https://doi.org/10.26599/IJCS.2023.9100002 International Journal of Crowd Science

# Modeling and Verification of Simulation-Oriented Digital Selves

Zhaotong Wang[1] and Hongbo Sun[1] ✉

## ABSTRACT

Networked life has now become one of our major life forms. In social networks, each individual has its own attributes and certain functions, which makes the current network present characteristics that the previous network did not have. The existing research believes that the structure and attributes of individuals in a network are the same, and they are in a single network at the same time. However, individuals in any social network may be in different networks at the same time and thus exhibit different behaviors, and such individuals are called digital selves. In this paper, we propose a simulation-oriented modeling method for digital selves, which allows them to be in multiple networks at the same time and to have their own decision-making mechanisms. The model consists of six parts, namely, pattern, affecter, decider, executor, monitor, and connector. After the verification of three simulation experiments, namely coevolutions, ecological structure evolution of an e-commerce market, and multi-information coevolution spreading, the model can be well applied in various scenarios, which verifies its feasibility and applicability.

## KEYWORDS

crowd network; digital self; simulation; modeling and verification

From horse migration in the animal world to "weak things united become strong" in human societies, group behaviors can be observed everywhere in daily lives. Compared with "fight alone", group behaviors often have a stronger ability to operate; this ability is called group intelligence[1], and the ability level reflects the group intelligence level. Group intelligence is presented in all aspects of science and is the research focus in various fields (e.g., biological, social, and computer). In fauna, it can reveal itself in collaborative foraging and migration, whereas in human societies, it can manifest itself in voting and meetings. Although the research interests of group intelligence are different, the purpose of its study is the same, i.e., how to apply this group intelligence in a rational way.

Along with the development of social networks, group intelligence has become more common and complex in human societies. In web-based industrial and daily life, each user can be deemed as a unit that participates in various activities on the web in a direct and self-directed way, which makes web communities have a very high level of group intelligence[2]. Moreover, web-based group intelligence has already demonstrated its power in many ways, such as assessing the loss of lives and properties caused by the Delta variant and public opinions around the world during Donald Trump's presidential campaign. Hence, this web-based mode of industrial and daily life is large-scale, open, self-organized, and evolving. Accordingly, Chai et al.[3] referred to it as "crowd intelligence" in our study, and users map the digital world as digital selves who are influenced by and react to crowd networks.

Crowd network integrates existing knowledge and methods from various scientific fields and disciplines, and this complexity obviously cannot be studied by traditional methods. Computer simulations can be free from obstacles from real systems to the greatest extent, so it has become an important method in studying crowd networks[4].

To efficiently describe the behaviors of intelligent subjects in a crowd network, Wang et al.[5] constructed a cyber-physical-psychological ternary fusion system and established a behavioral logic model of digital selves, which reflects the behaviors, purposes, and preferences of intelligent subjects in real systems.

Based on the behavioral logic model proposed by the above theory, this paper establishes a simulation-oriented general model of digital selves to satisfy the requirements of crowd network simulations and applies it in three areas to verify the feasibility and applicability of the model: coevolutions, ecological structure evolution of the e-commerce market, and multi-information coevolution spreading. The rest of this paper is organized as follows: Section 1 reviews existing complex simulation systems and related works on simulation modeling. Section 2 describes the simulation member model proposed in this paper. Section 3 describes three simulation studies based on this model. Section 4 draws the conclusions of the proposed model.

## 1 Related Work

Based on an in-depth analysis of complex simulation systems, this kind of simulation system often has the characteristics of large-scale, distributivity, interactivity, and openness. Mao et al.[6] designed a multi-dimensional interactive simulation system to study satellite communication applications. The system can be divided into three layers: satellite simulation application, distributed processing engine, and model/database. The engine layer mainly completes various functions, such as data protocol format conversion between federations, core service processing, and system process management. The model/database layer is the basic platform for system operations and provides data services for the distributed processing engine layer. The shipboard satellite communication system[7] solves key problems, such as time unification, fault simulation, and real-time visual display. The system is divided into four implementation layers: communication interface layer, data layer, functional entity layer,

1 School of Computer and Control Engineering, Yantai University, Yantai 264005, China
Address correspondence to Hongbo Sun, hsun@ytu.edu.cn

and human-computer interaction layer. Among them, the communication interface layer implements the data exchange among federation members according to the high-level architecture/run-time infrastructure (HLA/RTI) architecture. The satellite communication simulation system[8] consists of a network control center, a resource management center, and various satellite ground stations. The technical route is divided into three main lines. The first one is the main line of the task list. The mission scenarios and application models consist of decomposed mission planning. The second is the main line of physical models, and the goal is to form a complete model library of satellite communication systems. The third is the main line of scenarios, and the optimization of the system is achieved by building a multidimensional interactive system model library and processing engine.

The simulation model of the simulation system differs for different purposes, and designing a reasonable model structure helps in the development and operation of the simulation system. The traditional development based on the HLA architecture needs to consider which members and objects need to be constructed for the simulation system. The artillery element combat simulation system[9] constructs six federal members based on the analysis of artillery detachment combat activities. In this distributed system, the artillery battalion commander, deputy battalion commander, and scout are mapped as the observation warfare member, gun position member, and observer member, respectively. The simulation control member, simulation record member, and evaluation member are on a single simulation node. The system uses an advance mechanism based on time steps. The air-to-ground missile countermine simulation system[10] is divided into 13 federal members, namely, the system general control member, aircraft/integrated avionics fire control system simulation member, air-ground missile simulation member, ground-air weapon simulation member, target simulation member, electro-optical countermeasure simulation member, command automation simulation member, etc.

Due to the unique autonomy and collaboration of agents, the multi-intelligence body can adapt to the changing complex system by reorganizing its structure[11]. With the HLA system having the problem of a fixed structure, some scholars have attempted to combine the HLA system with multiple agents and use the inherent autonomy, initiative, interaction, and intelligence of the agents to make up for the shortcomings of HLA architectures. The military logistics simulation system[12] uses the distributed simulation HLA architecture and agent technology to design non-agent federation members and agent federation members to meet the system's intelligence requirements and make the system realistic on the basis of achieving interoperability and scalability of the military logistics system.

## 2 Simulation-Oriented General Model of Digital Selves

Based on the behavioral logic model of digital selves proposed by Wang et al.[5], this paper proposes a simulation-oriented general model of digital selves. In this model, the attributes of digital selves are designed for simulation purposes. The perception module is mapped as an affecter in crowd network simulations and used to influence deciders in making decisions. The mental module and decision-making module are mapped as the deciders and used to calculate the preference matrix of the deciders. The action module is mapped as an executor to execute the final decision based on the result of the decision-making module and perception module. To realize the interaction with other digital selves, a connector is added to the general model. In addition, a monitor is added to the general model to monitor and record the changes in some parameters during the crowd network simulations. Based on the above analysis, the behavioral logic model and simulation-oriented general model of digital selves are shown in Figs. 1 and 2, respectively.

### 2.1 Mental module and decision-making module corresponding to the decider

The main purpose of the mental module in the behavioral logic model is to receive and examine the information perceived by the perception module. After the mental operations (e.g., calculating the preference match between decision-makers and candidates in the voting process), the results are used to influence the decision-

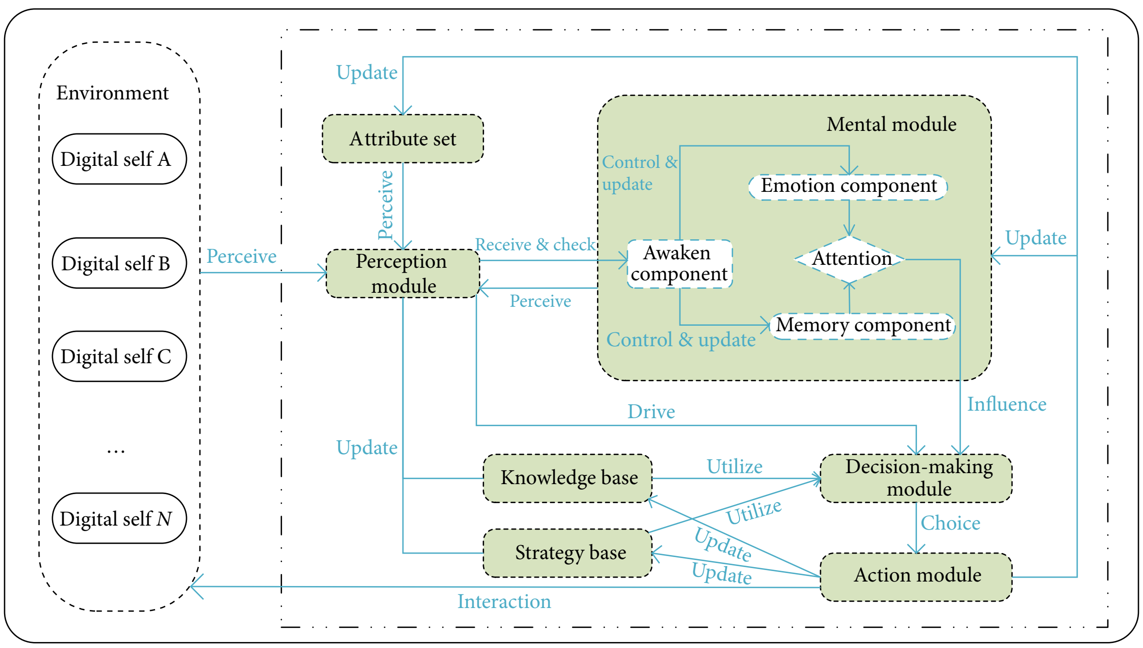



**Fig. 1 Behavioral logic model.**

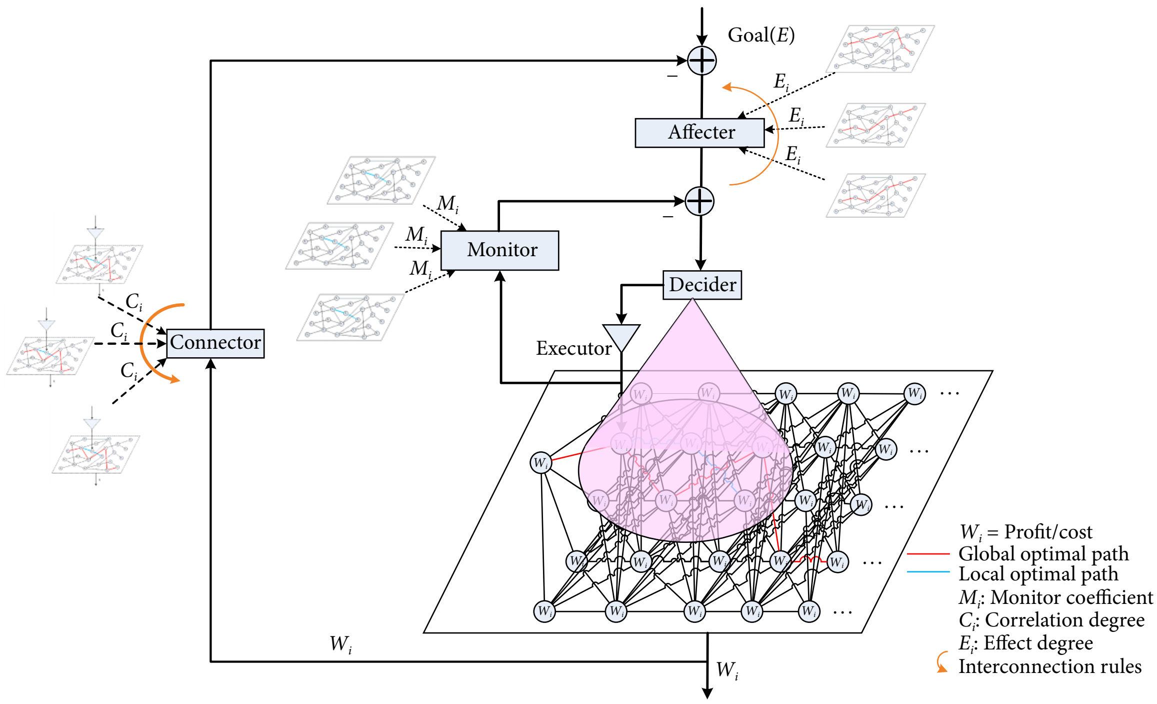



**Fig. 2 Simulation member model.**

making module. The decision-making module is driven by the perception module (corresponding to the affecter of the crowd network simulations) and influenced by the mental module, which is responsible for reasoning and decision-making. During the reasoning and decision-making process, the decision-making module needs to utilize the knowledge from the knowledge base and the strategies from the strategy base. Finally, the decision-making module transmits the decisions made by combining various types of knowledge and under the influence of mindfulness to the action module (corresponding to the executor in crowd network simulations). For the computation in the simulation, the mental module and decision-making module are mapped into the decider, which makes decisions by combining the resource situation and capabilities (reflecting depth, an aspect of endowment). The mental module, decision-making module, and decider correspondence are shown in Fig. 3.

## 2.2 Action module corresponding to the executor

The action module (executor) receives the decisions made by the decision-making module (decider) and selects different components to perform different decision tasks, thus updating the attribute set, knowledge base, strategy base, and mental module in digital selves. Finally, the interaction with other digital selves (corresponding to the connector in crowd network simulations) is completed. The action model is mapped into the executor, which executes choices according to the decisions of the decider (decision-making module and mental module) and the suggestions of the affecter (perception module) in a proportion influenced by the self-confidence level (volition component). The action module and executor correspondence are shown in Fig. 4.

## 2.3 Perception module corresponding to the affecter

The perception module senses the identity, supply, demand, and circle state information of the digital self itself and other digital selves from its own attribute set and the environment, respectively. Then this information is used to update the knowledge base and strategy base of digital selves and passed to the mental module

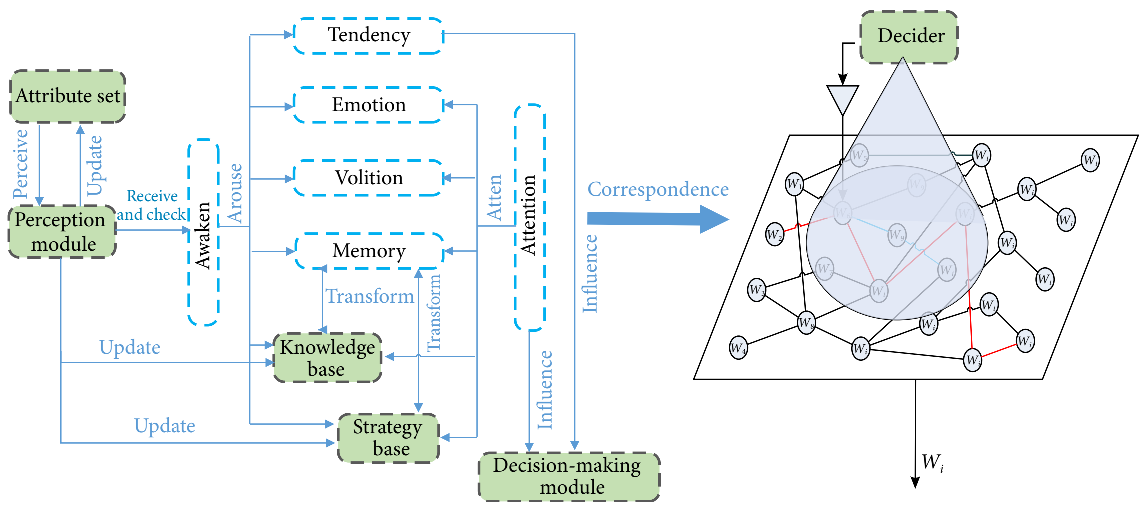



**Fig. 3 Mental module, decision-making module, and decider correspondence diagram.**

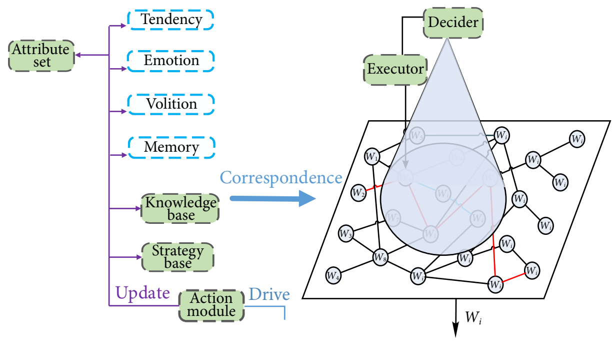



**Fig. 4 Action module and executor correspondence diagram.**

(corresponding to the decider in crowd network simulations). The perception module is mapped in crowd network simulations as an affecter, which is composed of several advisors. This condition influences the decisions of digital selves, and the intensity of its influence is determined by the interconnection rules of digital selves and advisors. The suggestions of the advisors are aggregated with the decisions of the decider at the executor. The specific mapping relationship is shown in Fig. 5.

### 2.4 Pattern

The pattern is an important part of the general model and contains all possible choice paths for digital selves. Before the simulation advances in each unit, this model inevitably faces multiple decision-making choices, and different path choices imply different conditions and costs. Due to the limitation of the intelligence level and resources, digital selves often can only find a local optimal path even if a global optimal path exists. The pattern is shown in Fig. 6.

### 2.5 Monitor

In crowd network simulations, it is necessary to monitor the parameters and control their changes, so a monitor is added to this model. In addition to monitoring the changes in parameters, the monitor also corrects deviations according to the specific simulation purpose, where the self-discipline level represents the self-correcting ability of digital selves. Meanwhile, the monitor's interference represents the external correcting ability, and its monitoring intensity is determined by the interconnection rules. The monitor is shown in Fig. 7.

### 2.6 Connector

To realize the interaction function between the digital self and other digital selves, this model adds connectors. By connecting with other digital selves, we can obtain the assignment of global variables in their decision process and the final decision results and update their relevant information in the next generation. However, there is no corresponding exclusive module in the behavioral logic model, but it can be used to complete the interaction with other digital selves in the action module. The connector belongs to the last step of the whole decision-making model and is the most critical step in the process of realizing the decision of the $i$-th generation to the $(i+1)$-th generation. The specific form is shown in Fig. 8.

## 3 Simulation Execution and Validation

Based on the general model proposed above, we conduct the following experiments separately to verify the reasonableness of the model.

### 3.1 Coevolutions simulation

In the field of coevolutions, the two main factors affecting the intelligence level of digital selves are the quality of their response to the task and the time to respond to the task, respectively. We thus use this quality-time model (QTM) based on the crowd intelligence level as the intelligence indicator of our evolutionary approach. The CIQ is the overall intelligence level of the intelligent subject in the QTM, which is calculated as follows:

$$\mathrm{CIQ} = \sum_{i}^{N} \frac{Q_i}{T_i},$$

where $Q_i$ denotes the comprehensive evaluation of the intelligent subject in the $i$-th task, and $T_i$ denotes the time taken by the intelligent subject to complete the $i$-th task.

The evolutionary method proposed in this experiment first initializes the number $N$ of digital selves and the quality and transmission time of digital selves. Among them, the quality of tasks completed by digital selves (i.e., comprehensive evaluation) $Q_i\,(i=1,\,2,\,\dots,\,N)$ is calculated using computing capacity,

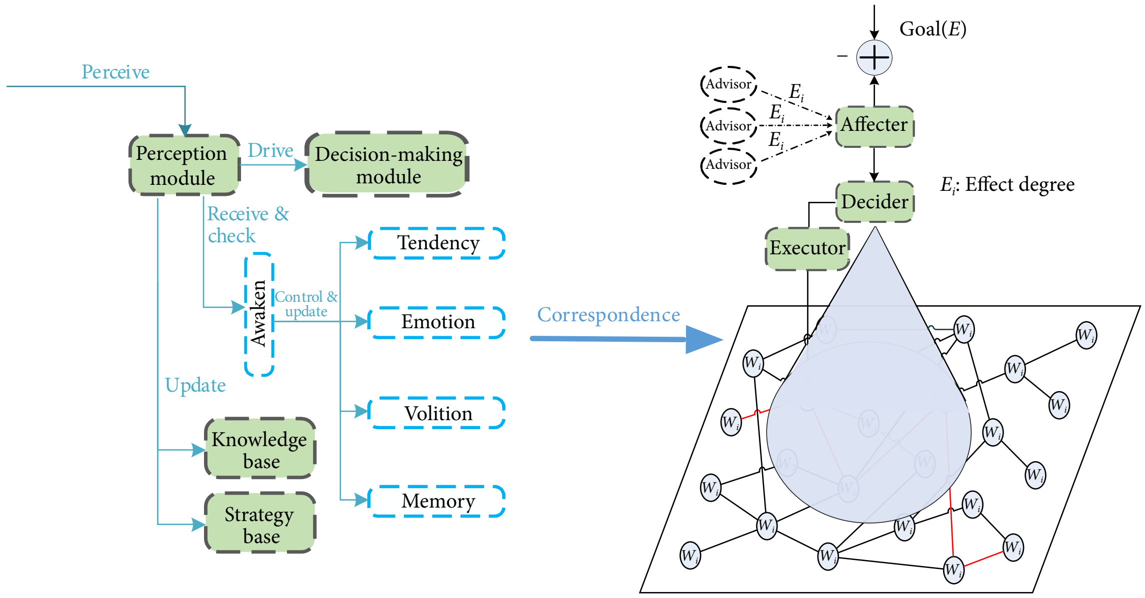



**Fig. 5 Perception module and affecter correspondence diagram.**

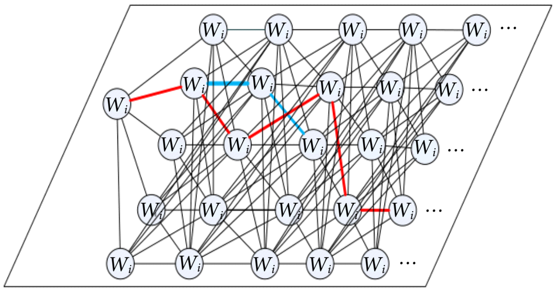


Fig. 6 Pattern demonstration of information spreading. $W_i$=Profit/cost.

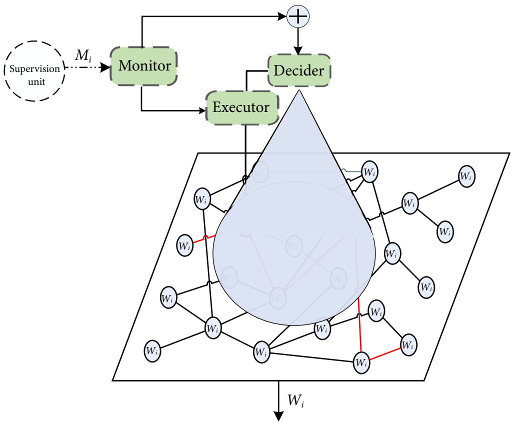



Fig. 7 Correction feedback mechanism by monitor. $M_i$ represents monitor coefficient.

storage capacity, and communication capacity. The time $T_i\,(i=1,\ 2,\ \ldots,\ N)$ is calculated according to the transmission time of digital selves.

$$Q_i = \text{comp}_i \times \text{cach}_i \times \text{comm}_i,\ i=1,\ 2,\ \ldots,\ N,$$

where $\text{comp}_i$ represents the computing capacity of the $i$-th digital self, $\text{cach}_i$ represents the storage capacity of the $i$-th digital self, and $\text{comm}_i$ represents the communication capability of the $i$-th digital self.

Then, digital selves are classified according to the intelligence level obtained by the QTM. The first step of classification is to initialize the number of clusters and set the cluster center to $\mu_j\,(j=1,\ 2,\ \ldots,\ k)$. The second step is to calculate the difference $I_{\text{dis}_i}$ between the intelligence level of each digital self $I_i$ and the cluster center $I_{\mu_j}$. The specific formula is

$$I_{\text{dis}_i} = \left|I_i - I_{\mu_j}\right|(i=1,\ 2,\ \ldots,\ N|j=1,\ 2,\ \ldots,\ k).$$

Digital selves are classified according to the minimum intelligence level difference, so each digital self is divided into the class of the center with the minimum difference. Then, the location of the cluster center is recalculated to ensure that the difference between the new cluster center and each digital self in the class is minimal. At the end of classification, the second step is performed until the cluster center converges. After the classification is completed, the initial crowd intelligence level is calculated, and the number of evolutionary iterations is set. During the iteration process, the idea of the simplified particle swarm optimization algorithm is adopted to calculate the intelligence level of digital selves. The equation is

$$x_i^{g+1} = x_i^g + c \times r \times (p_{\text{ec}} - x_i^g),$$

where $x_i^g$ denotes the intelligence level at the $g$-th iteration of the $i$-th digital self; $c$ is the learning factor, a non-negative constant; $r$ is a random number between $(0,1)$; and $p_{\text{ec}}$ denotes the intelligence level of the evolutionary center.

When the experiment reaches the maximum number of evolutions, the evolved crowd intelligence level is calculated and compared with the initial crowd intelligence level to analyze

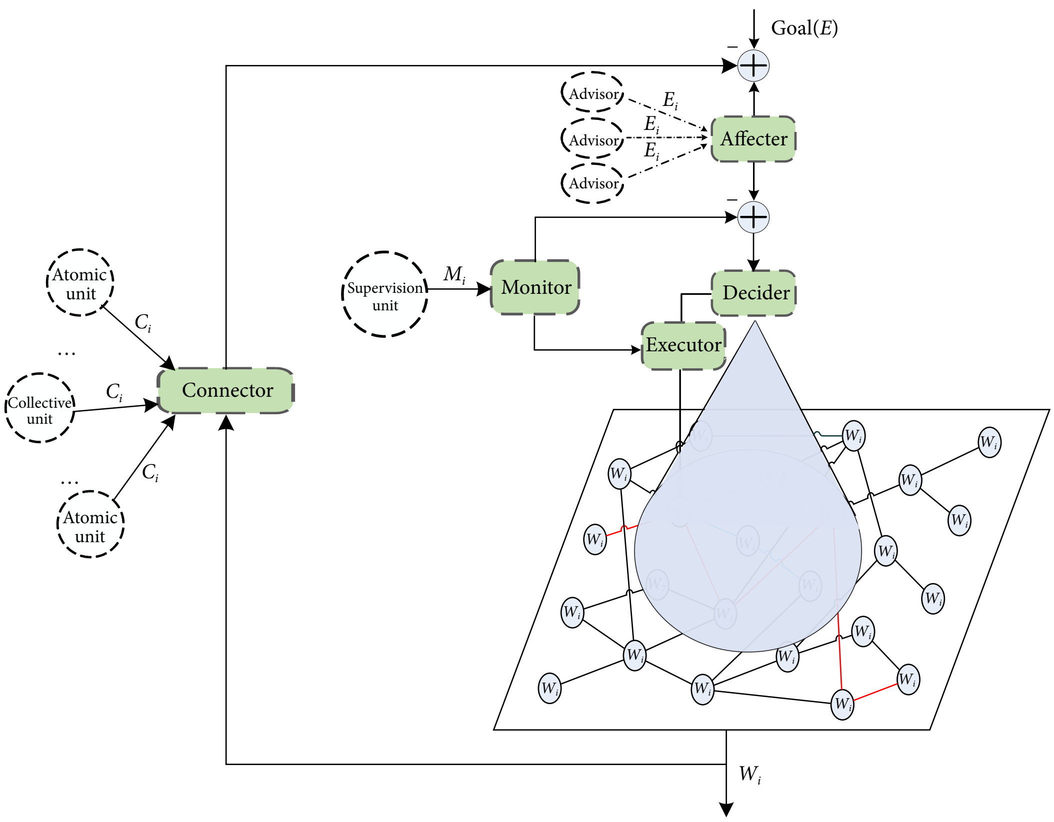



Fig. 8 Evolution feedback mechanism by connector. $C_i$ represents correlation degree.

whether the digital selves have evolved.

In each simulation iteration of the experiment, the advancing process of digital selves is as follows:

(1) The processing time of each assigned task is calculated according to its own computing capacity, storage capacity, and transmission capacity. This process is implemented by the executor in the simulation model (Algorithm 1).

(2) After all the tasks are processed, the top five digital selves with the highest intelligence level are selected in each cluster, and the individual with the highest $Q$ among the five digital selves is selected as the population evolution center of the crowd evolution method. Then, the evolution is performed according to the iterative formula of the crowd evolution method (Algorithm 2).

Assuming the same crowd intelligence level, the comparison experiment uses an evolutionary method based on the quadratic regression analysis, which calculates the positive difference between the digital selves. Based on the positive difference, different relationship weights are given to every digital self connected to these digital selves. Moreover, according to the magnitude of this weight, the evolutionary dynamics directions of digital selves are calculated, and the evolution of digital selves is achieved by the degree of coincidence between their workload and the evolutionary dynamic direction.

The experimental results of the two methods are shown in Fig. 9. Generation denotes the number of simulation iterations. As shown in Fig. 9, both methods based on the general model can achieve the positive evolution of the crowd intelligence level, but the experimental final intelligence level varies more. This is due to the fact that the experimental approach is limited to the in-cluster evolution, which has a limited evolutionary effect in clusters with a generally low intelligence level. By contrast, in the comparison

**Algorithm 1 Executor algorithm**

```
Input:   {task}     // A list of saved tasks needed to proceed
         unit       // Intelligence unit
Output:  {time_s}   // A list saved time spent on every task
    foreach task in {task} then
        time_s = 0
        time_s += task[comp]/unit[comp];
        time_s += task[cach]/unit[cach];
        time_s += task[comm]/unit[comm];
        append time_s to list {time_s};
    end foreach
return {time_s};
```

**Algorithm 2 Connector algorithm**

```
Input:   center   //An evolution center from a cluster
         unit     //An intelligence unit
Output:  unit     //An intelligence unit that finished evolution
  if |CIQ(unit) − CIQ(center)| > 0.12 then
     unit[comp] = unit[comp] + fac_learn × rand × (center[comp] − unit[comp]);
     unit[cach] = unit[comp] + fac_learn × rand × (center[cach] − unit[cach]);
     unit[comm] = unit[comm] + fac_learn × rand × (center[comm] − unit[comm]);
     unit[time] = unit[time] + fac_learn × rand × (center[time] − unit[time]);
  end if
return unit;
```

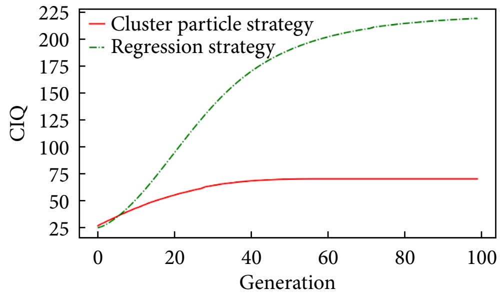



**Fig. 9 Global intelligence measure statistics.**

algorithm, the evolution of digital selves is driven by the difference in the ability of the digital self and the neighboring digital selves, and the upper limit of the digital self evolution will be higher in the case of high connectivity.

### 3.2 Simulation of the ecological structure evolution of the e-commerce market

In the e-commerce market, there are a large number of transaction subjects, such as manufacturers, service providers, and intermediaries, which are connected to one another through the network. Different transaction relations form different connection modes between different transaction subjects. The transaction behavior of each transaction subject determines the role played by the transaction subject in the market, and the transaction relations between these transaction subjects together constitute the e-commerce market structure. The driving force of the evolution of the e-commerce market structure is the market transaction efficiency, and the change in the transaction efficiency can affect the evolution process of the market structure. The experiment tries to predict the evolution of the e-commerce market structure by controlling the change in transaction efficiency and then proves the applicability of the model proposed in this paper.

The experiment first needs to initialize the simulation members with specified member models and member types, such as manufacturers, service providers, intermediaries, and three production and operation modes (self-production and self-marketing model, intermediaries resell mode, and manufacturers purchase transaction necessary services for the direct sales mode, respectively). Then, environment variables and global variables, including the income threshold $T$, simulation iteration number $M$, and transaction efficiency $k$, are inputted. Afterward, the simulation enters the first loop, and each simulation member calculates its own income according to its own production and operation modes. The second loop simulation member compares the income of neighboring members through the connector, selects the production and operation modes of the member with the largest income, and records it. The simulation member itself finally makes a decision to change its own production and operation modes through the suggestion of the advisors and its own self-confidence level. The executor performs the operation of merging, splitting, or keeping the status quo to change its own state and update the data of simulation members. Wait until the next iteration cycle to repeat this process to calculate the income, judge whether each simulation member has been calculated and compared, and record the number of various types of simulation members and the number of production and operation modes. This is the end of the second loop. Then, whether the experiment reaches the maximum number of iterations is judged, and the first layer of the cycle is ended. Finally, the data are recorded, and the experimental data are read and displayed.

The specific simulation advancement process is presented as follows:

(1) The suggestions of the advisors are compared with the production and operation modes specified after the previous iteration, and the advisors suggest the behavior of others according to their preferences (keeping, merging, splitting, and three production and operation modes). Finally, a simulation member will obtain the suggestions of all advisors and return the suggestions with the highest weight in the set, which will be called by the decider (Algorithm 3).

(2) Before the start of each iteration, the most profitable model from the three trading models is chosen for production and operation. Decisions on who to deal with, targets for completing transactions, and allocation of labor endowment for producing goods or services necessary for transactions based on resources and capabilities are made (Algorithm 4).

(3) The monitor corrects deviations from the goal or commitment, representing the external correction capability. The monitoring intensity is equally distributed, and the monitoring range is set. The monitor monitors the extent of deviations from the behavior and preferences of the congregate simulation members according to the preferences on the grid (Algorithm 5).

(4) The connector connects with other trading subjects, learns from the behavior results of other trading subjects, and acts as negative feedback in the next round of evolution. A certain number of other manufacturers are randomly selected to compare the profits. Then, the production and operation modes of manufacturers with high incomes and the accepted price of the service necessary for the transaction with a certain probability are copied (Algorithm 6).

$$P_i^p = \frac{I_i^p}{\sum_{i=1}^{n} I_i^p},\ P_i^s = \frac{I_i^s}{\sum_{i=1}^{n} I_i^s},\ P_i^m = \frac{I_i^m}{\sum_{i=1}^{n} I_i^m},$$

$$M = \max\{P_1^p,\ P_2^p,\ \ldots,\ P_n^p\},$$

where $I_i^p$ denotes the income of manufacturer $i$, $I_i^s$ denotes the income of the service provider $i$, $I_i^m$ denotes the income of the intermediary $i$, $P_i^p$ denotes the income ratio probability of the

**Algorithm 3 Affecter algorithm**

```
Input:   member     //Simulation member
         {AL}       //Advisor list
Output:  sug        //suggestion
  foreach AL in {AL} then
    {sug} += AL(member);
    end foreach
      select sug_max in {sug};
return sug_max;
```

**Algorithm 4 Decider algorithm**

```
Input:   mode      //Production and operation modes (a, b, c).
         sug       //Suggestion.
         Mode_c    //Impact intensity of other members' income.
         T         //Behavioral threshold.
Output:  OD        //Decision orders.
if sug > Mode_c && sug > sug_L then //Last round of impact intensity
  mode = sug;
else then
    mode=Mode_c;
end if
  if I ⩾ T then
    OD = "keep";
  else if (I < T) then
      if (The simulation member state is a single state) then
          OD = "combine";
      else if (The simulation member state is a full compound state)
then
          OD = "split";
      else if (random() × 100 < 50) then
          OD = "combine";
      else then
          OD = "split";
      end if
    end if
return OD;
```

**Algorithm 5 Monitor algorithm**

```
Input:   {ML}     //Monitor list
Output:  E_m      //Monitoring intensity
foreach ML in {ML} then
      {E_m} += ML();             //External monitoring intensity
end foreach
return E_m = random({E_m});      //Select an external monitoring
                                   intensity at random
```

**Algorithm 6 Decider algorithm**

```
Input:   {NL}     //Neighbor member list
         k        //Transaction efficiency
         P        //Sale price
         Q        //Sales quantity
         E        //Trading endowment
         w        //Wholesale price
         p_s      //Transaction required service sales unit price
         s_s      //Transaction required service sales quantity
         p_b      //Transaction required service purchase price
         s_b      //Transaction required service purchase quantity
Output:  I        //Income
         Mode_c   //The model of the most profitable member
  // Calculate income according to the mode
  if (currentSelfState== "a") then
      I = k × P × Q − E;
  else if (currentSelfState== "b") then
      I = k × P × Q − w × Q/k + k × p_s × s_s − p_b × s_b/k − E;
  else if (currentSelfState== "c") then
      I = k × P × Q − p_b × s_b/k − E;
  end if
  Mode_c = mode_NLmax
return I, Mode_c;
```

manufacturer $i$, $P_i^s$ denotes the income ratio probability of the service provider $i$, $P_i^m$ denotes the income ratio probability of the intermediary $i$, and $M$ denotes the production and operation modes of the manufacturer with maximum probability.

The experimental results of the evolution of the ecological structure of manufacturers, intermediaries, and service providers obtained by the simulation advancement model are shown in Fig. 10.

According to the experimental results, while proving the feasibility and applicability of our simulation model, it can also be interpreted and predicted for the real world. In the early stage of the development of the real-world e-commerce market, due to the low transaction efficiency, manufacturers select the production and operation modes of self-production and self-marketing, so there are no other two types of transaction subjects. With the improvement of transaction efficiency and the expansion of the product market, the original manufacturers differentiate into intermediaries and service providers. Moreover, due to the large market gap at the early stage of differentiation, the two types of subjects grow explosively immediately after their emergence. When the transaction efficiency reaches a certain height, and the market becomes saturated, the number of three types of subjects will be compressed to different degrees and stabilize. Among them, intermediaries will exist in the market with a very low proportion, which indicates that intermediaries will not be completely eliminated by the market in the future.

### 3.3 Simulation of multi-information coevolution spreading

In information spreading, there is a possibility of multi-information coevolution spreading simultaneously. To achieve the effective control of multi-information coevolution spreading, we conducted simulation experiments on the evolution of two relevant sub-events in the crowd network. The experiment proposes a mixed SIR (MSIR) model based on the traditional SIR model. The model will set the member property to be limited in individual energy, i.e., it cannot handle two events at the same time. Therefore, on the basis of single-event states $\{S_1, I_1, R_1\}$ and $\{S_2, I_2, R_2\}$, the states $\{I_1, I_2\}$ that spread $E_1$ and $E_2$ are excluded to obtain the mixture states of individuals: $\{S_1S_2, I_1S_2, R_1S_2, S_1I_2, R_1I_2, S_1R_2, I_1R_2, R_1R_2\}$. Figure 11 depicts the mixed-state transitions of the individuals, where $\beta_1$, $\gamma_1$, $\alpha_2$, $\delta_2$, $\beta_2$, $\gamma_2$, $\alpha_1$, and $\delta_1$ are the transition rates between the states.

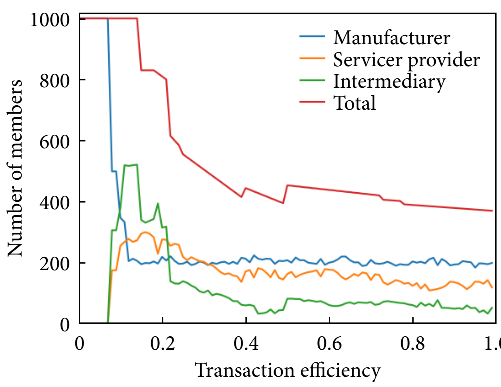



**Fig. 10 Simulation of member changes.**

Based on the principle of relevant sub-event evolution, the experiment used the modeling method proposed in this paper to set up the intelligent subject and its network structure, information, and other simulation members. Then, the network was used to make events evolve among the members and verify the MSIR model according to the changes of each property of the simulation members.

In the modeling of the simulation members, the main attributes are endo$_{\text{stored}}$, preferenceInfo, level$_c$, and level$_d$, where the attribute endo$_{\text{stored}}$ follows a normal distribution with parameters ($\mu$=20, $\sigma$=2) and takes values in the range ($\mu$−2$\sigma$, $\mu$+2$\sigma$). The attribute preferenceInfo is set to a normal distribution with ($\mu$=0.4, $\sigma$=0.23) and takes values in the range ($\mu$−2$\sigma$, $\mu$+2$\sigma$). The attribute level$_c$ is set to a normal distribution with ($\mu$=0.6, $\sigma$=0.33) with a normal distribution taking values in the range ($\mu$−2$\sigma$, $\mu$+2$\sigma$). The attribute level$_d$ is set to ($\mu$=0.5, $\sigma$=0.33) with a normal distribution taking values in the range ($\mu$−2$\sigma$, $\mu$+2$\sigma$) (Table 1).

The main advancement process of the simulation members in each simulation iteration is as follows.

(1) Each received event is integrated into the advised path according to the suggestions of the advisors. The process is implemented by the affecters in the simulation model (Algorithm 7).

(2) After combining a series of parameters, such as its own attributes (preferences, confidence level, and endowment), the decider calculates the corresponding decision results with the current event (Algorithm 8).

Figure 12a depicts the experimental results of the single information propagation, i.e., the SIR model. The experimental results of the SIR model are well simulated based on the modeling method proposed in this paper. In Fig. 12 b, the experimental results also meet the expected criteria of the MSIR model. As a result, the trends may be exhibited when two messages disseminate simultaneously. Moreover, the MSIR model based on this modeling method can effectively characterize the state changes of multi-information coevolution spreading.

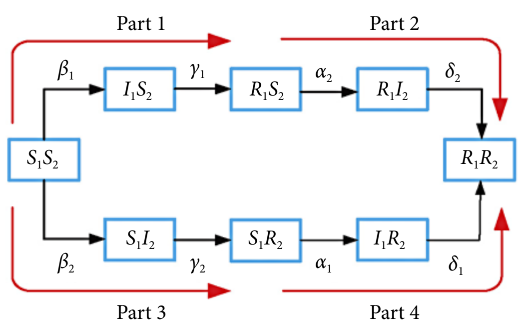



**Fig. 11 $E_1$ and $E_2$ state transition diagram.**

**Table 1 Simulation member properties.**

| Property name | Property explanation | Type | Value range |
|---|---|---|---|
| ID | Digital self ID | id | ID-0–ID-560 |
| endo$_{\text{stored}}$ | Store endowment | double | ($\mu$−2$\sigma$, $\mu$+2$\sigma$), $\mu$=20, $\sigma$=2 |
| preferenceInfo | Preference | double | ($\mu$−2$\sigma$, $\mu$+2$\sigma$), $\mu$=0.4, $\sigma$=0.23 |
| level$_c$ | Confidence level | double | ($\mu$−2$\sigma$, $\mu$+2$\sigma$), $\mu$=0.6, $\sigma$=0.33 |
| level$_d$ | Discipline level | double | ($\mu$−2$\sigma$, $\mu$+2$\sigma$), $\mu$=0.5, $\sigma$=0.33 |

**Algorithm 7 Affecter algorithm**

**Input:** message //Received message

advisorList

**Output:** Node //A node in a pattern

**if** angle$_{\text{message}}$ **in** angle$_{\text{monitoring}}$ **then**

node=silent;

**return** Node;

**end if**

{sug} = advisorList (sug); //Obtain sug from all advisors

**foreach** sug **in** {sug} **then**

**Switch**(sug) **then**

**case** forward: weight$_T$ ++;

**case** mass: weight$_B$ ++;

**case** silent: weight$_S$ ++;

**case** change$_M$: weight$_{CM}$ ++;

**case** change$_C$: weight$_{CC}$ ++;

**end foreach**

**return** Node = max(weight$_T$, weight$_B$, weight$_S$, weight$_{CM}$, weight$_{CC}$);

**Algorithm 8 Decider algorithm**

**Input:** Preferences

{mess [$i$]} //Stored information list

{message [$i$]} //Receiving information list

Level$_c$ //Confidence level

**Output:** Node //A node in a pattern

**foreach** $m_i$ **in** {message [$i$]} **then**

**if** ($m_i$ in {mess [$i$]}) **then**

//calculate intensity$_{m_{i+1}}$

intensity$_{m_{i+1}}$ = intensity$_{\text{mess}[i]}$ × Level$_c$ + intensity$_{m_i}$ × (1 − Level$_c$);

**else then**

intensity$_{m_{i+1}}$ = intensity$_{m_i}$;

**end if**

**end foreach**

**foreach** $p_i$ **in** {Preferences} **then**

Set indicator ∈ [−1,1];

**if**($m_{i+1}$ **in** {Preferences}) **then**

Proj = $m_{i+1}/p_i$;

**end if**

**end foreach**

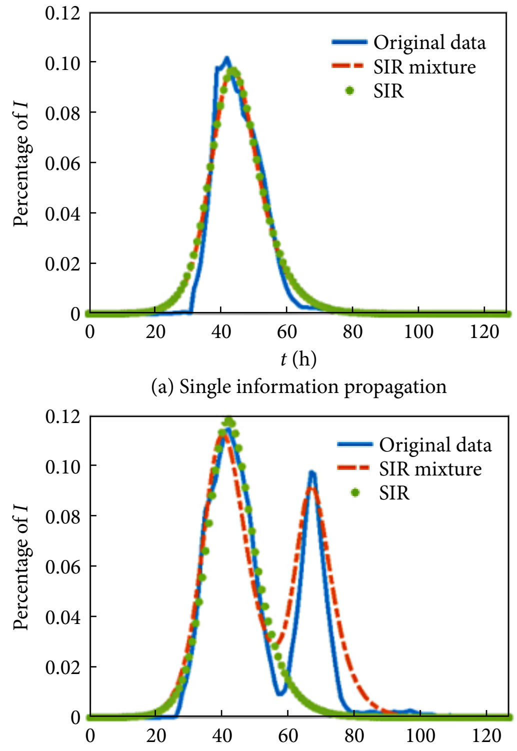
According to the Proj choose the Node;

**return** Node;

## 4 Conclusion

As one of the main models of the future economy and society, crowd network based systems are characterized by large scale, openness, self-organization, and evolution, which are also the challenges and difficulties faced by simulation. Accordingly, this paper proposes a simulation-oriented general model of digital selves in a crowd network, which consists of six parts, namely, pattern, affecter, decider, executor, connector, and monitor. Based on this model, experiments were conducted to verify its feasibility and applicability in three scenarios (coevolutions, ecological structure evolution of the e-commerce market, and multi-information coevolution spreading).

In these scenarios, the simulation results are fully supported by other research works. For example, the "coevolutions" experiment fits the results of the comparison method by computing the crowd intelligence level and redrawing its changes. The "ecological structure evolution of the e-commerce market" experiment recovers evolution trends and discovers some new characteristics of the structure evolution of the e-commerce market. The "multi-information coevolution spreading" experiment reappears the inhibitory effects from new information to spreading ones. Because all the simulations adopt the proposed model as the basic simulation model, the feasibility and applicability of the proposed model are well verified.




(a) Single information propagation

(b) Multi-information propagation

**Fig. 12 MSIR model validation results.**

## Acknowledgment

This work was partially supported by the National Key R&D Program of China (No. 2021YFE0111600).

## Dates

Received: 30 December 2022; Revised: 2 March 2023; Accepted: 3 March 2023

## References

[1] A. W. Woolley, I. Aggarwal, and T. W. Malone, Collective intelligence and group performance, *Curr. Dir. Psychol. Sci.*, vol. 24, no. 6, pp. 420–424, 2015.

[2] J. M. Leimeister, Collective intelligence, *Bus. Inf. Syst. Eng.*, vol. 2, no. 4, pp. 245–248, 2010.

[3] Y. Chai, C. Miao, B. Sun, Y. Zhang, and Q. Li, Crowd science and engineering: Concept and research framework, *International Journal of Crowd Science*, vol. 1, no. 1, pp. 2–8, 2017.

[4] H. Sun and M. Zhang, A reflective memory based framework for crowd network simulations, *International Journal of Crowd Science*, vol. 2, no. 1, pp. 74–84, 2018.

[5] S. Wang, L. Cui, L. Liu, X. Lu, and Q. Li, Projecting real world into crowdintell network: A methodology, *International Journal of Crowd Science*, vol. 3, no. 2, pp. 138–154, 2019.

[6] Z. Mao, L. Zhou, and Y. Chen, A satellite communication simulation system research based on HLA and MDIS, in *Proc. 4th Int. Conf. Computer Science and Application Engineering*, Sanya, China, 2020, pp. 1–6.

[7] R. Jin, Q. Zhao, X. Teng, and M. Liu, Research on simulation technology of shipborne satellite communication system, *Journal of Physics*: *Conference Series*, vol. 1650, no. 2, p. 022003, 2020.

[8] Z. Mao, Y. Chen, and L. Zhou, A multidimensional interactive techniques for satellite communication simulation application, in *Proc. 4th Int. Conf. Computer Science and Application Engineering*, Sanya, China, 2020, pp. 1–5.

[9] C. N. Kang and C. D. Zhang, Research on artillery element combat simulation system based on HLA, (in Chinese), *Computer Engineering and Design*, vol. 32, no. 6, pp. 2100–2103, 2011.

[10] M. Z. Yang and J. Yin, Study of an air-to-ground missile countermine simulation system, *Acta Armamentarii*, vol. 28, no. 5, pp. 576–580, 2007.

[11] Y. Liu and A. Zhang, Multi-agent system and its application in combat simulation, in *Proc. 2008 Int. Symp. on Computational Intelligence and Design*, Wuhan, China, 2008, pp. 448–452.

[12] S. Zhang, X. Yang, Y. Jiang, and H. J. Liu, Research on the multi-agent military logistics simulation system based on HLA, *Command Control and Simulation*, vol. 40, no. 4, pp. 91–94, 2018.