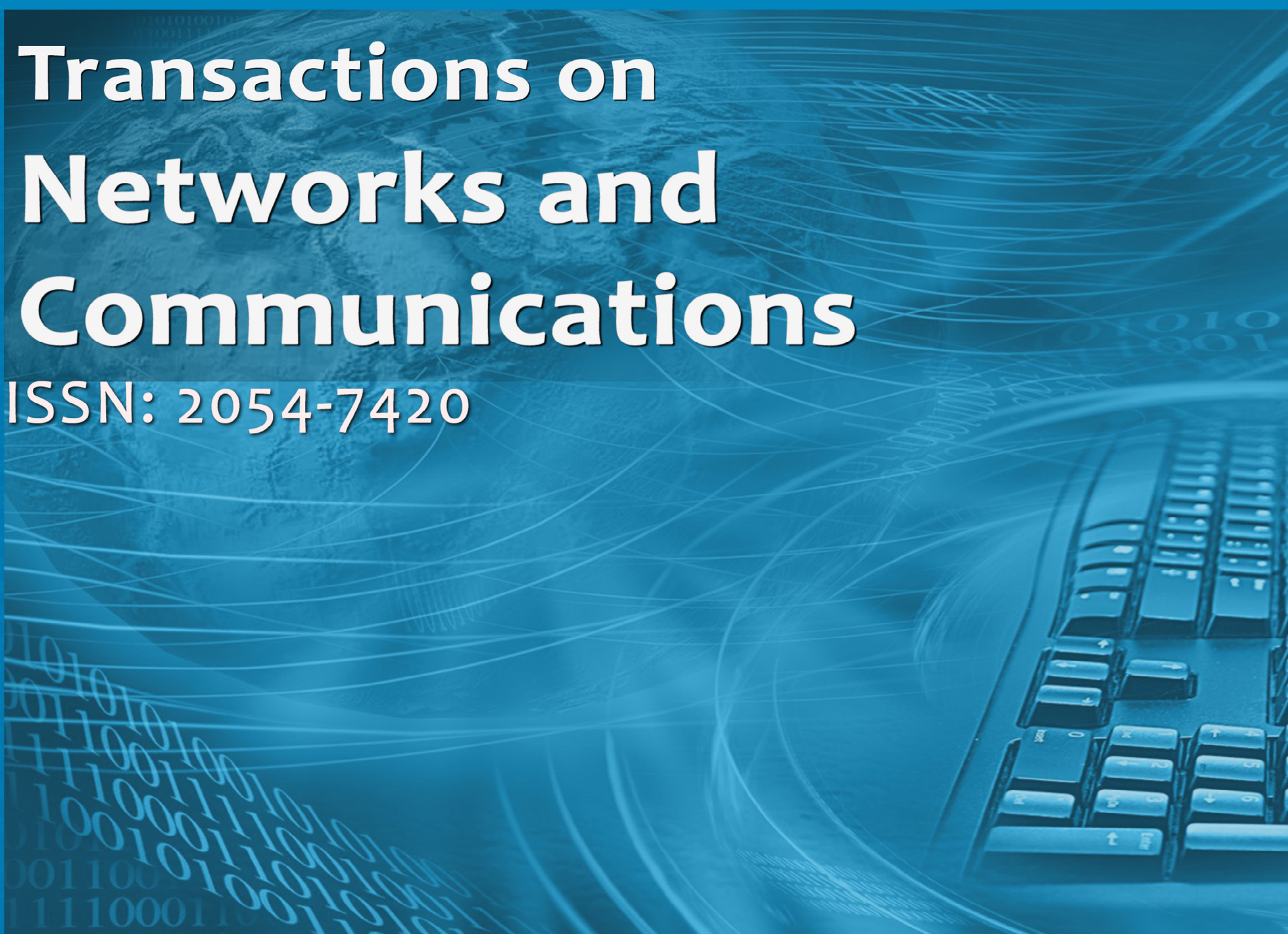
Volume 5 No. 5, Oct 2017



Transactions on Networks and Communications

ISSN: 2054-7420

Society for Sciences and Education - United Kingdom

TABLE OF CONTENTS

EDITORIAL ADVISORY BOARD

DISCLAIMER

All the contributions are published in good faith and intentions to promote and encourage research activities around the globe. The contributions are property of their respective authors/owners and the journal is not responsible for any content that hurts someone's views or feelings etc.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TNC TRANSACTIONS ON NETWORKS AND COMMUNICATIONS

VOLUME 5, NO. 5
ISSN: 2054 -7420

Stochastic Hybrid Dynamic Multicultural Social Networks

Kristina B. Hilton, G.S. Ladde
Department of Mathematics and Statistics, University of South Florida, Tampa, FL, United States;
kristinabuzzell@hotmail.com; gladde@usf.edu

ABSTRACT

In this work, we investigate the cohesive properties of a stochastic hybrid dynamic multi-cultural network under random environmental perturbations. By considering a multi-agent dynamic network, we model a social structure and find conditions under which cohesion and coexistence is maintained using Lyapunov's Second Method and the comparison method. In this paper, we present a prototype illustration that exhibits the significance of the framework and approach. Moreover, the explicit sufficient conditions in terms of system parameters are given to exhibit when the network is cohesive both locally and globally. The sufficient conditions are algebraically simple, easy to verify, and robust. Further, we decompose the cultural state domain into invariant sets and consider the behavior of members within each set. We also analyze the degree of conservativeness of the estimates using Euler-Maruyama type numerical approximation schemes based on the given illustration.

Keywords: Multi-agent Network; Cohesiveness; Lyapunov Second Method; Invariant Sets.

1 Introduction

The aim of this work is to explore and extend the cohesive properties of a dynamic network of multi-agents/members with a desired minimum safe distance between the members of the network [1-3] under the influence of both continuous and discrete-time stochastic perturbations. Dynamic network models play an important role in a variety of modeling applications. For example, economics, finance, engineering, management sciences, and biological networks have considered such large-scale dynamic models to investigate connectivity, stability, dynamic reliability, and convergence [4-7].

One of the concepts studied using a dynamic social network is that of consensus [8-11]. In such models, the conditions under which a group collectively comes to an agreement on an issue under consideration are studied. Another question of interest for such a network is when the group may divide into subgroups with an agreement reached within the subgroup but never reaching a consensus at an overall group level. Most of the work done in these areas look to develop consensus seeking algorithms and consider long term stability of the network in consideration [12-15].

The concepts of cohesion, coordination, and cooperation within a group are often multi-faceted, dynamic and complex, but are important concepts when trying to better understand how nations or communities function [16]. We seek to better understand the group dynamics of such a society in order to create policies and practices that encourage a sense of community among individuals from a variety of cultural backgrounds.

DOI: 10.14738/tnc.55.3643
Publication Date: 7th October 2017
URL: http://dx.doi.org/10.14738/tnc.55.3643

In fact, we systematically initiated the study of this issue [2, 3] to better understand the social dynamics of a group seeking to find such a balance under the influence of both continuous and discrete-time deterministic and stochastic perturbations. In doing so, we are interested in better understanding the cohesive properties of a multi-cultural social network. In this work, we further extend the developed results in the framework of hybrid stochastic dynamic model for which we explore the features of the network. By considering a hybrid dynamic [17], we are able to consider the impact that events both from external and internal stochastic fluctuations coupled with an intervention process on the network have on the cultural dynamics. The presented work is used to exhibit the quantitative and qualitative properties of the network. Further, the techniques used are computationally attractive and algebraically simple relating with the underlying network parameters. This feature plays an important role for planning and decision processes.

In Section 2, we present a general problem under consideration and the underlining assumptions. We then present an illustration of such a network in Section 3 to exhibit the role and scope of the underlying complexity with the simplicity without loss of generality. Using an appropriate energy function and the comparison method, upper and lower estimates on cultural states are established in Section 4. In Section 5, the long-term behavior of the solutions to the comparison equations are examined and we explore the study of the cultural state invariant sets in the context of the illustration presented in Section 3. In Section 6, we use numerical simulations to model the network and to better understand to what extent the analytically developed estimates in Section 5 are feasible. Overall, the presented results are conservative but are reliable and robust.

2 Problem Formulation

The network consists of *m* agents whose position at time *t* is represented by $x_i(t), i \in I(1,m) = \{1,2,\ldots,m\}$, with $x_i(t) \in \mathbb{R}^n$. In our model, this vector does not represent a geographical location but rather a cultural position of the *i*th member. That is to say, the vector x_i is a numerical representation of the *i*th member's beliefs or background on certain cultural or ethnic practices relevant to the network under study. Further, we assume that $\xi_{ij}, i, j \in I(1,m)$ is a normalized Wiener process such that $\xi_{ij} = \xi_{ji}$ and for $j \neq k, \xi_{ij}$ and ξ_{ik} are independent. We then consider a system of Itô-Doob type stochastic system of differential equations that describes the cultural state dynamic process:

$$\begin{cases} dx_i &= \sum_{j=1}^{m} f(t, x_i, x_i - x_j, k-1)dt + \sum_{j \neq i}^{m} \sigma(t, x_i - x_j, k-1) d\xi_{ij}(t), \\ \Delta x_i^k &= I\left(x_i^{k-1}(t_k^-, t_{k-1}, x_i^{k-1}), k\right), \qquad x_i^0\left(t_0\right) = x_i^0, \end{cases} \tag{1}$$

for $(t, x_i) \in [t_{k-1}, t_k) \times \mathbb{R}^n$ and $k \in I(1,\infty)$, where $x_i, x_j \in \mathbb{R}^n$ are continuous time dynamic states; $i, j \in I(1,m)$; f and σ are drift and diffusion rate coefficient functions, respectively; and $\Delta x_i^k = x_i^k - x_i^{k-1}$, where I in (1) stands for a discrete time intervention dynamic process. We will also make the following assumptions:

Assumption H_1: For

- $x_i^{k-1}(t_{k-1}) = x_i^{k-1}$ is an *n*-dimensional initial cultural state random vector defined on the complete probability space (Ω, F, P) and $k \in I(0,\infty)$ at the *k*th intervention time;
- x_i^{k-1} and $\xi_{ij}(t)$ are mutually independent for each $t_{k-1} \le t < t_k$ for $i \ne j$, $i, j \in I(1,m)$ and $k \in I(0,\infty)$;
- For $i, j \in I(1,m)$, $\xi_i(t) = (\xi_{i1}, \xi_{i2}, \ldots, \xi_{ij} \ldots, \xi_{im})^T$ is a *m*-dimensional normalized Wiener process of independent increments for $i \in I(1,m)$;
- ξ_{ij} are F_t -measurable for all $t \ge t_0$ and $\xi_{ij}(t+h) - \xi_{ij}(t)$ is independent of F_t , where F_t represents an increasing family of the smallest sub- σ algebra of F , i.e. $F_s \subset F_t$ if $t_0 < s < t$;
- $x_i(t_0)$ is F_{t_0} measurable;
- $\{t_k\}_{k=1}^{\infty}$ is a sequence of intervention time, and $t_k \to \infty$ as $k \to \infty$;
- f and σ are defined on: $\mathbb{R}_+ \times \mathbb{R}^n \times \mathbb{R}^n \times I(1,\infty)$ into $\mathbb{R}^n$ and continuous on $[t_{k-1}, t_k) \times \mathbb{R}^n \times \mathbb{R}^n$ for each $(t, x, y) \in [t_{k-1}, t_k) \times \mathbb{R}^n \times \mathbb{R}^n$;
- f and σ satisfy for each $k \in I(1,\infty)$ and for each $(t, x, y, k) \in [t_{k-1}, t_k) \times \mathbb{R}^n \times \mathbb{R}^n \times I(1,\infty)$

$$\begin{aligned} f(t,x,y,k-1) &\to f(t_k^-,x,y,k-1) \\ \sigma(t,x,y,k-1) &\to \sigma(t_k^-,x,y,k-1) \end{aligned} \tag{2}$$

as $t \to t_k^-$;

- $I : \mathbb{R}^n \times I(1,\infty) \to \mathbb{R}^n$ is a Borel measurable discrete time intervention function.

It is assumed that the initial value problem (1) for the system of stochastic differential equations has a solution process.

We wish to investigate the stochastic cohesive property of such a network. Further, we will explore the behavior of a member of the network based on the cultural state distance between a network member cultural state and the cultural state center of the network.

Below, we state a few definitions with regard the quantitative and qualitative behavior of the cultural network.

Definition 1: Let r_1 and r_2 be non-negative random functions for $t \in [t_{k-1}, t_k)$, $k \in I(1,\infty)$ such that $r_1 \le r_2$. We say that a stochastic multicultural dynamic network is:

i. *locally cohesive with probability* 1, if for any $N \in F_t$ such that $P(N) = 0, N \subset \Omega$ and for all $t \in [t_{k-1}, t_k)$,

$$r_1(t) \le \left\| x_i^{k-1}(t) - x_j^{k-1}(t) \right\| \le r_2(t), \tag{3}$$

for all $i, j \in I(1,m)$;

ii. *locally cohesive in probability*, if for all $i, j \in I(1,m)$, $t \in [t_{k-1}, t_k)$, and any $0 < \epsilon < 1$

$$P\left(\left\{\Omega : \left\|x_i^{k-1}(t) - x_j^{k-1}(t)\right\| < r_1(t) \text{or} \left\|x_i^{k-1}(t) - x_j^{k-1}(t)\right\| > r_2(t)\right\}\right) < \epsilon, \tag{4}$$

for all $i \in I(1,m)$;

iii. *locally cohesive in pth mean*, if for all $i, j \in I(1,m)$ and $t \in [t_{k-1}, t_k)$

$$E\left[r_1(t)\right] \le E\left[\left\|x_i^{k-1}(t) - x_j^{k-1}(t)\right\|^p\right] \le E\left[r_2(t)\right], \tag{5}$$

for all $i \in I(1,m)$.

If (i.), (ii.), or (iii.) exist for all $t \in [t_0, \infty)$, we say the network is globally cohesive with probability 1, in probability or in *p*th mean respectively.

Definition 2: We say that a stochastic multicultural dynamic network

i. *locally reaches a consensus with probability* 1, if there exists $N \subset F$ such that $P(N) = 0$ for all $\omega \in \Omega \setminus N$,

$$\lim_{t \to \infty} \left\|x_i^{k-1}(t) - \overline{x}^{k-1}\right\| = 0, \tag{6}$$

for $k \in I(1,\infty)$ and all $i, j \in I(1,m)$;

ii. *locally reaches a consensus in probability*, if for $\epsilon > 0$ and $t \in [t_{k-1}, t_k), k \in I(1,\infty)$

$$\lim_{t \to \infty} P\left(\left\{\left\|x_i^{k-1}(t) - \overline{x}^{k-1}\right\| > \epsilon\right\}\right) = 0, \tag{7}$$

for all $i \in I(1,m)$;

iii. *locally reaches a consensus in the pth mean*, if for $t \in [t_{k-1}, t_k), k \in I(1,\infty)$

$$\lim_{t \to \infty} E\left[\left\|x_i(t) - \overline{x}\right\|^p\right] = 0, \tag{8}$$

for all $i \in I(1,m)$.

If (i.), (ii.), or (iii.) exist for all $t \in [t_0, \infty)$, we say the network is reaches a global consensus with probability 1, in probability or in *p*th mean respectively.

Let x_i^{k-1} and x_j^{k-1} be cultural state random vectors for $i, j \in I(1,m)$ and $k \in I(1,\infty)$. For $t \in [t_{k-1}, t_k)$, we define the relative cultural state affinity with probability 1 sense by

$$\left\|x_i^{k-1}(t) - x_j^{k-1}(t)\right\|. \tag{9}$$

We note that the relative cultural state affinity in the a.s. sense exists as $\|\bullet\|$ is Borel measurable.

3 Prototype Dynamic Model

Let us define a prototype multicultural network dynamic model under the stochastic environmental perturbations described by the Itô-Doob type stochastic system of differential equations

$$\begin{cases} dx_i^{k-1} &= \Big[a_{k-1} \sum_{j=1}^{m} x_{ij}^{k-1} - q_{k-1} \left\| x_i^{k-1} - \bar{x}^{k-1} \right\|^2 \sum_{j=1}^{m} x_{ij}^{k-1} \\ & + b_{k-1} \sin\left\| x_i^{k-1} - \bar{x}^{k-1} \right\| \sum_{j=1}^{m} x_{ij}^{k-1} \exp\left[-\frac{\left\| x_{ij}^{k-1} \right\|^2}{c_{k-1}} \right] \Big] dt \\ & + \beta_{k-1} \sin\left\| x_i^{k-1} - \bar{x}^{k-1} \right\| \sum_{j=1}^{m} x_{ij}^{k-1} \exp\left[-\frac{\left\| x_{ij}^{k-1} \right\|^2}{c_{k-1}} \right] d\xi_{ij}, \\ x_i^k &= (1+\delta_i^{k-1}) x_i^{k-1}(t_k^-, t_{k-1}, x_i^{k-1}), \qquad x_i^0(t_0) = x_i^0, \end{cases} \tag{10}$$

for $t \in [t_{k-1}, t_k)$, $k \in I(0,\infty)$ and where $a_{k-1}, q_{k-1}, b_{k-1}, c_{k-1}$ and β_{k-1} are positive real numbers, and

$$x_{ij}^{k-1} = x_i^{k-1} - x_j^{k-1}, \tag{11}$$

We note that the solution process x_i of (10) is defined by

$$x_i(t, t_0, x_i^0) = \begin{cases} x_i^0(t, t_0, x_i^0) & t_0 \le t < t_1, \\ x_i^1(t, t_1, x_i^1) & t_1 \le t < t_2, \\ \vdots & \vdots \\ x_i^{k-1}(t, t_{k-1}, x_i^{k-1}) & t_{k-1} \le t < t_k, \\ \vdots & \vdots \end{cases} \tag{12}$$

Here, $\bar{x}^{k-1}$ is the center of the multicultural dynamic system (1) defined by:

$$\bar{x}^{k-1} = \frac{1}{m} \sum_{j=1}^{m} x_j^{k-1}(t), \qquad t \in [t_{k-1}, t_k), \tag{13}$$

and note that by substituting for x_i^{k-1} by $\bar{x}^{k-1}$ into (10), we have

$$
\begin{aligned}
d\bar{x}^{k-1} &= \Big[a_{k-1}\sum_{j=1}^{m}\left(\bar{x}^{k-1}-x_j^{k-1}\right)-q_{k-1}\left\|\bar{x}^{k-1}-\bar{x}^{k-1}\right\|^2\sum_{j=1}^{m}\left(\bar{x}^{k-1}-x_j^{k-1}\right) \\
&+b_{k-1}\sin\left\|\bar{x}^{k-1}-\bar{x}^{k-1}\right\|\sum_{j=1}^{m}\left(\bar{x}^{k-1}-x_j^{k-1}\right)\exp\left[-\frac{\left\|\bar{x}^{k-1}-x_j^{k-1}\right\|^2}{c_{k-1}}\right]\Big]dt \\
&+\beta_{k-1}\sin\left\|\bar{x}^{k-1}-\bar{x}^{k-1}\right\|\sum_{j=1}^{m}\left(\bar{x}^{k-1}-x_j^{k-1}\right)\exp\left[-\frac{\left\|\bar{x}^{k-1}-x_j^{k-1}\right\|^2}{c_{k-1}}\right]d\xi_{\bar{x}^{k-1}j}, \qquad (14) \\
&= a_{k-1}m\bar{x}^{k-1}-a_{k-1}\sum_{j=1}^{m}x_j^{k-1} \\
&= a_{k-1}m\bar{x}^{k-1}-a_{k-1}m\bar{x}^{k-1} \\
&= 0,
\end{aligned}
$$

for $t\in[t_{k-1},t_k)$, $k\in I(1,\infty)$. Thus $\bar{x}^{k-1}$ defined in (13) is a stationary center of the multicultural dynamic network on each interval $[t_{k-1},t_k)$. We define the transformation $z_i^{k-1}=x_i^{k-1}-\bar{x}^{k-1}$ and observe that $x_{ij}^{k-1}=z_i^{k-1}-z_j^{k-1}=z_{ij}^{k-1}$. Then the transformed network dynamic model corresponding to (10) is reduced to:

$$
\begin{cases}
dz_i^{k-1} = \left[a_{k-1}mz_i^{k-1}-q_{k-1}\left\|z_i^{k-1}\right\|^2 mz_i^{k-1}+b_{k-1}\sin\left\|z_i^{k-1}\right\|\sum_{j=1}^{m}z_{ij}^{k-1}\exp\left[-\frac{\left\|z_{ij}^{k-1}\right\|^2}{c_{k-1}}\right]\right]dt \\
\qquad +\beta_{k-1}\sin\left\|z_i^{k-1}\right\|\sum_{j=1}^{m}z_{ij}^{k-1}\exp\left[-\frac{\left\|z_{ij}^{k-1}\right\|^2}{c_{k-1}}\right]d\xi_{ij}, \qquad t\in[t_{k-1},t_k) \\
z_i^k = \left(1+\delta_i^{k-1}\right)z_i^{k-1}\left(t_k^-,t_{k-1},z_i^{k-1}\right), \qquad z_i^0(t_0)=z_i^0.
\end{cases} \qquad (15)
$$

The center $\bar{x}^{k-1}$ of the multicultural dynamic model (10) is reduced to the center zero in (15) over each interval $[t_{k-1},t_k)$ and $k\in I(1,m)$. For each $k\in I(1,\infty), a_k, b_k, c_k, q_k$ represent the weight of the social moderation attractiveness (q_k), the repulsive forces (a_k), the rate of decay of the long-range attractiveness (c_k), and the long-range attractiveness (b_k) between individual members and social groups. Further, the parameter β_k characterizes the random environmental perturbations. It exhibits both attractive and repulsive forces that are centered at the center of the network. The magnitude of the repulsive forces over $[t_{k-1},t_k)$ are described by $a_{k-1}m\left\|z_i^{k-1}\right\|$ and the magnitude of the long range deterministic attractive forces are characterized by

$$b_{k-1}\left\|\sum z_{ij}^{k-1}\exp\left[-\frac{\left\|z_{ij}^{k-1}\right\|^2}{c}\right]\right\|. \tag{16}$$

Further, $\sin\left\|z_i^{k-1}\right\|$ is the sine-cyclical influence of the *i*th member's relative distance to the center of the network. The stochastic term represents the environmental influence due to long-range attractive forces. In particular, in the case of a multi-cultural network, the noise captures the uncertainty generated due to the membership interactions and deliberations under the influence of the long-range cultural forces.

We remark that the solution process of (15) can be re-casted as (12). In order to study the multicultural dynamics (15), we use Lyapunov's Second Method in conjunction with the comparison method [18]. These methods are computationally attractive and provide a means of better understanding the movement and behavior of the state memberships of the network. By utilizing these methods, we are able to establish conditions for which we have both upper and lower estimates on the members' cultural state positions on the interval $[t_{k-1},t_k)$ for $k\in I(1,m)$. In this work, we assume that all inequalities are with probability 1.

4 Upper and Lower Comparison Equations

Using Lyapunov's Second Method and differential inequalities, we first seek a function $r_{k-1}(t,t_{k-1},u_{k-1})$ such that

$$\left\|z_i^{k-1}(t)\right\|\le r_{k-1}(t,t_{k-1},r_{k-1}),\qquad t\in[t_{k-1},t_k). \tag{17}$$

From Definition 1, relation (17) generates a concept of a locally upper-cohesive cultural network in the almost sure sense on the *k-1*th interval for $k\in I(1,\infty)$.

To this end, for $t\in[t_{k-1},t_k)$ let us choose an energy function V_{k-1} as:

$$V_{k-1}(z_i^{k-1})=\left\|z_i^{k-1}\right\|=\left(\left(z_i^{k-1}\right)^T z_i^{k-1}\right)^{\frac{1}{2}}. \tag{18}$$

We have previously shown [2, 3] that the differential of V_{k-1} in the direction of the vector field represented by (15) is

$$
\begin{aligned}
dV_{k-1} &= \frac{\left(z_i^{k-1}\right)^T dz_i^{k-1}}{\left\|z_i^{k-1}\right\|} + \frac{1}{2}\left[\frac{\left(dz_i^{k-1}\right)^T dz_i^{k-1}}{\left\|z_i^{k-1}\right\|} - \frac{\left(\left(z_i^{k-1}\right)^T dz_i^{k-1}\right)^2}{\left\|z_i^{k-1}\right\|^3}\right] \\
&= \frac{\left(z_i^{k-1}\right)^T \sum_{j=1}^{m} \phi_2(z_{ij}^{k-1}) d\xi_{ij}}{\left\|z_i^{k-1}\right\|} + LV(z_i^{k-1})dt,
\end{aligned}
\tag{19}
$$

where

$$
\begin{aligned}
\phi_1(z_i^{k-1}) &= a_{k-1} m z_i^{k-1} - q_{k-1}\left\|z_i^{k-1}\right\|^2 m z_i + b_{k-1}\sin\left\|z_i^{k-1}\right\| \sum_{j=1}^{m} z_{ij}^{k-1}\exp\left[-\frac{\left\|z_{ij}^{k-1}\right\|^2}{c_{k-1}}\right], \\
\phi_2(z_{ij}^{k-1}) &= \beta_{k-1}\sin\left\|z_i^{k-1}\right\| z_{ij}^{k-1}\exp\left[-\frac{\left\|z_{ij}^{k-1}\right\|^2}{c_{k-1}}\right],
\end{aligned}
\tag{20}
$$

and

$$
\begin{aligned}
LV_{k-1}(z_i^{k-1}) &= \frac{\left(z_i^{k-1}\right)^T \phi_1(z_i^{k-1})dt}{\left\|z_i^{k-1}\right\|} + \frac{\sum_{j=1}^{m}\phi_2^T(z_{ij}^{k-1})\phi_2(z_{ij}^{k-1})}{2\left\|z_i^{k-1}\right\|} - \frac{\left(\left(z_i^{k-1}\right)^T \sum_{j=1}^{m}\phi_2(z_{ij}^{k-1})\right)^2}{2\left\|z_i^{k-1}\right\|^3} \\
&= \Bigg[a_{k-1} m\left\|z_i^{k-1}\right\| - q_{k-1} m\left\|z_i^{k-1}\right\|^3 + \frac{b_{k-1}\sin\left\|z_i^{k-1}\right\|}{\left\|z_i^{k-1}\right\|}\sum_{j=1}^{m}\left(z_i^{k-1}\right)^T z_{ij}^{k-1}\exp\left[-\frac{\left\|z_{ij}\right\|^2}{c_{k-1}}\right] \\
&+ \frac{\beta_{k-1}^2\sin^2\left\|z_i^{k-1}\right\|\sum_{j=1}^{m}\left(z_{ij}^{k-1}\right)^T z_{ij}^{k-1}\exp\left[-\frac{2\left\|z_{ij}^{k-1}\right\|^2}{c_{k-1}}\right]}{2\left\|z_i^{k-1}\right\|} \\
&- \frac{\beta_{k-1}^2\sin^2\left\|z_i^{k-1}\right\|\sum_{j=1^m}\left(\left(z_i^{k-1}\right)^T z_{ij}^{k-1}\right)^2\exp\left[-\frac{2\left\|z_i^{k-1}\right\|^2}{c_{k-1}}\right]}{2\left\|z_i^{k-1}\right\|^3}\Bigg]dt.
\end{aligned}
\tag{21}
$$

In the following, we present a result that will be used subsequently.

Lemma 1: Let V_{k-1} be the energy function defined in (18) and z_i^{k-1} be a solution of the initial value problem defined in (15). Then, for each $i \in I(1,m), k \in I(1,\infty)$, and $t \in [t_{k-1}, t_k)$,

$$E\left[V_{k-1}(z_i^{k-1}(t+\Delta t)) - V_{k-1}(z_i^{k-1}(t)) \mid F_t\right] = LV_{k-1}(z_i^{k-1}(t))\Delta t, \tag{22}$$

where E stands for the expected value.

Proof: For each $k \in I(1,\infty)$, let $z_i^{k-1}(t, t_{k-1}, z_i(t_{k-1}))$ be the solution process of (15). Let F_t be an increasing family of sub$-\sigma$ algebras as previously defined and set

$$m(t) = E\left[V_{k-1}\left(z_i^{k-1}(t)\right) \mid F_t\right] = V\left(z_i^{k-1}(t)\right), \tag{23}$$

where the last equality holds as $z_i^{k-1}(t)$ is F_t measurable. Similarly, we have set

$$m(t+\Delta t) = E\left[V_{k-1}\left(z_i^{k-1}(t+\Delta t)\right) \mid F_t\right], \tag{24}$$

for all $\Delta t > 0$ sufficiently small such that $(t+\Delta t) \in [t_{k-1}, t_k)$. We consider

$$\begin{aligned} m(t+\Delta t) - m(t) &= E\left[V_{k-1}\left(z_i^{k-1}(t+\Delta t) - V_{k-1}\left(z_i^{k-1}(t)\right)\right) \mid F_t\right] \\ &= E\left[\frac{\partial V_{k-1}}{\partial z}\left(z_i^{k-1}(t)\right)\Delta z_i^{k-1}(t) + \frac{1}{2} tr\left(\frac{\partial^2 V_{k-1}}{\partial z^2}\left(\Delta z_i^{k-1}(t)\right)\left(\Delta z_i^{k-1}(t)\right)^T\right) \mid F_t\right] \\ &= E\left[dV_{k-1}\left(z_i^{k-1}(t)\right) \mid F_t\right]. \end{aligned} \tag{25}$$

This together with (19), yields

$$\begin{aligned} m(t+\Delta t) - m(t) &= E\left[LV_{k-1}\left(z_i^{k-1}(t)\right)\Delta t \mid F_t\right] \\ &= LV_{k-1}\left(z_i^{k-1}(t)\right)\Delta t, \end{aligned} \tag{26}$$

as $z_i^{k-1}(t)$ is F_t measurable. We note that for small Δt, we have

$$dm(t) = LV_{k-1}\left(z_i^{k-1}(t)\right)dt. \tag{27}$$

4.1 Upper Estimate of $LV_{k-1}(z_i^{k-1})$

We seek constraints on the parameters $a_{k-1}, b_{k-1}, c_{k-1}, q_{k-1}$ and β_{k-1}, $k \in I(1,\infty)$ for which we have an upper estimate on $V_{k-1}(z_i^{k-1})$. To this end, imitating the argument made in [3], an upper estimate of LV_{k-1} in (21) is

$$
\begin{aligned}
LV_{k-1} &\le q_{k-1} m \|z_i\| \Big(\frac{a_{k-1}}{q_{k-1}} \\
&+ \frac{4b_{k-1}(m-1)\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right] + \beta_{k-1}^2 (m-1) c_{k-1} \exp[-1]}{4q_{k-1}m} - \|z_i\|^2 \Big) \\
&\le q_{k-1} m V_{k-1}\left(\eta_{k-1}^2 - V_{k-1}^2\right) \\
&\le q_{k-1} m V_{k-1}\left(\eta_{k-1} - V_{k-1}\right)\left(\eta_{k-1} + V_{k-1}\right), \qquad (28)
\end{aligned}
$$

where η_{k-1} is defined by

$$
\eta_{k-1} = \left(\frac{a_{k-1}}{q_{k-1}} + \frac{4b_{k-1}(m-1)\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right] + \beta_{k-1}^2 (m-1) c_{k-1} \exp[-1]}{4q_{k-1}m} \right)^{\frac{1}{2}}. \qquad (29)
$$

From the inequality (28) utilizing the comparison method [18] and Lemma 1, we establish the following lemma. For each interval $[t_{k-1}, t_k)$ and $k \in I(1,\infty)$, the presented result establishes not only an upper bound but also the locally upper cohesive property almost surely. Hereafter, all inequalities and equalities are assumed to be valid with probability one.

Lemma 2:

Let V_{k-1} be the energy function defined in (18), $k \in I(1,\infty), t \in [t_{k-1}, t_k)$, and z_i^{k-1} be a solution of the initial value problem defined in (15). Let $r_{k-1}(t)$ be the maximal solution [18] of a random initial value problem

$$
du_{k-1} = \left[q_{k-1} m u_{k-1}\left(\eta_{k-1} - u_{k-1}\right)\left(\eta_{k-1} + u_{k-1}\right) \right] dt, \qquad u_{k-1}(t_{k-1}) = u_{k-1}, \qquad (30)
$$

where η_{k-1} is defined as in (29). For each $V_{k-1}(z_i^{k-1})$, $i \in I(1,m)$, and $k \in I(1,\infty)$ satisfying the differential inequality (28) and $V_{k-1}(z_i^{k-1}(t_{k-1})) \le u_{k-1}$, it follows that the multicultural dynamic network (10) is upper cohesive on $[t_{k-1}, t_k)$ with probability 1 and

$$
V_{k-1}(z_i^{k-1}(t)) \le r_{k-1}(t, t_{k-1}, u_{k-1}), \qquad (31)
$$

Proof:

From Lemma 1, (28), and the application of stochastic comparison theorem [18], with probability 1, it follows that

$$V_{k-1}(z_i^{k-1}(t)) \le r(t,t_{k-1},u_{k-1}), \tag{32}$$

when $V_{k-1}(z_i(t_{k-1})) \le u_{k-1}$. As the solution to (30) has an upper bound, the network is upper cohesive almost surely.

Remark 1: For each $k \in I(1,\infty)$, if the solution processes of (15) and (30) have a first moment, then the solution process of (15) is locally upper 1st moment cohesive. Furthermore, under the current inequality, it is indeed locally upper cohesive in the sense of probability.

4.2 Lower Estimate of $LV_{k-1}(z_i^{k-1})$

Next we consider the lower comparison equation. Using Lyapunov's Second Method and differential inequalities, we next seek a function $\rho_{k-1}(t,t_{k-1},u_{k-1})$ such that

$$\|z_i(t)\| \ge \rho(t,t_{k-1},\rho_{k-1}), \qquad t \in [t_{k-1},t_k). \tag{33}$$

Again, from Definition 1, relation (33) initiates a notion of a locally lower cohesive cultural dynamic network in the almost sure sense.

Using the energy function defined in (18) and relation (21), for $t \in [t_{k-1},t_k)$ it follows that

$$\begin{aligned} Lv_{k-1} &\ge a_{k-1}mV_{k-1} - q_{k-1}mV_{k-1}^3 - V_{k-1}(m-1)b_{k-1}\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right] \\ &\quad - \frac{\beta_{k-1}^2(m-1)c_{k-1}\exp\left[-1\right]}{4}V_{k-1} \\ &= q_{k-1}mV_{k-1}\Big(\frac{a_{k-1}}{q_{k-1}} \\ &\quad - \frac{4(m-1)b_{k-1}\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right] + \beta_{k-1}^2 c_{k-1}(m-1)\exp\left[-1\right]}{4q_{k-1}m} - V_{k-1}^2\Big). \end{aligned} \tag{34}$$

Assumption H₂: Assume there exists a positive number α_{k-1} such that

$$\alpha_{k-1} \le \left(\frac{a_{k-1}}{q_{k-1}} - \frac{4(m-1)b_{k-1}\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right] + \beta_{k-1}^2(m-1)c_{k-1}\exp\left[-1\right]}{4q_{k-1}m}\right)^{\frac{1}{2}}. \tag{35}$$

From (34), and noticing the fact that assumption H_2 implies

$$\frac{a_{k-1}}{q_{k-1}} > \frac{4(m-1)b_{k-1}\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right]+\beta_{k-1}^{2}(m-1)\exp\left[-1\right]}{4q_{k-1}m}, \tag{36}$$

it follows that

$$LV_{k-1} \geq q_{k-1}mV_{k-1}(\alpha_{k-1}-V_{k-1})(\alpha_{k-1}+V_{k-1}). \tag{37}$$

By inequality (37) and the comparison method [18] and Lemma 1, we establish the following lemma. The presented result provides the lower estimate that in turn establishes the locally lower cohesive property of (15).

Lemma 3: Let V_{k-1} be the energy function defined in (18), $k \in I(1,\infty)$, $t \in [t_{k-1},t_k)$, and z_i^{k-1} be a solution of the initial value problem defined in (15). Let $\rho_{k-1}(t)$ be the minimal solution [18] of a random initial value problem

$$du_{k-1} = q_{k-1}mu_{k-1}\left(\alpha_{k-1}-u_{k-1}\right)\left(\alpha_{k-1}+u_{k-1}\right)dt, \qquad u_{k-1}(t_{k-1}) = u_{k-1}, \tag{38}$$

where α_{k-1} is as defined in (35). For each $V_{k-1}(z_i^{k-1})$, $i \in I(1,m)$, and $k \in I(1,\infty)$ satisfying the differential inequality (37) and $V(z_i^{k-1}(t_{k-1})) \geq u_{k-1}$, it follows that the multicultural dynamic network (10) is lower cohesive on $[t_{k-1},t_k)$ with probability 1 and

$$V_{k-1}(z_i^{k-1}(t)) \geq \rho_{k-1}(t,t_{k-1},u_{k-1}). \tag{39}$$

Proof: From inequality (37) and Lemma 1 and the imitating the outline of the proof of Lemma 2, it follows that

$$V_{k-1}(z_i^{k-1}(t)) \geq \rho(t,t_{k-1},u_{k-1}) \tag{40}$$

provided that $V_{k-1}(z_i^{k-1}(t_{k-1})) \geq u_{k-1}$. As the minimal solution of (38) is a lower bound, the network is lower cohesive almost surely. Moreover, a remark similar to Remark 1 establishes the locally stochastic mean and probability of (15).

We note that comparison differential equations (30) and (38) each have a unique solution process. Therefore the maximal and minimal solutions of (30) and (38) are the unique solutions of the respective random initial value problems.

5 Long-term Behavior of Comparison Differential Equations and Invariant Sets

To appreciate the role and scope of Lemmas 2 and 3, we seek to better understand both the behavior of the network on each interval $[t_{k-1},t_k)$ and the long-term behavior of the network. For this purpose, for

$k \in (1,\infty)$, we find the closed form solutions of the comparison random initial value problems (30) and (38). Moreover, we analyze the qualitative properties of the solutions to the comparison equations. Using the comparison method [18], we are able to establish, quantitatively, the behavior of the individual member cultural dynamic states on the interval $[t_{k-1}, t_k)$. Using this, we also establish the overall long-term behavior of both individual member cultural dynamic states in the network as well as multicultural network state as a whole.

Following the method of finding the closed form solution process of the initial value problem [19], the solution of (38) is represented by

$$u_{k-1}(t, t_{k-1}, u_{k-1}) = \frac{u_{k-1}\nu}{\sqrt{u_{k-1}^2 + \left(\nu^2 - u_{k-1}^2\right)\exp\left[-2\nu^2 q_{k-1} m(t - t_{k-1})\right]}}. \tag{41}$$

As $z_i^k(t_k) = (1+\delta_i^{k-1}) z_i^{k-1}(t_k^-, t_{k-1}, x_i^{k-1})$ for $k \in I(0,\infty)$, we seek to write the initial position u_k in terms of u_0.

By squaring both sides and rearranging the terms, we can write the above as

$$\frac{u_{k-1}^2(t, t_{k-1}, u_{k-1})}{\nu^2 - u_{k-1}^2(t, t_{k-1}, u_{k-1})} = \frac{u_{k-1}^2 \exp\left[2\nu^2 q_{k-1} m(t - t_{k-1})\right]}{\nu^2 - u_{k-1}^2}. \tag{42}$$

We now set

$$y_{k-1}(t, t_{k-1}, y_{k-1}) = \frac{u_{k-1}^2(t, t_{k-1}, u_{k-1})}{\nu^2 - u_{k-1}^2(t, t_{k-1}, u_{k-1})}, \tag{43}$$

where $y(t_{k-1}) = y_{k-1}$ on the interval $[t_{k-1}, t_k)$. Next, we take the derivative of both sides

$$\begin{aligned}
dy_{k-1} &= \frac{2u_{k-1}(t,t_{k-1},u_{k-1})\left[\left(\nu^2 - u_{k-1}^2(t,t_{k-1},u_{k-1})\right) + 2u_{k-1}(t,t_{k-1},u_{k-1})(u_{k-1}^2(t,t_{k-1},u_{k-1})\right]du_{k-1}}{\left(\nu^2 - u_{k-1}^2(t,t_{k-1},u_{k-1})\right)^2} \\
&= \frac{2\nu^2 u_{k-1}(t,t_{k-1},u_{k-1})du_{k-1}}{\left(\nu^2 - u_{k-1}^2(t,t_{k-1},u_{k-1})\right)} \\
&= \frac{2\nu^2 u_{k-1}(t,t_{k-1},u_{k-1})\left(q_{k-1} m u_{k-1}(t,t_{k-1},u_{k-1})(\nu^2 - u_{k-1}^2(t,t_{k-1},u_{k-1}))\right)dt}{\left(\nu^2 - u_{k-1}^2(t,t_{k-1},u_{k-1})\right)^2} \\
&= 2\nu^2 q_{k-1} m \left(\frac{u_{k-1}^2(t,t_{k-1},u_{k-1})}{\nu^2 - u_{k-1}^2(t,t_{k-1},u_{k-1})}\right)dt \\
&= \left(2\nu^2 q_{k-1} m\right) y_{k-1} dt.
\end{aligned} \tag{44}$$

Therefore, on the interval $[t_{k-1}, t_k)$, the solution of **Error! Reference source not found.** is

$$y_{k-1}(t,t_{k-1},y_{k-1})=y_{k-1}\exp\left[2\nu^2 m q_{k-1}(t-t_{k-1})\right], \qquad y_0(t_0)=y_0. \tag{44}$$

Let $\Delta t_k = t_k - t_{k-1}$. When $k=1$, the solution of (44) on $[t_0,t_1)$ is

$$\begin{aligned} y_0(t,t_0,u_0) &= y_0\exp\left[2\nu^2 m q_0(t-t_0)\right] \\ y_0(t_1^-,t_0,u_0) &= y_0\exp\left[2\nu^2 m q_0\Delta t_1\right] \\ y_1(t_1) &= \left|1+\delta_i^0\right| y_0\exp\left[2\nu^2 m q_0\Delta t_1\right]. \end{aligned} \tag{45}$$

We assume that for $k-1\in I(1,\infty)$, the solution of (44) on $[t_{k-1},t_k)$ is

$$\begin{aligned} y_{k-1}(t,t_{k-1},y_{k-1}) &= \prod_{j=1}^{k-1}\left|1+\delta_i^{j-1}\right| y_0\exp\left[2\nu^2 m\left(\sum_{j=1}^{k-1} q_{j-1}\Delta t_{j-1}+(t-t_{k-1})\right)\right] \\ y_{k-1}(t_k^-,t_{k-1},u_{k-1}) &= \prod_{j=1}^{k-1}\left|1+\delta_i^{j-1}\right| y_0\exp\left[2\nu^2 m\sum_{j=1}^{k-1} q_{j-1}\Delta t_j\right] \\ y_k(t_k) &= \left|1+\delta_i^{k-1}\right| y_{k-1}(t_k^-,t_{k-1},u_{k-1}) \\ &= \prod_{j=1}^{k}\left|1+\delta_i^{j-1}\right| y_0\exp\left[2\nu^2 m\sum_{j=1}^{k} q_{j-1}\Delta t_j\right]. \end{aligned} \tag{46}$$

Then for $k\in I(1,m)$, the solution of (44) on $[t_k,t_{k-1})$ is

$$\begin{aligned} y_k(t,t_k,y_k) &= y_k\exp\left[2\nu^2 m q_k(t-t_k)\right] \\ &= \prod_{j=1}^{k}\left|1+\delta_i^{j-1}\right| y_0\exp\left[2\nu^2 m\left(\sum_{j=1}^{k} q_{j-1}\Delta t_j+(t-t_k)\right)\right], \end{aligned} \tag{47}$$

and

$$y_k(t_{k+1}^-,t_k,y_k)=\prod_{j=1}^{k}\left|1+\delta_i^{j-1}\right| y_0\exp\left[2\nu^2 m\sum_{j=1}^{k+1} q_{j-1}\Delta t_j\right], \tag{48}$$

so

$$\begin{aligned} y_{k+1}(t_{k+1}) &= \left|1+\delta_i^{k+1}\right| y_k(t_{k+1}^-,t_k,y_k) \\ &= \prod_{j=1}^{k+1}\left|1+\delta_i^{j-1}\right| y_0\exp\left[2\nu^2 m\sum_{j=1}^{k+1} q_{j-1}\Delta t_j\right]. \end{aligned} \tag{49}$$

Therefore, using mathematical induction, it follows that for any $k\in I(1,\infty)$,

$$\begin{aligned} y_{k-1}(t,t_{k-1},y_{k-1}) &= \prod_{j=1}^{k-1}\left|1+\delta_i^{j-1}\right| y_0 \exp\left[2\nu^2 m\left(\sum_{j=1}^{k-1} q_{j-1}\Delta t_{j-1} + (t-t_{k-1})\right)\right] \\ y_k(t_k) &= \prod_{j=1}^{k}\left|1+\delta_i^{j-1}\right| y_0 \exp\left[2\nu^2 m \sum_{j=1}^{k} q_{j-1}\Delta t_j\right]. \end{aligned} \quad (50)$$

From the definition of y_k and (50), for $k \in I(1,\infty)$ and $t \in [t_{k-1}, t_k)$

$$\begin{aligned} u_{k-1}^2(t,t_{k-1},u_{k-1}) &= \frac{\nu^2 y_{k-1}(t,t_{k-1},y_{k-1})}{1+y_{k-1}(t,t_{k-1},y_{k-1})} \\ &= \frac{\nu^2 \prod_{j=1}^{k-1}\left|1+\delta_i^{j-1}\right| u_0^2}{\prod_{j=1}^{k-1}\left|1+\delta_i^{j-1}\right| u_0^2 + \left(\nu^2-u_0^2\right)\exp\left[-2\nu^2 m\left(\sum_{j=1}^{k-1} q_{j-1}\Delta t_j + (t-t_{k-1})\right)\right]} \\ &= \frac{\nu^2 u_0^2}{u_0^2 + (\nu^2-u_0^2)\exp\left[-2\nu^2 m\left(\sum_{j=1}^{k-1} q_{j-1}\Delta t_j + (t-t_{k-1})\right)\right]\prod_{j=1}^{k-1}\left|1+\delta_i^{j-1}\right|^{-1}} \end{aligned} \quad (51)$$

and

$$u(t,t_k,u_k) = \frac{\nu u_0}{\left(u_0^2 + \left(\nu^2-u_0^2\right)\exp\left[-2\nu^2 m\left(\sum_{j=1}^{k} q_{j-1}\Delta t_{j-1} + (t-t_k)\right)\right]\prod_{j=1}^{k-1}\left|1+\delta_i^{j-1}\right|^{-1}\right)^{\frac{1}{2}}}. \quad (52)$$

Further, for $k \in (1,\infty)$,

$$u(t_k) = \frac{u_0 \nu}{\left(u_0^2 + (\nu^2-u_0^2)\exp\left[-2\nu^2 m \sum_{j=1}^{k+1} q_{j-1}\delta t_{j-1}\right]\prod_{j=1}^{k}\left|1+\delta_i^{j-1}\right|^{-1}\right)^{\frac{1}{2}}}. \quad (53)$$

By (53), taking the limit as $k \to \infty$, it follows that the initial positions u_{k-1} will converge and

$$\lim_{k\to\infty} u_{k-1} = \nu. \quad (54)$$

Further, by (52)

$$\lim_{k\to\infty} u(t,t_{k-1},u_{k-1}) = \nu. \quad (55)$$

Therefore, taking the limit of the upper comparison solution $r(t,t_0,u_0)$ at $t \to \infty$, the long term behavior of is such that

$$\lim_{t\to\infty} r(t,t_0,u_0)=\eta, \tag{56}$$

where

$$\eta=\limsup_{k\to\infty}\eta_{k-1},$$

if it exists and

$$\eta_{k-1}=\left(\frac{a_{k-1}}{q_{k-1}}+\frac{4b_{k-1}(m-1)\sqrt{\frac{1}{2}}\exp\left[-\frac{1}{2}\right]+\beta_{k-1}^2(m-1)c_{k-1}\exp[-1]}{4q_{k-1}m}\right)^{\frac{1}{2}}. \tag{57}$$

Thus, if the limit superior η exists, the solution process of (15) is globally upper cohesive in the a.s. sense on $[0,\infty)$.

Similarly, the limit of the solution of the lower comparison equation (38) as $t\to\infty$ is

$$\lim_{t\to\infty}\rho(t,t_0,u_0)=\alpha, \tag{58}$$

where

$$\alpha=\liminf_{k\to\infty}\alpha_{k-1} \tag{59}$$

and

$$\alpha_{k-1}\geq\left(\frac{a_{k-1}}{q_{k-1}}-\frac{4(m-1)b_{k-1}\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right]+\beta_{k-1}^2(m-1)c_{k-1}\exp[-1]}{4q_{k-1}m}\right)^{\frac{1}{2}}. \tag{60}$$

Moreover, the solution process of (15) is globally lower cohesive a.s. on $[t_0,\infty)$.

Using the long term behavior of the comparison equations in conjunction with Lemmas 2 and 3, we establish the following theorem.

Theorem: Let the hypotheses of Lemmas 2 and 3 be satisfied. Then the network is locally cohesive in the almost surely on $[t_{k-1},t_k)$ for $k\in I(1,\infty)$. If additionally η exists and is finite, then the network is globally cohesive in the almost surely on $[t_0,\infty)$.

Proof: From Lemmas 2 and 3,

$$\rho_{k-1}(t,t_{k-1},\rho_{k-1}) \le V_{k-1}(z_i^{k-1}(t)) \le r_{k-1}(t,t_{k-1},r_{k-1}) \tag{61}$$

with probability 1. Moreover, as the solution to the upper comparison equation is bounded above by η_{k-1} and the solution to the lower comparison equation is bounded below by α_{k-1}, the network is cohesive almost surely. Suppose that η exist and is finite. Then, we have

$$\rho(t,t_0,u_0) \le V(z_i(t,t_0,z_i^0)) \le r(t,t_0,u_0) \tag{62}$$

for $t \ge t_0$. As the solutions ρ and r are bounded, the network is globally cohesive with probability 1.

5.1 Invariant Sets

In the case of the hybrid stochastic dynamical network, we can first consider the behavior of the solution process on the interval $[t_{k-1},t_k)$. For $k \in I(1,\infty)$, let us denote

$$r_2 = \left(\frac{a_{k-1}}{q_{k-1}} - \frac{4(m-1)b_{k-1}\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right] + \beta_{k-1}^2(m-1)c_{k-1}\exp\left[-1\right]}{4q_{k-1}m} \right)^{\frac{1}{2}} \tag{63}$$

and

$$r_1 = \left(\frac{a_{k-1}}{q_{k-1}} + \frac{4b_{k-1}\left(m-1\right)\sqrt{\frac{c_{k-1}}{2}}\exp\left[-\frac{1}{2}\right] + \beta^2\left(m-1\right)c_{k-1}\exp\left[-1\right]}{4q_{k-1}m} \right)^{\frac{1}{2}}. \tag{64}$$

Further, let us define the following sets:

$$\begin{cases} A_{k-1} & = B\left(0,r_2\right) \\ B_{k-1} & = B^c\left(0,r_2\right) \cap B\left(0,r_1\right) \\ C_{k-1} & = B^c\left(0,r_1\right) \end{cases} \tag{65}$$

From the analysis developed in that section, we establish the following theorem for the solution on the interval $[t_{k-1},t_k)$.

Theorem: Let the hypotheses of Lemmas 2 and 3 be satisfied. Then almost surely,

i. the set $A_{k-1} \cup B_{k-1}$ is conditionally invariant relative to A_{k-1};

ii. the set B_{k-1} is self-invariant;

iii. the set $B_{k-1} \cup C_{k-1}$ is conditionally invariant relative to C_{k-1}.

Proof: Following the proof outlined in [3], the result follows directly.

By considering the limit as $k \to \infty$, we also establish the following result for the long-range invariant sets of (15).

For $k \in (1,\infty)$

$$\lim_{k\to\infty} u(t_{k-1}) = \lim_{k\to\infty} u(t, t_{k-1}, u_{k-1}) \tag{66}$$

for both the upper and lower comparison equations, then as $k \to \infty$

$$\alpha \leq \left\| z_i^{k-1}(t_{k-1}) \right\| \leq \eta \tag{67}$$

and

$$\alpha \leq \left\| z_i^{k-1}(t, t_{k-1}, z_i^{k-1}) \right\| \leq \eta \tag{68}$$

for sufficiently large $k \in (1,\infty)$. Thus, (15) exhibits long-range self-invariance for every member of the network.

In Section 6, we use numerical simulations to better understand the estimates and network behavior on the intervals $[t_{k-1}, t_k)$ for a finite number k.

6 Numerical Simulations

In this section, we consider numerical simulations for the multicultural dynamic network governed by the stochastic differential equation (15). We use a Euler-Maruyama [20-22] type numerical approximation scheme. We consider a network of six members, using the same initial position and varying the parameters a_{k-1}, b_{k-1}, and β_{k-1}, $k \in I(1,\infty)$. Further, we consider the case such that $\xi_{ij}^{k-1}(t)$ for $i, j \in I(1,6)$ is a one-dimensional Brownian motion process with mean of zero and variance of 1 over the interval $[0,1]$.

Often in a cultural network, events such as natural disasters, sudden political or economic changes, etc., can cause rippling effects in the cultural network. These changes can be characterized by the parametric changes in the stochastic differential equation (15). Therefore, we choose to simulate such a situation in the models in this section. Here, we choose 5 arbitrary times t_k on the interval $(0,1)$ for which the model experiences an intervention on the dynamic. Further, for each t_k, $k \in I(1,5)$, we set $x_i^k(t_k) = (1+\delta_i^k) x_i^k(t^-)$, where δ_i^k is a constant for fixed i and $k \in I(1,5)$, and consider the various scenarios based on changing the parameters a_k, b_k and β_k.

In order to consider the effects of changing the parametric quantity a_{k-1}, we consider various models for which $\beta_{k-1} = 2, b_{k-1} = 1, c_{k-1} = 2$, and $q_{k-1} = 1/7$ are held constant for $k \in I(1,5)$ and $a_k = a_{k-1} + 1$, $a_0 = 2$. The plot of the position $z_i(t)$ for $t \in [0,1]$ is given in Figure 1.

In order to consider the effects of changing the parametric quantity b_{k-1}, we consider the model for which $a_{k-1} = 2, \beta_{k-1} = 2, c_{k-1} = 2$, and $q_{k-1} = 1/7$ are held constant for $k \in I(1,5)$ and $b_k = b_{k-1} + 1$, $b_0 = 1$. Figure 2 exhibits the simulated positions of the members z_i.

In order to consider the effects of changing the parametric quantity β_{k-1}, we consider the model for which $a_{k-1} = 2, b_{k-1} = 1, c_{k-1} = 2$, and $q_{k-1} = 1/7$ are held constant for $k \in I(1,5)$ and $\beta_k = \beta_{k-1} + 1$, $\beta_0 = 2$. In Figure 3, we plot the positions of the members for $t \in [0,1]$.

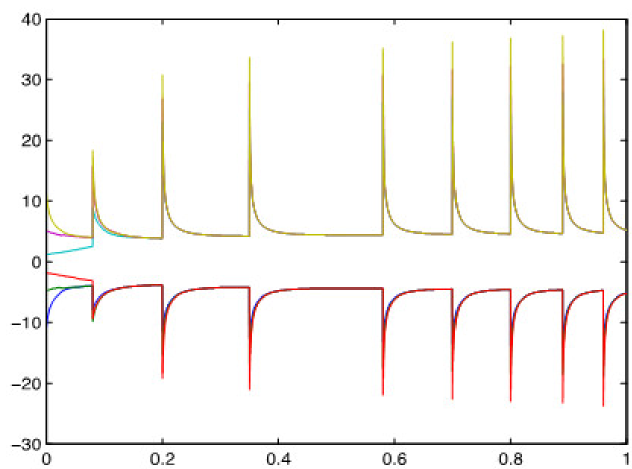


Figure 1: Euler-Maruyama approximation of the differential equation with six members and parameter ak=ak-1+1.

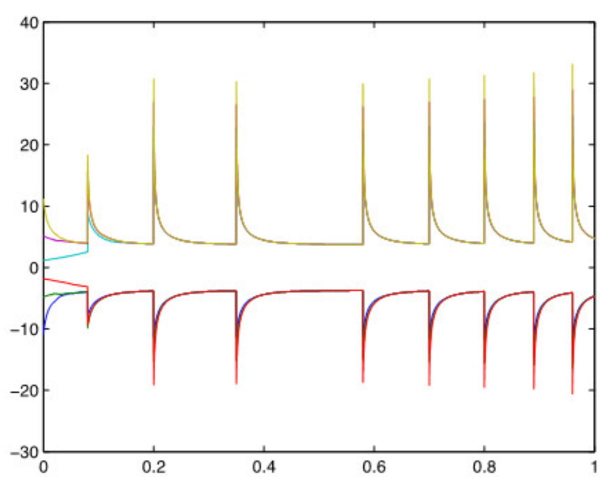


Figure 2: Euler-Maruyama approximation of the differential equation with six members and parameter bk=bk-1+1

In order to consider the effects of a change in the parametric quantity a_{k-1} and β_{k-1}, we consider the model for which $b_k = 1, c_{k-1} = 2$, and $q_{k-1} = 1/7$ are held constant for $k \in I(1,5)$ and $a_k = a_{k-1} + 1$, and $\beta_k = \beta_{k-1} + 1$, $\beta_0 = 2$. The plot of the member's positions of the simulated network is given in Figure 4.

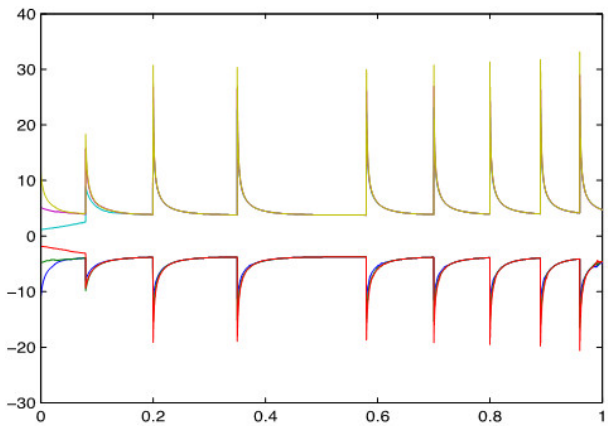


Figure 3: Euler-Maruyama approximation of the differential equation with six members and parameter βk=βk-1+1.

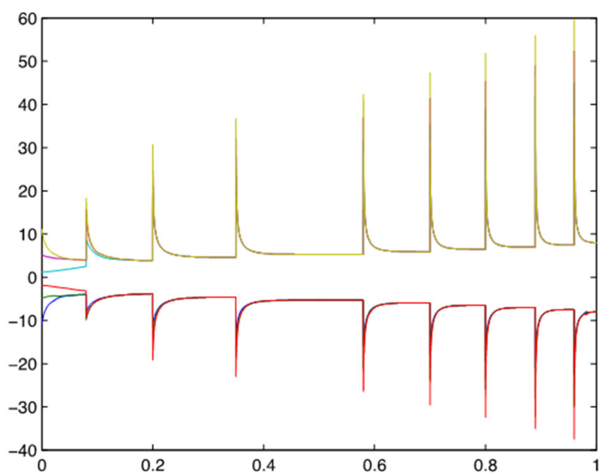


Figure 4: Euler-Maruyama approximation of the differential equation with six members and parameters ak=ak-1+1, bk=bk-1+1, and βk=βk-1+1.

7 Conclusion

Maintaining diversity while simultaneously fostering a sense of community membership, individual cultural identity, and cohesion is currently a goal among communities worldwide. It is important for members in a society to both feel as a part of the community in which they live and interact as well as feel free to embrace a strong sense of self and individuality. We seek to better understand the factors that play a role in obtaining such a balance by considering the impact of the repulsive and attractive forces influencing the multicultural network as in the previous work [2, 3]. Attractive influences can be thought of as attributes that bring people to active membership within the group. Social acceptance, gaining social status, economic opportunity, career growth, common purpose and membership, personal development, and a sense of mutual respect, trust and understanding are examples of attractive influences within a social cultural network. Repelling forces are attributes that create some desire for individuals to leave or be less involved in the group or to preserve some personal identity from one other with their individual magnitude of inner repulsive force. A desire to retain a sense of individuality, economic or emotional cost, interpersonal conflict within the group, or disagreement with parts of the overall philosophies of the group are forces that may be considered as repulsive forces. The goal of the presented multicultural dynamic network is model the balance sought by members of the network in achieving these types of objectives. By doing so, we can consider the impact that policies and environmental factors may have on such a network.

By considering a hybrid dynamic model, we are able to better understand the impacts of outside influences that occur within community members and the cultural impacts such events have on the modeled cultural network. We have considered change based on the parameters that allow the perturbed multicultural dynamic network to remain cohesive while retaining a cultural state that is distinctive from the cultural state center of the network. We established qualitative and quantitative conditions that are computationally attractive and verifiable. We also conducted simulations of the multicultural network that exhibit the influence of the random perturbations and intervention processes as well as demonstrate the long-term behavior of the multicultural network. The presented results provide a tool for planning, performance, and implementations of policies and procedures within a social network.

We are interested in further exploring similar multicultural networks in the context of better understanding the relative cultural affinity $\|x_{ij}\|$ between members within the network and not just the cultural affinity between the cultural state of a member relative to the center of the network. The goal is to better understand the environmental factors that help foster a sense of individuality and diversity between all members within the network while maintaining a cohesive structure.

ACKNOWLEDGEMENT

The authors would like to acknowledge the research support by the Mathematical Sciences Division, US Army Research Office, Grant No. W911NF-15-1-0182.

REFERENCES

[1] Chandra, J. and G.S. Ladde, *Collective behavior of multi-agent network dynamic systems under internal and external perturbations.* Nonlinear Analysis: Real World Applications, 2010. 11(3): p. 1330-1344.

[2] Hilton, K.B. and G.S. Ladde, *Deterministic Multicultural Dynamic Networks: Seeking a Balance between Attractive and Repulsive Forces.* International Journal of Communications, Network and System Sciences, 2016. 9(12): p. 582-602.

[3] Hilton, K.B. and G.S. Ladde, *Stochastic Multicultural Networks.* Dynamical Systems and Applications, 2017. In Press.

[4] Anabtawi, M.J., S. Sathananthan, and G.S. Ladde, *Convergence and stability analysis of large-scale parabolic systems under Markovian structural perturbations.* International Journal of Applied Mathematics, 2000. 2(1): p. 57-85.

[5] Ladde, G.S. and B.A. Lawrence, *Stability and convergence of large-scale stochastic approximation.* International Journal of Systems Science, 1995. 26(3): p. 595-618.

[6] Ladde, G.S. and D.D. Siljak, *Connective stability of large-scale stochastic systems.* International Journal of Systems Science, 1975. 6(8): p. 713-721.

[7] Siljak, D.D., *Large-scale Dynamic Systems: Stability and Structure*1978, New York, NY: Elsevier North-Holland.

[8] Acemoglu, D., et al., *Opinion fluctuations and disagreement in social networks.* Mathematics of Operations Research, 2013. 38(1): p. 1-27.

[9] DeGroot, M.H., *Reaching a consensus.* Journal of the American Statistical Association, 1974. 69(345): p. 118-121.

[10] Friedkin, N.E. *Complex objects in the polytopes of the linear state-space*. arXiv preprint arXiv:1401.5339, 2004. January 2014.

[11] Ma, H., *Literature survey of stability of dynamical multi-agent systems with applications in rural-urban migration.* American Journal of Engineering and Technology Research, 2013. 13(1): p. 131-140.

[12] Cao, Y., et al., *An overview of recent progress in the study of distributed multi-agen coordination.* IEEE Transactions on Industrial Informatics, 2013. 9(1): p. 427-438.

[13] Hu, H.-x., et al., *Group consensus in multi-agent systems with hybrid protocol.* Journal of the Franklin Institute, 2013. 350: p. 575-597.

[14] Huang, M. and J.H. Manton, *Coordination and consensus of networked agents with noisy measurements: stochastic algorithms and asymptotic behavior.* SIAM Journal on Control and Optimization, 2009. 48(1): p. 131-161.

[15] Zhu, Y.-K., X.-P. Guan, and X.-Y. Luo, *Finite-time consensus for multi-agent systems via nonlinear control protocols.* International Journal of Automation and Computing, 2013. 10(5): p. 455-462.

[16] Axelrod, R., *The Complexity of Cooperation: Agent-based Models of Competition*1997, Princeton, NJ: Princeton University Press.

[17] Ladde, G.S., *Hybrid Dynamical Inequalities and Applications.* Dynamical Systems and Applications, 2005. 14: p. 481-514.

[18] Ladde, G.S. and V. Lakshmikantham, *Random Differential Inequalities*1980, New York, NY: Academic Press.

[19] Ladde, A.G. and G.S. Ladde, *An Introduction to Differential Equations: Stochastic Modeling*. Vol. 2. 2013, Hackensak, NJ: World Scientific.

[20] Higham, D.J., *An algorithmic introduction to numerical simulation of stochastic differential equations.* SIAM Review, 2001. 43(3): p. 525-546.

[21] Higham, D.J. and P.E. Kloeden, *Maple and Matlab for stochastic differential equations in finance*, in *Programming Languages and Systems in Computational Economics and Finance*2002, Springer. p. 233-269.

[22] Kloeden, P.E. and E. Platen, *Numerical Solution of Stochastic Differential Equations*1992, New York, NY: Springer-Verlag.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TNC TRANSACTIONS ON NETWORKS AND COMMUNICATIONS

VOLUME 5, NO. 5
ISSN: 2054 -7420

An Achievement of High Availability and Low Cost on Data Center Infrastructure

Yen-Jen Chen and Po-I Lee
Dept. of Electronic Engineering, Ming Chi University of Technology, Taiwan, ROC
yjchen.net@gmail.com, M04158002@mail2.mcut.edu.tw

ABSTRACT

This study is to provide a design of low-cost and high-availability data center infrastructure for small/medium-scale businesses. The basic concept is to establish an infrastructure consisting of primary and backup sides through the clustering technology. By default, the primary side is active and responsible for data switching while the backup side is standby for the failure of the primary side. The design achieves the features of 1) providing full-level high availability (HA) at network, server, application, and management levels, 2) controlling the routing among network-level clusters to solve the "PPPOE connection racing" and "winding path" problems, 3) monitoring and recovering the objects in each level with an economic and effective way, and 4) handling events resulting from changing of object states in an event center. Finally, the experiment results are exhibited with five testing scenarios for verification and elaboration of the effectiveness of the HA design. The system can recover the failed objects and solve the routing problems of PPPOE connection racing and winding path among HA clusters automatically.

Keywords: Cloud; Data Center; High Availability (HA); Virtualization Clustering; Failover; Load Balance.

1 Introduction

With the development of Information Technology (IT) in recent years, the concepts of Cloud [1, 2, 19] have been utilized in the data centers of enterprises. Cloud concepts emphasizing on a High Availability (HA) [3] of a system, are usually realized by virtualization and clustering technologies [4]. In a large-scale enterprise, it can hire a lot of IT professionals and purchase an expensive HA solution to secure its data. However, for small/medium-scale companies, where the number of employees below 200, they cannot provide the resources as those of a large-scale one. There are many small/medium-scale businesses facing an IT security challenge as same as a large-scale one. The difference between them is that the large-scale one has larger traffic volume than those of the small/medium-scale ones. Therefore, a low-cost and high-availability design of a data center for small/medium-scale business is very essential [5].

This paper proposes a design of an economic, effective, and high-availability solution for a data center. The basic concept is to establish an infrastructure consisting of primary and backup sides through the clustering technology. By default, the primary side is active and responsible for data switching while the backup side is standby for the failure of the primary side. This can be implemented with clustering technology. However, there are several clusters in a data center and the routing between them is not

DOI: 10.14738/tnc.55.3777
Publication Date: 26th October 2017
URL: http://dx.doi.org/10.14738/tnc.55.3777

considered in the current clustering technology. The primary and backup devices in a cluster determine whether the other one is alive by issuing "heartbeat packets" [6] to each other. The devices also determine that each connected network segment works properly by checking whether the individual monitoring point, created at each segment, is alive. If an active device detects one segment failed, it will become standby and notify its backup device to be active. However, it is a huge effort to arrange a monitoring point in each segment and maintain it always up and running. An unstable monitoring point will make the devices in a cluster changing between active and standby frequently. A simple approach to detect the states of a device in network segments is to check the states of the device's interfaces connected to the network segments. In fact, this is an effective way since a network segment failed will not be recovered even if a standby device becomes active and take over the control. Although the technologies of Spanning Tree Protocol or EtherChannel are the failure recovery solutions for a network segment between two Layer 3 devices, these two techniques are not being emphasized in the current design.

To solve the problems mentioned above and consider a full-level of High Availability (HA) on a data center, the proposed design applies clustering technology to introduce HA in the network and server levels individually and Load Balancing (LB) [7] in the application level. In addition, it utilizes a monitor server to monitor the network-level devices for controlling the routing among clusters. The monitor server recovers the objects in network, server, and application levels by issuing reboot or restart commands to the objects. The monitor server also acts as an event center and uses Syslog protocol to record the events of state changes or transitions. The event center provides a web page to indicate the states of objects at the individual levels and other pages to display the details of events describing the state changes. In order to achieve management-level HA, dual monitor servers are designed and deployed to support the resilience of event center functionalities.

The proposed design for the data center infrastructure has the features of 1) providing full-level HA at network, server, application, and management levels, 2) controlling the routing among network-level clusters, 3) monitoring and recovering the objects in each level with an economic and effective way, and 4) handling events resulting from changing of object states in an event center. It can achieve a high-availability and low-cost infrastructure since the HA functions and monitor mechanisms are designed in these four levels as mentioned previously and implemented by free, open source software, such as Linux Virtual Server (LVS) [8], Vyatta Router and Firewall [9], and NMap Scanner [10]. The "low-cost" is also included in the design that events are only generated on state changes. This saves the space and time to store and de-duplicate events. The rest of the paper is organized as follows: Section 2 presents the background knowledge and related work of the proposed design. Section 3 illustrates and presents the details of the data center infrastructure design and its implementation. Section 4 shows the experiment results for verifying and elaborating of the approaches. The final section concludes the paper and describes the future work.

2 Background and Related Work

2.1 Background knowledge

High Availability is a commonly seen cluster mechanism. The goal of cluster is usually to maintain services and allow them to operate at high stability. Database servers in corporate environments often implement this mechanism by setting two database servers as a high availability cluster. If one database server is damaged due to uncontrollable reasons, the other database server can automatically takeover in a short period of time and the user will not notice any service interruption.

Load Balancing is another cluster mechanism allowing for a large number of service requests. The Front-end application server in a corporate environment often plays a role by allocating multiple application servers to simultaneously service a lot of requests from the client side. This mechanism works by accepting user requests in the load balancer, then using load-balancing rules [11] to allocate requests to application servers for processing. The structure is as shown in Figure1, where the real server refers to the application server.

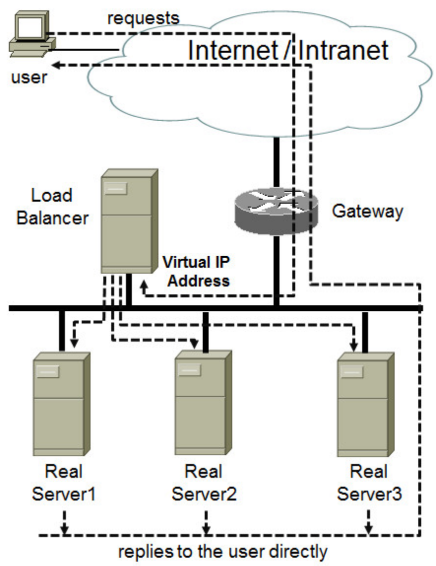


Figure1. Network and system architecture of Load Balancing

Heartbeat is as the word suggests. It is a mechanism responsible for monitoring cluster services. The heartbeat mechanism in a cluster node framework as shown in Figure2 will ping to detect if the opposing machine is alive. The heartbeat packet, e.g. one detection every 2 sec, is sent and if the opposing machine cannot be detected after a period of time, e.g. 30 sec, then the mechanism will determine operation failure for that cluster node and trigger a preset action such as taking over the services of the broken node. Heartbeat suite currently supports the following network signaling types:

- Unicast UDP over IPv4
- Broadcast UDP over IPv4
- Multicast UDP over IPv4
- Serial Link (Console Port)

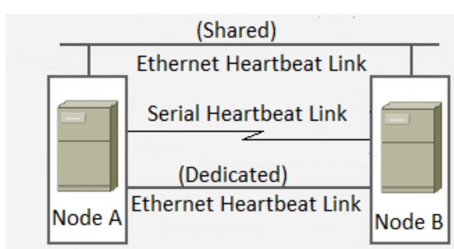


Figure2. Network architecture of Heartbeat mechanism

Failover mechanisms usually refer to the cluster services that are protected, such as server, operating system, network, application, etc. When an error or abnormality causes service interruption, the mechanism will automatically repair online services without human intervention. When using high availability heartbeat suite, the failover technology is achieved using "IP address takeover"; therefore, the active node of a cluster will have a virtual IP address to provide client-side services. When the client side sends a service request destined to the virtual IP address, the active node will reply the request. When problems occur in the active node, heartbeat functions will automatically toggle the service to standby node and give it the virtual IP address. Simultaneously, ARP packets are sent to local networks. The refreshed MAC address will perform smooth and error-free takeover of client-side related services.

Vyatta provides software-based virtual router, virtual firewall and VPN products for Internet Protocol networks (IPv4 and IPv6). A free download of Vyatta has been available since March 2006. The system is a specialized Debian-based Linux distribution with networking applications such as Quagga, OpenVPN, and many others. A standardized management console, similar to Juniper JUNOS or Cisco IOS, in addition to a web-based GUI and traditional Linux system commands, provides configuration of the system and applications. In recent versions of Vyatta, web-based management interface is supplied only in the subscription edition. However, all functionality is available through the connections of serial console or SSH/telnet protocols. The software runs on standard x86-64 servers.

2.2 Related work

In information technology, high availability refers to a system or component that is continuously operational for a desirably long length of time [12, 13]. In other words, the high availability is described through service level agreements and achieved through an architecture focusing on constant availability even in the face of failures at any level of the system [14]. In a data center, normally a redundant design at all levels ensures that no single component failure which impacts the overall system availability. While maintaining capability through load balancing, backup and replication, a mirrored facility, or a modular architecture replacing the monolithic one, it requires a significant investment and full monitoring [15].

On the other hand, in order to deploy high availabilities of a data center to protect its mission-critical services, the intelligent techniques need to be introduced or monitoring in the three main resource pools, i.e., computer, storage, and network, involving multi-layered approaches. The layered approach is the basic foundation of a data center design that seeks to improve scalability, performance, flexibility, resiliency and maintenance [16]. For example, in a Cisco approach, the layers of a data center design are the core, aggregation, and access layers. The core layer provides connectivity to multiple aggregation modules and provides a resilient Layer 3 (i.e., network layer) routing. The aggregation layer modules provide functions, such as firewall, load balancing to optimize and secure applications, intrusion detection, network analyses and more. Access layer provides both Layer 2 (data link layer) and Layer 3 topologies, fulfilling the various applications of servers. In addition, the primary data center design models are either multi-tier model or server cluster model for an enterprise or an academic and scientific community respectively. The models, at the aggregation and access layers, include "passive" redundancy built into data centers to overcome power or internet provider failures, as well as "active" redundancy that leverages sophisticated monitoring to detect issues and initiate failover procedures [14, 16]. Furthermore, additional log shipping can be adapted to achieve data storage protection [17].

As mentioned above, a lot of researches focus on the high availability and the relative performance in a single cluster. However, IBM Corp. published a patent [18] to solve the problem that the systems comprising cluster of identical servers are unable to provide a high availability processing environment with respect to web services for entire applications which are processed by such systems. In fact, such a system at best provides high availability for no more than localized portions of the application rather for the entire application. Usually, a web service system includes a web server cluster coupled to an application server cluster and the application server cluster coupled to a database server cluster. The system can function continuously while a local fault occurs in each cluster, but may not while a connection fails between two coupled clusters. IBM's solution is to add into the system a control server, which is linked to each server in each cluster with a connection path. Therefore, the control server can monitor the state of the link (or say, connection path) between the two servers respectively from two coupled clusters. If a link between two coupled clusters is down, the control server will inform the clusters to adjust their server status, either active or inactive for achieving high availability.

This paper also focuses on the high availability affected by the relationship between clusters. It further emphasizes the performance of routing path between clusters.

3 Economic Design of Full-level HA

The proposed design for a data center infrastructure has the features: full-level HA, system monitoring, recovering, and controlling, and event handling, which are presented in the following subsections.

3.1 HA model of data center infrastructure

The proposed HA design is based on a general model of data center infrastructure, as shown in Figure3, suitable for a small/medium-scale company (SMSC).

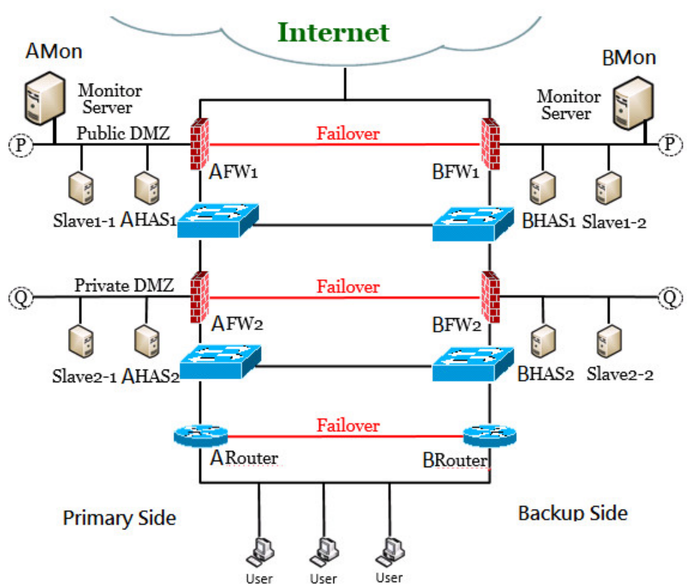


Figure3. HA model of data center infrastructure for SMSC

The design uses two tiers of firewalls to create the Public and Private DMZ network segments for accommodate the Internet servers and intranet servers, respectively. The Internet servers are utilized to settle the public-domain service applications such official Web, Email, FTP, etc. while the intranet servers to settle the private-domain ones such as HR, Finical, Asset systems, etc. In addition, one router tier is designed to create one or multiple internal network segments for the internal departments of company. The HA designs in network, server and application levels based on this model are shown as below.

3.1.1 Network-level HA

The network system in this model consists of the network devices of two-tier firewalls and one-tier routers. For network-level HA, each tier is grouped as a network cluster and divided into two sides: Primary and Backup. However, the public and private DMZ segments in Primary and Backup sides are connected in point Ᵽ and Ǭ, respectively. By default, the Primary-side firewalls and router are active in their network-level clusters while the Backup-side ones are standby. The internal user can issue a connection to the Internet through the active router (ARouter), then the active firewall 2 (AFW2), and finally the active firewall (AFW1). On the other side, the firewall BFW1 and BFW2 and the router BRouter are standby for backup. There is a failover network cable, between the active and standby devices of each tier, used as a passage for mutual monitoring. Once the active device fails and is unable to provide service, the virtual IP (VIP) address representing the cluster will be taken over by the backup device to realize uninterruptible network services. This is shown as Figure4. The network devices are implemented with general host computers running Vyatta system image. A Vyatta machine can act as a firewall or router and support HA functions.

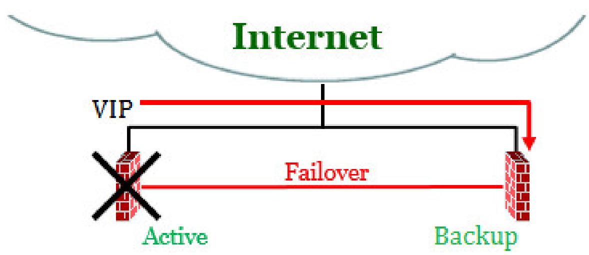


Figure 4. The association of VIP address changes on failover

There are two routing problems among the network-level clusters. First, many small/medium-scale companies use PPPOE connections in ADSL links to connect to the Internet. In the proposed model as shown in Figure3 , when firewalls AFW1 and BFW1, implemented as Vyatta machines, are in powered-on state and if their PPPOE connections are not controlled, both firewalls will fight for the PPPOE connections through ADSL and stall the network. In order to avoid this situation, a PPPOE connection coordination program, as shown in Figure5, must be set in AFW1 and BFW1 individually. The PPPOE program in the active firewall sets PPPOE commands to issue the PPPOE connection while the one in the standby firewall removes all PPPOE commands to close the PPPOE connection. The program thus avoids simultaneous PPPOE connections through ADSL. The PPPOE setting is never saved in configuration file. Thus, whenever the firewalls reboot, no PPPOE connection will be issued.

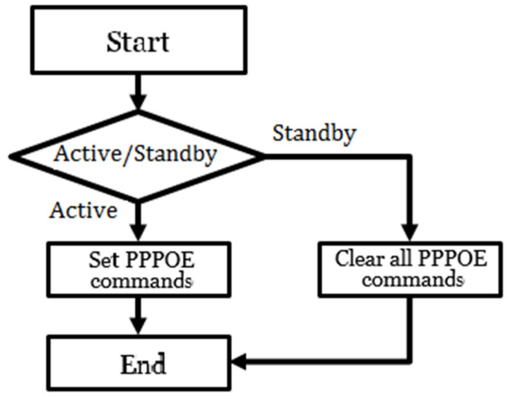


Figure 5. The PPPOE program in AFW1 and BFW1.

The second problem is that a winding path occurs while the routing among clusters is not controlled. Consider the scenario that AFW2 in a non-preempt HA reboots due to version upgrade. The BFW2 becomes active while AFW2 becomes standby. A winding path for an internal user's connection to the Internet occurs from ARouter, then through BFW2, and finally to AFW1. As shown in Figure3 , there are four switches in this path. If AFW2 finishes the reboot phase and works well, the better path is from ARouter, then through AFW2, and finally to AFW1 or from BRouter, then through BFW2, and finally to BFW1. There are only two switches in the individual paths. This routing can be adjusted by controlling the active/standby state of a network device. The monitor and control processes are designed in the monitor servers and describe in Section 3.2.

3.1.2 Server-level and application-level HA

In the server level, Linux Ubuntu is used to implement the server systems of the model. There are 8 Ubuntu servers created in the public DMZ and private DMZ network segments, named AHAS-1~2, BHAS-1~2, Slave1-1~2, and Slave2-1~2 (Figure3). In the public DMZ segment, AHAS-1 and BHAS-1 are grouped in the public server cluster and placed into the Primary and Backup sides, respectively. For server-level HA, AHAS-1 is active by default while BHAS-1 is standby for backup. This is implemented with Linux LVS and Keepalived utilities, which provide the Heartbeat mechanism to determine if the other side server is alive. Once the active server fails and is unable to provide service, the virtual IP (VIP) address representing the cluster will be taken over by the backup server. In the private DMZ segment, AHAS-2 and BHAS-1 is grouped in the private server cluster, whose HA operates as same as that of the public server cluster.

For application-level HA, an application service is distributed into the servers of DMZ segment, which dynamically share the service requests and thus achieve the HA of application service with the way of service Load Balancing (LB). In the public DMZ segment, by default, AHAS-1 acts as an active Master server while BHAS-1 acts as a standby Master, but they do not offer any application service. The active Master plays a role of load balancer to dispatch service requests to the Slave servers actually hosting the application service. Slave1-1 and Slave1-2 are the Slave servers in the public DMZ segment. The design of the LB adopts the Direct Routing (Figure1) architecture for reducing the traffic load of the Master servers. It is implemented with Linux LVS and Keepalived utilities, which are installed in the Master servers to provide the health checks of application services in the Slave servers and thus know how to dispatch service requests. For the private DMZ segment, the LB operation is implemented with the same way as that in the public DMZ.

3.2 Monitoring, Recovering, and Controlling HA

In the proposed design, the primary and backup devices in a cluster determine if the other one is alive by the Heartbeat mechanism. However, for network-level HA, the network devices additionally determine if each connected network segment works well by checking if the individual monitoring point, created in each segment, is alive. However, it is a large effort to arrange a monitoring point in each segment and maintain it always up. Instead, the design detects the states of network segments of a device by checking the states of its interfaces connected to network segments. A monitor cluster is created in the Public DMZ and consists of two Monitor servers, in the Primary and Backup sides, named AMon and BMon, respectively. By default, AMon is active to monitor the objects in network, server, and application levels and then recover them, if failures (also including HA failures) are detected, through reboot or restart commands in the Vyatta or Ubuntu systems. Simultaneously, BMon is standby for backup of AMon to achieve the management-level HA. The active Monitor server uses the Ping utility to monitor the objects in network and server levels while using the NMap utility to monitor the services in application level.

After the monitoring and recovering phases, the active Monitor server starts the HA control process. First, it checks the route to the Internet to prevent AFW1 and BFW1 from issuing the PPPOE connections simultaneously. The control flow is as shown in Figure6. Initially, it notifies the two firewalls to clear their PPPOE setting and then checks which firewall is active in HA. It notifies the active firewall to run the PPPOE program, as shown in Figure5, to issue the PPPOE connection. Once the connection establishes, the IP address is allocated to the PPPOE interface of active firewall. Then, the control process Pings the IP address to determine if the route to the Internet is successful. If the Ping detection fails, the process tries to reboot the firewall failing on PPPOE.

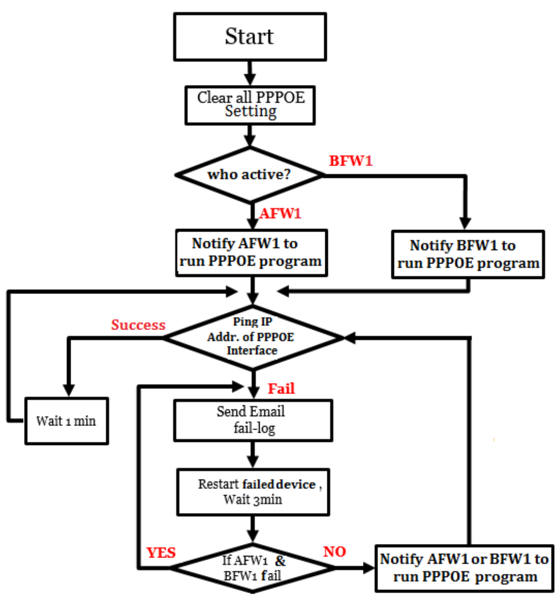


Figure 6. The control of PPPOE by Monitor server.

Secondly, the active Monitor server controls the routing among the network-level clusters to avoid a winding path through the clusters. The way is to control all active network devices in one side, either the

Primary or Backup side (Figure3), and all standby ones in the other side. The control process is called One-side-active process, as shown in Figure7. It tries to control the firewall successfully issuing PPPOE connection, either AFW1 or BFW1, together with other active network devices in the same side. Consider the following scenario. Assume that AFW1 has successfully established a PPPOE connection to the Internet. Obviously, AFW1 is active in its cluster. If AFW2 is standby in its cluster, the One-side-active process will reboot BFW2 to make AFW2 active and BFW2 thus standby.

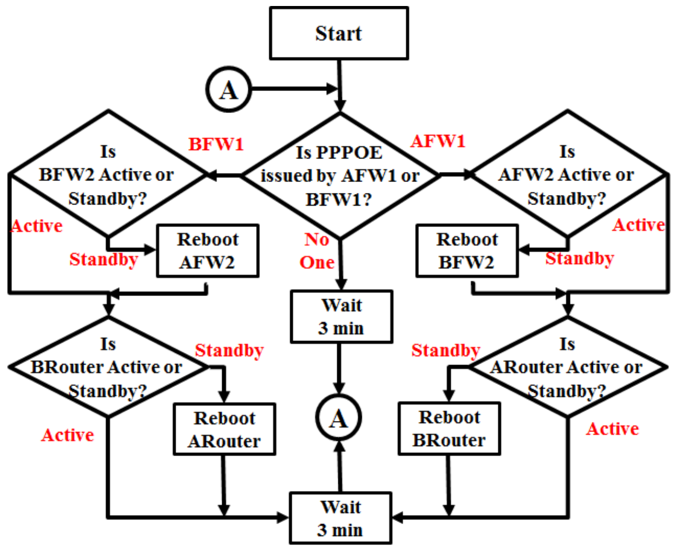


Figure 7. The control of One-side-active by Monitor server.

3.3 Event Handling

In the HA model of data center infrastructure, in order to coordinate the operations among the HA clusters of all levels, the concept of Event Center (EC) is introduced into and implemented in the Monitor server. As an EC, the Monitor server monitors and controls the whole data center and records and analyzes events to alert the administrator while a critical event occurs. Its system architecture is as shown in Figure8. The Monitor & Controller module undertakes the monitoring, recovering, and controlling of the objects in all levels for achieving full HA. The operation states of each object can be simply defined as Normal and Abnormal. An event from an object is defined as the state change of the object.

The monitor server is implemented with Linux Ubuntu system and utilizes the mechanisms of Syslog, Database, and Rotated Files to systematically record events. Whenever the monitor and control module detects a state change of an object, it creates an event to describe the state change. The event is sent to the local Syslog server with the Linux Logger utility or encapsulated as an email and then sent to a mail server. When the Syslog server receives the event, it can transfer the event to a remote Syslog server by the local Syslog client, save into the Rotated Files by the Logrotate utility, or store into a Database such as PostgreSQL. The local Syslog server and client module is implemented with the Rsyslog utility. The design uses state change to trigger the creation of an event and thus reduces the quantity of events. This is a low-cost design, compared to an event design which continuously generate the same-reason events from periodical detection of an Abnormal object until the object becomes Normal.

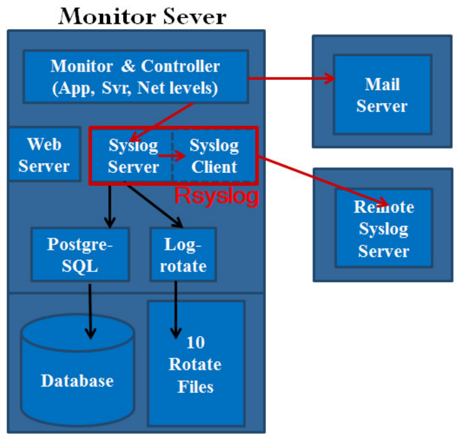


Figure 8. System architecture of Monitor Server.

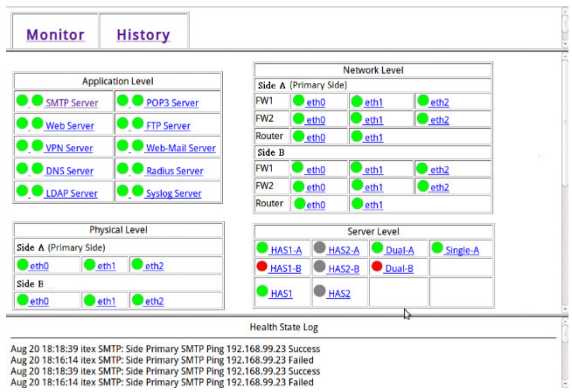


Figure 9. Web page shows the states of monitored objects.

The states of objects in all level are exhibited with the signs drawn in a web page, as shown in Figure9. The Normal and Abnormal states are presented as Green and Red signs, respectively. The logs record the stage changes of objects in different levels are exhibited in individual web pages, as shown in Figure10. These pages are formed by a web server as shown in Figure8, which is implemented with the Tomcat utility.

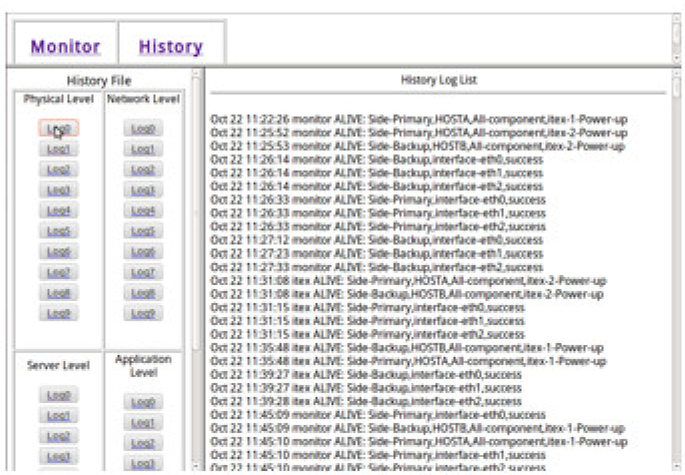


Figure10. Web page shows the historical events in each level.

4 Experiment results

In order to verify the proposed design works well, the data center infrastructure is implemented mainly with the Linux Ubuntu and Vyatta system software. The cluster HA functions in the two operating systems are used to provide individual-level cluster HA. The Monitor & Controller module designed in the Monitor server is to achieve a full HA by monitoring, recovering, and controlling the objects of all levels. It solves the routing problems of "PPPOE connection racing" and "winding path" among the network-level clusters. The experiment results exhibited in the following subsections explain how to verify the approaches.

4.1 Monitoring and recovering test

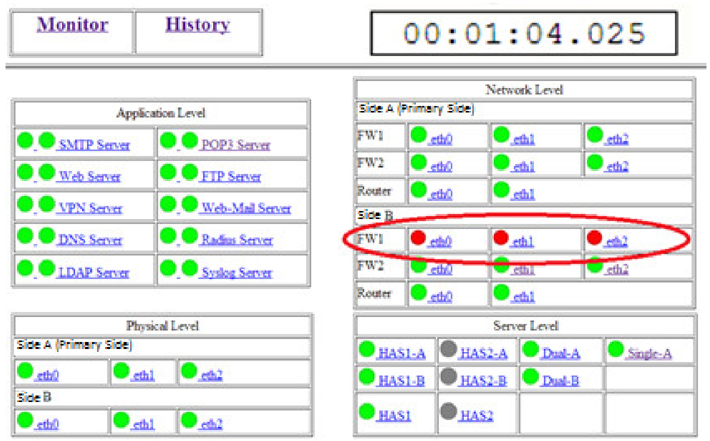


Figure 11. BFW1 is detected to be failed.

The testing scenario is as follows: The active side is the Primary side; that is, all the network devices in the Primary side are active while all the ones in the Backup side are standby. Close the standby firewall BFW1. The test is to check if the failure of BFW1 will be detected and then BFW1 will be recovered.

The time is reset to zero when BFW1 is closed. As shown in Figure11, BFW1 is detected to be failed at 1 min 4.025 sec. Then, the monitor server starts to recover BFW1 through rebooting. BFW1 is detected to be successful at 2 min 45.606 sec (Figure12).

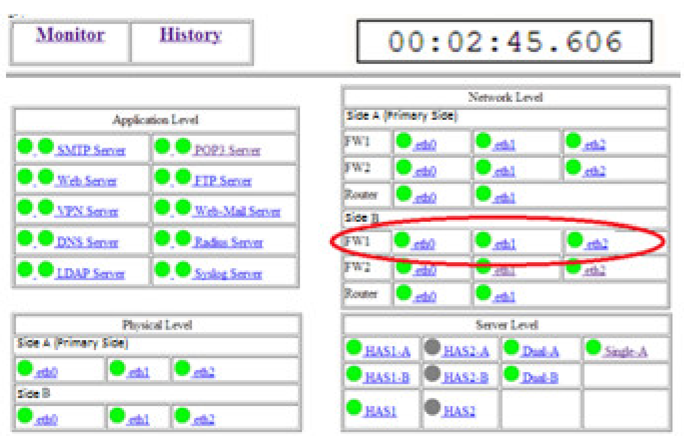


Figure12. BFW1 is recovered and detected to be successful

4.2 PPPOE connection test

The testing scenario is as follows: The active side is the Primary side. Close AFW1, the active firewall in the first tier. The test is to check if the Monitor server can detect the failure of AFW1, then recover it, and next notify BFW1 to establish the PPPOE connection since BFW1 has become active.

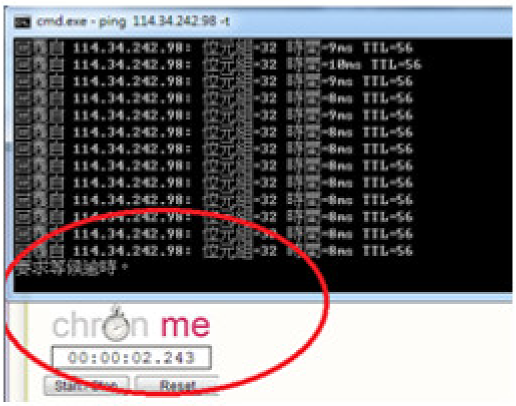


Figure13. The route to the Internet is failed

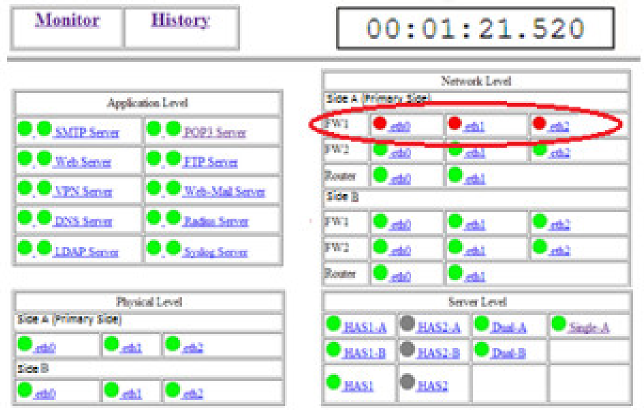


Figure14. AFW1 is detected to be failed.

The time is reset to zero when AFW1 is closed. As shown in Figure13, the PPPOE connection is failed at 2.243 sec. However, the Monitor server detects the failure of AFW1 after 1 min 21.520 sec (Figure14). It then recovers AFW1 through rebooting. AFW1 is detected to be successful after 2 min 26.020 sec (Figure15). Finally, the PPPOE connection is recovered at 3 min 4.009 sec (Figure16).

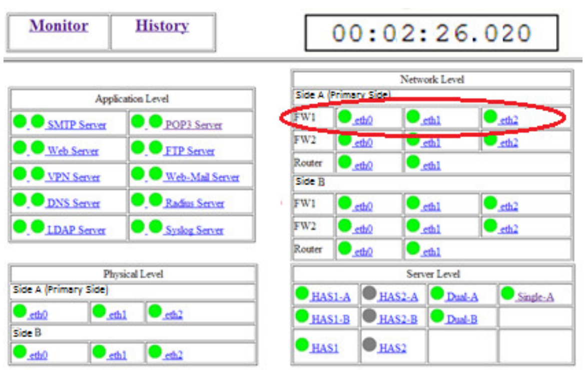


Figure15. AFW1 is recovered and detected to be successful

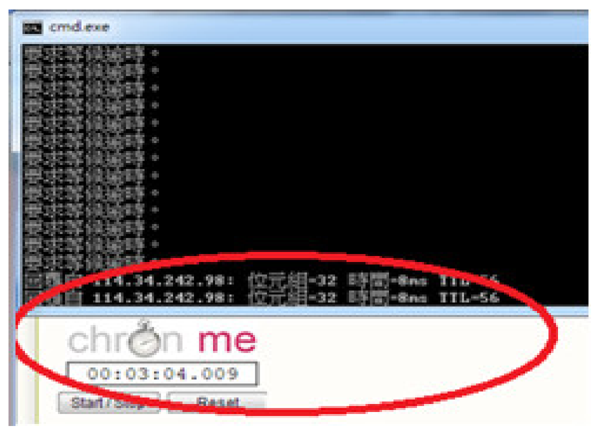


Figure16. The route to the Internet is recovered

4.3 One-side-active process test

This test scenario continues that of PPPOE connection test. When the PPPOE connection is recovered as shown in Figure16, One-side-active process will enforce BFW2 and BRouter to be active since BFW1 is active (Figure17), and issues the PPPOE connection.

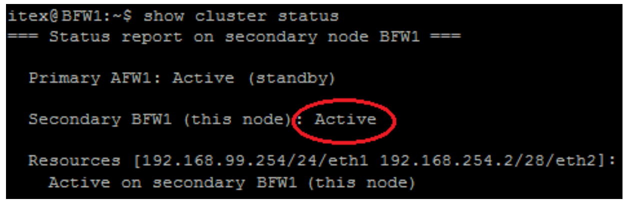


Figure 17. BFW1 is active while AFW1 is standby

Following the timing of PPPOE connection test, in which the time is reset to zero when AFW1 is closed, the scenario shows AFW2 and ARouter are rebooted and their failures are detected at 4 min 47.361 sec (Figure18). Then, they are detected to be successful at 5 min 58.760 sec (Figure19).

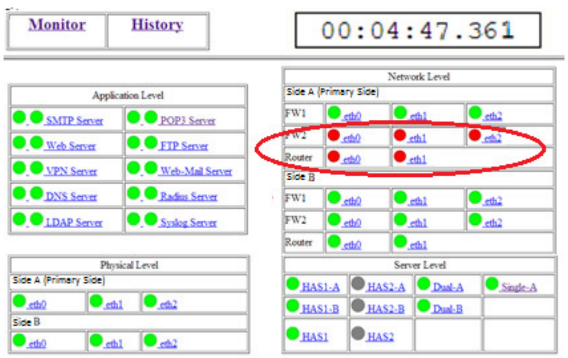


Figure18.One-side-active process reboots AFW2 and ARouter.

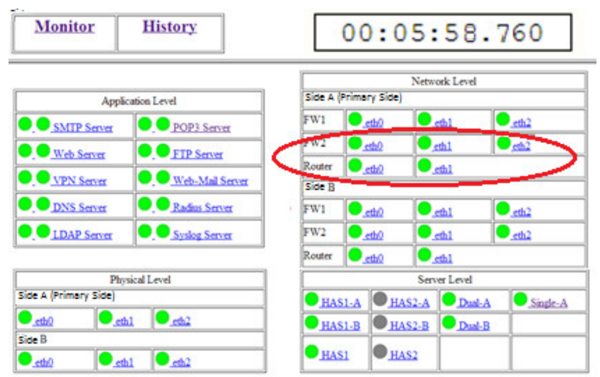


Figure19. AFW2 and ARouter reboot successfully.

4.4 Data continuity test

Vyatta devices can support stateful failover function to prevent a session from being broken while it encounters a HA failover. The proposed design enabling this function is to protect sessions from loss in the regulation enforced by One-side-active process.

The testing scenario is as follows: A user in the LAN segment between ARouter and BRouter gets data from the Ubuntu FTP site. While data is being transferred, ARouter is closed since it is active in its cluster. Due to stateful failover, BRouter undertakes the rest data transfer of the FTP connection. As shown in Figure 20 and 21, data continuity works well.

Figure 20. Data transfer is paused during failover.

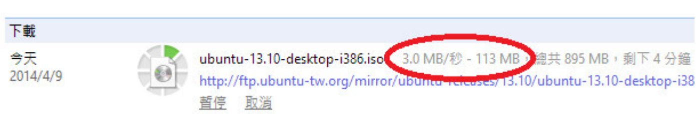


Figure 21. Data transfer is continued after failover.

4.5 Load Balancing test

Application-level HA is achieved with the Load Balancing (LB) mechanism applied on the servers hosting application service. In the testing scenario, an IE browser is used to issue connections to the VIP address, 192.168.99.20, of a web server cluster having two slave servers Slave1-1~2 in the public DMZ segment. Each connection is dispatched to one of the slave servers in a LB scheduling. The testing web pages are as shown in Figure22. If server Server1-2 is off, the connection dispatching will skip it and always select Slave1-1, as shown in Figure23.

Figure 22. Connections are dispatched to two slave servers

Figure 23. Connection dispatching skips an "off" server

5 Conclusions

The motivation is to provide a design of low-cost and high-availability data center infrastructure for small/medium-scale businesses. The proposed design has the features as follows: 1) providing full-level HA on network, server, application, and management levels, 2) controlling the routing among network-level clusters to solve the "PPPOE connection racing" and "winding path" problems, 3) monitoring and recovering the objects in each level with an economic and effective way, and 4) handling events resulting from object state changes with an event center. The design is low-cost since it is implemented with free software, such as Linux Virtual Server (LVS), Vyatta Router and Firewall, and NMap Scanner. In addition, the event center generates events only on state changes. This saves the space and time to store and de-duplicate events, caused by the same-reason source. The experiment results show the administrator can see the current and historical state changes of the objects in the data center infrastructure through the Event Center in the Monitor server. Its Monitor & Controller module can recover the failed objects and solve the routing problems of PPPOE connection racing and winding path among HA clusters automatically. The results also exhibits that the data continuity realized with stateful failover is still workable with the proposed design, especially at its One-side-active process.

The future work is to analyze and enhance the performance of the Monitor & Control module. This includes: 1) what is the proper lower and upper bounds of the monitoring interval to the objects of all levels, 2) how to shorten the time for silencing the rebooted or restarted objects for recovery, 3) finding better method of recovery instead of reboot.

REFERENCES

[1] Cegielski, C. G., Bourrie, D. M. & Hazen, B. T., Evaluating Adoption of Emerging IT for Corporate IT Strategy: Developing a Model Using a Qualitative Method. Information Systems Management, 2013. 30 (3): p. 235–249.

[2] Duan, Q., Yan, Y. & Vasilakos, A. V., A Survey on Service-Oriented Network Virtualization toward Convergence of Networking and Cloud Computing. IEEE Transactions on Network and Service Management, 2012. 9 (4): p. 373–392.

[3] Araujo, J. A., Lazaro, J., Astarloa, A., Zuloaga, A. & Garcia, A., High Availability Automation Networks: PRP and HSR ring implementations. IEEE International Symposium on Industrial Electronics (ISIE) 2012: p. 1197–1202.

[4] Sharkh, M. A., Jammal, M., Shami, A., & Ouda, A., Resource Allocation in a Network-Based Cloud Computing Environment: Design Challenges. IEEE Communications Magazine, 2013. 51 (11): p. 46–52.

[5] Bitar, N., Gringeri, S. & Xia, T. J., Technologies and protocols for data center and cloud networking. IEEE Communications Magazine, 2013. 51 (9): p. 24–31.

[6] Liao, C. F., Chang, H. C. & Fu, L. C., Message-Efficient Service Management Schemes for MOM-Based UPnP Networks. IEEE Transactions on Services Computing, 2013. 6 (2): p. 214–226.

[7] Lin, C. C., Chin, H. H. & Deng, D. J., Dynamic Multiservice Load Balancing in Cloud-Based Multimedia System. IEEE Systems Journal, 2014. 8 (1): p. 225–234.

[8] LVS. http://www.linuxvirtualserver.org as of October 11, 2015.

[9] Vyatta. Vyatta Community Documentation. http://www.vyatta.org/ as of August 15, 2014.

[10] NMap. http://nmap.org/ as of October 11, 2015.

[11] Membrey, P., Plugge, E. & Hows, D., Practical Load Balancing: Ride the Performance Tiger, 2012. 1st Ed. Apress.

[12] Rusu, L. & Smeu, A., Managing the Reliable Design of an Enterprise IT Network Infrastructure. Information Systems Management, 2010. 27 (3): p. 238–246.

[13] Radhakrishnan, R., Mark, K., & Powell, B., IT Service Management for High Availability. IBM Systems Journal, 2008. 47 (4): p. 549–561.

[14] Mahmood, A. & Rashid, I., Comparison of load balancing algorithms for clustered web servers. International Conference on Information Technology and Multimedia (ICIM) 2011: p. 1–6.

[15] IT Today. Is High-Performance Computing For You? IBM Systems Magazine, Power Systems, January. 2013.

[16] Cisco. Cisco Data Center Infrastructure 2.5 Design Guide. Cisco Validated Design. 2013.

[17] HP. Designing Disaster Tolerant High Availability Clusters. Manufacturing Part Number: B7660-90006. HP Document. 2014.

[18] Ahmed, I., Auvenshine, J. J., & Blackburn, J., System for Autonomic Monitoring for Web High Availability. IBM Corp. US 7996529 B2; Patent. August 9, 2011.

[19] Sim, K. M., Agent-based Approaches for Intelligent InterCloud Resource Allocation, IEEE Transactions on Cloud Computing, 2016: p. 1-14.

Society for Science and Education
United Kingdom

TNC Transactions on Networks and Communications

Volume 5, No. 5
ISSN: 2054 -7420

Searching Isomorphic Graphs

Anatoly D. Plotnikov
East Ukrainian National University,
Severodonetsk, Ukraine;
a.plotnikov@list.ru

ABSTRACT

To determine that two given undirected graphs are isomorphic, we construct for them auxiliary graphs, using the breadth-first search. This makes capability to position vertices in each digraph with respect to each other. If the given graphs are isomorphic, in each of them we can find such positionally equivalent auxiliary digraphs that have the same mutual positioning of vertices. Obviously, if the given graphs are isomorphic, then such equivalent digraphs exist. Proceeding from the arrangement of vertices in one of the digraphs, we try to determine the corresponding vertices in another digraph. As a result we develop the algorithm for constructing a bijective mapping between vertices of the given graphs if they are isomorphic. The running time of the algorithm equal to $O(n^5)$, where *n* is the number of graph vertices.

Keywords: graph, isomorphism, bijective mapping, isomorphic graphs, algorithm, graph isomorphism problem.

1 Introduction

Let L_n is the set of all *n*-vertex undirected graphs without loops and multiple edges.

Let, further, there is a graph $G = (V_G, E_G) \in L_n$, where $V_G = \{v_1, v_2, \dots, v_n\}$ is the set of graph vertices and $E_G = \{e_1, e_2, \dots, e_m\}$ is the set of graph edges. Local degree $\deg(v)$ of the vertex $v \in V_G$ is the number of edges, that is incident to the vertex v. Every graph $G \in L_n$ can be characterized by the vector $D_G = (\deg(v_{i1}), \deg(v_{i2}), \dots, \deg(v_{in})$ of the local vertex degrees, where $\deg(v_i) \leq \deg(v_{j)})$ if $i < j$.

Graphs $G = (V_G, E_G)$, $H = (V_H, E_H) \in L_n$ are called isomorphic if between their vertices there exists one-to-one (bijective) mapping $\varphi: V_G \leftrightarrow V_H$ such that if $e_G = \{v, u\} \in E_G$ then the corresponding edge is $e_H = \{\varphi(v), \varphi(u)\} \in E_H$, and conversely [1, 2]. We say that the mapping φ converts the graph G into the graph H and conversely.

The problem of determining the isomorphism of two given undirected graphs is used to solve chemical problems, and to optimize programs [3] - [6] and others. Effective (polynomial-time) algorithms for solving this problem were found for some narrow classes of graphs [7] - [9]. However for the general case, effective methods for determining the isomorphism of graphs are not known [10].

The purpose of this article is to propose a polynomial-time algorithm of searching isomorphic graphs.

DOI: 10.14738/tnc.55.3551
Publication Date: 26th October 2017
URL: http://dx.doi.org/10.14738/tnc.55.3551

2 Basic Definitions

Let there be a graph $G \in L_n$.

Choose some vertex $v \in V_G$. The set of all vertices of the graph G, adjacent to vertex v, we call the neighborhood of the 1st level of this vertex.

Suppose that we have constructed the neighborhood of $(k-1)$-th level of the vertex v. Then the set of all vertices, adjacent to at least one vertex $(k-1)$-th level, we call the neighborhood k-th level $0 \le k \le n-1$ of the vertex v. Such neighborhood we denote $N_G^k(v)$. For convenience, we assume that the vertex v forms the neighborhood of the zero level.

Found neighborhoods allow us to construct for the vertex v of the auxiliary directed graph $\vec{G}(v)$ in the following way.

Each neighborhood $N_G^k(v)$ of the graph G forms k-th line of the digraph. If the edge of the graph G connects the vertex v_1 of k_1-th level with the vertex v_2 of k_2 level, and $k_1 < k_2$, then this edge is replaced by the arc (v_1, v_2). If $k_1 = k_2$, then this edge is replaced by two arcs (v_1, v_2) and (v_2, v_1). We say that the constructed digraph is induced by the vertex v of the graph G.

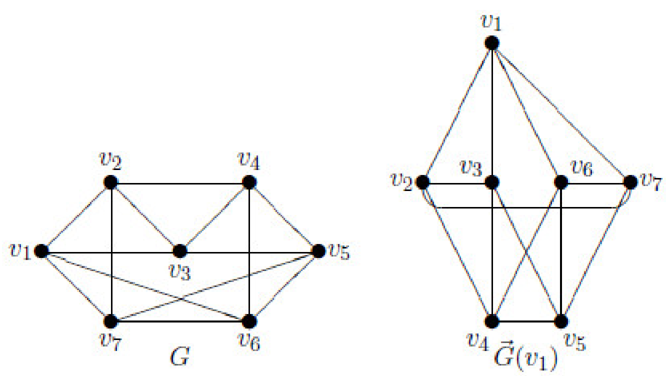


Figure. 1. The graph G and the auxiliary digraph $\vec{G}(v_1)$.

Figure 1 shows the graph G and the auxiliary digraph $\vec{G}(v_1)$, induced by the vertex v_1 (Here and below, the arcs of the digraph are directed from the top to the bottom, and the horizontal lines represent two mutually directed arcs).

Each vertex of the auxiliary graph will be characterized by two vectors.

The input characteristic of the vertex v of the digraph $\vec{G}(v)$ is called the vector $I(v) = (i_1, \dots, i_q)$, where the elements $i_1, \dots, i_q$ are the line numbers of the digraph, written in order of increase. These numbers indicate the lines, from which arcs come into the vertex v. If into the vertex v several arcs come from the same line, the line number is recorded in the vector $I(v)$ the corresponding number of times.

The output characteristic of a vertex v of the digraph $\vec{G}(v)$ is called the vector $O(v) = (o_1, \dots, o_q)$, where the elements $o_1, \dots, o_q$ are the line numbers of the digraph, also written in order of increase. These numbers indicate those lines, in which arcs come from vertex v. If from the vertex v several arcs come into vertices of the same line, then the line number are recorded in the vector $O(v)$ the corresponding number of times.

Characteristic of vertex v of the digraph $\vec{G}(v)$ will be called the input and output characteristics of this vertex. The characteristics of the two vertices v_1, v_2 are equal if them the input and output characteristics are equal, respectively.

Find the vertex characteristics of the digraph $\vec{G}(v_1)$, is shown in Figure 1. The results are presented in the Table 1.

Table 1. The vertex characteristics of the digraph $\vec{G}(v_1)$.

$I(v_1) = \varnothing$; $O(v_1) = (1,1,1,1)$;	$I(v_2) = (0,1,1)$; $O(v_2) = (1,1,2)$;	$I(v_3) = (0,1)$; $O(v_3) = (1,2,2)$;	$I(v_4) = (1,1,1,2)$; $O(v_4) = (2)$;
$I(v_5) = (1,1,1,2)$; $O(v_5) = (2)$;	$I(v_6) = (0,1)$; $O(v_6) = (1,2,2)$;	$I(v_7) = (0,1,1)$; $O(v_7) = (1,1,2)$.	

Let there are auxiliary directed graphs $\vec{G}(v_1)$ and $\vec{G}(v_2)$, induced by the vertices v_1 and v_2, possibly belonging to different graphs. The directed graphs $\vec{G}(v_1)$ and $\vec{G}(v_2)$ are called positionally equivalent if the lines of digraphs of the same level contain the same number of vertices having respectively equal (input and output) characteristics.

A vertex $x \in V_G$ will be called unique if the digraph $\vec{G}(v)$ does not exist another vertex with characteristics equal to the characteristics of the vertex x.

Note that the vertex $v \in V_G$, that induces the auxiliary digraph $\vec{G}(v)$, is always unique in this digraph.

3 The Basics of the Algorithm

Next, we will consider pairs of graphs $G, H \in L_n$, having equal number of vertices n, equal number of edges m and equal vectors of the local degrees $D_G = D_H$. It needs to determine the isomorphism of the given graphs and, if they are isomorphic, then find the bijective mapping φ between their vertices.

The idea of finding bijective mapping φ between the vertices of the vertex set of graphs $G, H \in L_n$ is the following. We assume that the graphs G and H are isomorphic. Naturally that the required mapping can only be found when graphs G and H is indeed isomorphic. If in the process of finding the mapping φ cannot be found, then the given graphs are not isomorphic.

Theorem 1. *Let the graphs G and H are isomorphic. Then there exist at least two vertices $v \in V_G$ and $u \in V_H$ such that induce two auxiliary positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$.*

Proof. The construction of the auxiliary directed graphs depends only on the location of graph vertices relative to the neighborhood of the induced vertex v or u and does not depend on the vertex names of the graphs G and H. Therefore, because graphs G and H are isomorphic and we have the identical procedure of constructing the auxiliary digraphs, we will be found the auxiliary positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$.■

Theorem 2. If graphs G and H are isomorphic and two auxiliary positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}$(u) are found, then any bijective mapping φ, which convert the graph G into the graph H (and conversely), is determined by pairs of vertices of the digraphs with equal characteristics.

Proof. The assertion of the Theorem 2 is true as if to assume contrary, we will get that the mapping φ converts one vertex to another with different characteristics. This is contrary to the concept of isomorphism of graphs.■

Corollary 1. Let the graphs G and H are isomorphic and two auxiliary positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$ are constructed. Let, further, it was found t unique vertices in these digraphs, the corresponding pairs of which have equal characteristics. Then, the pairs of these vertices belong to the mapping φ.

It is easy to understand that the search of vertex pairs that belong to the binary mapping φ, the equality of the vertex characteristics in the auxiliary digraphs are not sufficient if these vertices have incoming and/or outgoing arcs, connecting vertices of the same line of the auxiliary digraph.

In the isomorphic graphs G and H, we can find vertices $v \in V_G$ and $u \in V_H$ such that induce auxiliary positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$ respectively. In accordance with the Theorem 1, these vertices always exist.

The desired bijective mapping φ can be represented as the perfect matching in the bipartite graph, induced by the vertices of the auxiliary positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$. This bipartite graph we will call the virtual. In this bipartite graph, any vertex $x \in V_G$ is connected by the virtual edge with all vertices of the digraph $\vec{H}(u)$, which have same characteristics as x. It is clear that the bijective mapping will correspond to virtual perfect matching in the bipartite graph. Unfortunately, it is not every perfect matching corresponds to the bijective mapping φ.

Figure 2 shows the graph H and the auxiliary digraph $\vec{H}(u_1)$, induced by the vertex u_1.

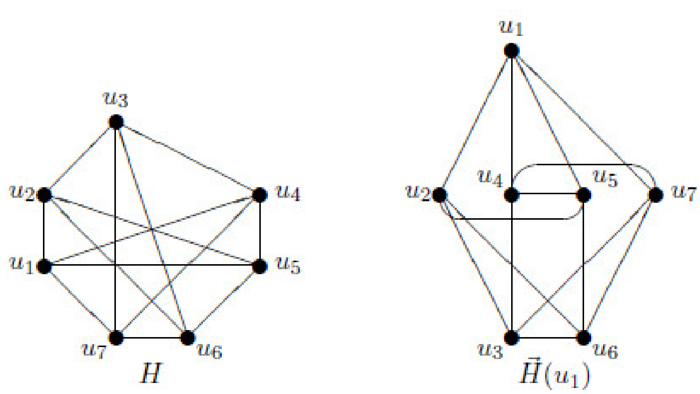


Figure. 2. The graph H and the auxiliary digraph $\vec{H}(u_1)$.

We find the vertex characteristics of the auxiliary digraph $\vec{H}(u_1)$. The results shown in the Table 2.

Table 2. The vertex characteristics of the digraph $\vec{H}(u_1)$.

$I(u_1)=\varnothing$; $O(u_1)=(1,1,1,1)$;	$I(u_2)=(0,1)$; $O(u_2)=(1,2,2)$;	$I(u_3)=(1,1,1,2)$; $O(u_3)=(2)$;	$I(u_4)=(0,1,1)$; $O(u_4)=(1,1,2)$;
$I(u_5)=(0,1,1)$; $O(u_5)=(1,1,2)$;	$I(u_6)=(1,1,1,2)$; $O(u_6)=(2)$;	$I(u_7)=(0,1)$; $O(u_7)=(1,2,2)$.	

It is easy to see that the auxiliary digraphs $\vec{G}(v_1)$ (see Figure 1) and $\vec{H}(u_1)$ are positionally equivalent. In these digraphs, there are only two unique vertices v_1 and u_1. Other vertices with equal characteristics form the virtual bipartite graphs, induced by the following vertex pairs: $\{v_3,v_6\},\{u_2,u_7\}$; $\{v_4,v_5\},\{u_3,u_6\}$ and $\{v_2,v_7\},\{u_4,u_5\}$.

Figure 3 shows the virtual bipartite graph for the auxiliary positionally equivalent digraphs $\vec{G}(v_1)$ and $\vec{H}(u_1)$.

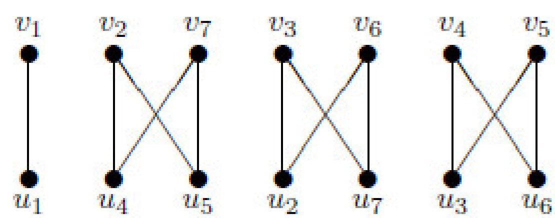


Figure. 3. The virtual bipartite graph.

In this case, we have one unique vertex in each of the digraphs $\vec{G}(v_1)$ and $\vec{H}(u_1)$. Therefore, the pair of vertices $\{v_1,u_1\}$ belongs to the desired mapping φ. Remove these vertices from the corresponding graphs. We will get graphs G_1 and H_1.

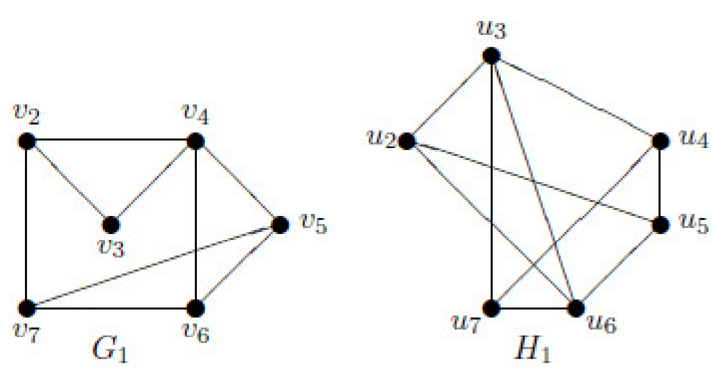


Figure. 4. The graphs G_1 and H_1.

Figure 4 shows the graphs G_1 and H_1.

Thus, we again have the problem of finding the bijective mapping $\varphi_1 \subset \varphi$, which converts the graph G_1 the graph H_1 and conversely.

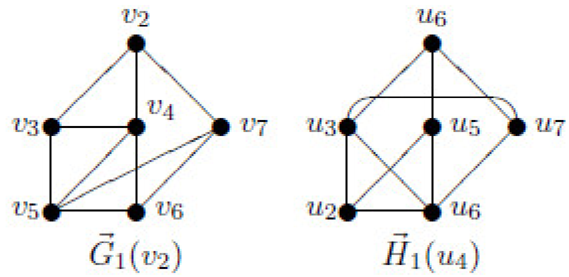


Figure 5. The digraphs $\vec{G}_1(v_2)$ and $\vec{H}_1(u_4)$.

In the graph G_1 we choose the vertex v_2, and construct the corresponding auxiliary digraph (see Figure 5). We find the vertex characteristics of the digraph $\vec{G}_1(v_2)$. The result of the calculation is placed in the Table 3.

Table 3. The vertex characteristics of the digraph $\vec{G}_1(v_2)$.

$I(v_2) = \varnothing$; $O(v_2) = (1,1,1)$;	$I(v_3) = (0,1)$; $O(v_3) = (1,2)$;	$I(v_4) = (0,1)$; $O(v_4) = (1,2,2)$;
$I(v_5) = (1,1,1,2)$; $O(v_5) = (2)$;	$I(v_6) = (1,1,2)$; $O(v_6) = (2)$;	$I(v_7) = (0)$; $O(v_7) = (2,2)$.

In the graph of H_1 we choose the vertex u_4, and construct the corresponding auxiliary graph (see Figure 5). Find the vertex characteristics of the digraph $\vec{H}_1(u_4)$. The result of the calculation is placed in the Table 4.

Table 4. The vertex characteristics of the digraph $\vec{H}_1(u_4)$.

$I(u_4) = \varnothing$; $O(u_4) = (1,1,1)$;	$I(u_3) = (0,1)$; $O(u_3) = (1,2,2)$;	$I(u_5) = (0)$; $O(u_5) = (2,2)$;
$I(u_7) = (0,1)$; $O(u_7) = (1,2)$;	$I(u_2) = (1,1,2)$; $O(u_2) = (2)$;	$I(u_6) = (1,1,1,2)$; $O(u_6) = (2)$.

Comparing the vertex characteristics of auxiliary digraphs, we see that the digraphs $\vec{G}_1(v_2)$ and $\vec{H}_1(u_4)$ is the positional equivalent. Moreover, in each of the constructed digraph, each vertex is unique.

At once, we can construct the mapping φ_1, choosing vertex pairs of the digraphs $\vec{G}_1(v_2)$ and $\vec{H}_1(u_4)$ with equal characteristics. Adding to φ_1 of the previously found pair $\{v_1, u_1\}$, we get the required mapping

$$\varphi = \{\{v_1, u_1\}, \{v_2, u_4\}, \{v_3, u_7\}, \{v_4, u_3\}, \{v_5, u_6\}, \{v_6, u_2\}, \{v_7, u_5\}\}.$$

To perform the verification of the obtained result, consider the matching edges of graphs G and H defined found the bijective mapping.

$$\begin{array}{ll} \{v_1, v_2\} \leftrightarrow \{u_1, u_4\} & \{v_3, v_4\} \leftrightarrow \{u_7, u_3\} \\ \{v_1, v_3\} \leftrightarrow \{u_1, u_7\} & \{v_3, v_5\} \leftrightarrow \{u_7, u_6\} \\ \{v_1, v_6\} \leftrightarrow \{u_1, u_2\} & \{v_4, v_5\} \leftrightarrow \{u_3, u_6\} \\ \{v_1, v_7\} \leftrightarrow \{u_1, u_5\} & \{v_4, v_6\} \leftrightarrow \{u_3, u_2\} \\ \{v_2, v_3\} \leftrightarrow \{u_4, u_7\} & \{v_5, v_6\} \leftrightarrow \{u_6, u_2\} \\ \{v_2, v_4\} \leftrightarrow \{u_4, u_3\} & \{v_5, v_7\} \leftrightarrow \{u_6, u_5\} \\ \{v_2, v_7\} \leftrightarrow \{u_4, u_5\} & \{v_6, v_7\} \leftrightarrow \{u_2, u_5\} \end{array}$$

All matches are correct.

4 Vertex Characteristics of the Digraphs

The example above illustrates our approach to solving problem of finding the bijective mapping φ between the vertices in isomorphic graphs G and H.

The essence of this approach consists in the following steps.

- In the graph G, choose the vertex v.
- Construct the auxiliary digraph $\vec{G}(v)$.
- In the graph H, choose the vertex u such that the auxiliary digraph $\vec{H}(u)$, which is positionally equivalent to the digraph $\vec{G}(v)$. If such vertex is not found, then terminate computation as graphs G and H are not isomorphic.
- In the positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$, find all unique vertices. Form vertex pairs (v, u), having equal characteristics, and record them. Delete all unique vertices from the graphs G and H.
- If the graphs are obtained after the removal of unique vertices is not empty, then repeat the above procedure again. Otherwise, recorded vertex pairs form the desired bijective mapping φ.

The conception of positionally equivalence of auxiliary digraphs requires clarification in the conditions when the algorithm in each iteration is dealing with changed set of vertices of the graph. This concept is due to the vertex characteristics of the considered digraphs. In the process of work of the search algorithm of the bijective mapping, we have to take into account the history of the choice of vertices for mapping φ in the previous iteration of the algorithm.

Let us explain the above by the example.

Suppose there are two graphs G and H (see Figure 6), bijective mapping the vertices of which we want to find.

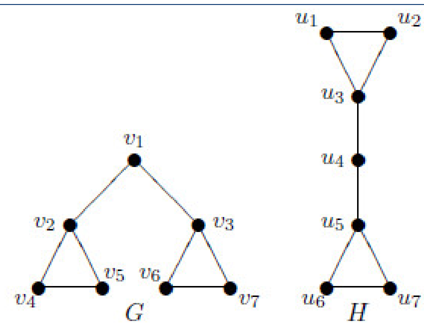


Figure. 6. The graphs G and H.

In the graph G, we choose the vertex v_1 and construct the corresponding auxiliary graph $\vec{G}(v_1)$. On form, it coincides with the figure of the graph G. Therefore, we immediately find the vertex characteristics of the digraph $\vec{G}(v_1)$.

The results are presented in the Table 5.

Table 5. The vertex characteristics of the digraph $\vec{G}(v_1)$

$I(v_1)=\varnothing;$ $O(v_1)=(1,1);$	$I(v_2)=(0);$ $O(v_2)=(2,2);$	$I(v_3)=(0);$ $O(v_3)=(2,2);$	$I(v_4)=(1,2);$ $O(v_4)=(2);$
$I(v_5)=(1,2);$ $O(v_5)=(2);$	$I(v_6)=(1,2);$ $O(v_6)=(2);$	$I(v_7)=(1,2);$ $O(v_7)=(2).$	

Further, when we construct the auxiliary directed graph for the vertices of the graph H, we find the digraph $\vec{H}(u_4)$, induced by the vertex u_4. It is easy to see that it will coincide with the figure of the digraph $\vec{G}_1(v_1)$. Find the vertex characteristics of this digraph.

The results are presented in the Table 6.

Table 6. The vertex characteristics of the digraph $\vec{H}(u_4)$.

$I(u_1)=(1,2);$ $O(u_1)=(2);$	$I(u_2)=(1,2);$ $O(u_2)=(2);$	$I(u_3)=(0);$ $O(u_3)=(2,2);$	$I(u_4)=\varnothing;$ $O(u_4)=(1,1);$
$I(u_5)=(0);$ $O(u_5)=(2,2);$	$I(u_6)=(1,2);$ $O(u_6)=(2);$	$I(u_7)=(1,2);$ $O(u_7)=(2).$	

It is easy to see that the auxiliary digraphs $\vec{G}(v_1)$ and $\vec{H}(u_4)$ positionally equivalent. They have by the single unique vertex of v_1 and u_4, respectively.

Remove the vertices v_1 and u_4 of graphs G and H. We obtain disconnected graphs G_1 and H_1 are shown in Figure 7.

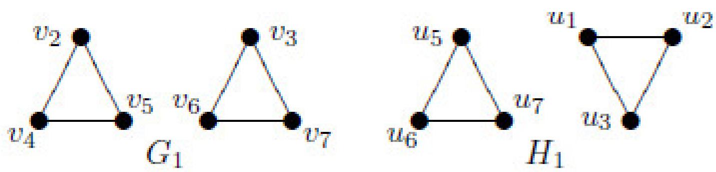


Figure 7. Graphs G_1 and H_1.

In the graph of G_1, we choose the vertex v_2 and construct the auxiliary digraph $\vec{G}_1(v_2)$. His form is the same as the first connected component of the graph. We find vertex characteristics of the digraph $\vec{G}_1(v_2)$. The results are presented in the Table 7.

Table 7. The vertex characteristics of the digraph $\vec{G}_1(v_2)$.

$I(v_1)=\varnothing$; $O(v_1)=(1,1)$;	$I(v_4)=(0,1)$; $O(v_4)=(1)$;	$I(v_5)=(0,1)$; $O(v_5)$.

In the graph H_1, we choose the vertex u_1 and construct the auxiliary digraph $\vec{H}_1(u_1)$. It is easy to see that its form will be coincide with the digraph $\vec{G}_1(v_2)$. We find vertex characteristics of the new digraph. The results are presented in the Table 8.

Table 8. The vertex characteristics of the digraph $\vec{H}_1(u_1)$.

$I(u_1)=\varnothing$; $O(u_1)=(1,1)$;	$I(u_2)=(0,1)$; $O(u_2)=(1)$;	$I(u_3)=(0,1)$; $O(u_3)=(1)$.

The resulting digraphs are positionally equivalent. Here vertices v_2 and u_1 are the unique, having equal characteristics. However, these pair of vertices (v_2, u_1) will not belong to the bijective mapping φ of graphs G and H.

It happened because when considering graphs G_1 and H_1 we did not take into account the previous stage of calculations, when it was found the couple of unique vertices v_1, u_4, remote sequently of the graphs G and H.

***

Each vertex of the auxiliary graph let's characterise by the old and new input and output characteristics in the following way.

Let in the given graphs G and H, positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$ were found. The calculated characteristics of the vertices of the digraphs, we fix for graph vertices G and H. Subsequently, after removing the unique vertices of the graphs, the vertex characteristics for new digraphs are finding and they join previously found and fixed characteristics.

For definiteness, we assume that the new vertex characteristics of the digraphs are always located on the first ``floor'' of the building from the vectors of input and output characteristics (stack). The previously found characteristics moved on one ``floor'' up.

We assume vertex characteristics of the digraphs equal if their vector characteristics are equal on the respective ``floors''. Similarly, two of the digraph call positionally equivalent if the line digraphs of the same level contain the same number of vertices having respectively equal to (input and output) characteristics.

We will write characteristics of vertices of the digraphs $\vec{G}_1(v_2)$ and $\vec{H}_1(u_1)$, obtained earlier.

We have the following characteristics for the digraphs $\vec{G}_1(v_2)$ (see Table 9).

Table 9. The vertex characteristics of the digraphs $\vec{G}_1(v_2)$.

$I(v_2) = (0);$	$I(v_4) = (1,2);$	$I(v_5) = (1,2);$
$I'(v_2) = \varnothing;$	$I'(v_4) = (0,1);$	$I'(v_5) = (0,1);$
$O(v_2) = (2,2);$	$O(v_4) = (2);$	$O(v_5) = (1);$
$O'(v_2) = (1,1);$	$O'(v_4) = (1);$	$O'(v_5) = (1).$

For the digraph $\vec{H}_1(u_1)$, we have the following characteristics (see Table 10).

Table 10. The vertex characteristics of the digraph $\vec{H}_1(u_1)$.

$I(u_1) = (1,2);$	$I(u_2) = (1,2);$	$I(u_3) = (0);$
$I'(u_1) = \varnothing;$	$I'(u_2) = (0,1);$	$I'(u_3) = (0,1);$
$O(u_1) = (2);$	$O(u_2) = (2);$	$O(u_3) = (2,2);$
$O'(u_1) = (1,1);$	$O'(u_2) = (1);$	$O'(u_3) = (1).$

It is easy to see that the constructed digraphs are not positionally equivalent.

In the graph of H_1, we choose the vertex u_3, and construct the auxiliary digraph $\vec{H}_1(u_3)$ and find the vertex characteristics of the digraph.

The results are presented in the Table 11.

Table 11. The vertex characteristics of the digraph $\vec{H}_1(u_3)$.

$I(u_1)=(1,2);$	$I(u_2)=(1,2);$	$I(u_3)=(0);$
$I'(u_2)=(0,1);$	$I'(u_2)=(0,1);$	$I'(u_3)=\varnothing;$
$O(u_1)=(2);$	$O(u_2)=(2);$	$O(u_3)=(2,2);$
$O'(u_1)=(1);$	$O'(u_2)=(1);$	$O'(u_3)=(1,1).$

In this case, we see that the digraphs $\vec{G}_1(v_2)$ and $\vec{H}_1(u_3)$ are positionally equivalent. The characteristics of pairs of vertices $(v_2,u_3), (v_4,u_1)$ and (v_5,u_2) are equal, and the pair of vertices (v_2,u_3) belongs to the bijective mapping φ between the vertices in the given graph G and H.

Theorem 3. *Time to compare the vertex characteristics of auxiliary digraphs, using the history of the graph changes, is $O(n^3)$.*

Proof. For comparison of the vertex characteristics of the auxiliary digraphs, consisting of a single ``floor'', it required, obviously, $O(n^2)$ time units. Therefore, for comparison of the vertex characteristics of the auxiliary digraphs, using the history of the graph changes, which consists of $O(n)$ ``floors'', it required $O(n^3)$ time.■

5 The Search Algorithm

We describe now the algorithm in more detail.

The input of the algorithm: graphs $=(V_G,E_G), H=(V_H,E_H)\in L_n$, isomorphism of which it is necessary to determine if it exists. We assume that these graphs have the same number of vertices and edges, as well as their vectors of local degrees D_G and D_H are equal.

The output of the algorithm: the determining the one-to-one correspondence P between the vertex sets of V_G and V_H, if it exists.

The algorithm for determining the bijective mapping between the vertices in the isomorphic graphs.

Step 1. Put $Q=G,\ S=H,\ P=\emptyset,\ N=n,\ i\coloneqq 1,\ j\coloneqq 1.$

Step 2. Choose the vertex $v_i\in V_Q$ in the graph Q.

Step 3. Construct the auxiliary digraph $\vec{Q}(v_i)$, using the graph Q.

Step 4. Find vertex characteristics of the auxiliary digraph $\vec{Q}(v_i)$.

Step 5. Choose the vertex $u_j\in V_S$ in the graph S.

Step 6. Construct the auxiliary digraph $\vec{S}(u_j)$, using the graph S.

Step 7. Find vertex characteristics of the auxiliary digraph $\vec{S}(u_j)$.

Step 8. Compare the vertex characteristics of the digraphs $\vec{Q}(v_i)$ and $\vec{S}(u_j)$ in the neighborhood of the vertices v_i and u_j of the same level.

Step 9. If the digraphs $\vec{Q}(v_i)$ and $\vec{S}(u_j)$ are not positionally equivalent then if $j < N$ then put $j := j + 1$ and go to Step 5 else stop the computations, as the graphs G and H are not isomorphic.

Step 10. If the digraphs $\vec{Q}(v_i)$ and $\vec{S}(u_j)$ are positionally equivalent then find the vertex sets $\{v_{j_1}, \ldots, v_{j_t}\}, \{u_{j_1}, \ldots, u_{j_t}\}$, unique in each of the digraphs.

Step 11. Put $P := P \cup \{(v_{i_1}, u_{j_1}), \ldots, (v_{i_t}, u_{j_t})\}$, $V_Q := V_G \setminus \{v_{j_1}, \ldots, v_{j_t}\}$, $V_S := V_S \setminus \{u_{j_1}, \ldots, u_{j_t}\}$, $N := N - t$.

Step 12. If $N \neq 0$, put $i := 1$, $j := 1$ and go to Step 2. Otherwise, stop the computations, because the bijective mapping between the vertices in isomorphic graphs G and H has constructed, the pairs of the respective vertices are stored in the set P.

Theorem 4. *The algorithm for determining the bijective mapping the vertices in isomorphic graphs finds the mapping if it exists.*

Proof. By Theorem 2 if graphs G and H are isomorphic then any bijective mapping φ, which convert the graph G into the graph H (and conversely), is determined by vertex pairs of the auxiliary positionally equivalent digraphs $\vec{G}(v)$ and $\vec{H}(u)$, having equal characteristics. If it is found t such vertices, then they explicitly define the elements of the bijective mapping φ. Note that among the vertices of the auxiliary positionally equivalent digraphs at least one vertex is unique in each of the digraphs. These are vertices, which induce the digraphs $\vec{G}(v)$ and $\vec{H}(u)$.

Deletion of the unique vertices from the graphs G and H reduces to obtaining the graphs which are also isomorphic. Therefore, the repetition of the above procedure to the obtained graphs will lead to the exhaustion of the vertex list of isomorphic graphs G and H.■

Theorem 5. *The running time of the algorithm for determining the bijective mapping the vertices in isomorphic graphs equal to* $O(n^5)$.

Proof. We determine the running time of the algorithm when performing steps 5—9.

Steps 5, 9 require to expend one unit of time for each step. Steps 6—7 require to expend $O(n^2)$ time units each. Step 8 requires to perform $O(n^3)$ time units. Therefore, n-multiple executing steps

5—9 require to expend $O(n^4)$ time units.

Single execution of Steps 2—12 requires, obviously, $O(n^4)$ time units and n-multiple execution require $O(n^5)$ time units.■

6 Conclusion

The results, presented in this article, show the fruitfulness of the method of positioning vertices of the given graphs.

One can allocate two features of our method.

- Each of the given graphs $G, H \in L_n$ is represented in the single form — in the form of the auxiliary digraph.
- The vertices of every digraph are positioned relative to each other without become attached to the vertex names of graphs.

It is possible that this approach will be used to solve other problems.

APPENDIX 1

We illustrate the proposed algorithm on another example.

Let two graphs $G, H \in L_n$ are given, the isomorphism of which we want to determine (see Figure A1).

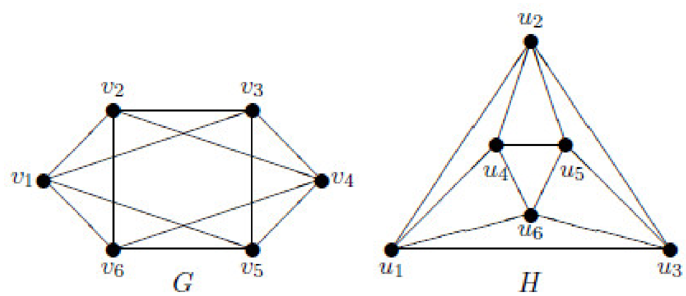


Figure A1. The given graphs G and H.

Choose the vertex v_1 in the graph G and construct the auxiliary digraph $\vec{G}(v_1)$ (see Figure A2).

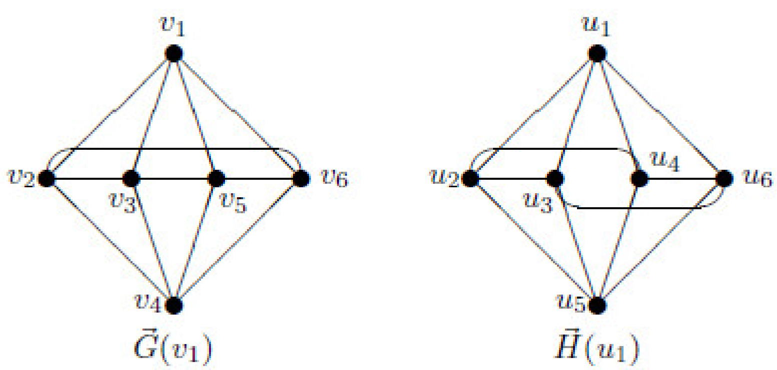


Figure A2. The auxiliary digraphs $\vec{G}(v_1)$ and $\vec{H}(u_1)$.

Find the vertex characteristics of the constructed digraph $\vec{G}(v_1)$. The results are located to the Table 12.

Table 12. The vertex characteristics of digraph $\vec{G}(v_1)$.

$I(v_1)=\varnothing;$ $O(v_1)=(1,1,1,1);$	$I(v_2)=(0,1,1);$ $O(v_2)=(1,1,2);$	$I(v_3)=(0,1,1);$ $O(v_3)=(1,1,2);$
$I(v_4)=(1,1,1,1);$ $O(v_4)=\varnothing;$	$I(v_5)=(0,1,1);$ $O(v_5)=(1,1,2);$	$I(v_6)=(0,1,1);$ $O(v_6)=(1,1,2).$

Choose the vertex u_1 in the graph H and construct the auxiliary digraph $\vec{H}(u_1)$ (see Figure A2).

Find the vertex characteristics of the newly constructed digraph $\vec{H}(u_1)$. The results are located to the Table 13.

Table 13. The vertex characteristics of digraph $\vec{H}(u_1)$.

$I(u_1)=\varnothing;$ $O(u_1)=(1,1,1,1);$	$I(u_2)=(0,1,1);$ $O(u_2)=(1,1,2);$	$I(u_3)=(0,1,1);$ $O(u_3)=(1,1,2);$
$I(u_4)=(0,1,1);$ $O(u_4)=(1,1,2);$	$I(u_5)=(1,1,1,1);$ $O(u_5)=\varnothing;$	$I(u_6)=(0,1,1);$ $O(u_6)=(1,1,2).$

It is easy to see that the constructed auxiliary digraphs $\vec{G}(v_1)$ and $\vec{H}(u_1)$ are positionally equivalent. The digraph $\vec{G}(v_1)$ has two unique vertices v_1 and v_4. They correspond to the unique vertices u_1, u_5 in the digraph $\vec{H}(u_1)$. Therefore, vertex pairs (v_1, u_1), (v_4, u_5) can be saved in the set P.

Delete from the graphs G and H the unique vertices. We get graphs G_1 and H_1 (see Figure A3).

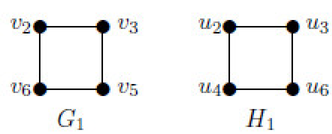


Figure A3. Graphs G_1 and H_1.

Choose the vertex v_2 in the graph G_1 and construct the auxiliary digraph $\vec{G}_1(v_2)$ (see Figure A4).

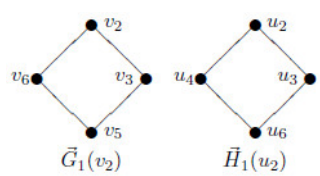


Figure A4. The digraphs $\vec{G}_1(v_2)$ and $\vec{H}_1(u_2)$.

Find the vertex characteristics of the constructed digraph. The results are located to the Table 14.

Table 14. The vertex characteristics of the digraph $\vec{G}_1(v_2)$.

$I(v_2) = (0,1,1);$	$I(v_3) = (0,1,1);$	$I(v_5) = (0,1,1);$	$I(v_6) = (0,1,1);$
$I'(v_2) = \varnothing;$	$I'(v_3) = (0);$	$I'(v_5) = (1,1);$	$I'(v_6) = (0);$
$O(v_2) = (1,1,2);$	$O(v_3) = (1,1,2);$	$O(v_5) = (1,1,2);$	$O(v_6) = (1,1,2);$
$O'(v_2) = (1,1);$	$O'(v_3) = (2);$	$O'(v_5) = \varnothing;$	$O'(v_6) = (2).$

Choose the vertex u_2 in the graph H_1 and construct the auxiliary digraph $\vec{H}_1(u_2)$ (see Figure A4).

The calculated vertex characteristics of the newly constructed digraph are located to the Table 15.

Table 15. The vertex characteristics of the digraph $\vec{H}_1(u_2)$.

$I(u_2) = (0,1,1);$	$I(u_3) = (0,1,1);$	$I(u_4) = (0,1,1);$	$I(u_6) = (0,1,1);$
$I'(u_2) = \varnothing;$	$I'(u_3) = (0);$	$I'(u_4) = (0);$	$I'(u_6) = (1,1);$
$O(u_2) = (1,1,2);$	$O(u_3) = (1,1,2);$	$O(u_4) = (1,1,2);$	$O(u_6) = (1,1,2);$
$O'(u_2) = (1,1);$	$O'(u_3) = (2);$	$O'(u_4) = (2);$	$O'(u_6) = \varnothing.$

We find that the constructed auxiliary digraphs $\vec{G}_1(v_2)$ and $\vec{H}_1(u_2)$ are positionally equivalent. The digraph $\vec{G}_1(v_2)$ has two unique vertices v_2 and v_5. They correspond to the unique vertices u_2, u_6 in the digraph $\vec{H}_1(u_2)$. Therefore, vertex pairs $(v_2, u_2), (v_5, u_6)$ can be saved in the set P.

Removing the unique vertices of the graphs G_1 and H_1, we obtain respectively the graphs G_2 and H_2 consisting of two isolated vertices each. It is clear that it will be obtained auxiliary positionally equivalent digraphs $\vec{G}_2(v_3)$ and $\vec{H}_2(u_3)$. Inducing vertices form the pair (v_3, u_3) of unique vertices that can be saved in the set P.

We also find that the pair of vertices (v_6, u_4) belongs to the set P.

Thus, the binary relation between vertices of isomorphic graphs G and H is:

$$\varphi = \{(v_1, u_1), (v_2, u_2), (v_3, u_3), (v_4, u_5), (v_5, u_6), (v_6, u_4)\}.$$

Perform the verification of the result.

$$\begin{array}{ll}
\{v_1, v_2\} \leftrightarrow \{u_1, u_2\}; & \{v_2, v_6\} \leftrightarrow \{u_2, u_4\}; \\
\{v_1, v_3\} \leftrightarrow \{u_1, u_3\}; & \{v_3, v_4,\} \leftrightarrow \{u_3, u_5\}; \\
\{v_1, v_5\} \leftrightarrow \{u_1, u_6\}; & \{v_3, v_5\} \leftrightarrow \{u_3, u_6\}; \\
\{v_1, v_6\} \leftrightarrow \{u_1, u_4\}; & \{v_4, v_5\} \leftrightarrow \{u_5, u_6\}; \\
\{v_2, v_3\} \leftrightarrow \{u_2, u_3\}; & \{v_4, v_6\} \leftrightarrow \{u_5, u_4\}; \\
\{v_2, v_4\} \leftrightarrow \{u_2, u_5\}; & \{v_5, v_6\} \leftrightarrow \{u_6, u_4\}.
\end{array}$$

REFERENCES

[1] Harary F. *Graph Theory*. Addison-Wesley, Reading, MA, 1969.

[2] West D.B. *Introduction to Graph Theory*, 2nd ed. Prentice Hall, Inc., NJ, 2001.

[3] Varmuza K. *Chemometrics in Practical Applications*. Rijeka, Croatia: InTech, 2012. – 326 p.

[4] Dehmer M., Grabner M. *The Discrimination Power of Molecular Identification Num-bers Revisited*. MATCH Commun. Math. Comput. Chem. – 2013. – V. 69. – No 3. – pp. 785–79.

[5] McKay B.D., Piperno A. *Practical Graph Isomorphism*. J. Symbolic Computation. – January 2014. – V. 60. – pp. 94–112.

[6] Wale N. and Karypis G. *Comparison of descriptor spaces for chemical compound re-trieval and classification*. In Proceedings of the International Conference on Data Mining, pages 678–689, Hong Kong, 2006.

[7] Aho A.V., Hopcroft J.E., and Ullman J.D. *The design and analysis of computer algorithms*. Addison-Wesley publishing company, N.Y., 1976.

[8] Garey M.R. and Johnson D.S. *Computers and Intractability*. W.H.Freeman and Company, San Francisco, 1979.

[9] Kukluk J. et al., *Planar Graph Isomorphism*, JGAA, 8(3) - 2004 - pp. 313–356.

[10] Babai L. *Graph Isomorphism in Quasipolynomial Time*. arXiv:1512.03547v1, 2015, 84 pages.