Article

Dynamic Community Discovery Method Based on Phylogenetic Planted Partition in Temporal Networks

Xiaoyang Liu [1,*] **, Nan Ding** [1] **, Giacomo Fiumara** [2] **, Pasquale De Meo** [3] **and Annamaria Ficara** [4]

[1] School of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054, China; dr.ding_1901@2019.cqut.edu.cn

[2] MIFT Department, University of Messina, Viale F. S. D'Alcontres 31, 98166 Messina, Italy; giacomo.fiumara@unime.it

[3] Department of Ancient and Modern Civilizations, University of Messina, Viale G. Palatucci 13, 98168 Messina, Italy; pasquale.demeo@unime.it

[4] Department of Mathematics and Computer Science, University of Palermo, Via Archirafi 34, 90123 Palermo, Italy; aficara@unime.it

[*] Correspondence: lxy3103@163.com

Abstract: As most of the community discovery methods are researched by static thought, some community discovery algorithms cannot represent the whole dynamic network change process efficiently. This paper proposes a novel dynamic community discovery method (Phylogenetic Planted Partition Model, PPPM) for phylogenetic evolution. Firstly, the time dimension is introduced into the typical migration partition model, and all states are treated as variables, and the observation equation is constructed. Secondly, this paper takes the observation equation of the whole dynamic social network as the constraint between variables and the error function. Then, the quadratic form of the error function is minimized. Thirdly, the Levenberg–Marquardt (L–M) method is used to calculate the gradient of the error function, and the iteration is carried out. Finally, simulation experiments are carried out under the experimental environment of artificial networks and real networks. The experimental results show that: compared with FaceNet, SBM + MLE, CLBM, and PisCES, the proposed PPPM model improves accuracy by 5% and 3%, respectively. It is proven that the proposed PPPM method is robust, reasonable, and effective. This method can also be applied to the general social networking community discovery field.

Keywords: temporal networks; community discovery; phylogenetic evolution; planted of partition

1. Introduction

1.1. Background

Complex network analysis is an interdisciplinary research field which can be applied in a lot of areas such as computer science [1,2] and social, biological and physical sciences [3–5], and it is capturing the attention of many scholars. A complex network is a simple graph defined as a set of nodes connected by a set of edges. Nodes can represent individuals or organizations. Edges are relational ties between two nodes, e.g., friendship relationships between two social users. Graphs are one of the most important and powerful data structures. Complex network analysis and modeling can be used to reveal patterns of social interaction, to study recommendation systems, or protein complexes and protein functional modules. By far the most basic tasks in complex networks are node identification, link prediction, and information dissemination. These tasks have received extensive research and attention. In addition, community structure discovery is also one of the most important tasks; it is usually defined as identifying tightly connected subgraphs from a complex network. Because communities help to reveal the structure–function relationship of the network, it has been studied extensively. For example, communities within cancer networks mark key pathways associated with cancer progression [6], and the communities in the

multi-layer transportation network correspond to common practices, which provides clues for airline management [7]. Therefore, a great deal of work has been carried out in the discovery of communities in the network [8–10]. A lot of work has been proposed for community discovery, existing algorithms either optimize predefined quantitative functions or acquire potential feature matrices for community detection. Typical methods include modularization-based methods [11], model-based methods [12,13] and random-walk-based methods [14–16]. S. Fortunato et al. [17,18] have conducted a comprehensive survey.

However, all these methods assume that the target network is static and ignore the timeliness of the network. In reality, many networks from society and nature are dynamic, meaning that the network structure changes over time; that is, it performs the dynamic network. More specifically, in a dynamic network, nodes may appear or disappear over time, and links between two nodes may also appear or disappear. For example, inter-personal relations often change due to individual behavior [19]. For another example, tumor cell migration leads to metastasis, which is crucial for the diagnosis and treatment of tumors [20]. Therefore, it is worthwhile to track how a community evolves in a dynamic network (also known as a dynamic or evolutionary community).

For dynamic network modeling, the most widely used method is to introduce explicit smoothing frameworks, which quantifies the similarity between snapshots in two subsequent steps by introducing the Temporal Smoothed Framework (TSF). Various TSF-based algorithms have been proposed to evolve the community by extending the static community discovery algorithm. For example, for topological connectivity, the Kim-Han algorithm [21] found dynamic communities by optimizing modularity, and the DYNMOGA method [22] is presented for the multi-objective genetic algorithm to simultaneously optimize clustering accuracy and clustering drift. Regarding matrix decomposition, the ESPC method [23] is used with matrix spectrum, the ECKF method [24] is proposed by using kernel ENMF, and the Se-NMF method [25] is used by a semi-supervised strategy to develop community testing. The Gr-NMF method [26] is adopted by graph-regularized NMF for community discovery in evolution. In the probabilistic model: FaceNet method [27] is researched by using Maximum a Posteriori (MAP) estimate, DSBM method [28] adopted the Bayesian method to obtain the evolving community by extending the random block model. According to the existing literature [29], there are six evolutionary events in the community (as shown in Figure 1), including birth, death, growth, contraction, merging, and splitting. Sometimes, a seventh event is added to these, i.e., continuing. Finally, an eight event was proposed by Cazabet and Amblard [30] and it is resurgence. A generic dynamic community discovery algorithm does not necessarily have to handle all these events, which can be differently managed in different works [31].
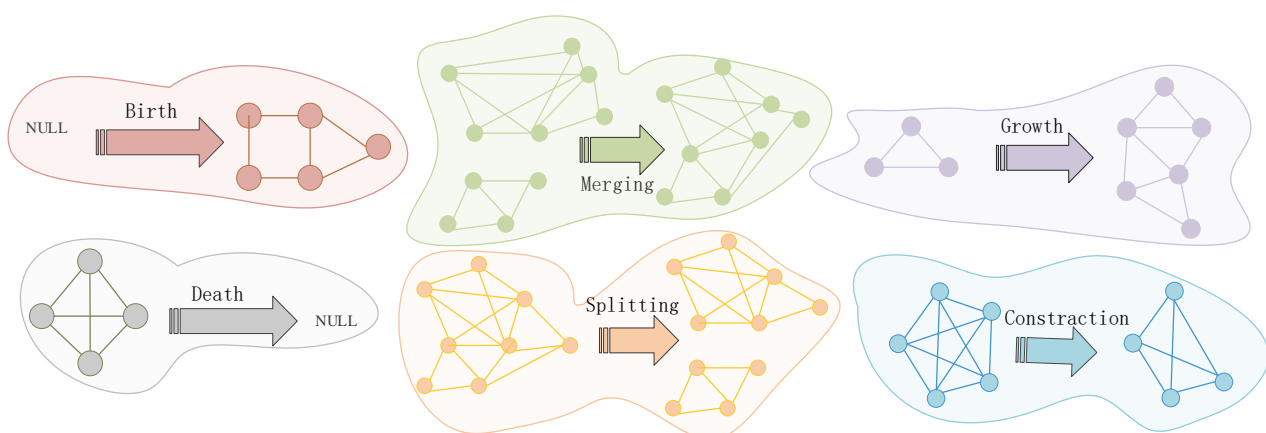

Figure 1. 6 evolutionary events in dynamic communities.

1.2. Motivation

While much work has been carried out to address the problem of dynamic community discovery, there are still some issues that need to be addressed.

Firstly, most existing dynamic network models assume that it is a hidden Markov structure, in this structure, when the current network state is given, a network snapshot at any given time is conditionally independent for all previous snapshots. This approach may not be flexible enough to replicate some of the observations in real network data.

Secondly, the dynamic system is used by filters, even for Gaussian distributions. However, after a nonlinear transformation, Gaussian terms are lost. Mean and covariance are the only measures computed by the filter. This is the result of a nonlinear transformation approximated by Gauss, and therefore this approximation may be poor.

Finally, how to combine the information of the community structure available at the previous moment with the information available at the current moment is an important question. In the traditional hidden Markov dynamic Bayesian network model, the probability of an edge appearing in a dynamic network is realized by the estimated state.

Therefore, this paper proposes a phylogenetic planted partition method, which uses the graph optimization strategy to continuously discover the evolving communities.

1.3. Our Work and Contributes

The main contributions of this paper can be summarized as follows:

(1) The time dimension is introduced into the typical stochastic block-model, and all states in the whole dynamic network system are treated as variables, and the observation equation is taken as the constraint between variables to construct an error function about the whole dynamic network system.

(2) By adopting the graph-based optimization strategy, the constraints in the entire motion trajectory can be considered once. In the linearization process, only the Jacobian matrix is calculated, and the calculation process is also relative to the entire motion trajectory. Therefore, the entire system evolution process is transformed into the nonlinear system optimization process.

(3) In natural ecosystems, inspired by the evolutionary thinking of species populations and combined with the typical probability model of stochastic block-model in community discovery, a phylogenetic planted partition method (PPPM) for dynamic community discovery is proposed.

(4) The proposed PPPM method in the two scenarios of artificial network and the real network is verified by experiments, which proves that the performance of the novel method is better than the four state-of-the-art methods (FaceNet, SBM + MLE, CLBM, and PisCES).

1.4. Organization

The remainder of this paper is structured as follows: In Section 2, related work is discussed; Section 3 introduces the proposed model in detail, describes the proposed PPPM method, and gives the derivation process. In Section 4, the experimental results of the novel PPPM method in the artificial dynamic network and real dynamic network are presented, and then compared with other existing models. Finally, in Section 5, some conclusions are given and future directions are discussed.

2. Related Work

According to the research of Aynaud et al. [32], the dynamic community discovery algorithms can be divided into four categories: coupling network, two-stage algorithm, evolutionary clustering, and probability model. However, Hartmann et al. [33] believed that all existing dynamic community discovery methods can be identified as online or offline methods. Rossetti and Cazabet [31] proposed a new survey on community detection in dynamic networks, which proposed the unique functions and challenges of dynamic community discovery algorithms.

The first kind of coupled network-based algorithm firstly builds the network by fusing edges at different times. Then, the classical static community detection algorithm is used to find the communities in the coupled network. For example, Agarwal et al. [34] discovered the ongoing events in the microblog message flow by adding edges between vertex instances at different times to build the coupling network, in which the dynamic community corresponds to the community in the built network. Because coupled networks cannot fully describe the dynamic characteristics of networks, these algorithms have been shown to accurately discover only short-cycle communities. To overcome this problem, the second kind of two-stage algorithm separated the community detection from the community dynamic, avoiding the coupling of the dynamic network.

Specifically, these algorithms used static community detection algorithms to find the community each time and then connected the community the next two times to extract the evolving community. Typical algorithms included GraphScope [35] and TRMMC [36] coupled networks and two-stage algorithms that detected dynamic communities in a dynamic network by simply extending static community detection methods and detecting dynamic communities in each operation dynamic network or static community. In general, these algorithms can achieve better performance in the case of weak network dynamics. In this case, the dynamic update method can accurately identify the dynamic community without running the community detection algorithm each time, and only need to update the previously discovered community. However, the accuracy of these algorithms is low.

The third type of dynamic community discovery method is related to clustering evolutionary, which is proposed by Chakrabarti et al. [37]. They extracted the implicit community structure in each snapshot, which is one of the most widely used methods for dynamic community discovery. The evolutionary clustering algorithm adopted the assumption of time smoothness. The community structure will not change much over a continuous-time slice. This time smoothing method can be used to overcome the randomness. Compared with other algorithms, the evolutionary community discovery algorithm aims to discover a smooth sequence of communities in a series of network snapshots (as can be seen in Figure 2). The overall objective function of the evolutionary algorithm can be decomposed into two parts: Snapshot Cost (*CS*) and Temporal Cost (*CT*) [38].

$$Cost = \alpha \cdot CS + (1 - \alpha) \cdot CT \tag{1}$$
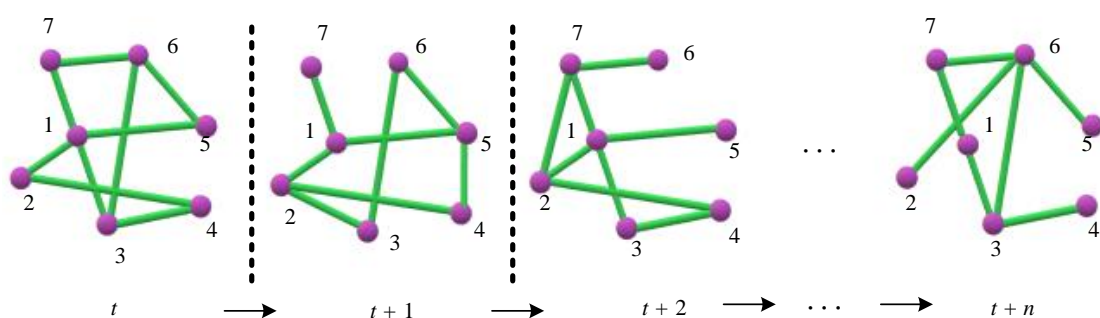

Figure 2. The series of network snapshots.

Among them, *CS* measures the adaptability of the community structure and network at the time t, while *CT* measures how similar the two community structures (the community structure is acquired at the time t to the structure is obtained at previous time $t - 1$). The parameter α is for balancing the importance between *CS* and *CT*. By introducing different object functions based on modularity, normalization of mutual information and spectrum clustering et al., this framework has been used in much of the literature [39,40] to discover communities in dynamic networks.

Folino and Pizzuti [39] formalized the dynamic community discovery algorithm as a multi-objective optimization problem, which maximized the clustering accuracy of the

current time step and minimized the clustering drift from one time step to one successive time step. Ma and Dong [40] proposed two evolutionary non-negative matrix decomposition (ENMF) frameworks and proved the equivalence relation between evolutionary module density and evolutionary spectrum clustering. In addition, they introduced a semi-supervised approach, which is called sE-NMF, that incorporated prior information into the ENMF.

Chi et al. [23] extended this idea with two frameworks of evolutionary spectral clustering, which are defined as Preserving Cluster Quality (PCQ) and Preserving Cluster Membership (PCM). Both frameworks have proposed the optimization and correction cost functions, but they differ in how to define the CT. In the PCQ framework, the CT is the cost of the clustering results at the time t applied to the similarity matrix at the time $t - 1$. In PCM, the CT is defined as a measure of the distance between the clustering results at the time t and $t - 1$. In the PCQ:

$$\max_{Z \in \mathbb{R}^{n \times k}} \alpha tr(Z^T W^{t-1} Z) + (1 - \alpha) tr(Z^T W^t Z), Z^T Z = I. \tag{2}$$

where W^{t-1} and W^t represent the adjacency matrix at the time $t - 1$ and t, respectively.

$$\alpha W^{t-1} + (1 - \alpha) W^t \tag{3}$$

Finally, the membership of community members can be obtained by calculating the eigenvector of Formula (3).

After the above work, an evolutionary community discovery algorithm is proposed to try to optimize the modified cost function in the definition. Since the user definition of snapshot and *CT* of community discovery results varies with community discovery algorithms, the aim of the above work is to solve the problem of how to select the parameter α, which can determine how much weight to assign to previous data or community discovery results.

Xu et al. [41] proposed an adaptive evolutionary clustering algorithm, using the following smooth approximation matrix $\hat{\Psi}^t$ to better estimate the network state.

$$\hat{\Psi}^t = \alpha^t \hat{\Psi}^{t-1} + (1 - \alpha^t) W^t \tag{4}$$

where the parameter α^t controls the rate of forgetting past information, so it is also defined as a forgetting factor.

Ma et al. [26] proposed a non-negative matrix decomposition for co-regularization evolution to identify dynamic communities under a time-smoothing framework.

$$O^t = L^t + \beta Q^t + \gamma R^t \tag{5}$$

where β and γ are regularized parameters.

In recent years, researchers have proposed some excellent techniques to improve the performance of dynamic community discovery algorithms. In the probability model, the researchers have put forward an innovative model, and this paper puts forward a new dynamic model, which is suitable for dynamic Bayesian networks, namely the system evolution partition transplantation model. In this method, the model parameters are tracked by using the graph of optimization strategy. Table 1 compares some recent representative dynamic social network discovery algorithms based on a probability model.

Table 1. Comparison of our work with previous model literatures.

Ref.	Year	Approach	Theory	Dataset
[42]	2015	Maximum Likelihood Estimation	Consistency	MIT Reality Mining
[43]	2015	Kalman Filtering + Local Search	/	Facebook wall posts
[44]	2016	Expectation Maximization	/	Internet AS graphs, Friendship networks
[45]	2016	Expectation Maximization	Detectability thresholds	Synthetic
[46]	2017	Expectation Maximization	Detectability thresholds	Synthetic
[47]	2017	Time-lag corrected	Convergence rates	Synthetic
[48]	2018	Aggregating SBM subroutines + MLE	Correctness/Stage-wise convergence rates	Enron emails, Facebook friendships
[49]	2020	Exhaustive grid search	Convergence rates	Synthetic
Our work	2022	Graph-based optimization	Convergence rates	MIT Reality Mining, Enron emails

3. Meterials

3.1. Formal Definition

In this paper, a novel phylogenetic planted partition model is defined for temporal social networks by the following definitions:

Definition 1. *A social network can be represented by a graph, G, on a set of nodes, V, and a set of edges, E. Nodes and edges are represented by an adjacency matrix A, where $\rho_{ij} = 1$ represents the existence of an edge from node $i \in V$ to node $j \in V - \{i\}$, and $\rho_{ij} = 0$ represents the absence of an edge. This paper assumes the network is a directed graph, which is generally $\rho_{ij} \neq \rho_{ji}$ and has no self-loop, namely $\rho_{ii} = 0$.*

Definition 2. *The positive integer n represents the number of nodes, and $\varepsilon = (\varepsilon_1, \dots, \varepsilon_k)$ is the probability vector on $[k] = \{1, \dots, k\}$ (k is the number of network communities). W is a symmetric matrix whose element is $k \times k$ between [0,1]. (X, G) is defined under the Stochastic Block-Mode (SBM) $SBM(n, \theta, W)$. X is a n dimensional random vector; it is independent and identically distributed under ε. Let the community set $C_r = C_r(X) = \{C_1, \dots, C_k\}, r \in [k]$ denotes the division of V into k communities. In this paper, we use the symbols p and q to indicate two generic communities, c_i and c_j, where i and j are the two nodes of a simple graph G of n vertices, which respectively belong to c_i (i.e., $i \in p$) and c_j (i.e., $j \in q$) at time t. Nodes i and j are connected according to a probability $\varepsilon_{c_i^t c_j^t}$, independently from the other node pairs. Therefore, the probability distribution of (X, G) [50] is as follows, where $G = (V, E)$.*

$$\Pr\{X = x\} = \prod_{i=1}^{n} \varepsilon_{c_i} = \prod_{r=1}^{k} \varepsilon_r^{|C_r(x)|} \tag{6}$$

$$\Pr\{E = y | X = x\} = \prod_{1 \leq i < j \leq n} \varepsilon_{c_i, c_j}^{y_{ij}} (1 - \varepsilon_{c_i, c_j})^{1 - y_{ij}} \tag{7}$$

$$= \prod_{1 \leq p < q \leq k} \varepsilon_{p, q}^{N_{pq}(x, y)} (1 - \varepsilon_{p, q})^{N_{pq}^c(x, y)} \tag{8}$$

where $N_{pq}(x, y)$ represents the observed value of the number of edges in the partition, which can be expressed as $N_{pq}(x, y) = \sum_{c_i = p} \sum_{c_j = q} y_{ij}^t$; $N_{pq}^c(x, y)$ represents the probability value of the number of edges in the partition, which can be expressed as follows.

$$N_{pq}^c(x, y) = \begin{cases} |p|(|p| - 1)/2 - N_{pp}(x, y) & p = q \\ |p||q| - N_{pq}(x, y) & p \neq q \end{cases} \tag{9}$$

Equation (8) can be rewritten by:

$$\Pr\{E = y | X = x\} = \exp\{\sum_{p=1}^{k} \sum_{q=1}^{k} [N_{pq}(x,y) \log \varepsilon_{p,q} + N_{pq}^c(x,y) \log(1 - \varepsilon_{p,q})]\} \tag{10}$$

Definition 3. *Define an evolutionary sequence of discrete time steps for social network (dynamic Bayesian network); the nodes and edges may appear or disappear with time. The temporal social networks can be expressed as $\{G^1, G^2, \ldots, G^T\}$, where $G^t = (V^t, E^t)$, superscript t represents the time step, V^t and E^t, respectively, in the time step are t collection of nodes and edges. Let $A^{(T)}$ represent the sequence of adjacency matrix on the node-set sequence $V^{(T)} = \cup_{t=1}^{T} V^t$, and let $c^{(T)}$ represent the sequence of community member membership vectors of the node. For this dynamic social network, the probability distribution of edges can be defined as:*

$$\Pr = (1 - \varepsilon_{pq}^{t-1})\varepsilon_{pq}^t \cdots \varepsilon_{pq}^{t+d-1}(1 - \varepsilon_{pq}^{t+d}) \tag{11}$$

For any pair of nodes $i \in p$ and $j \in q$ at $t-1$ and t, s.t. $\rho_{ij}^{t-1} = 1$. Namely, there is an edge from node i to the node j at the time $t-1$ and ρ_{ij}^t is Independent Identically Distributed (IID). The same is true for $\rho_{ij}^{t-1} = 0$. The mapping process of random sampling and probability allocation is shown in Figure 3.
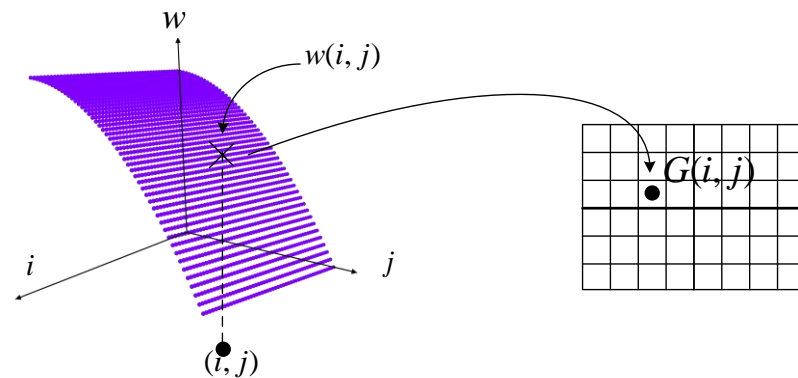

Figure 3. Randomly sample (i, j) and allocate ε with probability $\rho_{ij} = 1$, $\rho_{ij}^t \sim Bernoulli(\varepsilon_{c_i^t c_j^t}^t)$.

Figure 3 shows the hypothesis of this paper. There are only two possibilities for random events: existence or non-existence of edges. In this paper, we randomly sample (i, j) and allocate ε, with probability $\rho_{ij} = 1$, $\rho_{ij}^t \sim Bernoulli(\varepsilon_{c_i^t c_j^t}^t)$. Each term of the adjacency matrix A^t is independent. Therefore, Equation (10) can be rewritten as the likelihood form with parameter W^t.

$$f(A^t; W^t) = \exp\{\sum_{p=1}^{k} \sum_{q=1}^{k} [\alpha_{pq}^t \log \varepsilon_{pq}^t + \beta_{pq}^t \log(1 - \varepsilon_{pq}^t)]\} \tag{12}$$

where α_{pq}^t and β_{pq}^t are, respectively, denoted as follows:

$$\begin{cases} \alpha_{pq}^t = \sum_{c_i \in p} \sum_{c_j \in q} \rho_{ij}^t \\ \beta_{pq}^t = \begin{cases} |p|(|p|-1)/2 & p = q \\ |p||q| & p \neq q \end{cases} \end{cases} \tag{13}$$

3.2. A Migration Partitioning Model for Phylogenetic Evolution

The proposed dynamic community discovery method can track the status of the target over time to discover community results. Therefore, this paper constructs an observation model, which can be described by

$$S^t = W^t + Z^t, t = 0, 1, 2, \ldots \tag{14}$$

where Z^t is an independent Gaussian noise matrix with zero mean and variance $(\sigma_{pq}^t)^2 = \varepsilon_{pq}^t (1 - \varepsilon_{pq}^t)/\beta_{pq}^t$. This matrix reflects the transient variations caused by noise. In this paper, we assume that $Z^t, Z^{t-1}, \ldots$ are independent of each other.

In the dynamic system model, $S^{(t)}$ expresses the set of observed values, and $W^{(t)}$ represents the state of the sequence of observed values that generate noise in the dynamic system. This paper refines the final model by modeling the evolution of specified states over time. Because ε^t is a probability between $0 \sim 1$, and this paper deals with ε^t in logarithmic form, that is, $y^t = \log(\varepsilon_{pq}^t / (1 - \varepsilon_{pq}^t))$, then a time-series dynamic observation model of system evolution can be constructed as follows:

$$y^t = H^t y^{t-1} + z^t \tag{15}$$

where H^t denotes the state transition model, y^t represents the vector metric representation of matrix W^t, z^t implies the process noise, z^t is a random vector with zero mean and Θ^t is the covariance matrix. According to the vectorization expression of S^t and observation noise Z^t, the observation model (15) can be rewritten as:

$$s^t = g(y^t) + z^t \tag{16}$$

The logical activation function $g(\cdot)$ is handled by the Sigmod function, which is:

$$g(x) = \frac{1}{1 + e^{-x}} \tag{17}$$

This paper assumes that the initial state of the dynamic system obeys the Gaussian distribution, namely, $y^0 \sim N(\mu^0, \Theta^0)$. The nonlinear optimization problem in the time-series dynamic observation model of system evolution is constructed in this paper, which is the problem of calculating the Maximum a Posteriori (MAP) of c^t, while for Gaussian distributions, the maximization problem can be translated into the negative logarithm problem of minimizing the target probability c^t. Therefore, Equation (12) is converted into the following logarithmic likelihood:

$$\begin{aligned} \hat{c}^t &= \log f(A^t | y^t) \\ &= \sum_{p=1}^k \sum_{q=1}^k \{ \alpha_{pq}^t \log h(y_{pq}^t) + \beta_{pq}^t \log[h(1 - y_{pq}^t)] \} \end{aligned} \tag{18}$$

The following error function will be constructed:

$$e^t(y) = y^t - f(y^{t-1}, v^t, 0) \tag{19}$$

Then, minimize the quadratic form of the error function:

$$\min J(y) = \sum_{t=1}^T (\frac{1}{2} e^t(y)^T (W^t)^{-1} e^t(y)) \tag{20}$$

Finally, make the first-order expansion of $f(x)$:

$$f(x + \Delta x) \approx f(x) + J(x)\Delta x \tag{21}$$

where $J(x)$ is the derivative of $f(x)$ with respect to x, which is actually a matrix of $m \times n$, which is also Jacobian. The derivative problem can be turned into a recursive approximation problem; therefore, L–M method is adopted in this paper to determine the step size Δx, the L–M method avoids the non-singular and morbid state properties of the coefficient matrix of linear equations and can provide a more stable and accurate increment Δx. In the previous methods, as the approximate second-order Taylor expansion adopted in the GaUSs-Newton method could only have a good approximation effect near the expansion point, a trust-region is added to Δx. It should be noted that the trust-region should not be so large that the approximation is inaccurate. The approximate value in the trust region is considered to be valid; when it is outside of this region, the approximation might go wrong. The scope of the trust region is determined by the difference between the approximate model and the actual function. Determine rules: if the differences are small, let the scope be as large as possible; if the difference is large, narrow the approximation. Therefore, Equation (22) is used to judge whether the Taylor approximation is good enough or not.

$$\rho = \frac{f(x + \Delta x) - f(x)}{J(x)\Delta x} \tag{22}$$

where the numerator ρ is the decreasing value of the actual function, and the denominator is the decreasing value of the approximate model. If ρ is close to 1, then the approximation is good. If ρ is too small, meaning that the actual reduced value is far less than the approximate reduced value, then the approximate result is considered to be poor and the approximate range needs to be narrowed. On the contrary, when ρ is large, it means that the actual decline is larger than expected, and the approximate range can be enlarged.

Because the temporal dynamic observation model of system evolution constructed is nonlinear and df/dx is not easy to obtain, this paper intends to adopt an iterative method (if there is an extreme value, then convergence to approximation) to converge the approximation. The steps are shown in Algorithm 1.

Algorithm 1: Main procedure of the iterative method

---
1. Given an initial value x_0, radius r and parameter k
2. **for** the k-th iteration, solving:
$\min\limits_{\Delta x_k} \frac{1}{2}||f(x_k) + J(x_k)\Delta x_k||^2$, s. t. $||D\Delta x_k||_2 \leq r$
3.   Compute ρ
4.   **if** $\rho > 3/4$
5. $r = 2r$
6.   **else if** $\rho < 1/4$
7. $r = 0.5r$
8. $x_{k+1} = x_k + \Delta x_k$
9.   **if** convergence
10.   **break**
11. **end**

---

where the limiting condition r is the radius of the trust region. In Equation (21), the incremental range is limited to a sphere of radius r, which is seen as an ellipsoid after multiplying by D. D is taken as a non-negative diagonal matrix, usually with the square root of the diagonal element $J^T J$, and it is equivalent to directly constraining Δx in the ball.

$$\min\limits_{\Delta x_k} \frac{1}{2}||f(x_k) + J(x_k)\Delta x_k||^2 + \frac{\lambda}{2}||D\Delta x||^2 \tag{23}$$

where λ is the Lagrange multiplier. Finally, this paper needs to obtain the gradient by solving the objective function (23). Since it is an optimization problem with inequality constraints, the Lagrange multiplier is used in this paper to transform the objective function into an unconstrained optimization problem. Additionally, then the target function is transformed.

Let us expand out the square of the target function of (23).

$$
\begin{aligned}
&\tfrac{1}{2}\left\|f(x_k)+J(x_k)\Delta x_k\right\|^2+\tfrac{\lambda}{2}\left\|D\Delta x_k\right\|^2\\
&=\tfrac{1}{2}(f(x_k)+J(x_k)\Delta x_k)^T(f(x_k)+J(x_k)\Delta x_k)\\
&\quad+\tfrac{\lambda}{2}(D\Delta x_k)^T(D\Delta x_k)\\
&=\tfrac{1}{2}(\left\|f(x_k)\right\|_2^2+2f(x_k)^TJ(x_k)\Delta x_k+\Delta x_k{}^TJ(x_k)^TJ(x_k)\Delta x_k)\\
&\quad+\tfrac{\lambda}{2}(D^T\Delta x_k{}^TD\Delta x_k)
\end{aligned}
\tag{24}
$$

Then, solve the derivative of Δx_k in Equation (24) and set it to zero:

$$
2J(x_k)^Tf(x_k)+2J(x_k)^TJ(x_k)\Delta x_k+2\lambda D^TD\Delta x=0
\tag{25}
$$

The following equations are obtained:

$$
J(x_k)^TJ(x_k)\Delta x_k+\lambda D^TD\Delta x_k=-J(x_k)^Tf(x_k)
\tag{26}
$$

Let $J(x_k)^TJ(x_k)=H$, the right-hand side of the equation be defined as g, and the equation can be simplified as follows:

$$
(H+\lambda D^TD)\Delta x_k=g
\tag{27}
$$

In the initial time step of the algorithm, the proposed PPPM method is initialized with the spectral clustering algorithm; that is, the initial estimation of community is generated at the time $t=1$. The advantage of using the spectral clustering algorithm as the initialization algorithm here is that it can prevent the local search from falling into c^t poor local maximum in the initial time step. The main procedure of the proposed PPPM method can be shown in Algorithm 2.

Algorithm 2: The main procedure of PPPM

Input: $G=\{G^1,G^2,\ldots,G^T\}$, k//dynamic networks and the number of communities
Output: c^t//the community

1.   at $t=0$
2.   Initialize c^0 by using spectral clustering applied on W^0
3.   at $t>0$
4.   **if** iteration $\leq$ max iteration//hill-climbing algorithm
5. $\hat{c}_0^t\leftarrow-\infty$//negative Log of the best adjacent case till to a constant
6. $\dot{c}^t\leftarrow c^t$//currently being traversing case
7.   **for** $i=1$ **to** $|V^t|$ **do**//traverse all adjacent solutions
8.   **for** $j=1$ **to** k; s.t. $c_i^t\neq j$ **do**
9. $\dot{c}_i^t\leftarrow j$//change community of a node
10.   compute y^t using Equations (15)–(17)
11.   compute Log $\hat{c}_1^t$ using (18)
12.   **if** $\hat{c}_1^t>\hat{c}_0^t$ **then**//current case is the best case
13. $(\hat{c}_0^t,\bar{c}^t)\leftarrow(\hat{c}_1^t,\dot{c}^t)$
14. $\dot{c}_i^t\leftarrow c_i^t$//refresh community of current node
15.   **if** $\hat{c}_0^t>\hat{c}^t$ **then**//the best adjacent case is better than the current best case
16. $(\hat{c}^t,c^t)\leftarrow(\hat{c}_0^t,\bar{c}^t)$
17.   **else**//achieve a minimum
18.   break
19.   **end**
20.   **end**
21. **return** c^t

---

4. Results

In order to prove the rationality of the novel proposed method, four algorithms are compared, namely FaceNet [27], SBM + MLE [48], CLBM [49], and PisCES [51]. Firstly,

FaceNet was chosen because it was the first proposed dynamic web community discovery algorithm that could be compared as a baseline; secondly, SBM + MLE and CLBM were used because they are the latest proposed probabilistic model-based algorithms; finally, PisCES is also a recently proposed non-probabilistic model algorithm. In this paper, the indicators of the following two evaluation models are adopted.

(1)   Adjust Rand Index (*ARI*), $ARI \in [-1.1]$, if the value of *ARI* is closer to 1, it means better results.

$$ARI = \frac{RI - \mathrm{E}(RI)}{\max(RI) - \mathrm{E}(RI)} \tag{28}$$

where $\mathrm{E}(RI)$ represents the expected value of *RI* and $\max(RI)$ denotes the maximum value of *RI*.

(2)   Mean-squared errors (MSE), the smaller the value, the smaller the error, that is, the better the result.

$$MSE = \frac{1}{n} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 \tag{29}$$

where y_i is the real data, $\hat{y}_i$ expresses the fitting data, and *n* implies the number of samples.

Figure 4 can simulate the evolution process of an artificial dynamic network over time. The network consists of 156 nodes and 614 edges, and a total of eight time steps are set.
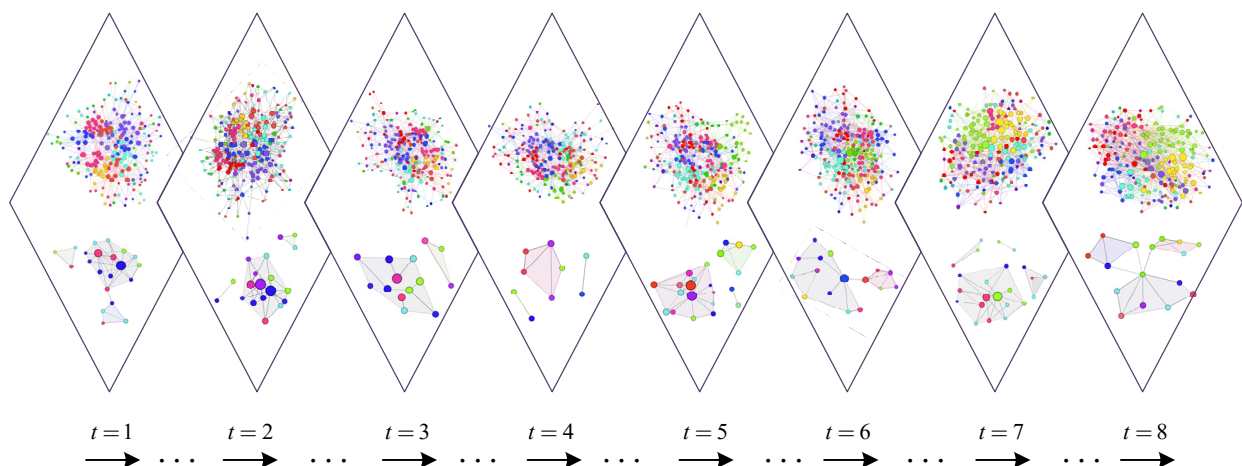

$t = 1 \quad t = 2 \quad t = 3 \quad t = 4 \quad t = 5 \quad t = 6 \quad t = 7 \quad t = 8$

Figure 4. The simulation of the evolution process of an artificial dynamic network.

In Figure 4, there are eight rhombic blocks, and the whole dynamic social network can be represented by the evolution of these eight rhombic blocks over time. The upper part represents the dynamic network of a time step, the lower part denotes the community where the current time step may exist, and the lower part is composed of the nodes with the highest degree of nodes of each color in the network absorbing nearby nodes to form larger nodes. (Absorption rule: connected with the node with the greatest degree and with the same color). For example, in the lower half $t = 1$, there are three submodules, each of which represents a possible community. More specifically, each submodule can be composed of nodes of different colors and sizes, and each color can represent nodes with the same characteristics in the dynamic social network. it can be seen that the community structure of dynamic social networks is phylogenetic over time.

4.1. Synthetic Networks

The artificial network is generated in this paper, which consists of 128 nodes, initially divided into four communities, where each community has 32 nodes. At the initial time step, the edge probability of the system evolution migration partition is set as $\varepsilon_{pp}^1 = 0.3109$ and $\varepsilon_{pq}^1 = 0.0765$ ($p, q = 1, 2, 3, 4$ and $p \neq q$). The initial covariance Θ^1 is set to the identity

matrix $0.04I$. The state vector G evolves according to the Gaussian random walk model, namely $H^t = I$ in Equation (15). This paper generates 25 and 50 time steps. At each time step, nodes are randomly selected to leave their communities and randomly assigned to one of the other three communities. Table 2 statistically compares the proposed PPPM method with the average *ARI* experimental results of multiple parameters of four representative models in an artificial network environment.

Table 2. The results of the proposed PPPM and representative model on the Mean ARI (synthetic data).

Time Step	Random Rate	Proposed PPPM	PisCES	CLBM	SBM + MLE	FaceNet
25		**0.65702**	0.50409	0.48838	0.56099	0.46667
50	10%	**0.56630**	0.38265	0.34544	0.49414	0.34324
25		**0.72236**	0.57972	0.54694	0.66961	0.50949
50	20%	**0.94401**	0.90616	0.89504	0.92753	0.88592
	Mean	**0.72242**	0.670	0.56895	0.663068	0.55133

In Table 2, bold font indicates that the result is the best. It can be clearly seen that the proposed PPPM method has the best performance under all parameters. It can be calculated that the average performance of the novel method is improved by 0.05 compared with the other four best models.

Figure 5 shows the comparison of average ARI results between the proposed method and four representative models in an artificial network environment.
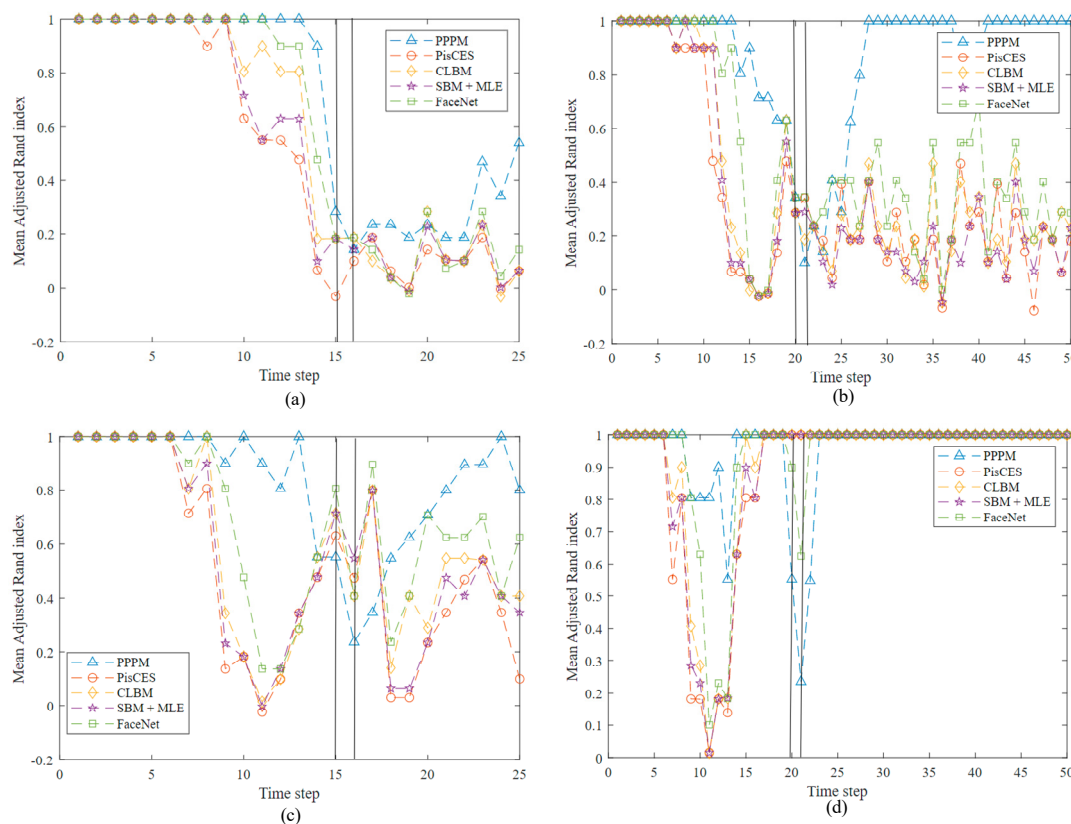

Figure 5. Comparison of the proposed model with 4 different models on Mean ARI (synthetic network). (**a**) indicates that the time step is 25 and the randomly selected parameter is 10%, (**b**) indicates that the time step is 50 and the randomly selected parameter is 10%, (**c**) indicates that the time step is 25 and the randomly selected parameter is 20%, (**d**) indicates that the time step is 50 and the randomly selected parameter is 20%.

As shown in Figure 5a, 25 time steps are generated by the artificial network. On each time step, the randomly selected parameter is set to 10%. In this experiment, the parameters of the noise item are changed at the 15th time step (the left line) and set back to the original state at the 16th time step (the right line). It is evident that in the 15th time step, the only two models with SBM and PPPM + MLE line charts show the correct change in trend, i.e., a downward trend, and the proposed PPPM method declines faster, and the increasing trend of PisPCES, CLBM, and FaceNet are unaffected and keep the previous state, after the 16th time step, which also can obviously show that compared with the other four kinds of models, the novel method callback trend is more obvious. This indicates that the proposed novel method has a more consistent response to noise terms.

In Figure 5b, the artificial network generates 50 time steps, randomly selects parameters and sets them to 10%, changes the parameters of the noise item at the 20th time step (the left line), and sets them back to the original state at the 21st time step (the right line). It is evident that in the 20th time step, the only two models with PPPM and CLBM line charts show the correct change trend, i.e., a downward trend, and PPPM declines faster, and PisPCES and FaceNet are on the rise, with SBM CLBM + MLE remaining unaffected and they to keep the previous state, after the 16th time step, which also can obviously show that compared with the other four kinds of models, PPPM callback trend is more apparent in terms of reverting to the previous state, which shows that this paper proposed the model of response that is more consistent in noise.

In Figure 5c, the artificial network generates 25 time steps, randomly selects parameters and sets them to 20%, changes the parameters of the noise item at the 15th time step (the left line), and sets them back to the original state at the 16th time step (the right line). It is obvious that at the 15th time step, the line graph of all models shows the correct trend of change, namely the downward trend. It is worth noting that the downward trend of PPPM is the most obvious. After the 16th time step, it is also obvious that compared with the other four models, the proposed novel method has a more obvious callback trend, which also indicates that the novel method has a more consistent response to the noise term.

Figure 5d shows that the artificial network generates 50 time steps, randomly selects the parameters and sets them to 20%, changes the parameters of the noise item at the 20th time step (the left line), and sets them back to the original state at the 21st time step (the right line). It is evident that in the 20th time step, only two models with PPPM and FaceNet line charts show the correct change trend downward trend, and PPPM decline faster, while the remaining three kinds of model, CLBM, SBM + MLE, PisPCES, are not affected, and keep the previous state; after the 16th time step, PPPM and FaceNet all can to go back to the previous state, which demonstrates that the proposed method has a more consistent response in noise.

In conclusion, in the artificial network, this paper proposed a dynamic community-found PPPM method compared with the other four kinds of a typical model. The model is tested in the perturbation parameter test (the noise is changed in a particular time step). The prediction accuracy of the model index (ARI) increased by 5% on average, and the experimental results show that the proposed model is robust.

4.2. Real-World Networks

4.2.1. MIT Reality Mining

This experiment is conducted on the MIT dataset [52]. The dataset is collected by recording the mobile phone activity of 94 students and employees over a year. The dataset built a dynamic network based on physical distance, which is measured by scanning nearby Bluetooth devices every 5 min. Data collected near the beginning and end of experiments with low participation rates are excluded in this experiment. Each time step corresponds to one week, so there are 37 time steps between August 2004 and May 2005. Figure 6 shows the mean-variance error results of the proposed novel method and four representative models under the artificial network.
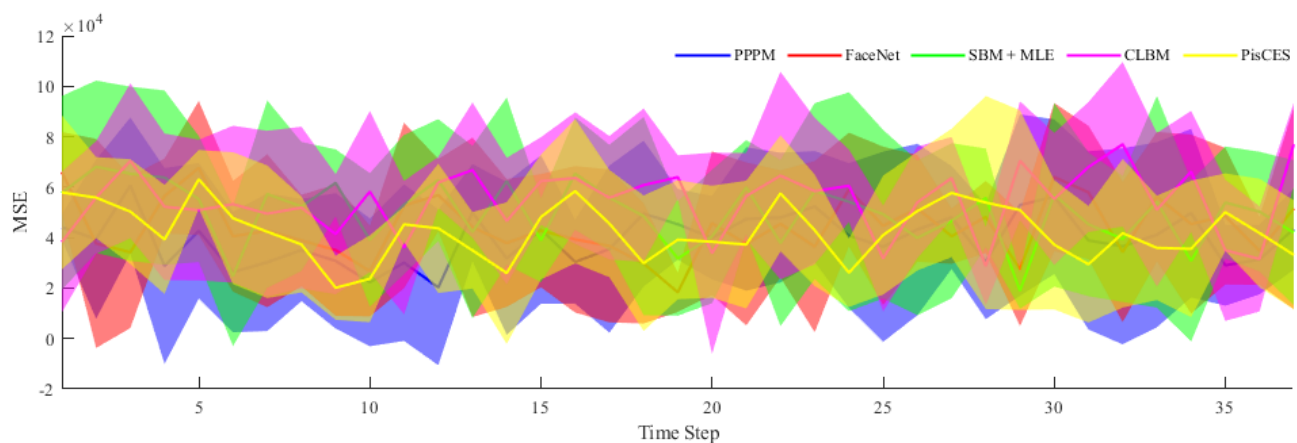
Figure 6. Comparison of the proposed method with 4 different models on MSE (synthetic network).

Figure 6 shows that, under the MSE evaluation index, the smaller the error, the better the result; that is, the closer the model image is to the *x*-axis. Obviously, compared with other colors (the other four models), the image with blue color (the proposed method in this paper) is closer to the *x*-axis; that is, the proposed PPPM method has a lower MSE value and a smaller error. Table 3 compares the average ARI results of the proposed method with those of the four representative models in the real network (MIT reality mining) environment.

Table 3. The results of the proposed PPPM method and representative model on the Mean ARI (real data).

	FaceNet	PPPM	SBM + MLE	CLBM	PisCES
Max	0.8991	**0.9555**	0.8678	0.8876	0.9412
75%	0.7125	**0.8005**	0.7856	0.6258	0.7811
Median	0.4902	**0.6523**	0.6215	0.4981	0.6536
25%	0.3154	**0.5111**	0.4992	0.2314	0.4902
Min	0.1002	**0.40**	0.2671	0.1487	0.3243

In Table 3, the bold font shows that the result is the best, you can clearly see that the proposed PPPM in all parameters (the maximum value; the first 75% of the value; the median; the first 25% of the value; minimum value) cases are the best and clear, the average performance of PPPM performance (median) than the best model is increased by 3% in the other four. Figure 7 shows the comparison of MARI values on the MIT dataset between the proposed method and four state-of-the-art models.
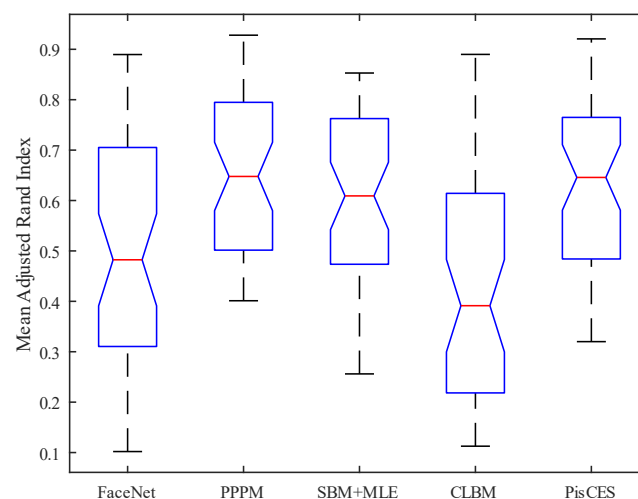

Figure 7. Comparison of the proposed model with 4 different models on MARI (reality network).

In Figure 7, the upper and lower edges of each box in the boxplot represent 25% and 75% values, respectively, and the middle red line denotes the median. It is obvious that PPPM, SEM + MLE, and the three boxes perform better than the other two model boxes. Among the three models with better performance, the PPPM box position is slightly higher than that of SEM + MLE and PisCES boxes, and the median value is also slightly higher than that of SEM + MLE and PisCES models. In conclusion, compared with the other four representative models in the real network, the proposed dynamic community PPPM method performs better under the two evaluation indexes of prediction accuracy and error.

4.2.2. Enron Email Data

The experiment is conducted on the dynamic social network, which is built by Enron [53], and it consisted of about 500,000 emails between 184 Enron employees from 1998 to 2002. The directional edge between the employee and the time point occurs if at least one email is sent within the first week. Each time step corresponds to an interval of 1 week. This dataset does not distinguish between emails sent to "recipients," "CC" or "BCC." In addition to email dataset, most employee roles (such as CEO, president, manager, employee) exist within the company and they are used as known communities. The first 56 weeks and the last 13 weeks are filtered because only a few emails are sent. Figure 8 compares the estimated community probability between a normal week and an event week. The higher the probability, the higher the community activity. Both the *x*-axis and the *y*-axis denote the estimated communities, and the color blocks on the diagonals express the activity within each community, and the color blocks of the diagonals imply the activity between each community.
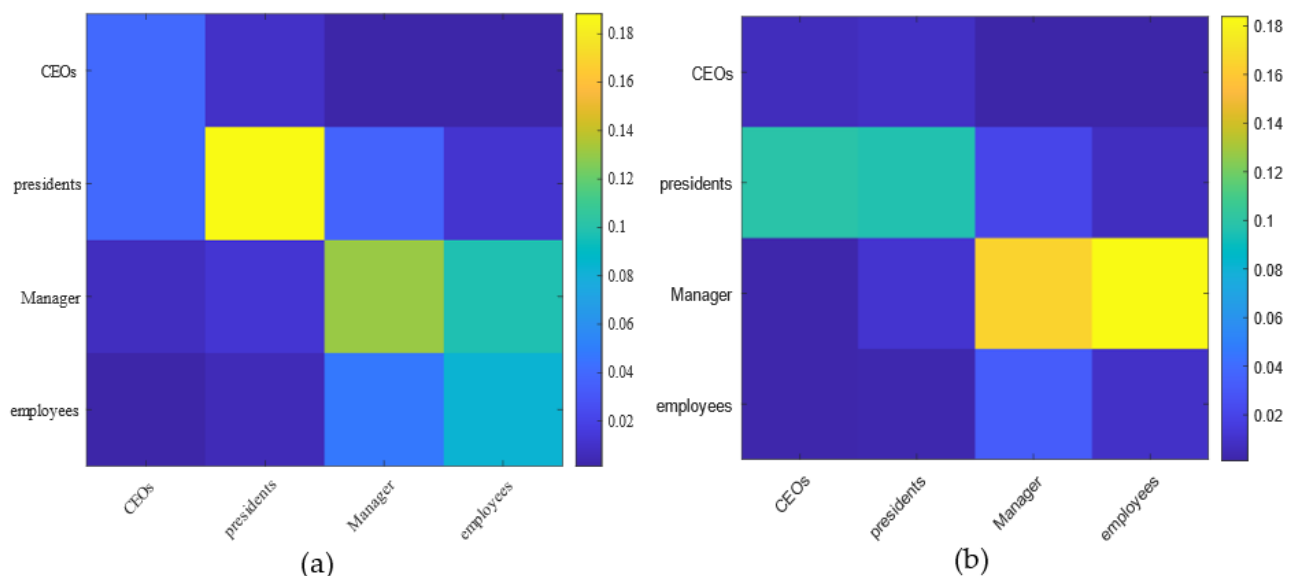

Figure 8. The comparison of community probability in normal week and event week. (**a**) indicates the normal week(week 59), (**b**) indicates the event week (the 89 weeks when CEO Jeffrey resigns).

As shown in Figure 8a, in a normal week (week 59), the president community is the most active, followed by managers and employees, and the CEO community is the least active. It is also worth noting that from the color block distribution of managers and employees, the two communities may merge into one large community. This phenomenon can be reflected in the fact that communication between department managers and employees is usually close, and managers and employees are more likely to get along with each other. As shown in Figure 8b, in the event week (the 89 weeks when CEO Jeffrey resigns), the most active community is that of the managers, followed by president community, and the brightest color block is the managers to the employee community. This is reflected in the fact that in real life when CEOs resign, the discussion is most intense among managers

because it is directly related to their personal interests. Discussions between managers and employees also proliferate for the same simple reason that it is indirectly related to the employees' personal interests. Figure 9 reveals the estimated edge connections between communities in Enron's email network under the proposed dynamic community's discovery approach PPPM and shows a 95% confidence interval (note: the lines on the left and right of the figure are for weeks 59 and 89, respectively).
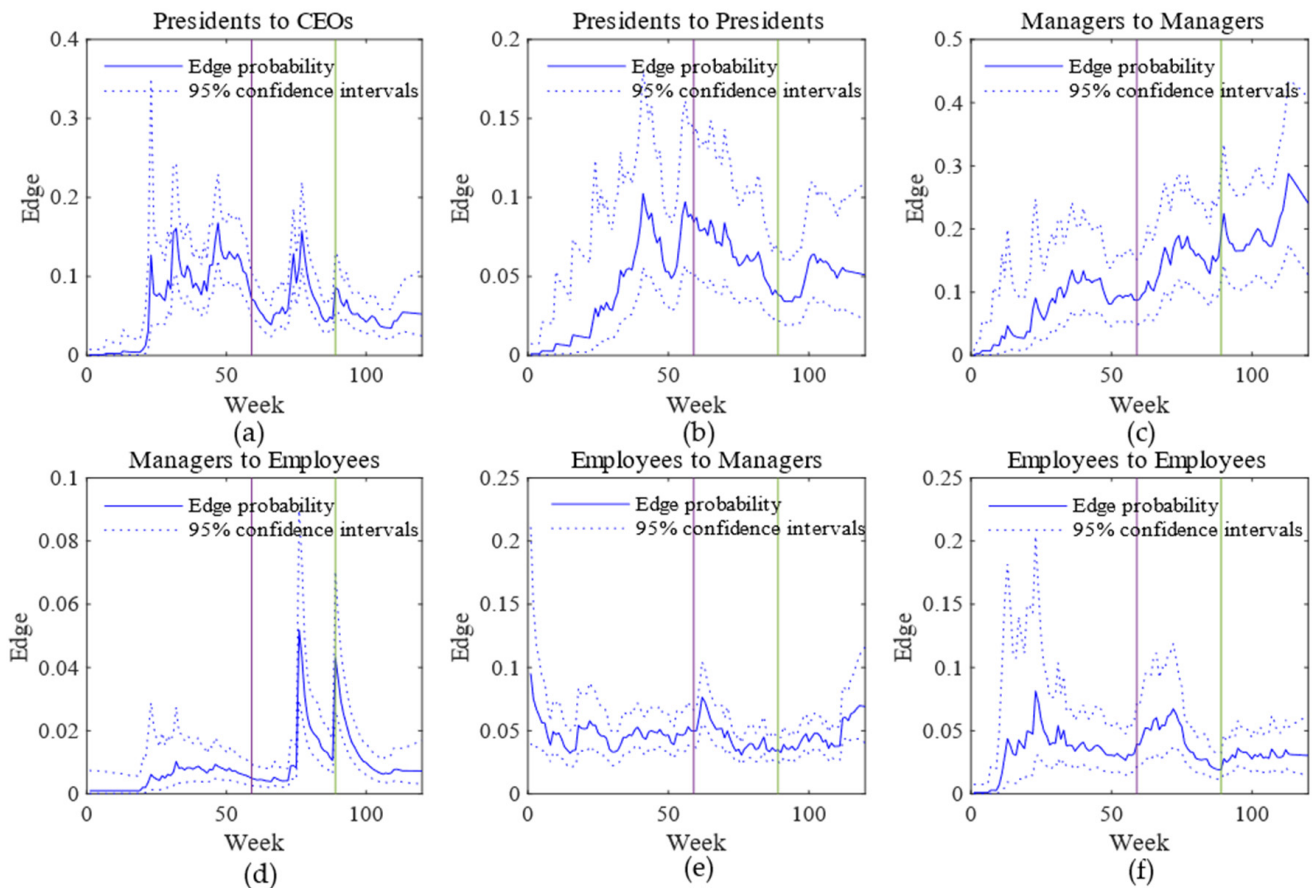

Figure 9. Probability of edges between communities on the Enron mail. (**a**–**f**) are the edge probabilities between different roles.

As shown in Figure 9a, it is the edge probability of presidents to CEOs; it can be seen that the presidents to CEOs edge probability increased slightly at week 59 (normal) and 89 (Jeffrey CEO resigned), which corresponds to presidents to CEOs activity (increased) from Figure 8a,b. Figure 9b shows the side probability within the president community. In the 59th and 89th weeks, the side probability inside the community shows a downward trend. This also corresponds to the active state (decreased activity) within the president community from Figure 8a,b. Figure 9c,d show the side probabilities between managers and the manager community and between the managers and the employee community, respectively. It can be seen that in the 59th and 89th weeks, the changing trend of the side probabilities of these two communities is consistent with that in Figure 9a.

Similarly, this change also corresponds to the changes in active state between managers and manager community and the employee community (increased activity) from Figure 8a,b. Figure 9e shows the edge probability between the employees and the manager community. It is not difficult to see that there is no obvious trend of change in week 59 and 89. Similarly, this situation also corresponds to the consistent change in the active state between the employees and the manager community from Figure 8a,b (there is no significant change in the activity). Finally, Figure 9f shows the edge probability between employees and the employee community. In week 59 and 89, similarly, the changing trend

of edge probability of these two communities is consistent with the change in Figure 9b; namely, it displays the downward trend. At the same time, it also corresponds to the consistent change in active state between employees and the employee community from Figure 8a,b (decreased activity).

To sum up, the proposed PPPM method can well reflect some phenomena existing in the real network, and the probability estimated by the novel method can make relatively consistent predictions with the advance of time and the occurrence of specific events.

5. Conclusions

The proposed model has practical theoretical and practical significance to mine and it also simulates deeper hidden information that is present in dynamic social networks. At present, the dynamic social network community discovery method cannot effectively represent the entire dynamic network evolution process. Therefore, inspired by the evolution theory of natural biosensors, this paper proposes a community discovery method based on phylogenetic planted partition. Firstly, the time dimension is added to the transplant partition model, all states in the whole dynamic network system are treated as variables, the observation equation is used as a constraint between variables, and an error function about the whole dynamic network system is constructed. Then, the quadratic form of the error function is minimized, which can abstract the observation results of the network more realistically. Secondly, a graph optimization strategy is used to consider the constraints in the whole motion trajectory at one time, and the Jacobian matrix is calculated during the linearization process. Because the calculation process is relative to the whole motion trajectory, the whole system evolution process is transformed into a nonlinear system optimization process. The gradient of the error function is obtained by using the L–M method, and then the iteration is carried out according to the direction of the gradient; finally, the proposed method is compared with four state-of-the-art representative models under two scenarios of artificial network and real network. The experimental results show that the PPPM method has better performance than the other four representative models in building a dynamic network model and mining dynamic network hidden information.

Next, this paper will consider how to integrate the multi-layer model mechanism into the proposed model and will study dynamic network hiding information with multi-layer information in future research.

Author Contributions: Conceptualization, X.L. and N.D.; methodology, N.D.; software, N.D.; formal analysis, N.D.; writing—original draft preparation, N.D.; writing—review and editing, G.F., P.D.M. and A.F.; visualization, N.D.; supervision, X.L.; project administration, X.L.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The calculated data presented in this work are available from the corresponding authors upon reasonable request. Figures 7–9 contain data available in Refs. [52,53].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Newman, M.E. The structure and function of complex networks. *SIAM Rev.* **2003**, *45*, 167–256. [CrossRef]
2. Dakiche, N.; Benbouzid, F.T.; Slimani, Y.; Benatchba, K. Tracking community evolution in social networks: A survey. *Inf. Process. Manag.* **2018**, *56*, 1084–1102. [CrossRef]
3. Girvan, M.; Newman, M.E. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA* **2002**, *99*, 7821–7826. [CrossRef] [PubMed]
4. Guruharsha, K.; Rual, J.F.; Zhai, B.; Mintseris, J.; Vaidya, P.; Vaidya, N.; Beekman, C.; Wong, C.; Rhee, D.Y.; Cenaj, O. A protein complex network of Drosophila melanogaster. *Cell* **2011**, *147*, 690–703. [CrossRef]
5. Pagani, G.A.; Aiello, M. The power grid as a complex network: A survey. *Physica A* **2013**, *392*, 2688–2700. [CrossRef]
6. Sanchez, F.; Mina, M. Oncogenic signaling pathways in the cancer genome atlas. *Cell* **2018**, *173*, 321–337. [CrossRef]
7. Baccaletti, S.; Bianconi, G.; Criado, R. The structure and dynamics of multilayer networks. *Phys. Rep.* **2010**, *544*, 1–122. [CrossRef]
8. Ma, X.; Sun, P.; Zhang, Z. An integrative framework for protein interaction and methylation data to discover epigenetic modules, IEEE/ACM Trans. *Comput. Biot. Bioinf.* **2019**, *16*, 1855–1866.
9. Ma, X.; Dong, D.; Wang, Q. Community detection in multi-layer networks using joint nonnegative matrix factorization. *IEEE Trans. Knowl. Data Eng.* **2019**, *31*, 273–286. [CrossRef]
10. Huang, Z.; Rege, X. Detecting community in attributed networks by dynamically exploring node attributes and topological structure. *Knowl. Based Syst.* **2020**, *196*, 105760. [CrossRef]
11. Džamić, D.; Aloise, D.; Mladenović, N. Ascent–descent variable neighborhood decomposition search for community detection by modularity maximization. *Ann. Oper. Res.* **2019**, *272*, 273–287. [CrossRef]
12. Karrer, B.; Newman, M.E. Stochastic blockmodels and community structure in networks. *Phys. Rev. E* **2011**, *83*, 016107. [CrossRef] [PubMed]
13. Wen, Y.M.; Huang, L.; Wang, C.D.; Lin, K.Y. Direction recovery in undirected social networks based on community structure and popularity. *Inform. Sci.* **2019**, *473*, 31–43. [CrossRef]
14. He, D.; Feng, Z.; Jin, D.; Wang, X.; Zhang, W. Joint identification of network communities and semantics via integrative modeling of network topologies and node contents. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 116–124.
15. Airoldi, E.M.; Blei, D.M.; Fienberg, S.E.; Xing, E.P. Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.* **2008**, *9*, 1981–2014.
16. Qiao, M.; Yu, J.; Bian, W.; Li, Q.; Tao, D. *Improving Stochastic Block Models by Incorporating Power-Law Degree Characteristic*; IJCAI: Melbourne, Australia, 2017; pp. 2620–2626.
17. Fortunato, S. Community detection in graphs. *Phys. Rep.* **2010**, *486*, 75–174. [CrossRef]
18. Fortunato, S.; Hric, D. Community detection in networks: A user guide. *Phys. Rep.* **2016**, *659*, 1–44. [CrossRef]
19. Rand, D.; Christakis, N. Dynamic social networks promote cooperation in experiments with humans. *Proc. Natl. Acad. Sci. USA* **2011**, *108*, 19193–19198. [CrossRef]
20. Chiang, A.; Massagie, J. Molecular basis of metastasis. *N. Engl. J. Med.* **2008**, *359*, 927–932. [CrossRef]
21. Kim, M.; Han, J. A particle-and-density based evolutionary clustering method for dynamic networks. *Proc. VLDB Endow.* **2009**, *2*, 622–633. [CrossRef]
22. Folino, F.; Pizzuti, C. An evolutionary multi-objective approach for community discovery in dynamic networks. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 1838–1852. [CrossRef]
23. Chi, Y.; Song, X.; Zhou, D.; Hino, K.; Tseng, B.L. On evolutionary spectral clustering. *ACM Trans. Knowl. Data Discov.* **2009**, *3*, 1–30. [CrossRef]
24. Wang, L.; Rege, M. Low-rank kernel matrix factorization for large-scale evolutionary clustering. *IEEE Trans. Knowl. Data Eng.* **2012**, *24*, 1036–1050. [CrossRef]
25. Ma, X.; Dong, D. Evolutionary nonnegative matrix factorization algorithms for community detection in dynamic networks. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 1045–1058. [CrossRef]
26. Ma, X.; Zhang, B.; Ma, C.; Ma, Z. Co-regularized Nonnegative Matrix Factorization for Evolving Community Detection in Dynamic Networks. *Inf. Sci.* **2020**, *528*, 265–279. [CrossRef]
27. Lin, Y.; Zhu, S. Analyzing communities and their evolutions in dynamic social networks. *ACM Trans. Knowl. Discov. Data* **2009**, *3*, 1–31. [CrossRef]
28. Yang, T.; Chi, Y. Detecting communities and their evolutions in dynamic social networks-a bayesian approach. *Mach. Learn.* **2011**, *82*, 157–189. [CrossRef]
29. Palla, G.; Barabási, A.L.; Vicsek, T. Quantifying social group evolution. *Nature* **2007**, *446*, 664–667. [CrossRef]
30. Cazabet, R.; Amblard, F. Dynamic Community Detection. In *Encyclopedia of Social Network Analysis and Mining*; Springer: New York, NY, USA, 2014; pp. 404–414.
31. Rossetti, G.; Cazabet, R. Community Discovery in Dynamic Networks: A Survey. *ACM Comput. Surv.* **2017**, *51*, 1–37. [CrossRef]
32. Aynaud, T.; Fleury, E.; Guillarme. Communities in evolving networks: Definitions, detection, and analysis techniques. In *Dynamics on and of Complex Networks*; Springer: New York, NY, USA, 2013; Volume 2, pp. 159–200.
33. Hartmann, T.; Kappes, A.; Wagner, D. Clustering Evolving Networks. In *Algorithm Engineering*; Springer: New York, NY, USA, 2016; Volume 9220, pp. 280–329.

34. Agarwal, M.; Ramamritham, K.; Bhide, M. Real time discovery of dense clusters in highly dynamic graphs: Identifying real world events in highly. dynamic environments. *Proc. VLDB Endow.* **2012**, *5*, 980–991. [CrossRef]
35. Tang, L.; Liu, H. Identifying evolving groups in dynamic multimode networks. *IEEE Trans. Knowl. Data Eng.* **2012**, *24*, 72–85. [CrossRef]
36. Sun, J.; Faloutsos, C. Graphscope: Parameter-free of large time evolving-graph. In Proceedings of the 13th Conference on Knowledge Discovery Data Mining, New York, NY, USA, 12–15 August 2007; pp. 687–696.
37. Chakrabarti, D.; Kumar, R.; Tomkins, A. Evolutionary clustering. In Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 20–23 August 2006; pp. 554–560.
38. Chi, Y.; Song, X.D.; Zhou, D.Y.; Koji, H.; Belle, L.T. Evolutionary spectral clustering by incorporating temporal smoothness. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 12–15 August 2007; pp. 153–162.
39. Folino, F.; Pizzuti, C. Multiobjective evolutionary community detection for dynamic networks. In Proceedings of the Conference on Genetic and Evolutionary Computation, Oregon, Portland, 7–11 July 2010; pp. 535–536.
40. Gong, M.G.; Zhang, L.J.; Ma, J.J. Community detection in dynamic social networks based on multi-objective immune algorithm. *J. Comput. Sci. Technol.* **2012**, *27*, 455–467. [CrossRef]
41. Xu, K.S.; Kliger, M.; Hero, A.O., III. Adaptive evolutionary clustering. *Data Min. Knowl. Discov.* **2014**, *28*, 304–336. [CrossRef]
42. Han, Q.; Kevin, X.; Edoardo, A. Consistent estimation of dynamic and multilayer block models. In Proceedings of the 32th International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 1511–1520.
43. Kevin, X. Stochastic block transition models for dynamic networks. In Proceedings of the 18th International Conference on Artificial Intelligence and Statistics, San Diego, California, USA, 9–12 May 2015; pp. 1079–1087.
44. Zhang, X.; Moore, C.; Newman, M.E.J. Random graph models for dynamic networks. *Eur. Phys. J. B* **2017**, *90*, 1–14. [CrossRef]
45. Amir, G.; Pan, Z.; Aaron, C.; Cristopher, M.; Leto, P. Detectability Thresholds and Optimal Algorithms for Community Structure in Dynamic Networks. *Phys. Rev. X* **2016**, *6*, 031005.
46. Sharmodeep, B.; Shirshendu, C. Spectral clustering for multiple dissociative sparse networks. *arXiv* **2017**, arXiv:1805.10594.
47. Paolo, B.; Fabrizio, L.; Piero, M.; Daniele, T. Detectability thresholds in networks with dynamic link and community structure. *arXiv* **2017**, arXiv:1701.05804.
48. Mehrnaz, A.; Theja, T. Block-Structure Based Time-Series Models for Graph Sequences. *arXiv* **2018**, arXiv:1804.08796.
49. Étienne, G.; Anthony, C.; Mustapha, L.; Hanane, A.; Loïc, G. Conditional Latent Block Model: A Multivariate Time Series Clustering Approach for Autonomous Driving Validation. *arXiv* **2020**, arXiv:2008.00946.
50. Emmanuel, A. Community Detection and Stochastic Block Models: Recent Developments. *J. Mach. Learn. Res.* **2017**, *18*, 1–86.
51. Liu, F.; Choi, D. Global spectral clustering in dynamic networks. *Proc. Natl. Acad. Sci. USA* **2018**, *115*, 927–932. [CrossRef] [PubMed]
52. Eagle, N.; Pentland, A.S.; Lazer, D. Inferring friendship network structure by using mobile phone data. *Proc. Natl. Acad. Sci. USA* **2009**, *106*, 15274–15278. [CrossRef] [PubMed]
53. Klimt, B.; Yang, Y. The enron corpus: A new dataset for email classification research. In Proceedings of the European Conference on Machine Learning, Pisa, Italy, 20–24 September 2004; Springer: Berlin, Germany, 2004; pp. 217–226.