Applied Computer Systems
ISSN 2255-8691 (online)
ISSN 2255-8683 (print)
2025, vol. 30, no. 1, pp. 166–173
https://doi.org/10.2478/acss-2025-0018
https://content.sciendo.com

RIGA TECHNICAL UNIVERSITY

Genetic Algorithm for Approximation of Equilibrium Strategies within Finite Space of Actions in Bimatrix Games for Improving Telecommunication Interactions

Vadim Romanuke*
Polish Naval Academy, Gdynia, Poland

***Abstract* – To improve the network interaction step-wise process, a genetic algorithm is suggested for finding more stable solutions in bimatrix games. The algorithm is based on using an approach of successive approximation to an equilibrium situation within a finite space of actions whose size directly depends on the number of game repetitions (network interactions). The algorithm has seven input parameters: the population size, the maximum number of generations, the number of generations for the early stop, the mutation rate, the number of bits per pure strategy, the number of maximum network interactions, and the number of the best chromosomes selected. The algorithm is more efficient for fewer network interactions, when the network peers obtain more stable and consistent strategies that encourage interaction itself rather than resigning from sending any information due to instability. The equilibrium concept is strengthened by introducing a criterion of mutual profitability, which is clearly a good tone and respect in network interactions (particularly, in P2P file-sharing networks). This criterion, expressed as a fitness value equal to negative maximum of potential losses, can be varied to alternatively evaluate the consequence of swerving from a given mixed strategy.**

***Keywords* – approximate equilibrium, bimatrix game, genetic algorithm, network interaction, strategic behaviour.**

I. Equilibrium and Finiteness

In telecommunications, non-cooperative games are used to model and analyse various scenarios where multiple entities (e.g., users, devices, service providers) have competing interests and must make decisions that affect one another [1], [2]. Finite non-cooperative games provide a powerful framework for modelling and analysing strategic interactions in telecommunications [3], [4]. By representing the decisions and payoffs of competing entities (players), these games help in understanding and predicting the behaviour of users and providers [5], [6]. This subsequently optimises resource allocation [7], pricing strategies [8], managing interference [9], peer-to-peer (P2P) file-sharing networks [10], and negotiating quality of service [6], [11].

While non-cooperative game theory helps in understanding dynamics of strategic behaviour and can suggest mechanisms (e.g., reputation systems [12], incentives [13], [14]) to encourage cooperation in P2P networks [10], [15]), the solution even in a finite game may be challenged if it is not of pure strategies. Indeed, any finite non-cooperative game has at least one solution, either in pure or mixed strategies. Commonly such a solution relies on the concept of the Nash equilibrium, by which any player cannot improve one's benefit (i.e., increase one's payoff) by individually retreating from an equilibrium situation. If there is a single Nash equilibrium situation and it is of mixed strategies, a practical implementation of the equilibrium encounters difficulties, when strategy probabilities are irrational numbers or rational fractions of big integers in numerator and denominator. The matter is the number of connections is finite anyway, during which statistical performance of the equilibrium situation comes out incomplete [13], [16], [17].

As an example, consider a P2P file-sharing network, where two peers can choose whether to share a file or not. Each peer benefits from sharing own files but incurs a cost for downloading files shared by others. The file of peer #1 is larger than that of peer #2. When both download (pure strategy #1), the respective payoffs are 100 and 45; when both share (pure strategy #2), the respective payoffs are 120 and 50. When peer #1 downloads and peer #2 shares, the respective payoffs are 75 and 55; when peer #1 shares and peer #2 downloads, the respective payoffs are 95 and 75. Such a benefit configuration exists during a short period of time, where the peer can make a limited number of shifts of one's pure strategy (from downloading to sharing and vice versa). The respective payoff matrices of the players are

$$\mathbf{A} = \begin{bmatrix} 100 & 75 \\ 95 & 120 \end{bmatrix} \tag{1}$$

and

* Corresponding author's e-mail: romanukevadimv@gmail.com
Article received 2025-10-08; accepted 2025-11-07

$$\mathbf{B}=\begin{bmatrix}45 & 55\\ 75 & 50\end{bmatrix}. \tag{2}$$

The bimatrix game with payoff matrices (1) and (2) has a single equilibrium situation which is

$$\left\{\mathbf{P}^*,\mathbf{Q}^*\right\}=\left\{\begin{bmatrix}\frac{5}{7} & \frac{2}{7}\end{bmatrix},\begin{bmatrix}\frac{9}{10} & \frac{1}{10}\end{bmatrix}\right\}. \tag{3}$$

Thus, peer #1 must download and share with probabilities $5/7$ and $2/7$, respectively. Peer #2 must download and share with probabilities $9/10$ and $1/10$, respectively. Theoretically, equilibrium strategies $\mathbf{P}^*$ and $\mathbf{Q}^*$ in (3) ensure that on average peer #1 receives payoff

$$v\left(\mathbf{P}^*,\mathbf{Q}^*\right)=\mathbf{P}^*\cdot\mathbf{A}\cdot\left(\mathbf{Q}^*\right)^{\mathrm{T}}=\frac{195}{2}=97.5 \tag{4}$$

and on average peer #2 receives payoff

$$u\left(\mathbf{P}^*,\mathbf{Q}^*\right)=\mathbf{P}^*\cdot\mathbf{B}\cdot\left(\mathbf{Q}^*\right)^{\mathrm{T}}=\frac{375}{7}\approx 53.5714. \tag{5}$$

However, the number of strategic interactions, as it is abovementioned, is limited and thus payoffs (4) and (5) are hardly reachable [16], [17]. Moreover, even if there are 10 possibilities to shift one's pure strategy, peer #1 cannot exactly implement probabilities $5/7$ and $2/7$. The closest faction to $2/7$ in this case is $3/10$, so peer #1 can implement a mixed strategy

$$\mathbf{P}_1=\begin{bmatrix}\frac{7}{10} & \frac{3}{10}\end{bmatrix}$$

rather than

$$\mathbf{P}^*=\begin{bmatrix}\frac{5}{7} & \frac{2}{7}\end{bmatrix}.$$

Then, the expected payoffs are

$$v\left(\mathbf{P}_1,\mathbf{Q}^*\right)=\mathbf{P}_1\cdot\mathbf{A}\cdot\left(\mathbf{Q}^*\right)^{\mathrm{T}}=\frac{195}{2}=97.5 \tag{6}$$

and

$$u\left(\mathbf{P}_1,\mathbf{Q}^*\right)=\mathbf{P}_1\cdot\mathbf{B}\cdot\left(\mathbf{Q}^*\right)^{\mathrm{T}}=\frac{1079}{20}=53.95. \tag{7}$$

It might seem that situation

$$\left\{\mathbf{P}_1,\mathbf{Q}^*\right\}=\left\{\begin{bmatrix}\frac{7}{10} & \frac{3}{10}\end{bmatrix},\begin{bmatrix}\frac{9}{10} & \frac{1}{10}\end{bmatrix}\right\} \tag{8}$$

is even better than equilibrium situation (3) because peer #1 does not lose anything and peer #2 receives a 0.7067 % higher payoff. That said, situation (8) is not equilibrium: peer #2 may want to swerve to a non-equilibrium strategy

$$\mathbf{Q}_1=\begin{bmatrix}1 & 0\end{bmatrix}$$

implying only downloading, which within the respective non-equilibrium situation

$$\left\{\mathbf{P}_1,\mathbf{Q}_1\right\}=\left\{\begin{bmatrix}\frac{7}{10} & \frac{3}{10}\end{bmatrix},\begin{bmatrix}1 & 0\end{bmatrix}\right\} \tag{9}$$

provides the expected payoffs

$$v\left(\mathbf{P}_1,\mathbf{Q}_1\right)=\mathbf{P}_1\cdot\mathbf{A}\cdot\left(\mathbf{Q}_1\right)^{\mathrm{T}}=\frac{197}{2}=98.5 \tag{10}$$

and

$$u\left(\mathbf{P}_1,\mathbf{Q}_1\right)=\mathbf{P}_1\cdot\mathbf{B}\cdot\left(\mathbf{Q}_1\right)^{\mathrm{T}}=66. \tag{11}$$

Despite payoffs (10) and (11) are higher than payoff (6) and (7), peer #1 subsequently may want to swerve to another non-equilibrium strategy

$$\mathbf{P}_2=\begin{bmatrix}1 & 0\end{bmatrix}$$

also implying only downloading to further increase one's payoff to 100, whereas the payoff of peer #2 drops down to 45. Moreover, in addition to that, when there are seven possibilities to shift one's pure strategy, peer #2 cannot exactly implement probabilities $9/10$ and $1/10$, and similar non-equilibrium skipping starts.

This has been a demonstration of instability of an equilibrium situation by a few possibilities of pure strategy shifting. Furthermore, by a fewer possibilities, the equilibrium inconsistency only builds up. Therefore, the players need a method to form their equilibrium strategies dependable on the number of repetitions of the game. This method should provide a better stability of a non-cooperative game solution, which would ensure a feasible equilibrium under a finite number of network connections, pings, transactions, bids, queries, etc.

II. Objective and Tasks

Due to Nash equilibria become infeasible (unstable) for a finite number of game repetitions, the objective is to suggest a method of finding more stable solutions in bimatrix games for improving telecommunication interactions. Such a method can be based on using an approach of successive approximation to an equilibrium situation within a finite space of actions whose size directly depends on the number of game repetitions. The approach of successive approximation must be based on a genetic algorithm. To reach the objective, the tasks listed below are to be fulfilled.

1. To build a genetic algorithm that finds approximate Nash equilibria in bimatrix games.
2. To correct the algorithm in its part, where chromosomes are formed as mixed strategies being probability distributions. The chromosome must satisfy a constraint of the finiteness of an action space.
3. To validate completeness of statistical performance of the approximate equilibria.
4. To discuss the merits and drawbacks of the suggested algorithm applied for improving telecommunication strategic interactions.
5. To conclude on the obtained results and outline promising ways of furthering the research.

III. Genetic Algorithm

The primary inputs to the genetic algorithm are real-valued positive payoff matrices $\mathbf{A} = \left[a_{ij} \right]_{M \times N}$ and $\mathbf{B} = \left[b_{ij} \right]_{M \times N}$, where M and N are the numbers of pure strategies of the first and second players, respectively. If initial payoff matrices contain negative entries, they are intentionally made positive by adding to each matrix a sufficiently great value (the same value is added to every matrix). In both bimatrix games (such games are called affine equivalent), the player has the same equilibrium strategies, while the respective payoffs differ by that value added to an initial matrix.

One of the best practices is to use a binary representation of the chromosome [18], [19]. For instance, a mixed strategy of two pure strategies can be represented as a six-bit string

'110101',

where the first three bits (1, 2, 3) correspond to the first pure strategy and the next three bits (4, 5, 6) correspond to the second pure strategy. In this three-bit pure strategy representation, there are $2^3 = 8$ possible integer values per pure strategy, between 0 and 7. In this particular example, the value of the first pure strategy ('110') is 6, and the value of the second pure strategy ('101') is 5. Upon the normalization, the mixed strategy becomes a vector

$$\begin{bmatrix} \frac{5}{11} & \frac{6}{11} \end{bmatrix}.$$

Therefore, the number of bits per pure strategy is another input parameter. Let it be denoted by b. The population size is denoted by S, and the maximal number of generations (steps or rounds of the genetic algorithm) is denoted by R. The mutation rate denoted by μ is a probabilistic value between 0 and 1, by which the algorithm decides whether the child as a result of the crossover operation will have mutations or not. Another input parameter is the number of the best chromosomes selected for crossover and mutations. Let this number be denoted by T.

At every generation, a score (fitness) is calculated for every chromosome of a population by using a fitness function [19]. The current-generation best chromosome is then found, at which the fitness is maximal. If this fitness is increased, it is stored along with the currently best chromosome. Then a new population is generated based on parameters T and μ. As all the generations are run over and done, the finally best chromosome is converted into mixed strategies of the players, which must approximate a Nash equilibrium situation. The players' payoffs at this situation are subsequently calculated.

The suggested genetic algorithm utilises six modules, each of which is used during every generation [18], [19]. These modules fulfil the following: population creation, conversion of a binary chromosome part to a mixed strategy, fitness calculation, crossover operation over two parental chromosomes (parents), mutation operation over a chromosome, selection of the best T chromosomes.

A population is created as a set of binary strings. The set size is S. Each string (being a chromosome) is of $(M + N)b$ bits. This module is invoked just once before the first generation starts.

In the binary-to-probability conversion module, a binary chromosome part is taken, which is a substring of either first Mb bits or next Nb bits. Every b bits in the substring are converted to an integer decimal value, which is subsequently divided by 2^b-1. Each decimal value is divided by the sum of all decimal values (whose number is either M or N), and thus a mixed strategy is obtained. If the sum is 0 (this happens only if the substring is of either Mb or Nb zero bits), then all the probabilities in a mixed strategy are equal to either $1/M$ or $1/N$.

The fitness calculation module takes a chromosome, calculates its respective mixed strategies

$$\mathbf{P} = \left[p_i \right]_{1 \times M}$$

and

$$\mathbf{Q} = \left[q_j \right]_{1 \times N},$$

where

$$\sum_{i=1}^{M} p_i = 1, \; p_i \geq 0 \;\; \forall\, i = \overline{1, M}$$

and

$$\sum_{j=1}^{N} q_j = 1, \; q_j \geq 0 \;\; \forall\, j = \overline{1, N},$$

and calculates the first and second players' payoffs

$$v(\mathbf{P}, \mathbf{Q}) = \mathbf{P} \cdot \mathbf{A} \cdot \mathbf{Q}^{\mathrm{T}}, \quad (12)$$

$$u(\mathbf{P}, \mathbf{Q}) = \mathbf{P} \cdot \mathbf{B} \cdot \mathbf{Q}^{\mathrm{T}}. \quad (13)$$

In addition, the first player's payoffs obtained by swerving from mixed strategy $\mathbf{P}$ to its M pure strategies are calculated:

$$v_i(\mathbf{Q}) = \sum_{j=1}^{N} a_{ij} q_j \ , \ i = \overline{1, M} \ . \tag{14}$$

The second player's payoffs obtained by swerving from mixed strategy $\mathbf{Q}$ to its N pure strategies are calculated similarly:

$$u_j(\mathbf{P}) = \sum_{i=1}^{M} b_{ij} p_i \ , \ j = \overline{1, N} \ . \tag{15}$$

Then, the fitness is calculated as

$$f = -\max\left\{ \frac{\max\limits_{i=\overline{1, M}} \{v_i(\mathbf{Q})\}}{v(\mathbf{P}, \mathbf{Q})}, \frac{\max\limits_{j=\overline{1, N}} \{u_j(\mathbf{P})\}}{u(\mathbf{P}, \mathbf{Q})} \right\}. \tag{16}$$

With fitness function (16), the genetic algorithm focuses on the players' mixed strategies in such a way that in the situation composed of these strategies the payoffs of the players are balanced as much as possible and neither player would want to swerve from this approximate balance. This, in fact, approximates an equilibrium situation in the bimatrix game.

The crossover operator takes two chromosomes (parents), for which an integer between 1 and $(M + N)b$ is randomly drawn. This integer referred to as the crossover point and denoted by h breaks each chromosome into two parts, which are interchanged making thus two new chromosomes. Namely, if

$$\mathbf{C}_1 = \left[c_1^{<k>} \right]_{1 \times (M+N)b}$$

and

$$\mathbf{C}_2 = \left[c_2^{<k>} \right]_{1 \times (M+N)b}$$

are the parents, then the two new chromosomes are

$$\mathbf{C}_3 = \left[c_3^{<k>} \right]_{1 \times (M+N)b} = \left\{ \left\{ c_1^{<k>} \right\}_{k=1}^{h}, \left\{ c_2^{<k>} \right\}_{k=h+1}^{(M+N)b} \right\} \tag{17}$$

and

$$\mathbf{C}_4 = \left[c_4^{<k>} \right]_{1 \times (M+N)b} = \left\{ \left\{ c_2^{<k>} \right\}_{k=1}^{h}, \left\{ c_1^{<k>} \right\}_{k=h+1}^{(M+N)b} \right\}. \tag{18}$$

The crossover operator is applied to a set of T selected chromosomes $\{\mathbf{C}_t\}_{t=1}^{T}$ as generation of a new population starts. In more detail, the crossover is done over $T - 1$ pairs of chromosomes

$$\{\mathbf{C}_t, \mathbf{C}_{t+1}\}_{t=1}^{T-1}.$$

Every parent and every child is subject to mutation at rate μ, where a value ρ of a uniformly distributed random variable on interval $(0; 1)$ is used. Value ρ is drawn for every bit of the chromosome. If

$$\rho < \mu \tag{19}$$

for bit k of chromosome

$$\mathbf{C} = \left[c^{<k>} \right]_{1 \times (M+N)b},$$

then this bit is inverted:

$$c^{<k>(\text{obs})} = c^{<k>}, \ c^{<k>} = 1 - c^{<k>(\text{obs})}.$$

The mutation operator is successively applied to $2 \cdot (T-1)$ children and T parents. In addition, condition (19) is checked for every parent. If (19) holds for a parental chromosome

$$\mathbf{C}_t = \left[c_t^{<k>} \right]_{1 \times (M+N)b}$$

then it is flipped as [20], [21]

$$\mathbf{C}_t^{*} = \left[c_t^{<k>*} \right]_{1 \times (M+N)b} \text{ by}$$
$$c_t^{<k>*} = c_t^{<(M+N)b-k+1>} \text{ for } k = \overline{1, (M+N)b} \tag{20}$$

and a new chromosome (20) is included into a new population along with the children and parents, regardless of whether they have had mutations or not.

The selection module takes a population $\{\mathbf{C}_t\}_{t=1}^{K}$ of K chromosomes and its respective fitnesses $\{f_t\}_{t=1}^{K}$ calculated by (16). It is worth noting that apart from the newly created chromosomes by crossover and mutation, number K includes also the parents from the previous generation, for which inequality (19) was not true. The fitness probabilities are calculated as

$$\lambda_t = f_t - \min_{t=\overline{1, K}} f_t + 2^{-52}, \ t = \overline{1, K}, \tag{21}$$

$$\lambda_t^{*} = \frac{\lambda_t}{\sum\limits_{r=1}^{K} \lambda_r}, \ t = \overline{1, K}. \tag{22}$$

Probabilities (22) are sorted in descending order, whereupon the first $T \subset \left\{ \overline{1, K} \right\}$ indices are taken out of this order. The subsequently selected subpopulation is

$$\{\mathbf{C}_t\}_{t \in T} \subset \{\mathbf{C}_t\}_{t=1}^{K}. \tag{23}$$

It is important that a pair of probability vectors differing by no more than 10^{-12} is not included in subpopulation (23). Only one of such vectors is included in it. This pairwise refinement makes subpopulation (23) more diverse. Besides, it speeds up the algorithm owing to probability vector repetitions are excluded.

It is clear that, due to (16), the fitness value does not exceed -1. Therefore, if $f = -1$ at a generation, the algorithm stops.

IV. Action Space Finiteness

The binary-to-probability conversion module is used within the fitness calculation module to calculate respective mixed strategies **P** and **Q** from a chromosome. The population of chromosomes is a subset of an action space, where each element is a situation in mixed strategies represented as a sequence of $(M+N)b$ bits. This space should be finite and its size directly depends on the number of game repetitions d. Hence, each mixed strategy should be of probabilities that could be represented as fractions with the same integer denominator d. Consequently, here is a correction in the binary-to-probability conversion module. As a mixed strategy of probabilities $\{r_l\}_{l=1}^{L}$ is formed upon dividing each decimal value by the sum of all decimal values (whose number is either M or N), a set of new values is calculated:

$$r_l^* = \frac{\psi(r_l \cdot d)}{d},\ l = \overline{1, L}, \tag{24}$$

where $\psi(\xi)$ rounds ξ to the nearest integer. From the set

$$J = \{l : r_l^* > 0\} \subset \{\overline{1, L}\} \tag{25}$$

of indices of nonzero values (24) a one index l_* is randomly chosen, whereupon assignment

$$r_{l_*}^* = 0 \tag{26}$$

is done. Then entry $r_{l_*}^*$ is re-assigned to the result of subtracting the sum of all values $\{r_l^*\}_{l=1}^{L}$ from 1:

$$r_{l_*}^* = 1 - \sum_{l=1}^{L} r_l^*. \tag{27}$$

Manipulations (25)–(27) may result in one or more negative values in set (24). To get rid of negative values in set (24), manipulations (25)–(27) are repeatedly fulfilled while set (24) contains a negative value.

However, it is probable that some chromosomes, within either the initial population or a new population, upon the conversion turn out to be the same due to rounding and negative value elimination by (24)–(27). Then chromosomes rendered to identical mixed strategy situations are removed from a population so that the population would not contain a pair of identical mixed strategy situations (the chromosomes representing a pair of identical mixed strategy situations can differ in one or more bits, but the difference vanishes due to rounding and negative value elimination). This is done to prevent non-uniform fitness evaluation.

V. Validation

For validating completeness of statistical performance of the approximate equilibria, payoff matrices $\mathbf{A} = \left[a_{ij}\right]_{M \times N}$ and $\mathbf{B} = \left[b_{ij}\right]_{M \times N}$ are randomly generated by drawing values ρ_{payoff} of a uniformly distributed random variable on interval $(0; 1)$. Each entry of these matrices is equal to

$$\psi(100 \cdot \rho_{\text{payoff}} + 5).$$

The payoff matrices are re-generated until the respective bimatrix game has no pure strategy equilibria. If a game has two or more equilibrium solutions, a solution whose value

$$\left| \frac{v(\mathbf{P}, \mathbf{Q})}{u(\mathbf{P}, \mathbf{Q})} - 1 \right| \tag{28}$$

is minimal is selected. This solution denoted by

$$\{\mathbf{P}^{**}, \mathbf{Q}^{**}\} \tag{29}$$

is compared to an approximate equilibrium returned by the genetic algorithm. The genetic approximate equilibrium being a function of denominator d is denoted by

$$\{\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d)\}. \tag{30}$$

Solutions (29) and (30) are compared by the four components:

$$v(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d)) - v(\mathbf{P}^{**}, \mathbf{Q}^{**}), \tag{31}$$

$$u(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d)) - u(\mathbf{P}^{**}, \mathbf{Q}^{**}), \tag{32}$$

$$v(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d)) + u(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d)) - \\ - v(\mathbf{P}^{**}, \mathbf{Q}^{**}) - u(\mathbf{P}^{**}, \mathbf{Q}^{**}), \tag{33}$$

$$\max\left\{ \frac{v(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d))}{u(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d))}, \frac{u(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d))}{v(\mathbf{P}^{*\sim}(d), \mathbf{Q}^{*\sim}(d))} \right\} - \\ - \max\left\{ \frac{v(\mathbf{P}^{**}, \mathbf{Q}^{**})}{u(\mathbf{P}^{**}, \mathbf{Q}^{**})}, \frac{u(\mathbf{P}^{**}, \mathbf{Q}^{**})}{v(\mathbf{P}^{**}, \mathbf{Q}^{**})} \right\}. \tag{34}$$

The algorithm is validated in MATLAB R2018a on dual-core processor Intel Core i5-7200U@2.50GHz with 11.8 GB of RAM.

Primarily, the matrix size has been varied as

$$\{M, N\} \in \{\{5, 5\}, \{10, 5\}, \{10, 10\}, \{15, 10\}, \{15, 15\}\}$$

By $b = 2$ and $b = 4$ with denominator $d = 100$. Here components (31)–(33) appear to be negative on average and they further decrease as the matrix size grows. Difference (34) is positive not exceeding 0.06, and the difference between $b = 2$ and $b = 4$ is hardly significant. Figure 1 shows how differences (31)–(33) vary within 10 % of the respective payoffs

$$v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right),\ u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right),\ v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)+u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right), \quad (35)$$

where the percentage of instances, where inequalities

$$\left|v\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)-v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)\right|<0.1\cdot v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right), \quad (36)$$

$$\left|u\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)-u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)\right|<0.1\cdot u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right), \quad (37)$$

$$\begin{aligned}&\left|v\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)+u\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)-\right.\\&\left.-v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)-u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)\right|<\\&<0.1\cdot\left(v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)+u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)\right)\end{aligned} \quad (38)$$

simultaneously hold, is plotted. Besides, when to inequalities (36)–(38) a requirement of inequality

$$\begin{aligned}&\max\left\{\frac{v\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)}{u\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)},\frac{u\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)}{v\left(\mathbf{P}^{*\sim}(d),\mathbf{Q}^{*\sim}(d)\right)}\right\}-\\&-\max\left\{\frac{v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)}{u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)},\frac{u\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)}{v\left(\mathbf{P}^{**},\mathbf{Q}^{**}\right)}\right\}\leq 0\end{aligned} \quad (39)$$

is added, the percentage of instances, where inequalities (36)–(39) simultaneously hold, drops (Fig. 1).

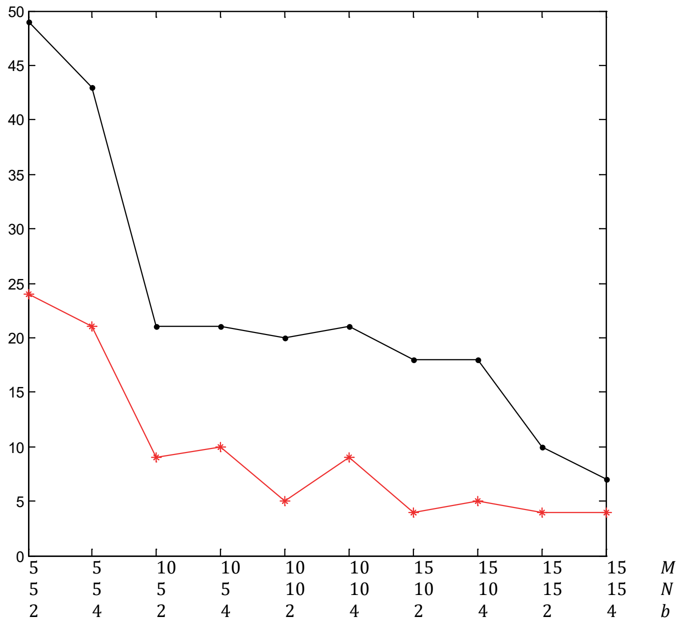

Fig. 1. The polyline percentage of instances, where inequalities (36)–(38) simultaneously hold (dots), along with the percentage of instances, where inequalities (36)–(39) simultaneously hold (stars), for $d = 100$. Both the dependencies have a decreasing trend, and adding the requirement of having a better payoff leverage by (39) shortens the percentage roughly twofold.

As the denominator is decreased (i.e., the number of game repetitions becomes fewer), components (31)–(33) tend to decrease. The percentages of inequalities (36)–(38) and of inequalities (36)–(39) are slightly higher for $d = 10$ and $d = 5$ (Fig. 2): whereas the average percentages are 22.8 % and 9.5 % in Fig. 1, the average percentages for $d = 10$ are 27.2 % and 10 %, and the average percentages for $d = 5$ are 31.2 % and 11.6 %.

Another factor is the rate of instances when components (31)–(33) are simultaneously positive: it is 8.2 %, 16.4 %, 39 % for $d = 100$, $d = 10$, $d = 5$, respectively. When to positive inequalities (31)–(33) the requirement of inequality (39) is added, the average percentages are 3.6 %, 5.8 %, 13 %, respectively.

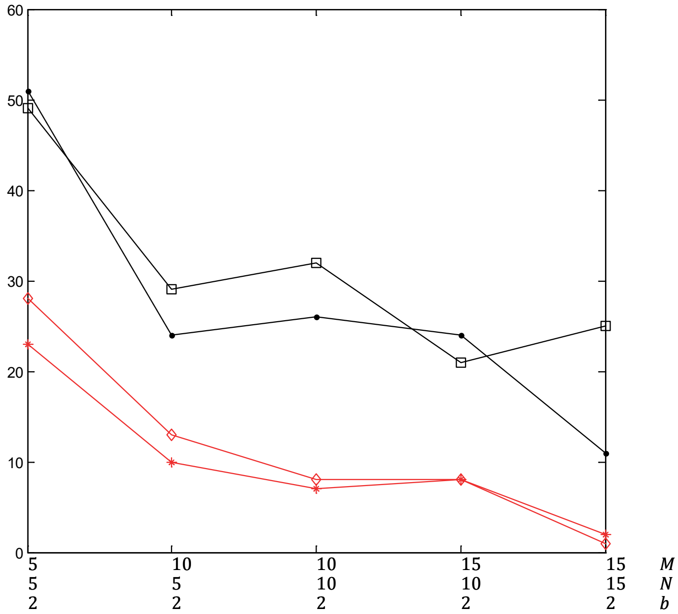

Fig. 2. The polyline percentage of instances, where inequalities (36)–(38) simultaneously hold (dots and squares for $d = 10$ and $d = 5$, respectively), along with the percentage of instances, where inequalities (36)–(39) simultaneously hold (stars and diamonds for $d = 10$ and $d = 5$, respectively). Both the dependencies have the same decreasing trend as those in Fig. 1.

VI. Discussion

The algorithm has seven input parameters: population size S, maximum number of generations R, the number of generations for the early stop e_{stop} [22], [23], mutation rate μ, bit length b, denominator d, and number T of the best chromosomes selected. The impact of the game size (referring to the payoff matrix size) and the bit length along with the denominator has been profoundly studied so far. The remaining five input parameters must be selected by using experience or rules of thumb. In the particular computer simulations (whose results are illustrated in Figs. 1 and 2) they are set to: $S = 2500$, $R = 1000$, $e_{\text{stop}} = 250$, μ is varied between 0.0125 and 0.025, $T = 50$. Apart from e_{stop} and μ, the population size, the maximum number of generations, and the number T of the best chromosomes are not optimal. However, their impact is far less strong than that of the mutation rate and the denominator.

The suggested algorithm is more efficient for fewer game repetitions. The average number of generations required for the algorithm to converge varies between 250 and 850, but it sluggishly increases as either of numbers M, N, b, d is increased. Typical plots of the learning curves, showing the highest fitness per generation, have staircase shape with pretty long last stair (Fig. 3), so the early stop number e_{stop} could have been set even below 250 for validation. It is a fair merit as fewer telecommunication strategic interactions are as likely as any other. The network peers obtain more stable and consistent strategies. This is expected to encourage interaction itself rather than resigning from sending any information due to instability.

Another merit is that the resulting approximate solution (30) is fairer owing to the fitness is calculated as (16). This imparts more equilibrium, so to say, into the peers' strategies. The approximate solution strategies become closer in terms of the respective payoffs [5], [17], [24], [25].

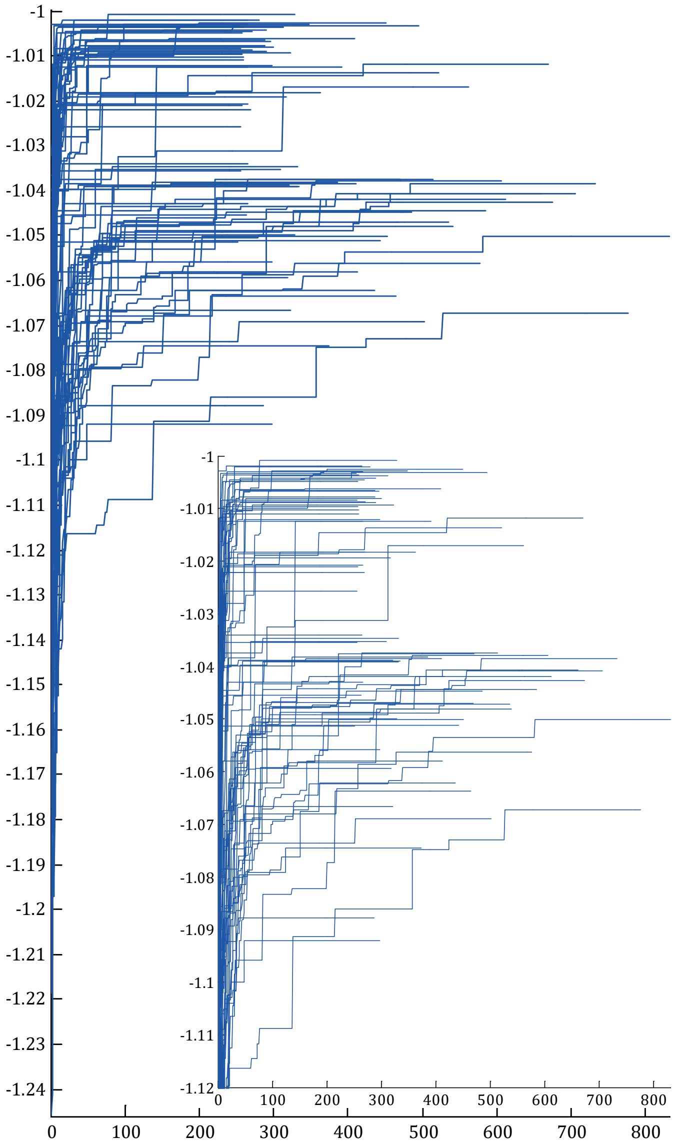

Fig. 3. Bunch of 80 learning staircase-shaped lines (with a zoom-in), showing the highest fitness per generation. The lines are plotted for varying the game size between 5×5 and 15×15 for $b \in \{2, 4, 6\}$ and $d \in \{5, 10, 100\}$.

A drawback is that the algorithm is basically for two peers. Nevertheless, this is just a primary study. The foundation of non-cooperative game solutions for three and more players is far more sophisticated and will require heavier statistical substantiation for building practically acceptable approximate equilibria [13], [17], [26]. Assessment of payoff matrices is a general drawback also (although it is not an issue from the algorithm itself) because more accurate assessment requires more information resources. In addition, as the number of pure strategies increases, the algorithm efficiency may start fading away.

VII. Conclusion

The suggested genetic algorithm is intended to find approximate equilibrium strategies in bimatrix games by knowing a finite number of game repetitions. The equilibrium concept is strengthened by introducing a criterion of mutual profitability, which is clearly a good tone and respect in telecommunication interactions. This criterion, expressed as fitness value (16), can be varied to alternatively evaluate the consequence of swerving from a given mixed strategy.

Formally speaking, the population size, the maximum number of generations, the early stop number, mutation rate, bit length, and the number of the best chromosomes could be fine-tuned for a given number of potential interactions d, but this is a computationally irrelevant task [27], [28]. It is simpler to go into adjusting the mutation rate upon appropriately selecting the population size, the maximum number of generations, and the number of the best chromosomes [29], [30]. Especially since those input parameters also depend on the payoff matrices [26], [31], [32].

The algorithm is completely flexible and applicable to strategic interaction problems in telecommunication [1], [2], [5], [25], [33], [34]: network connection attempts, pinging, executing transactions, making bids, querying, etc. are rationalized via successively selecting pure strategies in accordance with genetic approximate equilibrium (30). The fewer the number of interactions, the more efficiently the algorithm converges, outputting fairer approximate equilibria. The research will be furthered in improving the fitness criterion and adapting the algorithm for more players.

References

[1] C. Douligeris and R. Mazumdar, "A game theoretic perspective to flow control in telecommunication networks," *Journal of the Franklin Institute*, vol. 329, no. 2, pp. 383–402, Mar. 1992. https://doi.org/10.1016/0016-0032(92)90041-E

[2] J. Ma, N. Ning, and Z. Wang, "Analysis of Dynamic game with government subsidy in the telecommunication industry," *IFAC-PapersOnLine*, vol. 55, no. 10, pp. 2103–2108, 2022. https://doi.org/10.1016/j.ifacol.2022.10.018

[3] A. F. Zakaria, S. C. J. Lim, and M. Aamir, "A pricing optimization modelling for assisted decision making in telecommunication product-service bundling," *International Journal of Information Management Data Insights*, vol. 4, no. 1, Apr. 2024, Art. no. 100212. https://doi.org/10.1016/j.jjimei.2024.100212

[4] D. B. Sassi and M. Chaieb, "A game theory competitive intelligence solution stimulated from a Stackelberg game: A three players scenario," *Engineering Applications of Artificial Intelligence*, vol. 135, Sep. 2024, Art. no. 108825. https://doi.org/10.1016/j.engappai.2024.108825

[5] S. Nemmaoui, M. Baslam, and B. Bouikhalene, "Game theory analysis of biometric systems market: Users choice VS service providers revenue," *Theoretical Computer Science*, vol. 943, pp. 142–152, Jan. 2023. https://doi.org/10.1016/j.tcs.2022.12.004

[6] V. V. Romanuke, "An uplink power control routine for quality-of-service equalization in wireless data transfer networks constrained to equidistant power levels," *KPI Science News*, no. 2, pp. 46–56, May 2019. https://doi.org/10.20535/kpi-sn.2019.2.160199

[7] V. V. Romanuke, "Pure strategies Pareto efficient situations versus pure strategies Nash equilibrium situations by their stochastically constrained payoffs in dyadic game modeling of resources rational usage with

alternative choice of action," *Herald of Khmelnytskyi national university. Technical sciences*, no. 6, pp. 140–143, 2014.
[8] R. Soltani, M. Ashrafi, M. M. S. Esfahani, and H. Farvaresh, "Competitive pricing of complementary telecommunication services with subscriber churn in a duopoly," *Expert Systems with Applications*, vol. 237, Part C, Mar. 2024, Art. no. 121447. https://doi.org/10.1016/j.eswa.2023.121447
[9] V. V. Romanuke, "Multiple direction interference suppression by uniform linear phased array sidelobe efficient canceller," *Information and Telecommunication Sciences*, vol. 12, no. 1, pp. 33–40, Jun. 2021. https://doi.org/10.20535/2411-2976.12021.33-40
[10] M. B. Shareh, H. Navidi, H. H. S. Javadi, M. HosseinZadeh, "Preventing Sybil attacks in P2P file sharing networks based on the evolutionary game model," *Information Sciences*, vol. 470, pp. 94–108, Jan. 2019. https://doi.org/10.1016/j.ins.2018.08.054
[11] Z. Vahedi, S. J. S. M. Chabok, and G. Veisi, "Improving the quality of service indices of task allocation in mobile crowd sensing with fuzzy-based inverse Stackelberg game theory," *Intelligent Systems with Applications*, vol. 20, Nov. 2023, Art. no. 200291. https://doi.org/10.1016/j.iswa.2023.200291
[12] G. J. Mailath and L. Samuelson, *Repeated Games and Reputations: Long-Run Relationships*. Oxford University Press, 2006. https://doi.org/10.1093/acprof:oso/9780195300796.001.0001
[13] V. V. Romanuke, "Consistency of equilibrium stacks in finite uniform approximation of a noncooperative game played with staircase-function strategies," *Numerical Algebra, Control and Optimization*, vol. 14, no. 2, pp. 298–321, Jun. 2024. https://doi.org/10.3934/naco.2022027
[14] V. V. Romanuke, "Equilibrium stacks for a non-cooperative game defined on a product of staircase-function continuous and finite strategy spaces," *Journal of Mathematics and Applications*, vol. 46, pp. 79–114, 2023. https://journals.prz.edu.pl/jma/article/view/1526/1145
[15] D.-W. Huang, F. Luo, and J. Bi, "Uncertainty-aware prosumer coalitional game for peer-to-peer energy trading in community microgrids," *International Journal of Electrical Power & Energy Systems*, vol. 159, Aug. 2024, Art. no. 110021. https://doi.org/10.1016/j.ijepes.2024.110021
[16] V. V. Romanuke, "Convergence and estimation of the process of computer implementation of the optimality principle in matrix games with apparent play horizon," *Journal of Automation and Information Sciences*, vol. 45, no. 10, pp. 49–56, 2013. https://doi.org/10.1615/JAutomatInfScien.v45.i10.70
[17] V. V. Romanuke, "Adaptive finite approximation of continuous noncooperative games," *Journal of Automation and Information Sciences*, vol. 52, no. 10, pp. 31–41, 2020. https://doi.org/10.1615/JAutomatInfScien.v52.i10.20
[18] V. V. Romanuke, A. Y. Romanov, and M. O. Malaksiano, "Crossover operators in a genetic algorithm for maritime cargo delivery optimization," *Journal of ETA Maritime Science*, vol. 10, no. 4, pp. 223–236, 2022. https://doi.org/10.4274/jems.2022.80958
[19] V. V. Romanuke, A. Y. Romanov, and M. O. Malaksiano, "Pseudorandom number generator influence on the genetic algorithm performance to minimize maritime cargo delivery route length," *Pomorstvo. Scientific Journal of Maritime Research*, vol. 36, pp. 249–262, 2022. https://doi.org/10.31217/p.36.2.9
[20] V. V. Romanuke, "Parallelization of the traveling salesman problem by clustering its nodes and finding the best route passing through the centroids," *Applied Computer Systems*, vol. 28, no. 2, pp. 189–202, Dec. 2023. https://doi.org/10.2478/acss-2023-0019
[21] V. V. Romanuke, "Deep clustering of the traveling salesman problem to parallelize its solution," *Computers & Operations Research*, vol. 165, May 2024, Art. no. 106548. https://doi.org/10.1016/j.cor.2024.106548
[22] V. V. Romanuke, A. Y. Romanov, and M. O. Malaksiano, "A genetic algorithm improvement by tour constraint violation penalty discount for maritime cargo delivery," *System Research and Information Technologies*, no. 2, pp. 104–126, Jun. 2023. https://doi.org/10.20535/srit.2308-8893.2023.2.08
[23] V. V. Romanuke, "Traveling salesman problem parallelization by solving clustered subproblems," *Foundations of Computing and Decision Sciences*, vol. 48, no. 4, pp. 453–481, Dec. 2023. https://doi.org/10.2478/fcds-2023-0020
[24] V. V. Romanuke, "Refinement of acyclic-and-asymmetric payoff aggregates of pure strategy efficient Nash equilibria in finite noncooperative games by maximultimin and superoptimality," *Decision Making: Applications in Management and Engineering*, vol. 4, no. 2, pp. 178–199, 2021. https://doi.org/10.31181/dmame210402178r
[25] M. Hoseinpour and M.-R. Haghifam, "Prosumers' cost recovery in peer-to-peer electricity markets," *Electric Power Systems Research*, vol. 226, Jan. 2024, Art. no. 109934. https://doi.org/10.1016/j.epsr.2023.109934
[26] V. V. Romanuke, "Finite mixed extension of dyadic games and their fair solutions," *Applied Computer Systems*, vol. 30, no. 1, pp. 47–52, Jan. 2025. https://doi.org/10.2478/acss-2025-0006
[27] H. A. Yehoshyna and V. V. Romanuke, "Constraint-based Recommender system for commodity realization," *Journal of Communications Software and Systems*, vol. 17, no. 4, pp. 314–320, Oct. 2021. https://doi.org/10.24138/jcomss-2021-0102
[28] R. Fletcher, *Practical Methods of Optimization*, 2nd ed. J. Wiley and Sons, Chichester, 1987.
[29] L. D. Chambers, *The Practical Handbook of Genetic Algorithms*. Chapman and Hall/CRC, Boca Raton, FL, USA, 2000.
[30] A. LaTorre, J. M. Peña, V. Robles, and S. Muelas, "Using multiple offspring sampling to guide genetic algorithms to solve permutation problems," in *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*, ACM, New York, NY, USA, 2008, pp. 1119–1120. https://doi.org/10.1145/1389095.1389307
[31] A. Alizadeh, M. Esfahani, F. Dinar, I. Kamwa, A. Moeini, S. M. Mohseni-Bonab, and E. Busvelle, "A cooperative transactive multi-carrier energy control mechanism with P2P energy + reserve trading using Nash bargaining game theory under renewables uncertainty," *Applied Energy*, vol. 353, Part B, Jan. 2024, Art. no. 122162. https://doi.org/10.1016/j.apenergy.2023.122162
[32] A. Goswami and R. Gupta, "Stable reputation management system in P2P networks: An evolutionary perspective," *Journal of Computational Science*, vol. 78, Jun. 2024, Art. no. 102262. https://doi.org/10.1016/j.jocs.2024.102262
[33] M. Chołodecki, "The future EU postal regulation. What can be learnt from the telecommunication regulations," *Computer Law & Security Review*, vol. 52, Apr. 2024, Art. no. 105938. https://doi.org/10.1016/j.clsr.2024.105938
[34] M. A. I. Gazi, A. Al Mamun, A. Al Masud, A. Rahman bin S. Senathirajah, and T. Rahman, "The relationship between CRM, knowledge management, organization commitment, customer profitability and customer loyalty in telecommunication industry: The mediating role of customer satisfaction and the moderating role of brand image," *Journal of Open Innovation: Technology, Market, and Complexity*, vol. 10, no. 1, Mar. 2024, Art. no. 100227. https://doi.org/10.1016/j.joitmc.2024.100227

Vadim V. Romanuke was born in 1979. He graduated from the Technological University of Podillya in 2001. In 2006, he received the Degree of Candidate of Technical Sciences in Mathematical Modelling and Computational Methods. The Candidate Dissertation suggested a way of increasing interference noise immunity of data transferred over radio systems. The degree of Doctor of Technical Sciences in Mathematical Modelling and Computational Methods was received in 2014. The Doctor-of-Science Dissertation solved a problem of increasing efficiency of identification of models for multistage technical control and run-in under multivariate uncertainties of their parameters and relationships. In 2016, he received the status of a Full Professor.
He is a Professor of the Faculty of Mechanical and Electrical Engineering at the Polish Naval Academy. His current research interests concern decision making, game theory, semantic image segmentation, statistical approximation, and control engineering based on statistical correspondence. He is the author of 450 scientific articles, one monograph, one tutorial, and 16 methodical guidelines in Functional Analysis, Mathematical and Computer Modelling Master Thesis elaboration, Conflict-Controlled Systems, Algorithms and Data Structures, Data Analysis, Master Academic Practice. Before January 2018, Vadim Romanuke was the scientific supervisor of a Ukrainian budget grant work concerning minimization of water heat transfer and consumption.
Address for correspondence: 69 Śmidowicza Street, Gdynia, Poland, 81-127.
E-mail: romanukevadimv@gmail.com
ORCID iD: https://orcid.org/0000-0003-3543-3087