THE POPULARITY GAME

François Jenno
University of Montreal
francois.jenno@umontreal.ca

The Minh Nguyen
University of Montreal
the.minh.nguyen@umontreal.ca

April 24, 2023

ABSTRACT

In recent years, streaming services have become increasingly popular, revolutionizing the way people consume media such as movies, TV shows, and music. This trend can be attributed to several factors such as the increasing availability and accessibility of high-speed internet, the growing demand for personalized and on-demand content, and the convenience and affordability offered by streaming services. The COVID-19 pandemic has further accelerated the growth of streaming services as people spent more time at home and turned to streaming to keep themselves entertained. The demand for streaming services has also led to the emergence of new players. As a result, the market for streaming services has become highly competitive, with numerous players vying for a share of the market. We want to model this competition in the form of a Nash game among Stackelberg players. Each player solves a Stackelberg game where a streaming service, as a leader, wants to maximize its profit while its subscribers, as the followers, want to maximize their utilities. For the leader's problem, the objective function will depend on its subscription price that is fixed for simplification, the number of subscribers, and its spending to increase the quality of service such as original content production. On the other hand, each follower will face a knapsack problem where the goal is to choose the services that maximize the total utilities, given a certain budget. The utilities can be subjective to different groups of audiences. Because of the nature of this game, all the leaders are sharing a pool of followers, and a follower can subscribe to many leaders. We also assume that the followers do not compete or interact with each other. Intuitively, the leaders will want to gather as many followers as they can by investing more on their quality. This strategy decreases the objective function at first but it is expected to increase the subscribers and the profit in the long run. Our approach to solve this game is to implement a cutting plane method then solve for the best responses of leaders sequentially in the form of reformulated value function. There are three possible solutions for our algorithm: a Nash equilibrium where no leaders want to change their best responses, a local equilibrium that we define specifically for this game, and a case when no equilibrium found that can potentially provide some insights into the competition.

1 Introduction

Game theory is often useful to model situations in which there exists shared resources or interactions between different optimization components. Depending on the dynamic, many different models have been developed. Those can be regrouped in two main types: Cooperative games and non-cooperative ones. The first kind allows each entity, the players, to cooperate in order to optimize their individual objective functions, while the second ones prohibits such alliances or communications that may benefit players. The latter thus forms a good frame to model competitive dynamics, in which the gain of a player is usually correlated with a certain loss from other players, and can be used to represent the competition in a market where the entities' objective is to maximize profit, based on a certain set of decision variables pertinent to the problem, under a certain set of constraints that may be shared among the entities.

Formally, we define a game as a set of players $\{p_i, i = 1, ...n\}$ where each has to solve its own optimization problem $P_i, i = 1, ..., n$ with objective function $f_i(\{x_{ik}\})$ or utility $U_i(\{x_{ik}\}, \{x_{i'k'}\})$,, where $\{x_{ik}\}$ are decision variables or strategies for player i, under a certain set of constraints $g_i(\{x_{ik}\}) \leq b_{il}$. An optimum can be found when the solution $\{x^*_{ik},\ i = 1, ...n,\ k = 1, ..., h\}$ forms a *Nash equilibrium*, which is attained when no single player has incentive to deviate from their strategies given the other players' current strategies fixed. This is enclosed in the *best response* condition:

$$\left(U_i(\{x^*_{ik}\}, \{x^*_{i'k'}\}) \geq U_i(\{x_{ik}\}, \{x^*_{i'k'}\}) \qquad \forall \{x_{ik}\} \in \{x_{ik}\}^h \right)$$
$$\wedge$$
$$\left(U_i(\{x^*_{ik}\}, \{x^*_{i'k'}\}) \geq U_i(\{x^*_{ik}\}, \{x_{i'k'}\}) \qquad \forall \{x_{i'k'}\} \in \{x_{i'k'}\}^{h'} \right)$$

For bimatrix games for example, this can be solved by similar methods as in linear programming. A different dynamic can exist between the players, such as the case of a Stackelberg model. There, one player constitutes a dominant force (the leader), who sets their decision first, and the other players (the followers) *react* to it by optimizing their own objective function accordingly. This then requires to define a local optimum in a different manner. A *Stackelberg equilibrium* is attained when the followers *best reaction* matches the leader's expected one in their sequential optimization problems, and is usually solved by backward induction.

In a market, businesses interact with their customers via a *Stackelberg* type dynamic, whereas the competition between the former impose a *Nash equilibrium* type of solution for this part. There is a game combining both of these dynamics [1], as portrayed in figure 1.

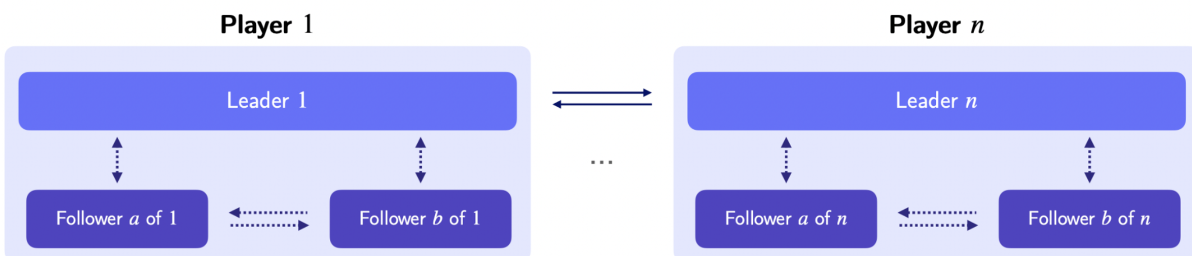

Figure 1: A combined Nash-Stackelberg model. Each player solves an independent Stackelberg game, and then Nash equilibrium are computed between the players.

Here each player solves a Stackelberg game with its followers, and the resulting strategies are optimized in a Nash type non-cooperative game. For the purpose of our model however, the dynamic is slighty different, as each stackelberg game is *not* idependent from the other. Indeed, competing streaming services, which we will model as leaders, interact with the same set of followers, which constitute the population, as can be seen in figure 2.

Various parameters and decisions could be prone to optimization in such a frame. For the leaders, they may be the subscription price p_i, the spending budget s_i, or many others, while for the followers, it could be their personal budget b_j or the combination of streaming services $\{x_{ij}\}$ they want to subscribe to. We will define in the following section exactly how we formulate our problem and any hypothesis or simplification we use, but an important feature to notice is that the followers are made to solve a knapsack problem, where the optimization is over the best discrete combination of variables (here the subscritions) that maximize a certain utility u_{ij} relevant to them (c_{ij}), under certain restraining conditions (here the limited budget b_j). The model can thus be described as a *global* Stackelberg game with multiple leaders, in which the shared followers reaction introduces the competition *implicitly*.

2 Mathematical Formulation

Modeling the game: we define the game as a Nash games among Stackleberg players where:

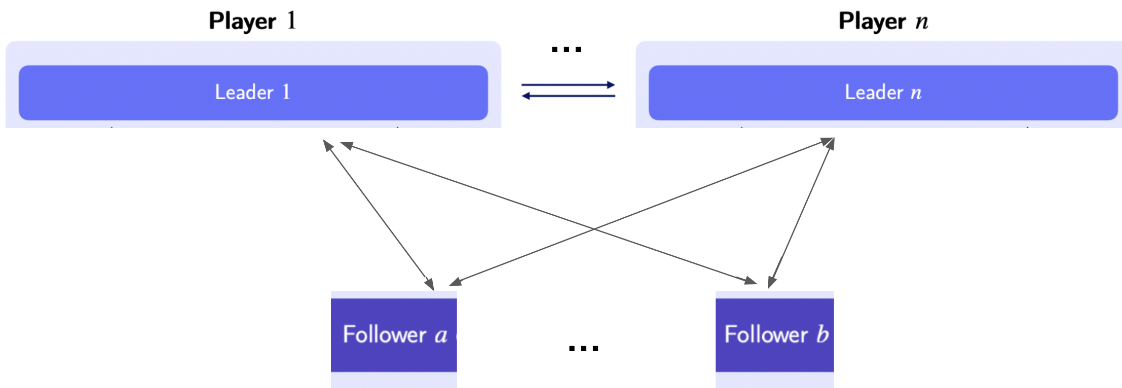

Figure 2: Updated Nash-Stackelberg model. Each player competes for the subscription of the same pool of followers, in a global Stackelberg game.

1. There are n first-mover streaming platforms as leaders. They all share a set of m users as followers at the lower level. These followers do not interact with each other. Leaders and followers hierarchically decide their strategies by solving optimization problems with known parameters.

2. Each follower j solves a knapsack problem where they subscribe to a combination of leaders $x_{1j}, ..., x_{nj} \in \{0, 1\}$ (n decision variables) that will maximize their total utilities. Their utilities for each leader are $u_{1j}, ..., u_{nj}$, which are affected by positive coefficients $c_{1j}, ..., c_{nj}$, positive constants $d_{1j}, ..., d_{nj}$ and each leader's investment on quality $s_1, ..., s_n$. Each follower has to choose the leaders that fit the budget b_j based on the their constant price $p_1, ..., p_n$.

3. Each leader i will try to maximize their profit by investing s_i (1 decision variable) for quality improvement. s_i is projected through a positive coefficient c_{ij} and a positive constant d_{ij}, which can be understood as how the spending affects the utilities u_{ij} of a follower j.

4. In this game, we fix the price p_i of each leader i to simplify the problem. This also reflects the observation that these services in reality rarely change their prices.

Each leader solves problem (P_i)

$$
\begin{aligned}
\max_{s_i, x, u_{ij}, j=1,...,m} \quad & p_i \sum_{j=1}^{m} x_{ij} - s_i && (1)\\
\text{s.t.} \quad & u_{ij} = c_{ij} s_i + d_{ij} \quad j = 1, ..., m && (2)\\
& s_i \geq 0 && (3)\\
& x_{ij} \in \underset{x_{1j},...,x_{nj}}{\text{argmax}} \left\{ \sum_{i=1}^{n} u_{ij} x_{ij} : \sum_{i=1}^{n} p_i x_{ij} \leq b_j, x_{1j}, ..., x_{nj} \in \{0, 1\} \right\} \quad j = 1, ..., m && (4)
\end{aligned}
$$

3 Algorithm

We perform a Value Function Reformulation for the problem (P_i) into (EP_i). $X_j \quad j = 1, ..., m$ is the collection of all feasible combination of services for a follower j. We want to apply cutting plane method by initializing $\overline{X_j} \subseteq X_j \quad j = 1, ..., m$ and adding more points to $\overline{X_j}$, which is equivalent to adding more constraints, after each iteration. We want to solve a relaxation of (EP_i) with $\overline{X_j} \quad j = 1, ..., m$. In terms of notation, a variable or parameter without subscript indicates all possible leaders and/or followers.

Problem (EP_i)

$$\max_{s_i,x,u_{ij},j=1,\dots,m} \quad p_i \sum_{j=1}^{m} x_{ij} - s_i \tag{5}$$

$$\text{s.t.} \quad u_{ij} = c_{ij}s_i + d_{ij} \quad j = 1,...,m \tag{6}$$

$$s_i \geq 0 \tag{7}$$

$$\sum_{i=1}^{n} u_{ij}x_{ij} \geq \sum_{i=1}^{n} u_{ij}\widehat{x_{ij}} \quad \forall \widehat{x_{ij}} \in X_j \quad j = 1,...,m \tag{8}$$

$$\sum_{i=1}^{n} p_i x_{ij} \leq b_j \quad j = 1,...,m \tag{9}$$

$$x_{ij} \in \{0,1\} \quad j = 1,...,m \tag{10}$$

Algorithm:
1: $k = 0$
2: **Input**: Initialized $s^k, u^k, \overline{X}^k$
3: **while true** (This loop gathers all the necessary points for $\overline{X}$ // apply cutting plane method)
4: $\quad k = k + 1$
5: $\quad$ Let $(s_i^k, u_{ij}^k, x^{ki} \quad j = 1,...,m)$ be an optimal solution to (EP_i) for $i = 1,...,n$ given $\overline{X}^{k-1}$ and u^{k-1}
// Note that x^{ki} is x at step k and this solution is specific to leader i
6: $\quad (x^{k*}) = BestReaction(s^k, u^k)$
7: $\quad$ **if** $\sum_{i=1}^{n} u_{ij}^k x_{ij}^{k*} > \sum_{i=1}^{n} u_{ij}^k \widehat{x_{ij}} \quad \forall \widehat{x_{ij}} \in \overline{X_j}$ **for any** $j = 1,...,m$
8: $\quad\quad$ Add constraint $\sum_{i=1}^{n} u_{ij}x_{ij} \geq \sum_{i=1}^{n} u_{ij}^k x_{ij}^{k*}$ to $(EP_i) \quad i = 1,...,n$
// update $\overline{X_j}$ for any j that satisfies the inequality in step 7
9: $\quad$ **end if**
10: **end while** if no new constraints are added. Now we have finalized $\overline{X}$
11: **for some iterations** (This loop will either end after some iterations or end earlier when 2 consecutive x^* are the same)
12: $\quad k = k + 1$
13: $\quad$ Let $(s_i^k, u_{ij}^k, x^{ki} \quad j = 1,...,m)$ be an optimal solution to (EP_i) for $i = 1,...,n$ given $\overline{X}$ and u^{k-1}
14: $\quad (x^{k*}) = BestReaction(s^k, u^k)$
15: $\quad$ **if** $x^{k*} == x^{(k-1)*}$
16: $\quad\quad$ **if** $s^k == s^{k-1}$
17: $\quad\quad\quad$ **return** $x^{(k-1)*}, s^{k-1}, u^{k-1}, p_i \sum_{j=1}^{m} x_{ij}^{(k-1)*} - s_i^{k-1} \quad i = 1,...,n$ **as Nash Equilibrium**
18: $\quad\quad$ **else**
19: $\quad\quad\quad$ **return** $x^{(k-1)*}, s^{k-1}, u^{k-1}, p_i \sum_{j=1}^{m} x_{ij}^{(k-1)*} - s_i^{k-1} \quad i = 1,...,n$ **as Local Equilibrium**
20: $\quad\quad$ **end if**
21: $\quad$ **end if**
22: **end for**
23: **return no Equilibrium** and potentially a certain number of cycled states

This algorithm consists of two main loops: **while** loop and **for** loop. **while** loop gathers all the necessary points for $\overline{X}$ as the application of the cutting plane method. After finalizing $\overline{X}$, we move to **for** loop. Inside this loop, we solve sequentially each player best response given the previous iteration's knowledge of u, then the followers will respond to the new u from all leaders' decisions by a new x^*, as a $n \times m$ matrix. It will try to achieve an equilibrium if certain conditions are met. If not, the loop will end and return no equilibrium after a set number of iterations.

Nash Equilibrium we want to define it as the solution when no leaders want to change their best responses s between iterations. In terms of algorithm, this happens when two consecutive s, and x^* correspondingly, are the same. Once we have $s^k == s^{k-1}$, then $u^k == u^{k-1}$. Since u is one of the input for calculating the best responses' of leaders in the next iteration, this condition $s^k == s^{k-1}$ is enough to guarantee that s, u, x^*, $p_i \sum_{j=1}^{m} x_{ij} - s_i \quad i = 1,...,n$ will stay the same for all iterations afterwards.

Local Equilibrium we want to define it as the solution when the best reaction from followers, x^*, are indifferent between 2 consecutive iterations, while the corresponding best responses from leaders, s, are not. In terms of algorithm, this happens when $x^{k*} == x^{(k-1)*}$ and $s^k != s^{k-1}$. To be specific, we choose the local equilibrium at iteration $k-1$ and the solutions are s^{k-1}, u^{k-1}, $x^{(k-1)*}$, with $p_i \sum_{j=1}^{m} x_{ij}^{(k-1)*} - s_i^{k-1} \quad i = 1, ..., n$. Before explaining the reason why we choose $k-1$ and not k, we start with an observation that given past knowledge of other leaders' response s^{k-1}, u^{k-1}, a leader i has a tendency to increase their spending to s_i^k as they believe they have a chance to win more followers given that the others do not change their spending. This leads to a state of competition (leaders gradually increase spending) until they realize they cannot compete anymore, which results in a new state where they initialize a new set of s, u and restart the competition. We define these states by their corresponding $x*$. All the consecutive iterations that share the same $x*$ would be considered to be in the same states. We choose the first iteration of each state with s^1, u^1, x^{1*} to be the local equilibrium. Let s^k, u^k, x^{k*} to be the solution for any iteration k in the same state. From our previously mentioned observation about leaders' tendency, $s^k > s^1$. Also, $x^{1*} == x^{1*}$ because of the same state. Consider the objective values of (EP_i), $p_i \sum_{j=1}^{m} x_{ij}^{k*} - s_i^k < p_i \sum_{j=1}^{m} x_{ij}^{1*} - s_i^1$ for every leader i. This proves that the first iteration's objective values are the local maximum of that state. Intuitively, once the program moves to a state (the followers' opinions stay the same), the leaders would have to play multiple times to get out of that state. Since this process does not yield the optimal results for all leaders, they have less incentive to change their responses and are content to stay in the first iteration of the state.

No Equilibrium Using the definition of state from local equilibrium, we also observed that a state can consist of only one iteration. As a result, sequentially playing the game multiple times indicates that the program is jumping from one state to another. Although we do not know if the number of total states is limited, we saw a case where the program cycles between two states (that will be discussed in the next section, and tutorial). For now, we set a number of iterations as a stopping criterion to end the program. The program could return the last state's results if a solution is needed. However, we truly believe that instead of picking a state randomly, we should gather all the cycled states, if possible, to have a clear understanding of the whole situation. For example, we have two leaders who are competing directly with each other and two corresponding states cycled by the program. Each state favors a leader by giving them high objective value. If we randomly return a solution just for the sake of it, we unintentionally prioritize a leader, which is against the nature of non-cooperative Nash game where players act selfishly. By knowing all the cycled states, we can make an agreement between two players to potentially have a consensus and net gains. Intuitively, the never-ending game between players resembles the perpetual competition in reality, and finding the middle ground with the knowledge of cycled states could be a great idea.

4 Results and Discussion

We use the Gurobi optimizer to solve both the upper and lower problems, where the former is solved on a restrained set, and the latter on the complete set X. The fixed-value parameters c_{ij}, d_{ij}, b_j and p_i are set randomly with fixed seed. Here are some of the results obtained for $n = 4$, $m = 5$:

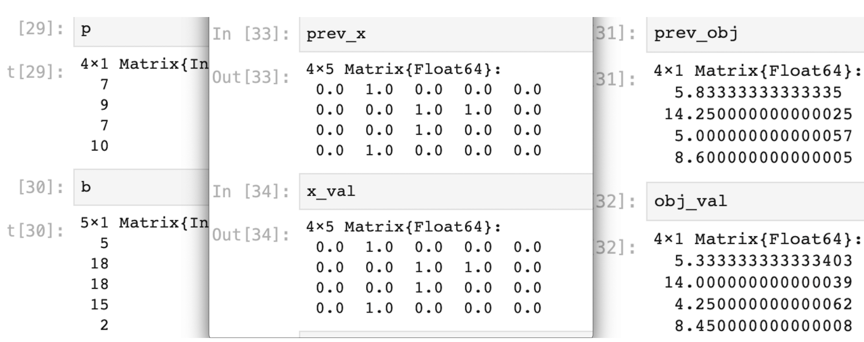

Figure 3: a)On the left: p_i values for the price of leader i and budjet b_j of follower j. b)In the middle: values of subscription decision x_{ij} of follower j to leader i ($x_{ij} \in \{0, 1\}$). Rows i are for leaders and Columns j for followers. c) On the right: Values of the objective function *before* (top) and *after* (bottom) the extra iteration after finding the optimal solution x_{ij}^{k*}.

We can see that for follower 1 whose budget $b_1 = 5$ is such that $b_1 < p_i, \ \forall i = 1, 2, 3, 4$ and thus he hasn't subscribed to any service ($x_{i1} = 0,\ \forall i = 1, 2, 3, 4$). For follower 2, we can see that their highest utility constants c_{i2}, d_{i2} are for leaders $i = 1, 4$, and indeed in the optimal solution $x_{12}, x_{42} = 1$.

While this seems to indicate that the model is correct and that the solver satisfies all the constraints, it does not show that the solution obtained is indeed a local optimum. The stopping criterion we used was such that the best reaction x^*_{k+1} doesn't improve the utility for the followers compared to any other $x_{ij} \in \bar{X}_j$, and thus at the optimal solution, $x^*_k = x^*_{k+1}$, but $f(x^*_{k+1}, s^*_{k+1})$ is not necessarily equal to $f(x^*_k, s^*_k)$, which is the condition for Nash equilibrium. However, this can still satisfy a local optimum if for any feasible direction, $f(x^*_{k+1}, s^*_{k+1}) < f(x^*_k, s^*_k)$. For this, we have implemented an additional fixed-number of iteration loop to evaluate the previous statement. We also use different seeds to test for the different predicted outcomes.

Local Equilibrium We obtain indeed that for the best reaction at an additional iteration $k + 1$, the optimal value for the upper problem *decreases* (figure 3c).

Nash equilibrium This doesn't mean that Nash equilibrium cannot be attained with our method, but only that they can't be guaranteed, as we can see with figure 4, where the objective value remains unchanged for all leaders when fixing the other leaders' decision variables, meaning a Nash equilibrium has been attained.

```
In [48]: prev_obj

Out[48]: 4×1 Matrix{Float64}:
           8.0
          -1.0
          17.00000000000004
           2.220446049250313e-16

In [49]: obj_val

Out[49]: 4×1 Matrix{Float64}:
           8.0
          -1.0
          17.00000000000004
           2.220446049250313e-16
```

Figure 4: Objective values for a *different initialization*, at the optimal solution (top) and after computing an *extra* iteration (bottom). As the values are the same, this is a Nash equilibrium.

No equilibrium Despite this, we notice that when the size of the problem increases, both in the number of leaders n and followers m, as in an increase in the number of decision variables or the values of the parameters, the program doesn't converge to a solution. For a particular initialization seed, we obtain a program that doesn't converge, and instead alternates between 2 solutions $\{x^*_{ij}\}$ and $\{s^*_i\}$, as can be seen in figure 5.

```
In [55]: sum(prev_x.*p, dims=2)-prev_s # prev obj value

Out[55]: 4×1 Matrix{Float64}:
           3.666666666666682
           5.4
          12.078125000220403
           7.375000000000215

In [56]: sum(x_val.*p, dims=2)-s_val # current obj value

Out[56]: 4×1 Matrix{Float64}:
           3.6666666666666803
           5.4
           4.78125000000016
          12.437500000293902
```

Figure 5: Objective values alternating between two sets of values in a instance where the program doesn't converge

Intuitively, this reflects the real-life situation in which companies keep competing and reacting to competitors incessantly, but if we would like to converge to a solution, we can consider including a threshold on the stopping criterion:

$$\left(p_i \sum_{j=1}^{m} x_{ij}^{k+1*} - s_i^{k+1*}\right) - \left(p_i \sum_{j=1}^{m} x_{ij}^{k*} - s_i^{k*}\right) \leq \epsilon \qquad \forall i = 1, ..., n$$

Such an ϵ-threshold can be fixed, or dynamic (its value depends on the recent variations of the decision variables). However, even with such threshold, there is no guarantee that it will satisfy any initial conditions. An alternative would be to consider looping through a fixed number of iterations, or until the variations start to stabilise, i.e. the average of variation in the values of the decision variables approaches 0.

5 Conclusions

In conclusion, we have modeled the competition between streaming services taking into consideration the influence from their subscribers. This game is in the form of a Nash game among Stackelberg players. Our contribution is to analyze the case where all leaders share the same pool of followers without any interaction between followers. In this project, we have proposed an algorithm to solve this game using value function reformulation and cutting plane method, which helps the program to reach a Nash equilibrium. We also define a local equilibrium when the leaders have no incentive to change their best responses in the short term. In the case of no equilibrium found, we provided our insights on the situation as we believe this could be beneficial in real life's application. For the future work, we would like to study in-depth and also develop the solutions for the case of local equilibrium and no equilibrium. We have also made several simplifications such as fixing leaders' prices, and ignoring the interaction between followers. We would like to consider these aspects in our further research.

References

[1] Margarida Carvalho et al. *When Nash Meets Stackelberg*. 2022. arXiv: 1910.06452 [cs.GT].