League of Legends Ban–Pick as a Min-Cost Flow Problem

Hanbyul (Han) Lee (he/him)

PhD Student, Applied Mathematics
Department of Mathematical and Statistical Sciences
University of Colorado Denver

November 30, 2025

hanbyul.lee@ucdenver.edu

Draft pick, but make it linear programming.

Motivation: Ban–Pick as an Optimization Problem

Why model ban–pick?

- Ban–pick phase determines the lineup before the game starts.
- Conceptual data we have:
  - Roles $R = \{\text{TOP, JGL, MID, ADC, SUP}\}$,
  - Available champions C and bans B,
  - Utilities $u_{r,c}$ for champion c on role r.
- *Question:* given $u_{r,c}$, bans, and eligibility, what is the optimal lineup for our team?
- We encode this as a structured min-cost flow problem.

LoL champion select screen (one lineup per role, no duplicates).

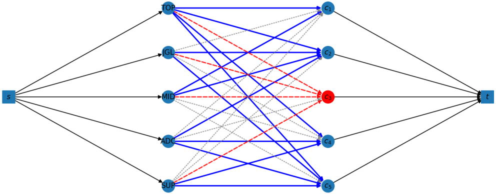
Model Ingredients & Network Structure

Data

- Roles $R = \{\text{TOP, JGL, MID, ADC, SUP}\}$, $|R| = 5$.
- Champion set $C \subseteq C_{\text{all}}$; team bans $B \subseteq C$.
- Eligibility $1_{r,c} \in \{0, 1\}$: 1 if champion c can be played on role r.
- Utility matrix $u \in \mathbb{R}^{R \times C}$; $u_{r,c}$ = performance of c on r.

Flow network

- Nodes: $V = \{s\} \cup R \cup C \cup \{t\}$.
- Arcs: $A_{sR} = \{(s, r) : r \in R\}$, $A_{RC} = \{(r, c) : r \in R, c \in C, 1_{r,c} = 1, c \notin B\}$, $A_{Ct} = \{(c, t) : c \in C\}$, $A = A_{sR} \cup A_{RC} \cup A_{Ct}$.
- Capacities: $u_a = 1$ for all $a \in A$.
- Supplies/demands: $b_s = |R|$, $b_t = -|R|$, $b_v = 0$ for $v \notin \{s, t\}$.
- Costs: $c_{s,r} = 0$, $c_{r,c} = -u_{r,c}$, $c_{c,t} = 0$.



Layered network: source–roles–champions–sink.

Min-Cost Flow Formulation

Decision variables

$$x_{s,r}\ (r \in R), \quad x_{r,c}\ ((r,c) \in A_{RC}), \quad x_{c,t}\ (c \in C), \quad 0 \le x_a \le 1\ \forall a \in A.$$

Objective

$$\min_x \sum_{(r,c)\in A_{RC}} (-u_{r,c})\, x_{r,c} \quad \Longleftrightarrow \quad \max_x \sum_{(r,c)\in A_{RC}} u_{r,c}\, x_{r,c}.$$

Flow-balance constraints

$$\begin{aligned}
\sum_{r\in R} x_{s,r} &= |R|, && \text{(source)}\\
\sum_{c:(r,c)\in A_{RC}} x_{r,c} - x_{s,r} &= 0, && \forall r \in R \text{ (roles)}\\
x_{c,t} - \sum_{r:(r,c)\in A_{RC}} x_{r,c} &= 0, && \forall c \in C \text{ (champions)}\\
\sum_{c\in C} x_{c,t} &= |R|, && \text{(sink).}
\end{aligned}$$

Interpretation

- Exactly $|R|$ units of flow go from s to t.
- Unit capacities enforce one champion per role and no duplicate champions.

Toy Example: Optimal Lineup

Toy utility matrix

$$u = \begin{array}{c|ccc} & G & D & A \\ \hline \text{TOP} & 8 & 7 & 4 \\ \text{MID} & 5 & 3 & 9 \end{array}$$

Feasible lineups (no duplicates)

- (TOP G, MID A): total utility $8 + 9 = 17$.
- (TOP D, MID A): total utility $7 + 9 = 16$.
- (TOP G, MID D): total utility $8 + 3 = 11$.

Min-cost flow solution

- Chooses (TOP G, MID A).
- Optimal total utility $= 17$, min total cost $= -17$.
- Corresponds to 2 units of integral flow from s to t via G and A.

Parametric Buff: When Does TOP Switch Champions?

Parametric setup

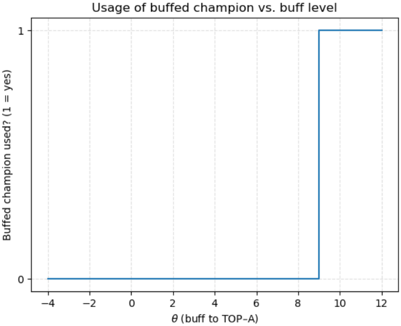
- Fix all utilities except TOP–A:

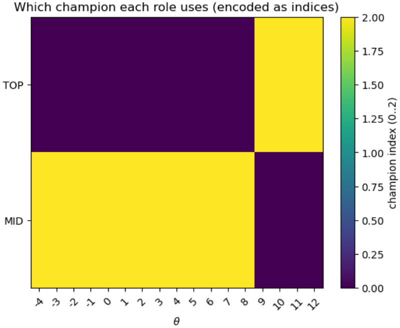
$$u_{\mathrm{TOP},A}(\theta) = u^{(0)}_{\mathrm{TOP},A} + \theta.$$

- For each θ, update costs $c_{r,c}(\theta) = -u_{r,c}(\theta)$ and re-solve the same min-cost flow model.

Behavior in the toy example

- For $\theta \leq 8$: optimal lineup is (TOP G, MID A), total utility $= 17$.
- For $\theta > 8$: lineup switches to (TOP A, MID G), total utility $= 9 + \theta$.
- The value function $-Z^*(\theta)$ is piecewise linear; the lineup changes exactly at $\theta = 8$.





Usage of the buffed champion (left) and role-swap heatmap (right).

Summary & Outlook

Summary

- Modeled LoL ban–pick for a single team as a structured min-cost flow problem.
- Layered network $s \to R \to C \to t$ with unit capacities enforces "one champ per role, no duplicates".
- Objective: choose role–champion assignments that maximize total utility.
- Parametric buff analysis yields piecewise-linear value functions and clear thresholds where the optimal lineup changes.

Implementation & extensions

- Implementable with standard LP / min-cost flow solvers in Python.
- Natural extensions: two-team game, sequential bans/picks, and game-theoretic models (extensive-form, Stackelberg, etc.).

Thanks! Questions or champion balance ideas?

Hanbyul (Han) Lee – hanbyul.lee@ucdenver.edu