SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2023, Vol. 66 112205:1–112205:20
https://doi.org/10.1007/s11432-021-3291-3

Toward the minimum vertex cover of complex networks using distributed potential games

Jie CHEN[1] & Xiang LI[2,1*]

[1]*Adaptive Networks and Control Lab, Department of Electronic Engineering, Fudan University, Shanghai* 200433*, China;*
[2]*Institute of Complex Networks and Intelligent Systems, Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University, Shanghai* 201210*, China*

Received 9 January 2021/Revised 29 March 2021/Accepted 2 June 2021/Published online 29 December 2022

Abstract Vertex cover of complex networks is essentially a major combinatorial optimization problem in network science, which has wide application potentials in engineering. To optimally cover the vertices of complex networks, this paper employs a potential game for the vertex cover problem, designs a novel cost function for network vertices, and proves that the solutions to the minimum value of the potential function are the minimum vertex covering (MVC) states of a general complex network. To achieve the optimal (minimum) covering states, we propose a novel distributed time-variant binary log-linear learning algorithm, and prove that the MVC state of a general complex network is attained under the proposed optimization algorithm. Furthermore, we estimate the upper bound of the convergence rate of the proposed algorithm, and show its effectiveness and superiority using numerical examples with representative complex networks and optimization algorithms.

Keywords minimum vertex cover, Nash equilibrium, convergence rate, scale-free network, random graph

Citation Chen J, Li X. Toward the minimum vertex cover of complex networks using distributed potential games. Sci China Inf Sci, 2023, 66(1): 112205, https://doi.org/10.1007/s11432-021-3291-3

1 Introduction

The vertex cover problem is a well-known nondeterministic polynomial (NP) hard problem that covers the minimum number of vertices of a general network, where at least one endpoint of each edge in the network is covered [1]. Therefore, it has gained wide interest in different research and potential application fields, such as wireless sensor networks [2], distributed optimization of collective intelligence [3], intelligent autonomous systems [4], and Internet of Things [5].

For the minimum vertex cover of a general network, previous decades have witnessed fruitful efforts in achieving (sub)-optimal solutions, where the connectivity patterns of a general network are important in the performance of network covering as an NP hard problem [6, 7]. Since the seminar work of small-world and scale-free networks in the 1900s [8, 9], the complexity rooted in the ubiquitous connectivity patterns of small-world and scale-free features in large-scale social and technical networks has been recognized due to its significant impact on collective system performance [10–12]. Thus, the problem developing optimization algorithms to find better or optimal solutions to the vertex cover problem is a challenge.

Toward proferring a solution, many efforts have been drawn from the approaches used in combinatorial optimization. Although the solution of the vertex cover problem of complex networks is not achieved in polynomial time, however, developing approximate algorithms with improved approximation factors is achieved [13–16]. For instance, Halperin [13] designed an improved algorithm with the approximation factor $2-(1-O(1))(2\mathrm{ln}\mathrm{ln}n/\mathrm{ln}n)$ for the vertex cover problem, where n is the number of network vertices. Karakostas [14] further reduced the approximation factor for the vertex cover problem to $2-O(1/\sqrt{\mathrm{log}n})$. Many heuristic algorithms such as evolutionary algorithm [17, 18], genetic algorithm (GA) [19], multi-objective evolutionary algorithm [20], quantum algorithm [21, 22], and branch-and-bound search algorithm [23], have been developed to find a more optimal solution for covering the network vertices.

* Corresponding author (email: lix2021@tongji.edu.cn)

However, most such optimization algorithms are centralized; i.e., the covering strategy for each individual (vertex) in the given network is not self-decided; rather, it only follows a central administrator's decision. For complex network structures, a large-scale network system can be dysfunctional due to the heavy computational burden and huge communication costs. Since each vertex inanimately acts, the performance of the whole network system relies on a central administrator, which is vulnerable to targeted attacks. In autonomous intelligent systems, the high-efficient requirement of a central administrator is difficult to satisfy and maintain. These emerging issues present the significance of designing efficient distributed algorithms to solve the vertex cover problem of a general complex network.

Recently, fruitful efforts have been devoted to distributed optimization algorithms for the vertex cover problem from the evolutionary game-based algorithms, where every vertex takes its independent decision to achieve the optimized covering of the entire network. An interactive snowdrift game framework between the vertices of a general network was first presented in [24] and proved the strict Nash equilibriums (SNEs) between the vertex covering (VC) and minimum vertex covering (MVC) states of a general network. Moreover, a memory-based best response (MBR) algorithm was employed to prove that the covering states of all vertices converge to an SNE [24]. Also, the MBR was extended to dynamical networks [25], and an extended variation of the MBR algorithm as the feedback-based best response algorithm for weighted vertex cover problem was systematically developed [26]. Sun et al. [27] used a potential game to describe the weighted vertex cover problem, proposed a relaxed greedy and memory-based algorithm, and similarly proved the convergence to an SNE.

Although such optimization algorithms are superior to other existing distributed algorithms, they only guarantee that the covering states of all vertices converge to an SNE. However, an SNE is inefficient for evaluating the optimality of an MVC state and achieves 50% suboptimality [28]. Therefore, we seek to develop a distributed optimization algorithm, such that the covering states of all vertices converge to the MVC state of a general complex network. This paper designs a novel individual cost function, and proves that the solutions to the minimum value of the potential function are the MVC states. Also, we propose a new game-based distributed algorithm: the time-variant binary log-linear learning algorithm (TVBLLA), which, as we prove in this paper, guarantees that the covering states of all vertices converge to the MVC state of a general complex network.

The main contributions of this paper are summarized as follows.

(1) We design a novel individual cost function with the cost value of each individual (vertex) bounded in [0,1], and prove that the solutions to the minimum value of the potential function are the MVC states of a general complex network. Since every individual (vertex) takes its local information to make covering decisions by itself, our newly designed individual cost function bridges the optimal solutions of the potential function and the optimal covering states of all vertices of a general complex network.

(2) We newly propose a distributed TVBLLA that outperforms the binary log-linear learning algorithm [29], which cannot guarantee the covering states of all vertices attain the VC state of a general complex network. We prove for the first time that the covering states of all vertices driven by the proposed TVBLLA converge with probability one to the MVC state of a general complex network.

(3) We estimate the convergence rate of the proposed TVBLLA and give an upper bound on the distance between the covering state after finite time steps and the MVC state. We employ extensive numerical experiments to verify the effectiveness of our proposed TVBLLA. Results prove affirmative and present its advantage to the representative optimization algorithms on various networks.

The rest of this paper is organized as follows: Section 2 introduces the preliminaries and provides guidelines for developing the problem. Section 3 presents the relationship between the solutions to a potential game and the vertex cover problem, and proves that the solutions to the minimum value of the potential function are the MVC states of a network. Section 4 proposes a TVBLLA. Section 5 analyzes the performance of TVBLLA, proves that the covering states of all vertices converge to the MVC state of a general network, and estimates the convergence rate of the TVBLLA. Section 6 provides numerical simulations to illustrate the effectiveness and advantages of TVBLLA. Section 7 presents the conclusion.

2 Preliminaries and problem formulation

2.1 Vertex cover problem

Given an undirected graph $\Xi = (V, E)$ with the set of vertices $V = \{1, 2, \ldots, n\}$ and the set of edges $E = \{e_{ij}\}$ $(i, j \in V, i \neq j)$, if there exists an edge from vertex i to vertex j, then $e_{ij} = 1$; otherwise, $e_{ij} = 0$. A vertex cover is defined as a set V_{VC} such that each $e_{ij}(e_{ij} \in E)$ has at least one endpoint (vertex) in the set V_{VC}. A minimum vertex cover V_{MVC} is a vertex cover with the minimum cardinality; i.e., the number of covered vertices is minimum.

Definition 1. Regard the vertex cover problem of a graph (network) as a game among all the vertices (players) of the graph denoted by $G = (V, X, F, E)$:

(i) $X = \prod_{i=1}^{n} X_i$, where $X_i = \{0, 1\}$ (0 and 1 represent the uncovered and covered states, respectively) is the strategy set of player (vertex) i and $x = (x_1, \ldots, x_n) \in X$, $x_i \in X_i$ is a strategy profile;

(ii) $F = \{f_1, \ldots, f_n\}$ is the set of cost functions, where f_i is the cost function of player i.

2.2 Nash equilibrium and potential game

Consider an n-player game on the finite strategy space X, and let $x_{-i} = (x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$ denote the strategy profile of all players except player i. Thus, the cost function of player i can be written as $f_i(x) = f_i(x_i, x_{-i})$.

Definition 2 (Nash equilibrium [30])**.** A Nash equilibrium $\bar{x} = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)$ is such a strategy profile where no player can unilaterally change its strategy to decrease its cost, i.e.,

$$f_i(\bar{x}_i, \bar{x}_{-i}) \leqslant f_i(x'_i, \bar{x}_{-i}), \tag{1}$$

where $\bar{x}_i \in \bar{x}$, and $\forall x'_i \neq \bar{x}_i$. If Eq. (1) holds strictly for every $x'_i \neq \bar{x}_i$, $\bar{x}$ is an SNE.

Definition 3 (Potential game [31])**.** A strategic game $G = (V, X, F, E)$ is a potential game if there exists a function $\varphi : X \to \mathbb{R}$, such that

$$f_i(x'_i, x_{-i}) - f_i(x''_i, x_{-i}) = \varphi(x'_i, x_{-i}) - \varphi(x''_i, x_{-i}), \tag{2}$$

for every player $i \in V$, and $\forall x'_i, x''_i \in X_i$, where φ is a potential function of G.

2.3 Inhomogeneous Markov chains

A discrete-time Markov chain on the finite strategy space X with time-dependent transition probability matrices $P^1, P^2, \ldots$ is inhomogeneous [32–34]. The probability of the state transition during z steps starting from time t is defined as

$$P^{t,z} = \prod_{i=t}^{t+z-1} P^i, \tag{3}$$

where $P^{t,z}$ contains element $P^{t,z}_{i,j}$.

Note that a transition probability matrix is a stochastic matrix, and all elements in the matrix are nonnegative.

Definition 4 (Scrambling matrix [32])**.** A stochastic matrix P is a scrambling matrix if for any two rows, α and ξ, there exists at least one column γ such that both $P_{\alpha,\gamma} > 0$ and $P_{\xi,\gamma} > 0$.

Moreover, the scrambling power of a stochastic matrix P is defined as $\mathrm{sp}(P)$ [32]:

$$\mathrm{sp}(P) = \min_{\alpha,\alpha'} \sum_{\gamma} \min(P_{\alpha,\gamma}, P_{\alpha',\gamma}). \tag{4}$$

For a stochastic matrix P, if some power of P has all positive entries, then P is a regular stochastic matrix [32]. Besides, for each $t \in \mathbb{N}^+$, if a stochastic matrix P^t is regular with the same arrangement of non-zero elements, then there exists an integer $c > 0$ such that $P^{t,c}$ is a scrambling matrix, i.e., $\forall t > 0, \mathrm{sp}(P^{t,c}) > 0$, where c is a scrambling coefficient [32].

Definition 5 (Ergodicity coefficient [33])**.** The ergodicity coefficient of a stochastic matrix P is defined as

$$\tau(P) = 1 - \mathrm{sp}(P). \tag{5}$$

Lemma 1 ([33]). For a stochastic matrix P and a vector $\boldsymbol{y} \in \mathbb{R}^m$ such that $\sum_{i=1}^m \boldsymbol{y}_i = 0$,

$$\|\boldsymbol{y}P\| \leqslant \tau(P)\|\boldsymbol{y}\|. \tag{6}$$

Lemma 2 ([33]). For any two stochastic matrices A and B,

$$\tau(AB) \leqslant \tau(A)\tau(B). \tag{7}$$

If a Markov chain is strongly ergodic, then there exists a unique stochastic row vector π^*, called stationary distribution, such that $\lim_{z\to\infty} \pi P^{t,z} = \pi^*$ for any initial distribution π [34]. The criteria for weak ergodicity and strong ergodicity of an inhomogeneous markov chain are introduced as follows.

Lemma 3 ([32]). A Markov chain is weakly ergodic if and only if there exists a subdivision at time steps $t_1, t_2, \ldots$, such that $\sum_{l=1}^{\infty} \mathrm{sp}(P^{t_l,z_l})$ diverges, i.e., $\sum_{l=1}^{\infty} \mathrm{sp}(P^{t_l,z_l}) > \infty$, where $z_l = t_{l+1} - t_l$.

Lemma 4 ([34]). A Markov process $\{P^t\}$ is strongly ergodic if the following requirements hold:

(i) For each t, P^t is a regular matrix with stationary distribution π^t;

(ii) The Markov process $\{P^t\}$ is weakly ergodic;

(iii) The stationary distribution π^t satisfies

$$\sum_{t=0}^{\infty} \sum_{x\in X} \|\pi_x^t - \pi_x^{t+1}\| < \infty. \tag{8}$$

2.4 Problem formulation

In this paper, we aim at minimizing the global objective function as [17]

$$f(x) = \sum_{i=1}^{n} \left[x_i + \mu(1-x_i)\sum_{j=1}^{n}(1-x_j)e_{ij} \right], \tag{9}$$

where $\mu > n/2$ is the penalty parameter of the uncovered edge of a general complex network.

We design the cost function of each player i as

$$f_i(x) = \frac{x_i + 2\mu(1-x_i)\sum_{j=1}^{n}(1-x_j)e_{ij}}{2\mu\Delta}, \tag{10}$$

where Δ is the maximum degree of a general complex network.

Lemma 5. Given an undirected graph Ξ and a constant μ, a strategic game $G = (V, X, F, E)$ is the potential game, where $F = \{f_i\}_{i\in V}$ with f_i given by (10), and the potential function is

$$\varphi(x) = \frac{f(x)}{2\mu\Delta}. \tag{11}$$

We leave out the proof as it is similar to that of Theorem 2 in [27], and the only difference is the expression of $f_i(x)$ and $\varphi(x)$.

Note that given an undirected graph Ξ and a constant μ, a solution to the minimum value of (11) is also the solution to the minimum value of (9).

3 Relationship between potential game and minimum vertex cover

Let V_{MIN} denote the set of covered vertices in the solution to the minimum value of (9), and $\Omega_{\mathrm{MIN}} := \{V_{\mathrm{MIN}}\}$. Let $\Omega_{\mathrm{VC}} := \{V_{\mathrm{VC}}\}$, $\Omega_{\mathrm{MVC}} := \{V_{\mathrm{MVC}}\}$, $\Omega_{\mathrm{SNE}} := \{V_{\mathrm{SNE}}\}$ denote the sets of VCs, MVCs, SNEs, respectively.

Proposition 1. The necessary and sufficient condition for an SNE is that for each vertex: (i) if $x_i = 0$, then $x_j = 1$, $\forall j \in \Gamma_i$, where Γ_i is the neighbor set of player i; (ii) if $x_i = 1$, then $x_j = 0$, $\exists j \in \Gamma_i$.

Proof. (Sufficiency) Suppose a strategy profile $x = (x_1, \ldots, x_n)$ satisfying the conditions (i) and (ii) of Proposition 1.

(i) For each player i, if $x_i = 0$, $x_j = 1$, $\forall j \in \Gamma_i$, we have

$$f_i(0, x_{-i}) - f_i(1, x_{-i}) = \frac{-1}{2\mu\Delta} < 0.$$

Thus, player i will not change its strategy from 0 to 1.

(ii) For each player i, if $x_i = 1$, $x_j = 0$, $\exists j \in \Gamma_i$, and player i has m ($m \geqslant 1$) neighbors with strategy 0, then

$$f_i(1, x_{-i}) - f_i(0, x_{-i}) = \frac{1 - 2\mu m}{2\mu\Delta} \leqslant \frac{1 - 2\mu}{2\mu\Delta} < 0.$$

Thus, player i will not change its strategy from 1 to 0. From Definition 2, $x = (x_1, \ldots, x_n)$ is an SNE.

(Necessity) Suppose $x = (x_1, \ldots, x_n)$ is an SNE. From Definition 2, for each player i, $f_i(x_i, x_{-i}) < f_i(x_i', x_{-i})$, $\forall x_i' \neq x_i$. Assume player i has m neighbors with strategy 0.

(i) If $x_i = 0$, then

$$f_i(0, x_{-i}) - f_i(1, x_{-i}) = \frac{2\mu m - 1}{2\mu\Delta} < 0$$

holds. We have $m < \frac{1}{2\mu} < 1$; thus, $m = 0$. Thus, if $x_i = 0$, then $x_j = 1$, $\forall j \in \Gamma_i$.

(ii) If $x_i = 1$, then

$$f_i(1, x_{-i}) - f_i(0, x_{-i}) = \frac{1 - 2\mu m}{2\mu\Delta} < 0$$

holds. We have $m > \frac{1}{2\mu} > 0$; thus, $m \geqslant 1$. Thus, if $x_i = 1$, then $x_j = 0$, $\exists j \in \Gamma_i$.

From Proposition 1, for any SNE, there is no 0-0 edge in the network. Therefore, an SNE must be a VC state of a general complex network; i.e., we have $\Omega_{\mathrm{SNE}} \subseteq \Omega_{\mathrm{VC}}$.

If a strategy profile $x = (x_1, \ldots, x_n)$ satisfies that at least one endpoint of each edge is covered, then, we define the strategy profile x as a covered solution of (9); otherwise, it is defined as an uncovered solution of (9).

Theorem 1. Consider a potential game $G = (V, X, F, E)$ with $\varphi(\cdot)$ and $f_i(\cdot)$. We have

(i) $f_{\mathrm{VC}}^{\max} \leqslant n < f_{\mathrm{UVC}}^{\min}$, where $f_{\mathrm{VC}}^{\max}$ is the maximum value of (9) in all covered solution, and $f_{\mathrm{UVC}}^{\min}$ is the minimum value of (9) in all uncovered solution;

(ii) $f_i(x) \in [0, 1]$, $\forall i \in V, x \in X$;

(iii) $\Omega_{\mathrm{MIN}} = \Omega_{\mathrm{MVC}} \subseteq \Omega_{\mathrm{SNE}} \subseteq \Omega_{\mathrm{VC}}$.

Proof. Since the number of network vertices is finite, there exist finite covered solutions to (9). Let $X^{\mathrm{vc}} = \{x^{\mathrm{vc},1}, \ldots, x^{\mathrm{vc},g}\}$ denote the set of covered solutions. Moreover, let $X^{\mathrm{uvc}} = \{x^{\mathrm{uvc},1}, \ldots, x^{\mathrm{uvc},h}\}$ denote the set of uncovered solutions. Thus, for any $x^{\mathrm{vc}} \in X^{\mathrm{vc}}$ and $x^{\mathrm{uvc}} \in X^{\mathrm{uvc}}$, we have $\{k | x_k^{\mathrm{vc}} = 1, x_k^{\mathrm{vc}} \in x^{\mathrm{vc}}\} \in \Omega_{\mathrm{VC}}$, and $\{k | x_k^{\mathrm{uvc}} = 1, x_k^{\mathrm{uvc}} \in x^{\mathrm{uvc}}\} \notin \Omega_{\mathrm{VC}}$.

(i) For any $x^{\mathrm{vc}} \in X^{\mathrm{vc}}$, we have

$$f(x^{\mathrm{vc}}) = \sum\nolimits_{i=1}^{n} x_i^{\mathrm{vc}}. \tag{12}$$

Thus, $f_{\mathrm{VC}}^{\max} = n$.

For any $x^{\mathrm{uvc}} \in X^{\mathrm{uvc}}$, there exists at least one uncovered edge with both endpoints being 0. Assume $x_i^{\mathrm{uvc}} = 0$ and $\exists j \in \Gamma_i, x_j^{\mathrm{uvc}} = 0$, where Γ_i is the neighbor set of player i. Then, we have

$$f(x^{\mathrm{uvc}}) \geqslant 2\mu \sum\nolimits_{j=1}^{n} (1 - x_j^{\mathrm{uvc}}) e_{ij} \geqslant 2\mu > n.$$

Thus, $f_{\mathrm{UVC}}^{\min} > n = f_{\mathrm{VC}}^{\max}$.

(ii) For any player i, if $x_i = 0$, we have

$$0 \leqslant f_i(x) = \frac{2\mu \sum_{j=1}^{n} (1 - x_j) e_{ij}}{2\mu\Delta} \leqslant \frac{k_i}{\Delta} \leqslant 1,$$

where k_i is the degree of player i.

If $x_i = 1$, we have

$$0 < f_i(x) = \frac{1}{2\mu\Delta} < \frac{1}{n\Delta} < 1.$$

Thus, $f_i(x) \in [0,1], \forall i \in V$.

(iii) In order to prove that $\Omega_{\mathrm{MIN}} = \Omega_{\mathrm{MVC}}$, we will prove that a solution to the minimum value of (9) is an MVC state, and an MVC state is also a solution to the minimum value of (9). Therefore, we prove it as sufficient and necessary conditions.

(Sufficiency) Suppose $x^{\min} = (x_1^{\min}, \ldots, x_n^{\min})$ is a solution to the minimum value of (9). Then

$$f(x^{\min}) \leqslant f(\tilde{x}), \quad \forall \tilde{x} \neq x^{\min}. \tag{13}$$

If $x^{\min} \notin X^{\mathrm{vc}}$, by the part (i) of Theorem 1, then $f(x^{\min}) > f(x^{\mathrm{vc}})$, $\forall x^{\mathrm{vc}} \in X^{\mathrm{vc}}$. It is a contradiction to (13). Thus, $x^{\min}$ is a covered solution. If $x^{\min}$ is not a minimum covered solution, then $f(x^{\min}) > f(x^{\mathrm{mvc}})$, where $\{k|x_k^{\mathrm{mvc}} = 1, x_k^{\mathrm{mvc}} \in x^{\mathrm{mvc}}\} \in \Omega_{\mathrm{MVC}}$. It is also a contradiction to (13). Thus, $x^{\min}$ is the MVC state of a general complex network.

(Necessity) Suppose $x^{\mathrm{mvc}} = (x_1^{\mathrm{mvc}}, \ldots, x_n^{\mathrm{mvc}})$ is the MVC state of given complex network, but not the solution to the minimum value of (9). Then, there exists a strategy profile $\tilde{x}$ such that

$$f(x^{\mathrm{mvc}}) > f(\tilde{x}). \tag{14}$$

According to (12), we have

$$f(x^{\mathrm{mvc}}) < f(x'), \quad \forall x' \in X^{\mathrm{vc}} \backslash X^{\mathrm{mvc}},$$

where X^{mvc} is the set of the MVC states. Thus, $X^{\mathrm{mvc}} \subseteq X^{\mathrm{vc}}$.

According to the conclusion of (i), we have

$$f(x') < f(x''), \quad \forall x'' \notin X^{\mathrm{uvc}}.$$

Thus, we have

$$f(x^{\mathrm{mvc}}) < f(x), \quad \forall x \notin X^{\mathrm{mvc}},$$

which is a contradiction to (14). Thus, $\Omega_{\mathrm{MIN}} = \Omega_{\mathrm{MVC}}$.

Suppose a strategy profile $x^* = (x_1^*, \ldots, x_n^*)$ is a solution to the minimum value of (9), but not an SNE. According to Definition 2, there exists at least one player who changes its strategy to decrease the cost, that is

$$f_i(x_i^*, x_{-i}^*) > f_i(x_i', x_{-i}^*), \quad \exists x_i' \neq x_i^*.$$

From Lemma 5, we have

$$\varphi(x_i^*, x_{-i}^*) > \varphi(x_i', x_{-i}^*),$$

and thus, x^* is not a solution to the minimum value of (9), which is a contradiction to the hypothesis. Thus, x^* is an SNE, and we have $\Omega_{\mathrm{MIN}} \subseteq \Omega_{\mathrm{SNE}}$.

According to the conclusion of Proposition 1, we have $\Omega_{\mathrm{SNE}} \subseteq \Omega_{\mathrm{VC}}$. Then, $\Omega_{\mathrm{MIN}} = \Omega_{\mathrm{MVC}} \subseteq \Omega_{\mathrm{SNE}} \subseteq \Omega_{\mathrm{VC}}$.

Remark 1. (1) The part (i) of Theorem 1 is used to verify whether the covering states of all vertices satisfy the VC state of a general complex network. Besides, Eq. (9) presented in this paper is to transform the vertex cover problem into a combinatorial optimization problem.

(2) Since the change of each player's cost function $f_i(\cdot)$ can be mapped to the potential function $\varphi(\cdot)$, the vertex cover problem can be solved in a distributed algorithm. More importantly, the cost value of each individual is bounded in $[0,1]$, which is an indispensable condition for the following proposed TVBLLA to guarantee that the covering states of all vertices converge to the MVC state of a general complex network.

4 Optimization approach: TVBLLA

In this section, we newly design a TVBLLA. The procedure of the TVBLLA is summarized at Algorithm 1.

In Step 3, the time-dependent $p_{x_i}^t$ at time step t is given as

$$p_{x_i}^t(\beta^t) = \frac{\mathrm{e}^{\beta^t f_i(x_i, x_{-i}^{t-1})}}{\sum_{\tilde{x}_i \in \Lambda_i^t} \mathrm{e}^{\beta^t f_i(\tilde{x}_i, x_{-i}^{t-1})}}, \tag{15}$$

Algorithm 1 TVBLLA

Step 1: At each time step t, player $i \in V$ is randomly chosen and allowed to update its strategy; other players repeat their previous strategies, i.e., $x_{-i}^t = x_{-i}^{t-1}$.

Step 2: Player i selects one trial strategy $\hat{x}_i^t$ from its constrained strategy set $C_i(x_i^{t-1})$ with probability:

$$p(\hat{x}_i^t = x_i) = \frac{1}{n_{\max}}, \quad \forall x_i \in C_i(x_i^{t-1}) \setminus x_i^{t-1},$$

$$p(\hat{x}_i^t = x_i^{t-1}) = 1 - \frac{|C_i(x_i^{t-1})| - 1}{n_{\max}},$$

where $n_{\max} := \max_{i \in V} |C_i(x_i^k)|, \forall k \geqslant 0$.

Step 3: Player i puts $\hat{x}_i^t$ and x_i^{t-1} in set Λ_i^t, and selects one strategy x_i from Λ_i^t with the probability $p_{x_i}^t$ for the current time step, i.e., $x_i^t = x_i$.

where the time-varying coefficient $\beta^t \leqslant 0$ represents the greedy level at time step t. When $\beta^t = 0$, player i randomly selects a strategy from Λ_i^t. When $\beta^t \to -\infty$, player i is more possible to select the best response BR_i^t [24,35] from Λ_i^t, where $BR_i^t = \arg\min_{x_i \in \Lambda_i^t} f_i(x_i, x_{-i}^{t-1})$.

In this paper, we design β^t as a function approaching to minus infinity when time step $t \to +\infty$, namely,

$$\beta^t = -\frac{\ln(\lambda t + 1)}{c}, \tag{16}$$

where $c \in \mathbb{N}^+, \lambda \geqslant 1$.

We give the following two assumptions on the constrained strategy set.

Assumption 1 (Feasibility)**.** For any player $i \in V$, and any strategy pair $x_i^0, x_i^z \in X_i$, there exists a sequence of strategies $x_i^0 \to x_i^1 \to \cdots \to x_i^z$ satisfying $x_i^k \in C_i(x_i^{k-1})$ for all $k \in \{1, 2, \ldots, z\}$.

Assumption 2 (Reversibility)**.** For any player $i \in V$, and any strategy pair x_i^1, x_i^2,

$$x_i^2 \in C_i(x_i^1) \Leftrightarrow x_i^1 \in C_i(x_i^2).$$

In practice, if a player's strategy set X_i is a fixed set, we can set $C_i(x_i^{t-1}) = X_i$, which satisfies Assumptions 1 and 2. If a player's strategy set X_i is variable, the two assumptions are not necessarily satisfied.

Under Assumptions 1 and 2, the TVBLLA with time-varying β^t induces an inhomogeneous Markov chain $\mathcal{M}_{\beta^t} = \{\{x^t\}, X, P^t\}$, where the transition probability matrix is $\{P_{x',x''}^t\}_{x',x'' \in X}$, and P^t depends on time step t. It is easy to check that, for $\forall t \geqslant 0$, the arrangements of non-zero elements of the transition probability matrix P^t are the same. $P_{x',x''}^t$ is the transition probability from the strategy profile x' to the strategy profile x'' at time step t.

(1) When $x'_{-i} = x''_{-i}$ and $x'_i \neq x''_i$, $P_{x',x''}^t$ is expressed as

$$P_{x',x''}^t = \frac{p_{x''_i}^t(\beta^t)}{|V| \cdot n_{\max}} = \frac{1}{|V| \cdot n_{\max}} \cdot \frac{\mathrm{e}^{\beta^t f_i(x''_i, x'_{-i})}}{\mathrm{e}^{\beta^t f_i(x''_i, x'_{-i})} + \mathrm{e}^{\beta^t f_i(x'_i, x'_{-i})}}, \tag{17}$$

where $\frac{1}{|V|}$ represents the probability that player i is selected to update its strategy and $|V| = n$.

(2) When $x' = x''$, each player has the same opportunity to be selected for updating the strategy, and the selected player updates its strategy, which is the same as its previous strategy. Thus,

$$P_{x',x''}^t = \sum_{i=1}^{n} \frac{1}{|V|} \left[1 - \frac{|C_i(x'_i)| - 1}{n_{\max}} + \sum_{\tilde{x}_i \in C_i(x'_i) \backslash x'_i} \frac{1}{n_{\max}} \cdot \frac{\mathrm{e}^{\beta^t f_i(x''_i, x'_{-i})}}{\mathrm{e}^{\beta^t f_i(x''_i, x'_{-i})} + \mathrm{e}^{\beta^t f_i(\tilde{x}_i, x'_{-i})}} \right]. \tag{18}$$

(3) In other cases, at least two players update their strategies at the same time step. Thus,

$$P_{x',x''}^t = 0. \tag{19}$$

If we fix β^t in (15) as a constant, i.e., $\beta^t = \bar{\beta}, \forall t \geqslant 0$, then the TVBLLA degrades to the binary log-linear learning algorithm (BLLA) [29], and the probability $p_{x_i}^t(\bar{\beta})$ does not depend on time step t, but on the joint strategy of all its neighbors. It means that in this case, the Markov chain $\mathcal{M}_{\bar{\beta}} = \{\{x^t\}, X, P\}$ with the transition probability matrix $\{P_{x',x''}\}_{x',x'' \in X}$ induced by the BLLA is a homogeneous Markov chain [32–34]. Note that $P_{x',x''}$ does not depend on time step t.

Proposition 2. Consider a potential game G with $\varphi(\cdot)$ and $f_i(\cdot)$. Then the BLLA induces a homogeneous Markov chain $\mathcal{M}_{\bar{\beta}}$ on X, which has a unique stationary distribution π with the coordinates:

$$\pi_x = \frac{\mathrm{e}^{\bar{\beta}\varphi(x)}}{\sum_{\tilde{x}\in X} \mathrm{e}^{\bar{\beta}\varphi(\tilde{x})}}. \tag{20}$$

We leave out the proof as it is similar to that of Theorem 6.2 in [36]. The difference is the expression of $P_{x',x''}$.

Now, before we analyze the convergence of the TVBLLA in Section 5, we give the following proposition to point out the main defect of the BLLA, whose convergence to the VC state cannot be guaranteed.

Proposition 3. Consider a potential game G with $\varphi(\cdot)$ and $f_i(\cdot)$. The BLLA cannot guarantee that the covering states of all vertices converge to the VC state of a general complex network.

Proof. The BLLA induces a homogeneous Markov chain $\mathcal{M}_{\bar{\beta}}$. It has a unique stationary distribution $\pi=\{\pi_{x^1},\ldots,\pi_{x^{|X|}}\}$, where $x^j \in X$. According to (20), we have $\pi_{x^j} > 0, \forall x^j \in X$. Then there exists a positive probability for each element in the stationary distribution. In other words, any strategy profile x may appear on the finite strategy space X. Thus, the BLLA cannot guarantee that the covering states of all vertices converge to the VC state of a general complex network.

5 Convergence analysis of the TVBLLA

5.1 Convergence to the MVC state

Theorem 2. Consider a potential game G with $\varphi(\cdot)$ and $f_i(\cdot)$. There exists a value c in (16) such that the TVBLLA with time-varying β^t can guarantee that the covering states of all vertices converge with probability one to the MVC state of a general complex network, i.e,

$$\lim_{t\to\infty} p\left\{x^t \in \left\{x^*|f(x^*) = \min_{x\in X} f(x)\right\}\right\} = 1.$$

Proof. (i) At a given time step t, β^t is a fixed value $\tilde{\beta}^t$ by (16). And if $\beta^t = \tilde{\beta}^t, \forall t \geqslant 0$, then the TVBLLA induces a homogeneous Markov chain, and it has a unique stationary distribution $\pi^t = \{\pi^t_{x^1},\ldots,\pi^t_{x^{|X|}}\}$, where $x^z \in X$, which satisfies $\pi^t = \pi^t P^t$. Thus, for any column j of the transition probability matrix P^t, we have $\pi^t_j = \lim_{k\to\infty}(P^t_{i,j})^k > 0$, where $(P^t_{i,j})^k$ is the k power of matrix $P^t_{i,j}$. Thus, at a given time step t, P^t is a regular matrix.

(ii) At a given time step t, the transition probability matrix P^t is regular with the same arrangements of non-zero elements. Thus, there exists some $c > 0$ such that $P^{t,c}$ is scrambling,

$$\mathrm{sp}(P^{t,c}) \geqslant \min\nolimits_{x',x''\in X} P^{t,c}_{x',x''},$$

where $P^{t,c}_{x',x''}$ denotes the positive elements of $P^{t,c}$.

By substituting (16) into (15). We have

$$p^t_{x_i}(\beta^t) = \frac{\mathrm{e}^{\beta^t f_i(x_i,x^{t-1}_{-i})}}{\sum_{\tilde{x}_i\in\Lambda^t_i} \mathrm{e}^{\beta^t f_i(\tilde{x}_i,x^{t-1}_{-i})}} = \frac{(\lambda t+1)^{-\frac{f_i(x_i,x^{t-1}_{-i})}{c}}}{\sum_{\tilde{x}_i\in\Lambda^t_i}(\lambda t+1)^{-\frac{f_i(\tilde{x}_i,x^{t-1}_{-i})}{c}}} = \frac{1}{\sum_{\tilde{x}_i\in\Lambda^t_i}(\lambda t+1)^{\frac{f_i(x_i,x^{t-1}_{-i})-f_i(\tilde{x}_i,x^{t-1}_{-i})}{c}}}.$$

From part (ii) of Theorem 1, we have

$$p^t_{x_i}(\beta^t) \geqslant \frac{1}{\sum_{\tilde{x}_i\in\Lambda^t_i}(\lambda t+1)^{\frac{1}{c}}} = \frac{1}{2(\lambda t+1)^{\frac{1}{c}}}.$$

Further, we have

$$P^t_{x',x''}(\beta^t) \geqslant \frac{1}{2|V|n_{\max}(\lambda t+1)^{\frac{1}{c}}}.$$

Next, let $P^t_{x',x''} := P^t_{x',x''}(\beta^t)$. We have

$$P^{t,c}_{x',x''} = \sum_{x^1\in X}\cdots\sum_{x^{c-1}\in X} P^t_{x',x^1}P^{t+1}_{x^1,x^2}\cdots P^{t+c-1}_{x^{c-1},x''} \geqslant \frac{1}{2^c|V|^c n^c_{\max}\left(\lambda(t+c-1)+1\right)},$$

$$\mathrm{sp}(P^{t,c}) \geqslant \min_{x',x''\in X} P^{t,c}_{x',x''} \geqslant \frac{1}{2^c|V|^c n^c_{\max}\,(\lambda(t+c-1)+1)},$$

$$\sum_{l=1}^{\infty}\mathrm{sp}(P^{t_l,c}) \geqslant \frac{1}{2^c|V|^c n^c_{\max}}\sum_{l=1}^{\infty}\frac{1}{\lambda(t_l+c-1)+1}.$$

The time sequence is subdivided into the following blocks $\{T_k\}_\infty$:

$$T_1=\{1,\ldots,c\},\quad T_2=\{c+1,\ldots,2c\},\quad \ldots.$$

Thus, $t_z=(z-1)c+1, z\in\mathbb{N}^+$.

Let $t_l=(l-1)c+1$. Thus, $t_{l+1}-t_l=c$. We have

$$\sum_{l=1}^{\infty}\mathrm{sp}(P^{t_l,c}) \geqslant \frac{1}{2^c|V|^c\lambda c n^c_{\max}}\sum_{l=1}^{\infty}\frac{1}{l+1/(\lambda c)} > \infty.$$

Thus, $\sum_{l=1}^{\infty}\mathrm{sp}(P^{t_l,c})$ diverges. According to Lemma 3, the TVBLLA with time-varying β^t in (16) induces an inhomogeneous Markov chain $\mathcal{M}_{\beta^t}$, which is weakly ergodic.

(iii) Then, we prove that the following inequality holds:

$$\sum_{t=0}^{\infty}\sum_{x\in X}\|\pi^t_x-\pi^{t+1}_x\|<\infty.$$

Substituting (16) into (20), we have

$$\pi^t_x=\frac{1}{\sum_{\tilde{x}\in X}\mathrm{e}^{\beta^t(\varphi(\tilde{x})-\varphi(x))}}=\frac{1}{\sum_{\tilde{x}\in X}(\lambda t+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}}. \tag{21}$$

Then

$$\pi^t_x-\pi^{t+1}_x=\frac{\sum_{\tilde{x}\in X}\big[(\lambda(t+1)+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}-(\lambda t+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}\big]}{\sum_{\tilde{x}\in X}(\lambda t+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}\sum_{\tilde{x}\in X}(\lambda(t+1)+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}}.$$

Considering $\varphi(x)$ and $\varphi(\tilde{x}),\forall\tilde{x}\in X$, there exist two cases: (a) $\varphi(x)\leqslant\varphi(\tilde{x})$, $\forall\tilde{x}\in X$; (b) $\varphi(x)>\varphi(\tilde{x})$, $\exists\tilde{x}\in X$.

For case (a), we know that $x\in X^*$ is a solution to the minimum value of (11), where X^* is the set of the solutions to the minimum value of (11). Thus, we have

$$\pi^t_x-\pi^{t+1}_x\leqslant 0,\quad \forall t\geqslant 0.$$

For case (b), we know that $x\notin X^*$ is not a solution to the minimum value of (11). Define

$$X^- := \{\tilde{x}\in X|\varphi(x)>\varphi(\tilde{x})\};\quad X^+ := \{\tilde{x}\in X|\varphi(x)\leqslant\varphi(\tilde{x})\}.$$

When $\forall\tilde{x}\in X^-$, we have

$$\pi^t_x-\pi^{t+1}_x>0,\quad \forall t\geqslant 0.$$

When $\forall\tilde{x}\in X^+$, we have

$$\pi^t_x-\pi^{t+1}_x\leqslant 0,\quad \forall t\geqslant 0,$$

$$\lim_{t\to\infty}\Big[(\lambda(t+1)+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}-(\lambda t+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}\Big]=0.$$

Thus, for case (b), there exists some $t'>0$ such that

$$\pi^t_x-\pi^{t+1}_x>0,\quad \forall t\geqslant t'.$$

Further, we have

$$\sum_{t=0}^{\infty}\sum_{x\in X}\|\pi^t_x-\pi^{t+1}_x\|=\sum_{t=0}^{t'}\sum_{x\in X}\|\pi^t_x-\pi^{t+1}_x\|+\sum_{t=t'+1}^{\infty}\sum_{x\in X}\|\pi^t_x-\pi^{t+1}_x\|,$$

$$\sum_{t=t'+1}^{\infty}\sum_{x\in X}\|\pi_x^t-\pi_x^{t+1}\|=\sum_{x\in X^*}\left(\lim_{t\to\infty}\pi_x^t-\pi_x^{t'+1}\right)+\sum_{x\notin X^*}\left(\pi_x^{t'+1}-\lim_{t\to\infty}\pi_x^t\right).$$

According to (21), we have

$$\sum_{x\in X^*}\lim_{t\to\infty}\pi_x^t=\frac{1}{|X^*|},\quad \sum_{x\notin X^*}\lim_{t\to\infty}\pi_x^t=0,$$

$$\sum_{t=0}^{\infty}\sum_{x\in X}\|\pi_x^t-\pi_x^{t+1}\|<\infty.$$

Then, according to Lemma 4, the TVBLLA with time-varying β^t in (16) induces an inhomogeneous Markov chain $\mathcal{M}_{\beta^t}$, which is strongly ergodic.

Thus, there exists a unique stationary distribution π^* of $\mathcal{M}_{\beta^t}$. According to (20), as $t\to+\infty$, $\beta^t\to-\infty$, if a coordinate π_x^* in the stationary distribution π^* satisfies $\pi_x^*>0$, then the strategy profile x is the solution to the minimum value of potential function $\varphi(x)$. Therefore, by employing the TVBLLA, the covering states of all vertices converge with probability one to the MVC state of a general complex network.

Note that parameter c plays a key role in the convergence of the TVBLLA. Specifically, the parameter c can be chosen as a minimum value $\bar{c}$ such that $\mathrm{sp}(P^{t,\bar{c}})>0,\forall t>0$. Let us turn back again to the inhomogeneous Markov chain $\mathcal{M}_{\beta^t}$. Matrix P^t has the same arrangements of non-zero element for all t. Hence, $\mathrm{sp}(P^{t,c})>0$ with any t, if and only if $\mathrm{sp}(P^{0,c})>0$ with $\beta^t=0$. According to [37], parameter c can be chosen as the minimum value $\bar{c}$ such that

$$\lfloor\bar{c}/n\rfloor\geqslant\frac{\ln(2+\epsilon)}{\ln(n^n)-\ln(n^n-1)},\tag{22}$$

where $\lfloor r\rfloor$ is the largest integer not exceeding r, and ϵ is a small positive value.

Remark 2. Compared with the existing distributed optimization algorithms which can only guarantee to obtain an SNE as t goes to infinity, the TVBLLA in this paper is the first distributed optimization algorithm to guarantee the MVC state of a general complex network as t goes to infinity.

5.2 Convergence rate estimation

We further analyze the convergence rate of the TVBLLA. Firstly, we give a proposition as follows.

Proposition 4. Given x and y, we have

(i) If $x\in[1,+\infty)$ and $y\in(0,1)$, then $(x+1)^y-x^y\leqslant yx^{y-1}$, and $x^y-(x-1)^y\geqslant yx^{y-1}$;

(ii) If $x\in[0,1)$, then $\ln(1-x)\leqslant-x$;

(iii) If $x\in[1,+\infty), y\geqslant x$, then $\sum_{k=x}^{y}\frac{1}{k}\geqslant\ln(\frac{y+1}{x})$.

Proof. (i) Using Newton's generalized binomial theorem, we have

$$\left(1+\frac{1}{x}\right)^y=\sum_{k=0}^{\infty}\binom{y}{k}\left(\frac{1}{x}\right)^k=1+\binom{y}{1}\left(\frac{1}{x}\right)+\binom{y}{2}\left(\frac{1}{x}\right)^2+\cdots.$$

Since $x\in[1,+\infty)$ and $y\in(0,1)$,

$$\begin{aligned}\binom{y}{2k}\left(\frac{1}{x}\right)^{2k}+\binom{y}{2k+1}\left(\frac{1}{x}\right)^{2k+1}&=y(y-1)\cdots(y-2k+1)\cdot\frac{1}{(2k)!}\left(\frac{1}{x}\right)^{2k}\frac{(2k+1)x+(y-2k)}{(2k+1)x}\\&=\sigma_k<0,\ k<\infty.\end{aligned}$$

Since

$$\lim_{k\to\infty}\binom{y}{k}\left(\frac{1}{x}\right)^k=0,$$

$$\left(1+\frac{1}{x}\right)^y\leqslant1+\binom{y}{1}\left(\frac{1}{x}\right)=1+\frac{y}{x},$$

which leads to

$$(x+1)^y-x^y\leqslant yx^{y-1}.\tag{23}$$

Analogously,

$$\left(1-\frac{1}{x}\right)^y = 1-\binom{y}{1}\left(\frac{1}{x}\right)+\binom{y}{2}\left(\frac{1}{x}\right)^2-\binom{y}{3}\left(\frac{1}{x}\right)^3+\cdots,$$

$$x^y-(x-1)^y \geqslant yx^{y-1}. \tag{24}$$

(ii) Let $h(x)=\ln(1-x)+x$. Since $h'(x)=\frac{x}{x-1}<0$ and $x\in[0,1)$, we have $h(x)\leqslant h(0)=0$, which leads to

$$\ln(1-x)\leqslant -x. \tag{25}$$

(iii) When $y=x$, since $\frac{1}{x}\in[0,1]$,

$$\sum_{k=x}^{y}\frac{1}{k}=\frac{1}{x}\geqslant \ln\left(\frac{y+1}{x}\right)=\ln\left(1+\frac{1}{x}\right).$$

When $y>x$, we have

(a) For $y=m$, the following inequality holds:

$$\sum_{k=x}^{m}\frac{1}{k}\geqslant \ln\left(\frac{m+1}{x}\right);$$

(b) For $y=m+1$,

$$\sum_{k=x}^{m+1}\frac{1}{k}=\sum\nolimits_{k=x}^{m}\frac{1}{k}+\frac{1}{m+1},$$

$$\ln\left(\frac{m+2}{x}\right)=\ln\left(\frac{m+1}{x}\right)+\ln\left(1+\frac{1}{m+1}\right).$$

Since $\frac{1}{m+1}\geqslant \ln(1+\frac{1}{m+1})$,

$$\sum_{k=x}^{m+1}\frac{1}{k}\geqslant \ln\left(\frac{m+2}{x}\right).$$

Next, we will estimate an upper bound on the distance between the covering state after finite time steps and the MVC state of a general complex network. Let v^m denote the state probability vector of an inhomogeneous Markov chain $\mathcal{M}_{\beta^t}$ at time step $m<\infty$, namely $v^m=v^0P^{0,m}$. The stationary distribution of an inhomogeneous Markov chain $\mathcal{M}_{\beta^t}$ is π^*. Then, the decomposition is

$$v^{mc}-\pi^*=\left(v^{mc}-\pi^0P^{0,mc}\right)+(\pi^0P^{0,mc}-\pi^{mc})+(\pi^{mc}-\pi^*), \tag{26}$$

where π^t is the stationary distribution of the $\mathcal{M}_{\beta^t}$ with any time step t. Note that at a given time step t, β^t is a fixed value $\tilde{\beta}^t$ by (16), and we can replace $\bar{\beta}$ in (20) with the fixed value $\tilde{\beta}^t$ to calculate π^t_x, where $\pi^t_x\in\pi^t$. Then

$$\|v^{mc}-\pi^*\|\leqslant\|v^{mc}-\pi^0P^{0,mc}\|+\|\pi^0P^{0,mc}-\pi^{mc}\|+\|\pi^{mc}-\pi^*\|. \tag{27}$$

Theorem 3. The TVBLLA with time-varying β^t induces an inhomogeneous Markov chain $\mathcal{M}_{\beta^t}$, where the transition probability matrix is P^t. The convergence rate can be estimated as

$$\|v^{mc}-\pi^*\|\leqslant\frac{\theta}{m^{\delta}}, \tag{28}$$

where $\theta>0$, $\delta=\frac{\min\{b,\delta'\}}{\min\{b,\delta'\}+1}$, $b=\frac{1}{\lambda c2^c|V|^c n^c_{\max}}$, $\delta'=\min_{x\notin X^*}\frac{\varphi(x)-\varphi(x^*)}{c}$, $x^*=\arg\min_{x\in X}\varphi(x)$.

Proof. We analyze each term on the right-hand side of (27).

(i) For the first term, according to Lemma 1, we have

$$\|v^{mc}-\pi^0P^{0,mc}\|=\|(v^0-\pi^0)P^{0,mc}\|\leqslant\tau(P^{0,mc})\|v^0-\pi^0\|. \tag{29}$$

According to Lemma 2, we have

$$\tau(P^{0,mc}) = \tau(P^{0,(a-1)c}P^{(a-1)c,(m-a+1)c}) \leqslant \tau(P^{(a-1)c,(m-a+1)c}) = \tau(P^{(a-1)c,c}P^{ac,c}\cdots P^{(m-1)c,c})$$
$$\leqslant \prod\nolimits_{k=a}^{m} \tau(P^{(k-1)c,c}) = \prod\nolimits_{k=a}^{m} \left(1 - \mathrm{sp}(P^{(k-1)c,c})\right), \quad a \geqslant 1.$$

Since

$$\mathrm{sp}(P^{(k-1)c,c}) = \frac{(kc)\mathrm{sp}(P^{(k-1)c,c})}{kc} \geqslant \frac{1}{2^c kc\lambda |V|^c n_{\max}^c},$$

there exists a value $b = \frac{1}{2^c\lambda c|V|^c n_{\max}^c} \in (0,1)$ such that

$$b/k \leqslant \mathrm{sp}(P^{(k-1)c,c}) < 1.$$

Then, we have

$$\tau(P^{0,mc}) \leqslant \prod\nolimits_{k=a}^{m} \left(1 - \mathrm{sp}(P^{(k-1)c,c})\right) \leqslant \prod\nolimits_{k=a}^{m} (1 - b/k) = \mathrm{e}^{\sum_{k=a}^{m} \ln(1-b/k)}.$$

According to parts (ii) and (iii) of Proposition 4, we have

$$\sum_{k=a}^{m} \ln\left(1 - \frac{b}{k}\right) \leqslant -\sum_{k=a}^{m} \frac{b}{k} \leqslant -b\ln\left(\frac{m+1}{a}\right) = \ln\left(\frac{a}{m+1}\right)^b.$$

We obtain

$$\tau(P^{0,mc}) \leqslant \left(\frac{a}{m+1}\right)^b = \frac{l}{(m+1)^b}, \tag{30}$$

where $l = a^b \in [1,+\infty)$, and $b \in (0,1)$.

(ii) For the second term, we have $\pi^m P^{m,1} = \pi^m$. Let $V(m) := \pi^0 P^{0,m} - \pi^m$, where $V(0) = 0$. Then,

$$\begin{aligned} V(mc) &= \pi^0 P^{0,mc} - \pi^{mc} = \pi^0 P^{1,mc-1} - \pi^{mc} \\ &= \pi^0 P^{1,mc-1} - \pi^1 P^{1,mc-1} + \pi^1 P^{1,mc-1} - \pi^{mc} = \cdots \\ &= \sum_{k=1}^{m}\sum_{s=1}^{c} (\pi^{kc-s} - \pi^{kc-s+1}) P^{kc-s+1,s-1} P^{kc,(m-k)c}. \end{aligned}$$

According to Lemmas 1 and 2, we have

$$\|V(mc)\| \leqslant \sum_{k=1}^{m} \tau(P^{kc,(m-k)c}) \sum_{s=1}^{c} \|\pi^{kc-s} - \pi^{kc-s+1}\|. \tag{31}$$

According to the proof of Theorem 2, there exists some t' such that for any $t \geqslant t'$, π_x^t is an increasing or a decreasing function. Specifically, when $x \in X^*$, π_x^t is an increasing function with t; when $x \notin X^*$, π_x^t is a decreasing function with t. Then,

$$\begin{aligned} \|V(mc)\| \leqslant &\sum_{k=1}^{t'} \tau(P^{kc,(m-k)c}) \sum_{s=1}^{c} \|\pi^{kc-s} - \pi^{kc-s+1}\| \\ &+ \sum_{k=t'+1}^{m} \tau(P^{kc,(m-k)c}) \sum_{s=1}^{c} \|\pi^{kc-s} - \pi^{kc-s+1}\|. \end{aligned}$$

According to (30), we have

$$\tau(P^{kc,(m-k)c}) \leqslant \left(\frac{k+1}{m+1}\right)^b, \tag{32}$$

where $b \in (0,1)$.

There exists some value $C' > 0$ such that

$$\sum_{k=1}^{t'} \tau(P^{kc,(m-k)c}) \sum_{s=1}^{c} \|\pi^{kc-s} - \pi^{kc-s+1}\| \leqslant \frac{C'}{(m+1)^b}.$$

For any $t \geqslant t'$, we have

$$\|\pi^{t+1} - \pi^t\| = \sum_{x \in X^*} (\pi_x^{t+1} - \pi_x^t) - \sum_{x \notin X^*} (\pi_x^{t+1} - \pi_x^t),$$

$$\sum_{x \in X^*} \pi_x^t + \sum_{x \notin X^*} \pi_x^t = 1, \quad \forall t \geqslant 0. \tag{33}$$

Thus, we have

$$\begin{aligned}\|\pi^{t+1} - \pi^t\| &= \sum_{x \in X^*} (\pi_x^{t+1} - \pi_x^t) - \left(1 - \sum_{x \in X^*} \pi_x^{t+1}\right) + \left(1 - \sum_{x \in X^*} \pi_x^t\right)\\ &= 2\left(\sum_{x \in X^*} \pi_x^{t+1} - \sum_{x \in X^*} \pi_x^t\right),\end{aligned}$$

$$\sum_{t=t'}^{\infty} \|\pi^{t+1} - \pi^t\| \leqslant 2.$$

So, for any $k \geqslant t' + 1$, we have

$$\begin{aligned}&\tau(P^{kc,(m-k)c}) \sum_{s=1}^{c} \|\pi^{kc-s} - \pi^{kc-s+1}\|\\ &= 2\tau(P^{kc,(m-k)c}) \left(\sum_{x \in X^*} \pi_x^{kc} - \sum_{x \in X^*} \pi_x^{(k-1)c}\right).\end{aligned}$$

Let $\pi_*^{kc} = 1 - \sum_{x \in X^*} \pi_x^{kc}$. By part (i) of Proposition 4, we have

$$\begin{aligned}\sum_{k=t'+1}^{m} \tau(P^{kc,(m-k)c}) \sum_{s=1}^{c} \|\pi^{kc-s} - \pi^{kc-s+1}\| &\leqslant \frac{2}{(m+1)^b} \sum_{k=t'+1}^{m} (k+1)^b (\pi_*^{(k-1)c} - \pi_*^{kc})\\ &\leqslant \frac{2}{(m+1)^b} \left[\sum_{k=t'+1}^{m} (k+1)^b \pi_*^{(k-1)c} - \sum_{k=t'+1}^{m-1} (k+1)^b \pi_*^{kc} - (t'+1)^b \pi_*^{t'c} + (t'+1)^b \pi_*^{t'c}\right]\\ &= \frac{2}{(m+1)^b} \left[\sum_{k=t'+1}^{m} ((k+1)^b - k^b) \pi_*^{(k-1)c} + (t'+1)^b \pi_*^{t'c}\right]\\ &\leqslant \frac{2}{(m+1)^b} \left[b \sum_{k=t'+1}^{m} \frac{\pi_*^{(k-1)c}}{k^{1-b}} + (t'+1)^b \pi_*^{t'c}\right].\end{aligned}$$

Let $C = C' + 2(t'+1)^b \pi_*^{t'c}$. We have

$$\|V(mc)\| \leqslant \frac{C'}{(m+1)^b} + \frac{2b}{(m+1)^b} \sum_{k=t'+1}^{m} \frac{\pi_*^{(k-1)c}}{k^{1-b}} + \frac{2(t'+1)^b \pi_*^{t'c}}{(m+1)^b} = \frac{C}{(m+1)^b} + \frac{2b}{(m+1)^b} \sum_{k=t'+1}^{m} \frac{\pi_*^{(k-1)c}}{k^{1-b}}.$$

Next, we analyze π_*^t, $t \geqslant t'$.

$$\pi_*^t = \sum_{x \notin X^*} \pi_x^t = \frac{\sum_{x \notin X^*} \mathrm{e}^{\beta^t \varphi(x)}}{\sum_{\tilde{x} \in X} \mathrm{e}^{\beta^t \varphi(\tilde{x})}} = \frac{\sum_{x \notin X^*} (\lambda t + 1)^{-\frac{\varphi(x)}{c}}}{\sum_{\tilde{x} \in X} (\lambda t + 1)^{-\frac{\varphi(\tilde{x})}{c}}}$$

$$= \sum_{x\notin X^*} \frac{1}{\sum_{\tilde{x}\in X} (\lambda t+1)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}} \leqslant \sum_{x\notin X^*} \frac{1}{\sum_{\tilde{x}\in X} (\lambda t)^{\frac{\varphi(x)-\varphi(\tilde{x})}{c}}}.$$

Let $s^* := |X^*|$, $s := |X|$. Thus, $|X\backslash X^*| = s - s^*$. Further,

$$\pi_*^t \leqslant \frac{s-s^*}{s(\lambda t)^{\delta'}} = \frac{d}{t^{\delta'}}, \tag{34}$$

where

$$\delta' = \min_{x\notin X^*} \frac{\varphi(x)-\varphi(x^*)}{c}, \quad x^* = \arg\min_{x\in X} \varphi(x), \quad d = \frac{s-s^*}{s\lambda^{\delta'}}.$$

From part (ii) of Theorem 1, $\delta' \in (0,1)$. By (34), we have

$$\begin{aligned}\|V(mc)\| &\leqslant \frac{C}{(m+1)^b} + \frac{2bd}{(m+1)^b} \sum_{k=t'+1}^{m} \frac{1}{k^{1-b}(kc-c)^{\delta'}} \\ &\leqslant \frac{C}{(m+1)^b} + \frac{2bd}{(m+1)^b c^{\delta'}} \sum_{k=t'+1}^{m} \frac{1}{(k-1)^{1-b}(k-1)^{\delta}},\end{aligned}$$

where

$$\delta = \frac{\min\{b,\delta'\}}{\min\{b,\delta'\}+1} < \min\{b,\delta'\}, \quad (b-\delta) \in (0,1).$$

According to part (i) of Proposition 4, we have

$$\begin{aligned}\|V(mc)\| &\leqslant \frac{C}{(m+1)^b} + \frac{2bd}{(m+1)^b c^{\delta'}} \sum_{k=t'}^{m} k^{(b-\delta)-1} \\ &\leqslant \frac{C}{(m+1)^b} + \frac{2bd}{(b-\delta)(m+1)^b c^{\delta'}} \sum_{k=t'-1}^{m} [(k+1)^{(b-\delta)} - k^{(b-\delta)}] \\ &\leqslant \frac{C}{(m+1)^b} + \frac{2bd}{(b-\delta)(m+1)^b c^{\delta'}} [(m+1)^{(b-\delta)} - (t'-1)^{(b-\delta)}] \\ &\leqslant \frac{C}{(m+1)^b} + \frac{2bd}{(b-\delta)(m+1)^{\delta} c^{\delta'}}.\end{aligned}$$

(iii) For the third term, for $m \geqslant t'$, we have

$$\|\pi^{mc} - \pi^*\| = 2\left(\sum_{x\in X^*} \pi_x^* - \sum_{x\in X^*} \pi_x^{mc}\right) \leqslant 2 - 2\sum_{x\in X^*} \pi_x^{mc} = 2\pi_*^{mc} \leqslant 2d/(c^{\delta'} m^{\delta'}).$$

Based on the above analysis, we have

$$\begin{aligned}\|v^{mc} - \pi^*\| &\leqslant \|v^{mc} - \pi^0 P^{0,mc}\| + \|\pi^0 P^{0,mc} - \pi^{mc}\| + \|\pi^{mc} - \pi^*\| \\ &\leqslant \frac{l}{(m+1)^b}\|v^0 - \pi^0\| + \frac{C}{(m+1)^b} + \frac{2bd}{(b-\delta)(m+1)^{\delta} c^{\delta'}} + \frac{2d}{c^{\delta'} m^{\delta'}} \\ &\leqslant \frac{l}{m^{\delta}}\|v^0 - \pi^0\| + \frac{C}{m^{\delta}} + \frac{2bd}{(b-\delta) m^{\delta} c^{\delta'}} + \frac{2d}{c^{\delta'} m^{\delta}} = \frac{\theta}{m^{\delta}},\end{aligned}$$

where

$$l \in [1,+\infty), \quad \delta = \frac{\min\{b,\delta'\}}{\min\{b,\delta'\}+1}, \quad b = \frac{1}{\lambda c 2^c |V|^c n_{\max}^c},$$
$$\delta' = \min_{x\notin X^*} \frac{\varphi(x)-\varphi(x^*)}{c}, \quad x^* = \arg\min_{x\in X} \varphi(x), \ \theta > 0.$$

Remark 3. Theorem 3 tells that the upper bound of the convergence rate of the TVBLLA is limited by parameters δ' and b. Specifically, the upper bound increases exponentially with the decrease of δ' and b. Thus, in finite time steps, i.e., $m < \infty$, if the scale of a distributed networking system is large, i.e., n is large, then δ will be small, and the convergence rate of the TVBLLA is slower.

6 Simulation experiment and analysis

To evaluate the effectiveness and advantage of the TVBLLA, we consider (1) the comparison of the TVBLLA with the MBR and relaxed greedy and memory-based algorithm (RGMA), where the memory length $\mathrm{ml} = 1$ for the MBR and RGMA, since we may regard the TVBLLA as an optimization algorithm with the memory length of 1 as well; (2) the comparison of the TVBLLA with some representative existing optimization algorithms, where we set the memory length $\mathrm{ml} > 1$ for the MBR and RGMA to achieve their best performance; (3) the comparison of the TVBLLA with a heuristic local search algorithm on three benchmarks.

The constrained strategy set $C_i(x_i^t)$ is given by X_i, $\forall i \in V$, which satisfies Assumptions 1 and 2, and the parameter μ is set to $(n+1)/2$. All numerical simulations are realized on the same computer with 3.20-GHz CPU and 16.0-G RAM. For each algorithm, we independently repeat 100 runs of the simulations on each network for each data of $f(x)$, and obtain the minimum value $f_{\min}$, the average value $\bar{f}$, the maximum value $f_{\max}$, the range $R = f_{\max} - f_{\min}$, and the standard deviation σ.

According to (22), the minimum value $\bar{c}$ will tend to infinity in the large-scale networks, which causes the time-varying factor β^t decreases very slowly. Due to the finite time in practice, we relax the parameter c with a preferred convergent rate as proved in Theorem 3 in the following simulation experiments; that is, c takes a small value. Coefficient λ is set to $1\mathrm{E}+8$.

6.1 Comparison with the MBR and RGMA (ml = 1)

In this subsection, we compare the TVBLLA with the MBR and RGMA on a variety of representative complex networks, including the ER random networks [38], the WS small-world networks [8], the BA scale-free networks [9], and the 2-D grid (regular lattices) networks [39]. The four types of networks are described as follows.

(1) ER $n\langle k\rangle$ denotes the ER networks with n vertices, where $\langle k\rangle$ is the average degree of a general complex network.

(2) WS(I) and WS(II): For WS(I), we select $\mathrm{pr} = 0.1$; and for WS(II), $\mathrm{pr} = 0.5$, where pr is the probability to randomly rewire each edge of a small-world network.

(3) BA(I) and BA(II): The Π_i of BA(I) is $\frac{k_i}{\sum_j k_j}$; and Π_i of BA(II) is $\frac{\ln k_i}{\sum_j \ln k_j}$, where Π_i is the probability that a new vertex is connected to an existing vertex i.

(4) Grid n denotes regular lattices with n vertices.

In addition, the MBR and RGMA are summarized as follows.

MBR [24, 40]. The main idea of the MBR is that each vertex rationally records the best responses of the latest step into its memory, which will randomly select a strategy from the memory for the next time step.

RGMA [27]. The main idea of the RGMA is that each vertex records the temporal strategy by the relaxed greedy rule of the latest step into its memory, which will select a strategy from the memory with probability ρ for the next time step.

The memory length ml of the MBR and RGMA is set to 1, and the mutation probability ρ of the RGMA is set to 1 (a larger ρ can lead to a better solution [27]). The number of time steps of each run for three algorithms in this subsection, i.e., the TVBLLA, the MBR, and the RGMA is set to the same as $8\mathrm{E}+7$, and the results are recorded in Table 1. From Table 1, we can see that the $\bar{f}$ of the TVBLLA is much smaller than that of the RGMA on all examples. Moreover, the MBR with $\mathrm{ml} = 1$ fails to reach a stable state of all examples even after the maximum number of running time steps.

6.2 Comparison with the existing typical algorithms

We compare the TVBLLA with several optimization algorithms on a variety of networks including the MBR, RGMA, GMA, BLLA, and GA, where the GMA, BLLA, and GA are summarized as follows.

GMA. This is the game-based memetic algorithm (which is a hybrid algorithm) [41]. Firstly, each chromosome locally evolves le times by the asynchronous updating rule, and repeats each chromosome evolution $g_{\max}^{\mathrm{GMA}}$ generations.

BLLA [29]. Here, $\bar{\beta}$ is fixed as a constant as compared with the TVBLLA. If $\bar{\beta} \to -\infty$, each player tends to choose the best response.

GA. The genetic algorithm is a typical centralized algorithm [42–44], and the fitness function is set by (9).

Table 1 Comparison of the TVBLLA, MBR, and RGMA: the average value $\bar{f}$ a)

Algorithms ml=1	Networks							
	ER $(n/\langle k\rangle/c)$				WS(I) $(n/\langle k\rangle/c)$			
	500/4 /1E−3	1000/4 /5E−4	1000/8 /3E−4	2000/4 /2.5E−4	500/4 /1.5E−3	1000/4 /1E−3	1000/8 /5E−4	2000/4 /4E−5
TVBLLA	**261.87**	**529.33**	**650.83**	**1059.15**	**318.02**	**638.72**	**752.93**	**1278.01**
RGMA	295.31	588.16	715.82	1187.09	348.41	687.27	801.18	1385.06
MBR	–	–	–	–	–	–	–	–
	WS(II) $(n/\langle k\rangle/c)$				BA(I) $(n/\langle k\rangle/c)$			
	500/4 /1.5E−3	1000/4 /6E−4	1000/8 /4.5E−4	2000/4 /3E−4	500/4 /3E−4	1000/4 /5E−5	1000/8 /3E−5	2000/4 /2.5E−5
TVBLLA	**282.46**	**568.88**	**683.57**	**1141.57**	**207.81**	**419.71**	**540.19**	**840.01**
RGMA	320.34	627.05	743.30	1266.91	234.62	455.83	611.72	897.28
MBR	–	–	–	–	–	–	–	–
	BA(II) $(n/\langle k\rangle/c)$				Grid (n/c)			
	500/4 /3E−4	1000/4 /1.5E−4	1000/8 /1.5E−4	2000/4 /5E−5	100 /2E−2	500 /4E−3	1000 /2E−3	2000 /6E−4
TVBLLA	**223.82**	**458.27**	**598.33**	**908.64**	**50.27**	**251.56**	**503.24**	**1005.53**
RGMA	248.86	492.29	653.07	996.43	64.33	305.69	626.11	1251.58
MBR	–	–	–	–	–	–	–	–

a) The bold numbers are the best results.

Table 2 Comparison of the TVBLLA, MBR, RGMA, GMA, BLLA, and GA: the average value $\bar{f}$/range R/standard deviation σ a)

Networks		Algorithms					
		TVBLLA	MBR	RGMA	GMA	BLLA	GA
ER $(n/\langle k\rangle)$	500/4	**261.87/3/0.86**	268.67/8/2.03	271.39/10/2.64	273.87/5/1.28	279.28/8/1.98	310.90/11/3.03
	1000/4	**529.41/4/1.18**	539.04/9/2.71	543.42/11/2.88	548.33/6/1.77	562.84/9/2.61	622.48/18/4.35
	1000/8	**650.99/4/1.23**	667.57/12/3.34	675.30/17/4.17	682.31/7/2.02	695.42/10/2.92	788.90/22/5.37
	2000/4	**1059.15/4/1.45**	1072.36/15/3.98	1088.54/18/4.45	1107.88/8/2.11	1122.73/19/4.67	1254.91/32/7.75
WS(I) $(n/\langle k\rangle)$	500/4	**318.02/4/1.35**	323.14/6/1.84	326.11/7/2.07	329.45/5/1.56	336.38/8/2.15	357.30/11/3.16
	1000/4	**638.81/4/1.41**	649.16/9/2.53	655.97/10/2.75	664.73/5/1.58	674.08/9/2.72	725.10/16/4.01
	1000/8	**752.95/5/1.62**	765.65/11/2.98	775.20/12/3.20	786.70/6/1.92	790.62/10/2.98	848.90/21/5.25
	2000/4	**1278.01/5/1.84**	1299.16/15/3.92	1321.41/18/4.35	1338.55/10/2.95	1353.26/25/6.66	1455.36/29/7.21
WS(II) $(n/\langle k\rangle)$	500/4	**282.46/3/0.81**	291.50/6/1.85	294.90/8/2.20	296.80/5/1.53	303.94/6/1.81	332.32/10/2.85
	1000/4	**568.85/4/1.03**	584.68/7/2.08	589.54/8/2.35	593.04/5/1.86	607.58/9/2.53	672.56/15/3.87
	1000/8	**683.59/4/1.27**	711.44/8/2.32	723.64/9/2.70	728.02/6/2.01	735.48/11/3.01	799.50/20/5.06
	2000/4	**1141.57/6/1.42**	1164.59/13/3.78	1178.11/17/4.15	1198.90/12/3.23	1219.48/19/4.86	1343.61/27/6.91
BA(I) $(n/\langle k\rangle)$	500/4	**207.81/3/0.63**	214.60/5/1.42	216.78/6/1.83	215.01/4/1.22	219.08/6/1.70	246.60/9/2.53
	1000/4	**419.75/3/0.75**	426.90/6/1.83	432.08/8/2.09	436.89/4/1.31	444.90/9/2.75	506.36/11/3.02
	1000/8	**540.22/4/1.11**	556.82/9/2.47	566.36/13/3.29	572.40/5/1.52	580.20/10/3.02	662.83/16/4.05
	2000/4	**840.01/4/1.33**	850.60/11/3.31	862.30/14/3.70	891.26/6/1.86	896.60/15/3.91	1029.27/23/5.45
BA(II) $(n/\langle k\rangle)$	500/4	**223.82/3/0.77**	231.58/4/1.36	235.92/6/1.77	236.71/3/1.14	239.88/8/2.08	281.70/10/2.79
	1000/4	**458.34/3/0.99**	468.68/7/2.14	472.44/10/2.83	475.55/4/1.38	484.80/11/3.11	568.14/12/3.31
	1000/8	**598.39/4/1.05**	611.64/10/2.84	618.46/14/3.78	624.15/5/1.67	640.44/12/3.37	744.55/17/4.22
	2000/4	**908.64/4/1.37**	928.75/11/3.19	939.66/13/3.18	949.18/8/2.03	973.59/16/4.08	1144.32/25/6.37
Grid (n)	100	**50.27/1/0.55**	55.28/9/2.50	56.24/10/2.63	56.33/4/1.22	59.96/8/2.02	64.63/20/4.98
	500	**251.56/4/0.96**	278.54/20/5.13	283.46/22/5.64	287.07/9/2.78	299.52/10/2.89	323.44/27/7.04
	1000	**503.28/5/1.23**	556.14/24/6.31	568.26/28/7.27	576.21/11/3.23	605.02/12/3.32	656.10/30/8.25
	2000	**1004.53/7/1.85**	1122.68/27/7.30	1154.88/29/7.41	1167.41/12/3.45	1218.05/18/4.35	1312.24/42/10.89

a) The bold numbers are the best results.

Since each algorithm has different parameters, we present the best effect of parameters on each algorithm, whose parameters are given in detail as follows.

(1) MBR. ml $= 20$.

(2) RGMA. ml $= 20$, $\rho = 1$.

(3) GMA. The population size: $m_{\text{size}}^{\text{GMA}} = 100$; the times of local evolution: le $= 10$; the mutation rate: 0.5.

(4) BLLA. $\bar{\beta} = -1\text{E}+5$.

(5) GA. The population size: $m_{\text{size}}^{\text{GA}} = 100$; the mutation rate: 0.5.

The $\bar{f}$ and R of six algorithms are recorded in Table 2, where $\bar{f}$ of the TVBLLA is from Table 1. Note that in all collected experiment data, the running time steps of all algorithms are set to $8\text{E}+7$. The ratio of $f_{\min}$, $\bar{f}$ and $f_{\max}$ of each algorithm to the network size n is plotted in Figures 1–4.

As shown in Table 2, the TVBLLA obtains the best average value $\bar{f}$ on all exampled networks, and the $\bar{f}$ of the MBR is smaller than those of the other four algorithms. Moreover, the TVBLLA also performs the best range R and the standard deviation σ on all examples. Specifically, the R and σ of the GMA is smaller than those of the other four algorithms.

We further examine the convergence rates of all exampled algorithms, which present a reverse trend as compared with those of $\bar{f}$ and R. The $\bar{f}$ of different algorithms on the networks ER, WS(II), BA(II) and Grid are shown in Figures 5–8. Note that the the minimum number of covered vertices of the Grid

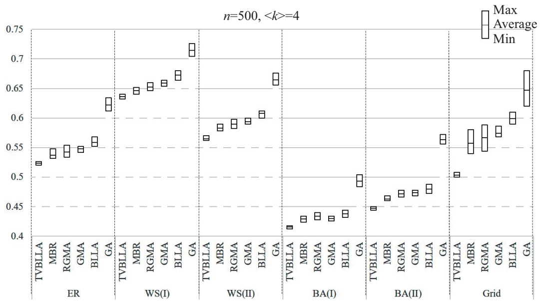

Figure 1 The ratio of $f_{\min}$, $\bar{f}$ and $f_{\max}$ of each algorithm to the network size $n = 500$, where the average degree $\langle k\rangle$ of ER, WS, and BA networks is 4.

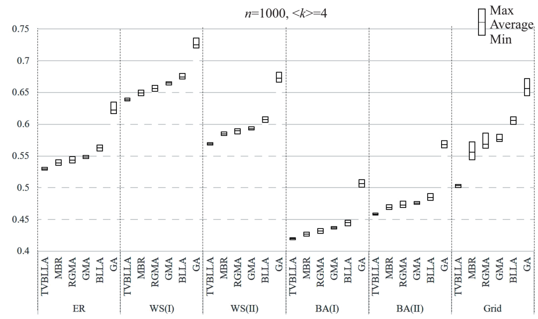

Figure 2 The ratio of $f_{\min}$, $\bar{f}$ and $f_{\max}$ of each algorithm to the network size $n = 1000$, where the average degree $\langle k\rangle$ of ER, WS, and BA networks is 4.

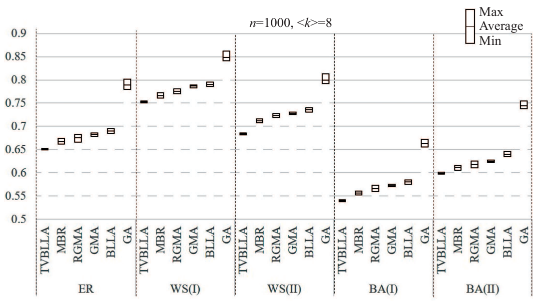

Figure 3 The ratio of $f_{\min}$, $\bar{f}$ and $f_{\max}$ of each algorithm to the network size $n = 1000$, where the average degree $\langle k\rangle$ of ER, WS, and BA networks is 8.

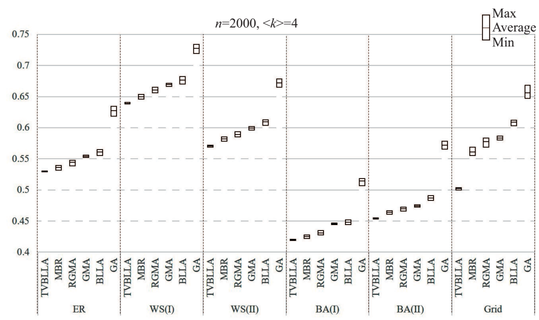

Figure 4 The ratio of $f_{\min}$, $\bar{f}$ and $f_{\max}$ of each algorithm to the network size $n = 2000$, where the average degree $\langle k\rangle$ of ER, WS, and BA networks is 4.

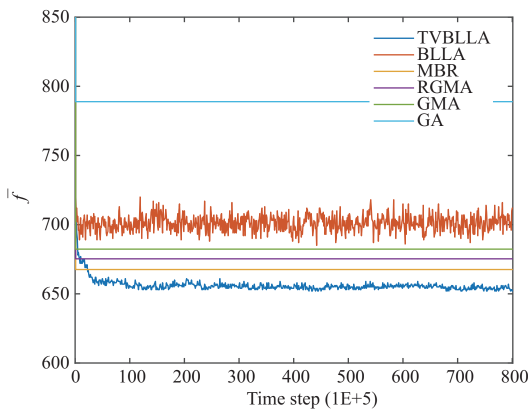

Figure 5 The ER network. Comparison among different algorithms, where $\bar{f}$ is sampled every 1E + 5 time steps.

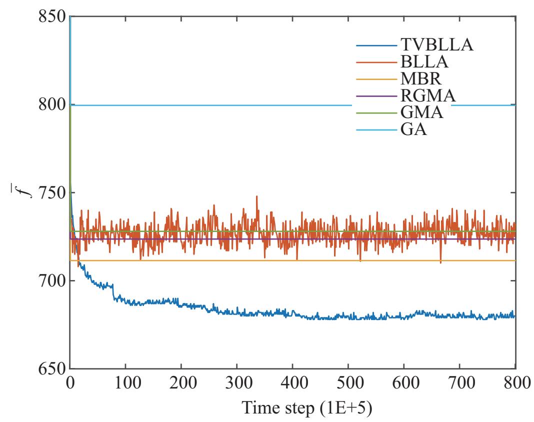

Figure 6 The WS(II) network. Comparison among different algorithms, where $\bar{f}$ is sampled every 1E + 5 time steps.

1000 is 500. As shown in Figures 5–8, we can see that, although the convergence rates of the TVBLLA and BLLA are slower than those of the other algorithms, the TVBLLA can achieve the best results in terms of $\bar{f}$ in finite time steps.

6.3 Comparison with the heuristic local search algorithm on three standard benchmarks

We also compare the proposed TVBLLA with a heuristic local search algorithm for solving the vertex cover problem, i.e., the so-called MetaVC, on three standard benchmarks, including DIMACS-HARD[1)],

1) http://lcs.ios.ac.cn/ caisw/Resource (DIMACS%20complementary%20graphs.tar.gz).

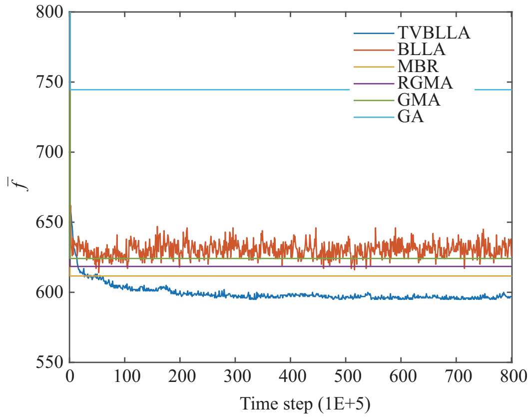

Figure 7 The BA(II) network. Comparison among different algorithms, where $\bar{f}$ is sampled every $1\mathrm{E}+5$ time steps.

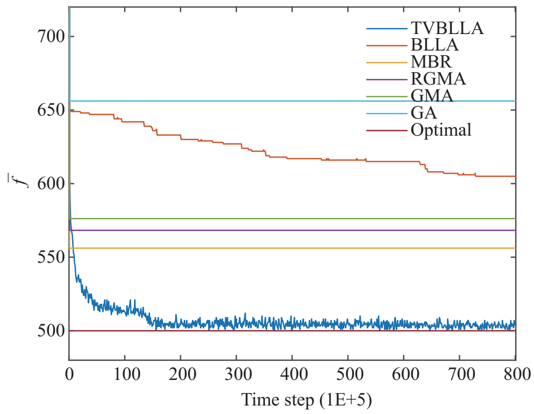

Figure 8 The Grid network. Comparison among different algorithms, where $\bar{f}$ is sampled every $1\mathrm{E}+5$ time steps.

Table 3 Comparison of the TVBLLA and MetaVC on the three benchmarks

Benchmarks	Network name	Optimal value	Algorithms	
			TVBLLA	MetaVC
			success rate: sr (%)/average run time (s): $\bar{T}$	
DIMACS-HARD	brock400_4	367	95/500.61	100/0.17
	brock800_4	774	93/503.26	100/201.06
	MANN_a45	690	94/501.37	100/20.02
	brock400_2	371	95/498.82	100/2.81
	brock800_2	776	92/496.35	67/708.59
	C2000.9	1920	94/500.13	10/1102.35
	C4000.5	3982	92/502.98	100/89.71
	MANN_a81	2221	91/499.65	87/606.83
			TVBLLA	MetaVC
			success rate: sr (%)/average run time (s): $\bar{T}$	
BHOSLIB-HARD	frb53-24-1	1219	93/500.82	92/545.36
	frb53-24-2	1219	94/501.36	100/105.88
	frb53-24-3	1219	95/500.08	100/26.72
	frb53-24-4	1219	93/499.67	100/126.73
	frb53-24-5	1219	94/502.81	100/19.60
	frb56-25-1	1344	92/500.96	98/345.67
	frb56-25-2	1344	92/498.68	95/402.53
	frb56-25-3	1344	93/497.64	100/58.12
	frb56-25-4	1344	95/498.32	100/23.84
	frb56-25-5	1344	94/503.67	100/11.82
	frb59-26-1	1475	92/497.99	85/574.56
	frb59-26-2	1475	94/501.23	45/1021.83
	frb59-26-3	1475	93/500.64	100/302.36
	frb59-26-4	1475	95/499.78	92/475.91
	frb59-26-5	1475	94/502.64	100/29.18
			TVBLLA	MetaVC
			$f_{\min}(\bar{f})$/average run time (s): $\bar{T}$	
REAL-WORLD-HARD	socfb-CMU	4986	4986(4988.92)/1998.15	4986(4986.00)/282.28
	web-webbase-2001	2651	2651(2652.88)/1006.35	2652(2652.00)/1.54
	socfb − UCSB37	11261	11261(11263.09)/3008.72	11261(11261.06)/172.14
	socfb-UConn	13230	13230(13234.71)/3496.53	13230(13230.05)/501.06

BHOSLIB-HARD[2)], and REAL-WORLD-HARD[3)]. Ref. [45] reported the MetaVC achieved the best performance on most of the testing instances. The comparison results are recorded in Table 3. The sr is the number of times and the optimal value was successfully obtained divided by the total number of runs [45], and c is set to $1\mathrm{E}-4$. From Table 3, we can see that for benchmarks DIMACS-HARD and BHOSLIB-HARD, the sr values of TVBLLA are all over 90%, while those values of MetaVC are quite unstable with variant performance. For benchmark REAL-WORLD-HARD, although the $\bar{f}$ values of TVBLLA are larger than those of the MetaVC, they are also very close to the optimal value. In sum, although the MetaVC performs well in some of the examples, the TVBLLA achieves much more stable and satisfactory performance as compared with the MetaVC in finite time on all examples. Note that on the comparison of the average run time of the TVBLLA and the MetaVC for the most instances of Table 3, we record that the average run time of the TVBLLA is much longer than that of the MetaVC, which

2) http://lcs.ios.ac.cn/ caisw/Resource (benchmarks/graph-benchmarks.htm).
3) http://lcs.ios.ac.cn/ caisw/Resource (realword%20graphs.tar.gz).

is mainly caused by two reasons: (i) The updating/learning mechanism is different; i.e., the updating mechanism of the TVBLLA is a serial updating mechanism, while the MetaVC is a parallel updating mechanism; (ii) The parameters of the MetaVC need to be pre-tuned to reach the optimial configuration, while the average run time of the MetaVC does not include the time spent on tuning the parameters; yet, the TVBLLA does not require such a pre-tuned/training phase.

7 Conclusion

To solve the vertex cover problem in complex network systems, many distributed optimization algorithms have been presented. However, a distributed optimization algorithm realizing the MVC state of a general complex network is unsatisfied. This paper proposed the TVBLLA for solving the vertex cover problem from a potential game perspective, proved that the solutions to the minimum value of the potential function are the MVC states of a general complex network, whose convergence to the MVC state of a general complex network has been guaranteed and verified by extensive examples of general complex networks.

Since the proposed TVBLLA requires only one player to update its strategy at each time step, further attention is paid to ensure each vertex updates its strategy parallelly. In more practical situations, designing new distributed optimization algorithms to achieve robust self-stabilization performance [46, 47] is desired.

Acknowledgements This work was partly supported by the National Natural Science Foundation of China (Grant Nos. 61751303, 71731004) and National Science Fund for Distinguished Young Scholars of China (Grant No. 61425019).

References

1 Hochbaum D S. Approximation algorithms for the set covering and vertex cover problems. SIAM J Comput, 1982, 11: 555–556
2 Xu E, Ding Z, Dasgupta S. Target tracking and mobile sensor navigation in wireless sensor networks. IEEE Trans Mobile Comput, 2013, 12: 177–186
3 Tan Y, Ding K. A survey on GPU-based implementation of swarm intelligence algorithms. IEEE Trans Cybern, 2016, 46: 2028–2041
4 Yu J J Q, Lam A Y S. Autonomous vehicle logistic system: joint routing and charging strategy. IEEE Trans Intell Transp Syst, 2018, 19: 2175–2187
5 Ansere J A, Han G, Liu L, et al. Optimal resource allocation in energy-efficient Internet-of-Things networks with imperfect CSI. IEEE Internet Things J, 2020, 7: 5401–5411
6 Bai X L, Yun Z Q, Xuan D, et al. Optimal patterns for four-connectivity and full coverage in wireless sensor networks. IEEE Trans Mobile Comput, 2010, 9: 435–448
7 Coppersmith D, Vishkin U. Solving NP-hard problems in 'almost trees': vertex cover. Discrete Appl Math, 1985, 10: 27–45
8 Watts D J, Strogatz S H. Collective dynamics of 'small-world' networks. Nature, 1998, 393: 440–442
9 Barabási A L, Albert R. Emergence of scaling in random networks. Science, 1999, 286: 509–512
10 Wang X F, Chen G R. Complex networks: small-world, scale-free and beyond. IEEE Circ Syst Mag, 2003, 3: 6–20
11 Newman M E J. The structure and function of complex networks. SIAM Rev, 2003, 45: 167–256
12 Boccaletti S, Latora V, Moreno Y, et al. Complex networks: structure and dynamics. Phys Rep, 2006, 424: 175–308
13 Halperin E. Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs. SIAM J Comput, 2002, 31: 1608–1623
14 Karakostas G. A better approximation ratio for the vertex cover problem. In: Proceedings of International Colloquium on Automata, Languages, and Programming. Berlin: Springer, 2005. 1043–1050
15 Wang J X, Li W J, Li S H, et al. On the parameterized vertex cover problem for graphs with perfect matching. Sci China Inf Sci, 2014, 57: 072107
16 Qiu Z P, Wang P B. Parameter vertex method and its parallel solution for evaluating the dynamic response bounds of structures with interval parameters. Sci China Phys Mech Astron, 2018, 61: 064612
17 Khuri S, Bäck T. An evolutionary heuristic for the minimum vertex cover problem. In: Proceedings of Genetic Algorithms within the Framework of Evolutionary Computation, 1994. 86–90
18 Kratsch S, Neumann F. Fixed-parameter evolutionary algorithms and the vertex cover problem. Algorithmica, 2013, 65: 754–771
19 Oliveto P S, He J, Yao X. Analysis of the $(1+1)$-EA for finding approximate solutions to vertex cover problems. IEEE Trans Evol Computat, 2009, 13: 1006–1029
20 Friedrich T, He J, Hebbinghaus N, et al. Approximating covering problems by randomized search heuristics using multi-objective models. Evolary Computation, 2010, 18: 617–633
21 Chang W-L, Ren T-T, Feng M. Quantum algorithms and mathematical formulations of biomolecular solutions of the vertex cover problem in the finite-dimensional hilbert space. IEEE Transon Nanobiosci, 2015, 14: 121–128
22 Li H S. Quantum vertex algebras and quantum affine algebras. Sci Sin Math, 2017, 47: 1423–1440
23 Weigt M, Hartmann A K. Typical solution time for a vertex-covering algorithm on finite-connectivity random graphs. Phys Rev Lett, 2001, 86: 1658–1661
24 Yang Y, Li X. Towards a snowdrift game optimization to vertex cover of networks. IEEE Trans Cybern, 2013, 43: 948–956
25 Li A, Tang C B, Li X. An evolutionary game optimization to vertex cover of dynamic networks. In: Proceedings of the 33rd Chinese Control Conference, 2014. 2757–2762
26 Tang C, Li A, Li X. Asymmetric game: a silver bullet to weighted vertex cover of networks. IEEE Trans Cybern, 2018, 48: 2994–3005

27 Sun C, Sun W, Wang X, et al. Potential game theoretic learning for the minimal weighted vertex cover in distributed networking systems. IEEE Trans Cybern, 2019, 49: 1968–1978
28 Vetta A. Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions. In: Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. 416–425
29 Arslan G, Marden J R, Shamma J S. Autonomous vehicle-target assignment: a game-theoretical formulation. J Dynamic Syst Measurement Control, 2007, 129: 584–596
30 Nash J F. Equilibrium points in n-person games. Proc Natl Acad Sci USA, 1950, 36: 48–49
31 Monderer D, Shapley L S. Potential games. Games Economic Behav, 1996, 14: 124–143
32 Hajnal J, Bartlett M S. Weak ergodicity in non-homogeneous Markov chains. Math Proc Camb Phil Soc, 1958, 54: 233–246
33 Dobrushin R L. Central limit theorem for nonstationary Markov chains. I. Theor Probab Appl, 1956, 1: 65–80
34 Isaacson D L, Madsen R W. Markov Chains: Theory and Applications. New York: Wiley, 1976
35 An B, Lesser V. Characterizing contract-based multiagent resource allocation in networks. IEEE Trans Syst Man Cybern B, 2010, 40: 575–586
36 Young P H. Individual Strategy and Social Structure: An Evolutionary Theory of Institutions. Princeton: Princeton University Press, 1998
37 Tatarenko T. Log-linear learning: convergence in discrete and continuous strategy potential games. In: Proceedings of the 53rd IEEE Conference on Decision and Control, 2014. 426–432
38 Erdős P, Rényi A. On the evolution of random graphs. Publ Math Inst Hung Acad Sci, 1960, 5: 17–60
39 Szabó G, Fáth G. Evolutionary games on graphs. Phys Rep, 2007, 446: 97–216
40 Young H P. The evolution of conventions. Econometrica, 1993, 61: 57–84
41 Wu J, Shen X, Jiao K. Game-based memetic algorithm to the vertex cover of networks. IEEE Trans Cybern, 2019, 49: 974–988
42 Bhasin H, Ahuja G. Harnessing genetic algorithm for vertex cover problem. Int J Comput Sci Eng, 2012, 4: 218–223
43 Renders J M, Flasse S P. Hybrid methods using genetic algorithms for global optimization. IEEE Trans Syst Man Cybern B, 1996, 26: 243–258
44 Juang C F. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. IEEE Trans Syst Man Cybern B, 2004, 34: 997–1006
45 Luo C, Hoos H H, Cai S, et al. Local search with efficient automatic configuration for minimum vertex cover. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence, 2019. 1297–1304
46 Radenkovic M S, Michel A. Robust adaptive systems and self stabilization. IEEE Trans Automat Contr, 1992, 37: 1355–1369
47 Klinkhamer A, Ebnenasir A. Shadow/puppet synthesis: a stepwise method for the design of self-stabilization. IEEE Trans Parallel Distrib Syst, 2016, 27: 3338–3350