Computing Stackelberg Equilibrium for Cancer Treatment

Sam Ganzfried
Ganzfried Research
sam.ganzfried@gmail.com

Abstract

Recent work by Kleshnina et al. has presented a Stackelberg evolutionary game model in which the Stackelberg equilibrium strategy for the leading player corresponds to the optimal cancer treatment [4]. We present an approach that is able to quickly and accurately solve the model presented in that work.

1 Introduction and problem formulation

In this section we review the Stackelberg evolutionary game dynamic model of cancer evolution previously studied [4]. There are two players: a follower and a leader. The leader is a physician who selects amounts of two different drugs to use for therapy, m_1 and m_2. The follower is a cancer population consisting of three cell types: 0-type denotes a cell that does not develop resistance to either drug, 1-type cells are resistant to just drug 1, and 2-type cells are resistant to just drug 2. The follower selects a population size for each type, denoted x_0, x_1, x_2, as well as a trait for each type, denoted u_0, u_1, u_2. It is assumed that each of these variables are nonnegative, with the $u_i \in [0, 1]$. It is also assumed that they are all implicitly functions of time t. Note that u_0 does not appear anywhere in the analysis so can be ignored.

For each cell type i there is a fitness function $G_i(u_i, \mathbf{m}, \mathbf{x})$ that the follower is trying to maximize. We assume that the dynamics of the population x are governed by

$$\dot{\mathbf{x}} = \mathbf{G}(t)\mathbf{x}.$$

In order to ensure that we are in equilibrium of the ecological dynamics we must have that $\dot{x_i} = 0$ for $i = 0, 1, 2$. Thus, the follower is selecting u_1 that maximizes G_1, u_2 that maximizes G_2, and x_0, x_1, x_2 that ensure equilibrium of the ecological dynamics. The leader, knowing that the follower will subsequently select their actions in this way, selects m_1, m_2 to maximize a quality of life function $Q(\mathbf{m}, \mathbf{u}, \mathbf{x})$.

Thus, we can formulate the problem of determining the optimal strategies for both players as follows:

$$\begin{array}{rl} \max_{\mathbf{m}^*, \mathbf{u}^*, \mathbf{x}^*} & Q(\mathbf{m}^*, \mathbf{u}^*, \mathbf{x}^*) \\ \text{s.t.} & \dot{x}_i^* = 0 \text{ for } i = 0, 1, 2 \\ & u_i^* \in \arg\max_{u_i} G_i(u_i, \mathbf{m}^*, \mathbf{x}^*) \text{ for } i = 1, 2 \\ & \mathbf{m}^* \geq 0, \mathbf{x}^* \geq 0, 0 \leq \mathbf{u}^* \leq 1 \end{array} \tag{1}$$

This general model is instantiated by the following functional forms for the fitness functions G_0, G_1, G_2 and quality of life function Q. Note that the model presentation is slightly different between the paper [4] and the implementation in the code repository [3]. We will be using the model presented in the code.

$$G_0 = r_{\max}\left(1 - \frac{\alpha_{00}x_0 + \alpha_{01}x_1 + \alpha_{02}x_2}{K}\right) - d - \frac{m_1}{k_1} - \frac{m_2}{k_2}$$

$$G_1 = r_{\max}e^{-g_1u_1}\left(1 - \frac{\alpha_{10}x_0 + \alpha_{11}x_1 + \alpha_{12}x_2}{K}\right) - d - \frac{m_1}{b_1u_1 + k_1} - \frac{m_2}{k_2}$$

$$G_2 = r_{\max} e^{-g_2 u_2} \left(1 - \frac{\alpha_{20} x_0 + \alpha_{21} x_1 + \alpha_{22} x_2}{K}\right) - d - \frac{m_1}{k_1} - \frac{m_2}{b_2 u_2 + k_2}$$

$$Q = Q^{\max} - c \left(\frac{x_0 + x_1 + x_2}{K}\right)^2 - w_1 m_1^2 - w_2 m_w^2 - r_1 u_1^2 - r_2 u_2^2$$

The model has several parameters, whose interpretations are summarized in Table 1. Note that in the code additional parameters a_0, a_1, a_2, a_3 are defined, with

$$\begin{bmatrix} \alpha_{00} & \alpha_{01} & \alpha_{02} \\ \alpha_{10} & \alpha_{11} & \alpha_{12} \\ \alpha_{20} & \alpha_{21} & \alpha_{22} \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & a_1 \\ a_2 & a_0 & a_3 \\ a_2 & a_3 & a_0 \end{bmatrix}$$

Parameter	Interpretation
$r_{\max}$	Max cell growth rate
g_i	Cost of resistance strategy (cell type) i
α_{ij}	Interaction coefficient between cell types i and j
K	Carrying capacity
d	Natural death rate
k_i	Innate resistance that may be present before drug exposure
b_i	Benefit of the evolved resistance trait in reducing therapy efficacy
$Q^{\max}$	Quality of life of a healthy patient
w_i	Toxicity of drug i
r_i	Effect of resistance rate of cell type i
c	Weight for impact of tumor burden vs. drug toxicity/treatment-induced resistance rate

Table 1: Interpretations of model parameters

2 Prior approach

In this section we present the approach described in Github repository created by the authors [3]. They first solve for expressions for x_i in terms of u_i, m_i such that the condition for equilibrium of the ecological dynamics is satisfied. This involves solving a system of three equations $G_i x_i = 0$ with unknowns x_1, x_2, x_3. They calculate the following analytical solution:

$$x_0^* = \frac{K(X_{01} + X_{02} + X_{03})}{r_{\max}(a_0 - a_3)(a_0^2 - 2a_1 a_2 + a_0 a_3)},$$

where X_{01}, X_{02}, X_{03} denote the following quantities:

$$X_{01} = (a_0^2 - a_3^2)\left(-d - \frac{m_1}{k_1} - \frac{m_2}{k_2 + r_{\max}}\right)$$

$$X_{02} = a_1(a_3 - a_0)e^{g_1 u_1}\left(-d - \frac{m_2}{k_2} + e^{-g_1 u_1} r_{\max} - \frac{m_1}{k_1 + b_1 u_1}\right)$$

$$X_{03} = a_1(a_3 - a_0)e^{g_2 u_2}\left(-d - \frac{m_1}{k_1} + e^{-g_2 u_2} r_{\max} - \frac{m_2}{k_2 + b_2 u_2}\right)$$

$$x_1^* = \frac{K(X_{11} + X_{12} + X_{13})}{r_{\max}(a_0 - a_3)(a_0^2 - 2a_1 a_2 + a_0 a_3)}$$

$$X_{11} = a_2(a_3 - a_0)\left(-d - \frac{m_1}{k_1} - \frac{m_2}{k_2} + r_{\max}\right)$$

$$X_{12} = (a_0^2 - a_1a_2)e^{g_1u_1}\left(-d - \frac{m_2}{k_2} + e^{-g_1u_1}r_{\max} - \frac{m_1}{k_1 + b_1u_1}\right)$$

$$X_{13} = (a_1a_2 - a_0a_3)e^{g_2u_2}\left(-d - \frac{m_1}{k_1} + e^{-g_2u_2}r_{\max} - \frac{m_2}{k_2 + b_2u_2}\right)$$

$$x_2^* = \frac{K(X_{21} + X_{22} + X_{23})}{r_{\max}(a_0 - a_3)(a_0^2 - 2a_1a_2 + a_0a_3)}$$

$$X_{21} = a_2(a_3 - a_0)\left(-d - \frac{m_1}{k_1} - \frac{m_2}{k_2} + r_{\max}\right)$$

$$X_{22} = (a_1a_2 - a_0a_3)e^{g_1u_1}\left(-d - \frac{m_2}{k_2} + e^{-g_1u_1}r_{\max} - \frac{m_1}{k_1 + b_1u_1}\right)$$

$$X_{23} = (a_0^2 - a_1a_2)e^{g_2u_2}\left(-d - \frac{m_1}{k_1} + e^{-g_2u_2}r_{\max} - \frac{m_2}{k_2 + b_2u_2}\right)$$

While it is not given in the model formulation, the code assumes that m_1 and m_2 fall in the interval $[\underline{m}, \overline{m}]$, with $\underline{m} = 0$, $\overline{m} = 1$. They discretize the space so that m_1 and m_2 are multiples of $h = 0.1$, and iterate over all combinations, of which there are $\frac{1}{h^2}$. For each combination of values they calculate the optimal values of u_1, u_2 as follows. They assume that the optimal value of u_1 is where $\frac{\partial G_1}{\partial u_1} = 0$, and the optimal value of u_2 is where $\frac{\partial G_2}{\partial u_2} = 0$. They calculate the following expressions for the partial derivatives:

$$\frac{\partial G_1}{\partial u_1} = -g_1 r_{\max} e^{-g_1u_1}\left(1 - \frac{\alpha_{10}x_0 + \alpha_{11}x_1 + \alpha_{12}x_2}{K}\right) + \frac{m_1b_1}{(k_1 + b_1u_1)^2}$$

$$\frac{\partial G_2}{\partial u_2} = -g_2 r_{\max} e^{-g_2u_2}\left(1 - \frac{\alpha_{10}x_0 + \alpha_{11}x_1 + \alpha_{12}x_2}{K}\right) + \frac{m_2b_2}{(k_2 + b_2u_2)^2}$$

To find values of u_1, u_2 for which these derivatives equal zero, they perform an optimization to minimize the quantity $\left|\frac{\partial G_1}{\partial u_1}\right| + \left|\frac{\partial G_2}{\partial u_2}\right|$. They perform this optimization in Matlab using the fminsearchbnd function, using bounds $0 \le u_1 \le 1, 0 \le u_2 \le 1$. Note that the above expressions for x_i^* which are in terms of m_i and u_i are substituted into the expressions for $\frac{\partial G_i}{\partial u_i}$, so that u_i is the only variable in the objective. This procedure results in optimal values $u_1^*(m_1, m_2), u_2^*(m_1, m_2)$ for each combination of values for (m_1, m_2). The values for x_0^*, x_1^*, x_2^* are determined by these values.

Finally, they iterate over all $\frac{1}{h^2}$ of these combinations of values to determine which one maximizes the value of Q. This determines the optimal values of m_1^*, m_2^* for the leader, which in turn determine the optimal strategies for the follower. The overall procedure is summarized in Algorithm 1.

Algorithm 1 Prior approach

for $m_1 = 0$ to 1 by increments of h **do**

  for $m_2 = 0$ to 1 by increments of h **do**

    Calculate values of u_1, u_2 that minimize $\left|\frac{\partial G_1}{\partial u_1}\right| + \left|\frac{\partial G_2}{\partial u_2}\right|$, where $0 \le u_i \le 1$ and the x_i satisfy the equations for the equilibrium of the ecological dynamics.

Calculate the values for (m_1, m_2) out of the $\frac{1}{h^2}$ possibilities for which the corresponding optimal variables maximize the value of Q.

3 Limitations of prior approach

The prior approach has several significant limitations. The first is that it does not check that the calculated values for x_i^* for given values of u_i and m_i are biologically sensible. Since the x_i correspond to populations, they must be nonnegative.

Another limitation is that the coarse discretization of values for m_i means that only a small number of possibilities are considered. This also means that the running time will potentially be large, since we must perform $\frac{1}{h^2}$ separate optimizations.

Another significant limitation is that it is assumed that G_i is maximized when $\frac{\partial G_i}{\partial u_i} = 0$, and the boundary cases when it is maximized at $u_i = 0$ or 1 are ignored.

A final limitation is that the procedure invokes the fminsearch algorithm in Matlab, which is not even guaranteed to find a local minimum, let alone a global minimum.

4 New approach

We now describe our new approach that addresses the limitations of the prior approach. We will formulate a single quadratic program that corresponds to the full optimization problem and solve it using Gurobi's nonconvex MIQCP solver which has a guarantee of global optimality (subject to numerical precision).

First we have the main decision variables x_i, u_i, m_i with $x_i \geq 0$, $m_i \geq 0$, $0 \leq u_i \leq 1$. The objective function Q is a quadratic function of these variables. Next we encode the conditions for equilibrium of the ecological dynamics. We must define several auxiliary variables to do this.

First define $\eta_i = g_i u_i$, and $\tau_i = e^{\eta_i}$ for $i = 1, 2$. For the latter, we use Gurobi's addGenConstrExp function that uses a piecewise linear approximation for the exponential function. We set these variables to be nonnegative. To provide tighter upper bounds we can set $\eta_i \leq g_i$, $\tau_i \leq e^{g_i}$, since $u_i \leq 1$. We next define the auxiliary variable $\gamma_i = \frac{m_i}{k_i + b u_i}$. We can do this by including the quadratic constraint $k_i\gamma_i + b u_i \gamma_i - m_i = 0$ for $i = 1, 2$. Using these variables we can now encode the conditions for equilibrium of ecological dynamics using constraints that are quadratic in the variables.

Next we must encode the conditions that u_i is a maximizer of G_i. To do this we define several additional auxiliary variables. We define $\sigma_i = e^{-g_i u_i}$ by adding in the constraint $\sigma_i \tau_i = 1$. Next we define $\beta_i = \frac{1}{(k_i + b_i u_i)^2}$. We can do this by including the quadratic constraint $\beta_i - b_i^2 u_i^2 - 2k_i b_i u_i - k_i^2 = 0$. Finally we define $\omega_i = \frac{m_i}{(k_i + b_i u_i)^2}$ by including the quadratic constraint $\omega_i \beta_i - m_i = 0$. Using these variables, we can now encode the expressions for $\frac{\partial G_i}{\partial u_i}$ that are quadratic in the variables.

Recall that we are trying to select $u_i \in [0, 1]$ to maximize G_i, for $i = 1, 2$. We can do this by introducing two Lagrange multipliers $\lambda_{i1} \geq 0$, $\lambda_{i2} \geq 0$. Then the KKT optimality condition is equivalent to the following three constraints:

$$\frac{\partial G_i}{\partial u_i}(u_i) + \lambda_{i1} - \lambda_{i2} = 0$$

$$\lambda_{i1}(u_i - 0) = 0$$

$$\lambda_{i2}(u_i - 1) = 0$$

These constraints are all quadratic in the variables and ensure that we find $u_i \in [0, 1]$ that maximizes G_i regardless of whether it is at the boundary or at an interior solution with the derivative equal to zero.

Our full formulation is given below. Here the X_{ij} correspond to the same quantities as before and are just defined to simplify presentation, not as new variables.

$$
\begin{aligned}
\max_{\mathbf{m},\mathbf{u},\mathbf{x}} \quad & Q^{\max} - \frac{c(x_0^2+x_1^2+x_2^2+2x_0x_1+2x_0x_2+2x_1x_2)}{K^2} - w_1m_1^2 - w_2m_2^2 - r_1u_1^2 - r_2u_2^2 \\
\text{s.t.} \quad & x_0 = \frac{K(X_{01}+X_{02}+X_{03})}{r_{\max}(a_0-a_3)(a_0^2-2a_1a_2+a_0a_3)} \\
& X_{01} = \left(a_0^2 - a_3^2\right)\left(-d - \frac{m_1}{k_1} - \frac{m_2}{k_2} + r_{\max}\right) \\
& X_{02} = a_1(a_3 - a_0)\left(-\tau_1 d - \frac{\tau_1 m_2}{k_2} + r_{\max} - \tau_1\gamma_1\right) \\
& X_{03} = a_1(a_3 - a_0)\left(-\tau_2 d - \frac{\tau_2 m_1}{k_1} + r_{\max} - \tau_2\gamma_2\right) \\
& x_1 = \frac{K(X_{11}+X_{12}+X_{13})}{r_{\max}(a_0-a_3)(a_0^2-2a_1a_2+a_0a_3)} \\
& X_{11} = a_2(a_3 - a_0)\left(-d - \frac{m_1}{k_1} - \frac{m_2}{k_2} + r_{\max}\right) \\
& X_{12} = (a_0^2 - a_1a_2)\left(-\tau_1 d - \frac{\tau_1 m_2}{k_2} + r_{\max} - \tau_1\gamma_1\right) \\
& X_{13} = (a_1a_2 - a_0a_3)\left(-\tau_2 d - \frac{\tau_2 m_1}{k_1} + r_{\max} - \tau_2\gamma_2\right) \\
& x_2 = \frac{K(X_{21}+X_{22}+X_{23})}{r_{\max}(a_0-a_3)(a_0^2-2a_1a_2+a_0a_3)} \\
& X_{21} = a_2(a_3 - a_0)\left(-d - \frac{m_1}{k_1} - \frac{m_2}{k_2} + r_{\max}\right) \\
& X_{22} = (a_1a_2 - a_0a_3)\left(-\tau_1 d - \frac{\tau_1 m_2}{k_2} + r_{\max} - \tau_1\gamma_1\right) \\
& X_{23} = (a_0^2 - a_1a_2)\left(-\tau_2 d - \frac{\tau_2 m_1}{k_1} + r_{\max} - \tau_2\gamma_2\right) \\
& -g_i r_{\max}\sigma_i\left(1 - \frac{\alpha_{i0}x_0+\alpha_{i1}x_1+\alpha_{i2}x_2}{K}\right) + b_i\omega_i + \lambda_{i1} - \lambda_{i2} = 0 \text{ for } i = 1,2 \\
& u_i\lambda_{i1} = 0 \text{ for } i = 1,2 \\
& u_i\lambda_{i2} - \lambda_{i2} = 0 \text{ for } i = 1,2 \\
& \eta_i = g_iu_i \text{ for } i = 1,2 \\
& \tau_i = e^{\eta_i} \text{ for } i = 1,2 \\
& k_i\gamma_i + bu_i\gamma_i - m_i = 0 \text{ for } i = 1,2 \\
& \sigma_i\tau_i = 1 \text{ for } i = 1,2 \\
& \beta_i - b_i^2u_i^2 - 2k_ib_iu_i - k_i^2 = 0 \text{ for } i = 1,2 \\
& \omega_i\beta_i - m_i = 0 \text{ for } i = 1,2 \\
& m_i \geq 0 \text{ for } i = 1,2 \\
& x_i \geq 0 \text{ for } i = 1,2 \\
& 0 \leq u_i \leq 1 \text{ for } i = 1,2,3 \\
& 0 \leq \eta_i \leq g_i \text{ for } i = 1,2 \\
& 0 \leq \tau_i \leq e^{g_i} \text{ for } i = 1,2 \\
& \gamma_i \geq 0 \text{ for } i = 1,2 \\
& \sigma_i \geq 0 \text{ for } i = 1,2 \\
& \beta_i \geq 0 \text{ for } i = 1,2 \\
& \omega_i \geq 0 \text{ for } i = 1,2 \\
& \lambda_{ij} \geq 0 \text{ for } i = 1,2 \text{ and } j = 1,2
\end{aligned}
$$

This formulation addresses the limitations of the prior approach. It ensures that all quantities are biologically relevant by imposing nonnegativity constraints on corresponding variables. It allows m_i to take on arbitrary nonnegative values, not a small set of discretized values. It involves solving a single optimization problem instead of $\frac{1}{h^2}$ separate optimization problems. It uses KKT conditions to ensure that the values of u_i that maximize G_i are found regardless of whether they are interior or boundary solutions. And the approach guarantees finding a global optimum since that is guaranteed by Gurobi's nonconvex MIQCP solver.

5 Experiments

We ran experiments with both approaches on a problem instance using the same parameter values as the prior approach [3], which are provided in Table 2. All experiments were done on a single core of a laptop using Windows 11. Experiments with the prior approach used Matlab version 24.1.0.2689473 (R2024a) Update 6 [2], and experiments with the new approach were done using Gurobi version 11.03 [1] with Java version 14.0.2. For the optimizations in the prior approach, Matlab's function fminsearchbnd was called using parameters TolX = 1×10^{-12}, MaxFunEvals = 1000. The results are shown in Table 3. We can see that the prior approach found a solution with a negative value for x_1^*, which is not biologically sensible. The prior approach took nearly five minutes while our new approach took less than two seconds.

Parameter	Value
$r_{\max}$	0.45
g_1	0.5
g_2	0.5
a_0	1
a_1	0.15
a_2	0.9
a_3	0.9
K	10,000
d	0.01
k_1	5
k_2	5
b_1	10
b_2	10
$Q^{\max}$	1
w_1	0.5
w_2	0.2
r_1	0.4
r_2	0.4
c	0.5

Table 2: Parameter values used in experiments

	Prior approach	New approach
m_1^*	0.4	0.40837
m_2^*	0.5	0.46579
u_1^*	0.19015	0.21361
u_2^*	0.3123	0.28554
x_0^*	5634.3774	5749.8474
x_1^*	-360.2658	1.3366
x_2^*	1316.2683	950.5000
Q^*	0.59936	0.59780
Running time (seconds)	282.69	1.65

Table 3: Experimental results for both approaches

6 Conclusion

We presented a new approach for computing Stackelberg equilibrium strategies in a Stackelberg evolutionary game dynamic model of cancer evolution previously studied. Our approach is based on solving a new quadratic program formulation. We noted several limitations of the approach used by prior work which are addressed by our approach. When we compared the approaches on a sample instance our approach ran significantly faster and the prior approach output a solution that is not biologically relevant. As more complex game-theoretic and optimization models are being formulated for problems in biology and cancer treatment in particular, it is important to develop efficient algorithms for accurately solving them. While we focused on one instantiation presented in prior work, our approach is applicable more generally to computing optimal strategies in Stackelberg evolutionary games.

References

[1] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2024.

[2] The MathWorks Inc. Matlab version: 24.1.0.2689473 (R2024a) update 6, 2024.

[3] Maria Kleshnina. `https://github.com/kleshnina/SEGopinion`, 2023.

[4] Maria Kleshnina, Sabrina Streipert, Joel Brown, and Kateřina Staňková. Game theory for managing evolving systems: Challenges and opportunities of including vector-valued strategies and life-history traits. *Dynamic Games and Applications*, 13:1–26, 2023.