(\alpha)\), or \(a_i \in B_n(\beta)\).

In addition, \(u\) is the best utility player 1 can get against \(\beta\) using a pure strategy. Hence, \(u\) is the utility player 1 gets for his best response against \(\beta\). Similar argument for player 2. \(\Rightarrow\) \((\alpha, \beta)\) have utility payoff \((u, v)\).

Noise approach to solving LCPs:

1. Enumerate all pairs of possible supports: 
   \((2^n - 1) \cdot (2^n - 1)\) such pairs.

2. For each pair \((\text{supp}(\alpha), \text{supp}(\beta))\), simplify / count to LCP as follows:

   Replace conditions of the form
   \[ \alpha(a_i)(u - U_n(a_i, \beta)) = 0 \]

   by
   \[ \left\{ \begin{array}{ll}
   u - U_n(a_i, \beta) = 0 & \text{if } a_i \in \text{supp}(\alpha) \\
   \alpha(a_i) = 0 & \text{if } a_i \notin \text{supp}(\alpha)
   \end{array} \right. \]

   Solve they for player 2. Tied objects, find \(\beta_{\text{MSNE}}\).
Then, we have a LP. We can solve this using any LP solver and use solutions to LPs as solutions to the original LP.

Levin-Howard algorithm is a direct way of solving such games.