E-Commerce Autocorrelation Decision Model Based on Defect Matrix Constraints

Juan Xi  
School of Foreign Languages, Handan University, Handan 056005, Hebei, China

Haojie Xia  
School of Mechanical and Electronic Engineering, Handan University, Handan 056005, Hebei, China

Jun Yang  
School of Foreign Languages, Handan University, Handan 056005, Hebei, China

Abstract  
Decision delay is an important index of e-business decision-making. Given a decision node caching and the number of decision-making information copies, how to select the appropriate decision node to support decision information becomes a key problem. In this paper, E-Commerce autocorrelation Decision based on Defect Matrix Constraint Method (ECDDMC) is proposed. ECDDMC considers the average time of encounter between the decision node and target decision node in each time slot as a random variable. Based on the statistical properties of the random variable, ECDDMC obtains a rule to stop observation and support the decision information. A simple threshold structure is presented in which the decision support information is supported when the average time of encounter between the decision node and target decision node on a certain time slot is less than a given threshold, ECDDMC can tradeoff between the smaller encounter interval and the waiting cost, the decision-making delay of the minimum decision information in the sense of mathematical expectation is introduced. The ECDDMC's e-commerce environment model, the proof process of existence of the defect matrix constraint and the calculation method are introduced. The simulation experiment results show that ECDDMC is superior to other methods in decision-making success rate and decision delay and so has obvious advantages.

Keywords: E-commerce Decision-Making, Decision Algorithm, Optimal Stop, Decision Delay, Decision-Making Success Rate.

1. INTRODUCTION

E-commerce decision-making (opportunistic network) is a decision-making process simulation for the decision-making node motion (Chen, Rungruengsamrit and Rajkumar, 2013). In the segmented e-commerce environment, decision-making process of information decision is transmitted. In e-business decision, if the current decision node and decision table have not next-hop decision node to the target decision node, and cache the decision information and find the appropriate forwarding opportunity with the movement of the decision node. The traditional e-business environment decision becomes unstable or even invalid in the e-business environment. Therefore, the research on mobility modeling and decision - making of decision nodes becomes a hot issue in e - business decision - making.

For example, the decision node carrying decision information in literature (Hamner, Seth and Steele, 2013) supports decision information once every time it encounters a new decision node, which is similar to that of decision tree in decision tree. Flooding the forwarding mode obviously leads to huge cache and communication overhead. To this end, the paper (Nikishin and Pankina, 2015) based on the concept of votes (tickets) control e-commerce environment, the number of copies of decision information, that is, decision-making information dissemination stage, which supports decision-making nodes within the number of votes, and then enters the waiting stage, waiting for the decision node to meet the target and directly transmitting the decision information. This blindness in the decision-making information dissemination stage restricts the performance of e-commerce environment. Introduces a probabilistic prediction mechanism to evaluate the forwarding effectiveness of the decision node, and only forwards the decision information to the decision node with better forwarding efficiency. The redundancy mechanism of these defective matrix constraining methods often leads to large cache cost. Cache management mechanism of e-business decision which can be used to delete replicas of decision information which have been confirmed or lower priority to meet the cache constraint. Overall, the redundancy mechanism can improve the decision performance to a certain extent, but it is limited by the decision space and...
decision time constraint, too many copies of the number of decision-making information but led to longer queuing time and more frequent decision-making information deletion operation, which led to decision-making information decision-making delay increased and the decision-making success rate and other issues.

Under the condition of limiting the number of copies of decision information, the higher the forwarding efficiency of the decision node carrying the decision information, the better the performance of e-commerce environment. Therefore, the core problem of defect matrix constraint is how to select the appropriate decision node support Decision-making information, which is a decision-making problem that needs to use the knowledge to optimize the calculation. Using Spray and Wait as an example, the algorithm that supports decision information to the decision node by time is a greedy algorithm that considers only the encounter time interval Local optimization algorithm: It is possible to miss decision nodes that are encountered later but with higher forwarding efficiency because of the limitation of the number of replicas. Aiming at this problem, this paper proposes a new method for defective matrix constraint decision making based on optimal stopping method algorithm to tradeoff between latency and forwarding effectiveness to optimize the overall delay of decision information decisions.

Defect matrix constraint algorithm can be divided into several categories: one is a simple flooding methods, such as the literature (Hamner, Seth and Steele, 2013; Nikishin and Pankina, 2015; Kenneth and Rebecca, 2012) only the implementation of blind support without choice, too much decision information copy of the decision-making algorithm leads to such e-commerce environment performance. Such as literature (Kemény and Simón, 2016) and (Lee and Lee, 2015) when the decision node in the two meet the time to update the probability of meeting between the two decision nodes, and then predict the decision-making node of the forward. The probability of encounter is the key technique, OPF is based on the hypothesis that the probability of decision node obey the exponential distribution (Aubry, De Maio and Pallotta, 2012), but (Kimura, Kudo and Tanaka, 2016) the tracking result of the real decision node shows that the probability of meeting between decision nodes is more and the results of the test bed show that the probability of encounter is affected by many factors (Aubry and De Maio and Pallotta, 2012), which leads to the difficulty of predicting the probability of encounter and the accuracy is low. Therefore, the probability-based forecasting method is not practical. In order to build a community, Bubble detects the length of contact time between decision nodes, but does not evaluate the weight of the contact and data interaction. SDM (Nakamura, Kitaoka and Akiyama, 2012) and (Beyer and Finck, 2012) However, we know that the decision-making nodes of the taxi-like decision-making nodes do not have obvious community attributes. Therefore, the decision-making method that depends on the social attribute can be invalidated in some cases.

Similar to our working methods (Lee and Hyun and Kim, 2016; Jahanshahi, Rezaei and Nawaser, 2012), also use the optimization theory to select the decision node of the next-hop decision-making through the establishment of a mathematical model (Lee, Hyun and Kim, 2016), using the optimal control theory to decide when to forward the decision information. In (Jahanshahi, Rezaei and Nawaser, 2012), we use the classic secretary problem (Yao, Ruohomaa and Xu, 2012) to select the optimal forwarding decision node. However, theorem 2.1 of the literature (Yao, Ruohomaa and Xu, 2012) points out that the probability of choosing the optimal decision node is e^{-1}. In contrast, the ECDDMC (Yao, Ruohomaa and Xu, 2012) this problem is avoided by using the average encounter time as the optimization goal.

This paper first introduces the e-commerce environment model and decision algorithm framework used in e-commerce decision-making, and then introduces the forwarding decision algorithm based on defect matrix constraints.

2. E-COMMERCE DECISION-MAKING MODER AND DECISION-MAKING ALGORITHM FRAMEWORK

2.1. E-commerce Decision-Making Model

All NUM decision nodes that move in a limited area form a decision node set V. Each decision node v \in V is equipped with an omnidirectional transmitting and receiving antenna, and the communication distance is symmetric between decision and decision nodes. Decision nodes communicate decision information only when they meet each other. All decision nodes have the same communication bandwidth and all decision nodes can generate decision information to the given decision node. All the decision information has the same size, and the decision node holding the decision information stores the decision information in the cache. The cache size of all decision nodes is finite and equal. It is assumed that the E-business environment runs in time units with length T. The decision nodes idle in the decision-making node announce the configuration information of the decision node in each time slot and detect the neighbor decision nodes. The decision information of the two decision-making nodes can fulfill all the decision-making information supporting work in one meeting time period. Every entity of decisive information has the remaining decision number H and the remaining life time TTL two parameters. When the decision information is supported between the two decision-making nodes, the decision information of the decision information is reduced by 1 on both decision nodes, and the decision
information of H decreasing to 0 can only be supported by the target decision node. The TTL value decreases with the time slot, when TTL decreases to 0, the decision node holding the decision information deletes it from the cache.

2.2. Decision Algorithm Framework

Decision node j ∈ V checks the decision-making channel in every time slot. If the decision-making channel is idle, it transmits advertisement decision information which includes the number of decision nodes i and the average time-to-encounter vector of the decision node to all other decision nodes \( \bar{T}_i = (t_{i \rightarrow 1}, t_{i \rightarrow 2}, \ldots, t_{i \rightarrow \text{NUM}}) \), which \( t_{i \rightarrow d} (d \in V) \) represents the average time interval between decision nodes i and d which are decision nodes maintained by the node i. If the decision node j receives the decision information of the decision node i successfully, it first updates the decision node distance matrix \( \bar{T}_{all} \). \( \bar{T}_{all} \) is a matrix of NUM × NUM size, the average time between each decision node and other decision nodes is recorded. The decision node j then puts \( \bar{T}_i \) the column vector as a column vector into the matrix \( \bar{M}_j \). Each column of \( \bar{M}_j \) the decision node j corresponds to the average encounter time vector received by the decision node j over a time slot.

Obviously, the d-th row (represented by vectors \( \bar{M}_{j,d} = (t_{i \rightarrow d}, t_{2 \rightarrow d}, \ldots) \)) is the average time \( t_{i \rightarrow d} \) between the decision node i and the decision node d encountered by a decision node j over a series of successive time slots. This series of decision knowledge \( t_{i \rightarrow d} \) will be used for judging the forwarding effect of the encountered decision node to the target decision node d. Figure 1 is the data structure that the decision node j maintains.

\[
\begin{bmatrix}
0 & t_{1 \rightarrow 2} & \cdots & t_{1 \rightarrow \text{NUM}} \\
t_{2 \rightarrow 1} & 0 & \cdots & t_{2 \rightarrow \text{NUM}} \\
\vdots & \vdots & \ddots & \vdots \\
t_{\text{NUM} \rightarrow 1} & t_{\text{NUM} \rightarrow 2} & \cdots & 0
\end{bmatrix}
\]

(a) The decision node distance matrix

\[
\begin{bmatrix}
0 & t_{2 \rightarrow 1} & \cdots \\
t_{1 \rightarrow 2} & 0 & \cdots \\
\vdots & \vdots & \ddots \\
t_{1 \rightarrow \text{NUM}} & t_{2 \rightarrow \text{NUM}} & \cdots
\end{bmatrix}
\]

(b) the matrix of observed values of \( T_d \)

Figure 1. The data structure maintained by decision node j

Decision node j receives decision information of decision node i in a certain time slot and updates the corresponding data structure, and then decides which decision information in cache will be supported to decision node i according to the autocorrelation decision method. If the decision node j decision information to support the decision number of the remaining decision-making node i, then i and j in the decision-making information of the remaining decision-making will become H - 1. When H is 0, the decision information can only support the purpose of decision-making node d. Therefore, the maximum support number of each entity of decision-making information in the e-commerce environment is 2H.

3. AUTOCORRELATION DECISION-MAKING METHOD BASED ON DEFECT MATRIX CONSTRAINT

3.1. Minimum Expected Delay Decision Problem

In e-commerce decision-making, given the decision-making overhead limit, the decision-making protocol pursues a higher success rate and a lower average decision delay. Generally speaking, the higher the decision-making delay is, the lower the success rate is. The longer the existence of the environment is, the greater the likelihood that the decision node will be removed from the cache. In addition, considering the high complexity of multi-objective optimization, this paper reduces decision delay as the optimization goal of autocorrelation decision.

As described above, decision node j receives the decision information of decision node i in the N-th time slot and then determines whether to support the decision information of decision node d to decision node i according to the observation value in current time slot \( t_{i \rightarrow d} \). There are two possibilities: First, decision nodes x in the follow-up timeslots \( t_{x \rightarrow d} \) are reported to be larger, then the decision information is supported to the decision node i can get lower decision delay; Secondly, Decision node y encountered at the N+1-th time slot, so \( t_{y \rightarrow d} + T < t_{i \rightarrow d} \) that the decision node y can be delivered to the decision node with the smaller delay, but may
be due to the number of copies of the constraints cannot be supported again to the decision node \( y \). Therefore, the decision node \( j \) need to select the appropriate time slot to support decision-making information in order to obtain a smaller decision delay, the problem can be modeled as an objective to minimize the delay to optimize the target problem.

Since the decision information is supported among multiple decision nodes, the decision node which holds the decision information \( m \) in some time slot constitutes the set \( V_m \). The autocorrelation decision goal of the decision node is to select \( m \) in time slot and the sequence of random variables \( X_T \).

For the revenue function (3), according to the expectation maximization algorithm, we use the defect matrix constraint to prove Proposition 1. The MED (minimum expected delay) is a known quantity. Here, for convenience of calculation, we stipulate that if there is no decision node in an observation time slot, we assume that the joint distribution and the income function of the sequence are independent and identically distributed.

Among them, the average time of encountering decision nodes \( i \) and goal decision node \( d \) in each time slot and the sequence of random variables \( X_T = T_{i \rightarrow d} - T_{i \rightarrow j} \) is independent and identically distributed.

Then for the MED problem, the defect matrix constraint is to select a moment \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \) so that:

\[
X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \tag{1}
\]

or:

\[
X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \tag{2}
\]

Among them,

\[
X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \tag{3}
\]

It means the income function of the MED problem. Under Hypothesis 1, we also assume that \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \) obey the independent identical distribution.

According to the above definition, the average encounter time \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \) of decision nodes \( i \) to \( d \) is a random variable, and \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \) is a known quantity. Here, for convenience of calculation, we stipulate that if there is no decision node in an observation time slot, we consider that the decision node observes itself. So random variable \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \) at the same time, because of the sparse e-commerce environment, we assume that there is at most one decision node for each observation. If more than one decision node is observed at the same time, a smaller decision node in \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \) will be taken as the decision node.

Proposition 1.

(1) The MED problem has a stop rule \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \), and:

\[
X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \tag{4}
\]

(2) \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \) is the solution of the following equation:

\[
X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \tag{5}
\]

[.] In this case, only the positive part is taken.

Proposition 1 show that as long as the observed value satisfies a certain threshold, the decision information can be supported to the currently encountered decision node.

3.3. Demonstration and Solution of the Existence of Defective Matrix Constraints

We use the defect matrix constraint to prove Proposition 1. For the revenue function (3), according to the e-commerce decision model described in section 1.1, we can know that any decision node needs to be probed at least once to support the decision information, so \( N \) is a non-zero natural number, that is \( X_N = T_{i \rightarrow d} - T_{i \rightarrow j} \), and then we use the general solution method to find the stop rule:

Step 1, proving the existence of a defect matrix constraint.

According to theorem 3.1 in (Yao and Ruohomaa and Xu, 2012), the defect matrix constraint exists when the following two conditions are satisfied:
The operation process of ECDDMC is as follows:

1. MED problem, we design and implement an Electronic commerce decisions based on defect matrix constraint
2. Step 1: Solve the integral equation
3. Assumptions and Rationality Analysis of Defect Matrix Constraints
4. According to the solution method of Problem 4.1 and theorem 3.2 in the paper (Yao and Ruohomaa and Xu, 2012), we can know that for the income function of the MED problem $X_N = T_{t+sd} - T_{t+ad}$, if the distribution is independent of time, the defect matrix constraint presents a threshold structure. That is, suppose $X_N = T_{t+sd} - T_{t+ad}$ represent the expected return of the defect matrix constraint, then $X_N = T_{t+sd} - T_{t+ad}$ the decision node needs to continue to observe and that $X_N = T_{t+sd} - T_{t+ad}$ the decision node should stop watching and support the decision information. With the assumption 1, then the stop rule becomes:

$$X_N = T_{t+sd} - T_{t+ad}$$

According to the optimization formula of theorem 3.1 in (Yao and Ruohomaa and Xu, 2012), the stopping rule $X_N = T_{t+sd} - T_{t+ad}$ is solved by

$$X_N = T_{t+sd} - T_{t+ad}$$

Make:

$$X_N = T_{t+sd} - T_{t+ad}$$

Combining equations (8) and (9) yields:

$$X_N = T_{t+sd} - T_{t+ad}$$

Where, F represents the distribution of $X_N = T_{t+sd} - T_{t+ad}$.

For discrete random variables $X_N = T_{t+sd} - T_{t+ad}$, we have:

$$X_N = T_{t+sd} - T_{t+ad}$$

In summary, proposition 1 was proved.

The calculation method for the defect matrix constraint threshold.

There are two ways to calculate the threshold $X_N = T_{t+sd} - T_{t+ad}$: The first method is to consider $X_N = T_{t+sd} - T_{t+ad}$ the discrete variables, all possible $X_N = T_{t+sd} - T_{t+ad}$ are arranged in descending order, and then use a linear search to search $X_N = T_{t+sd} - T_{t+ad}$ to meet the formula (11) value, the complexity of this calculation method $O(n)$, n is the number of decision nodes. Another method which $X_N = T_{t+sd} - T_{t+ad}$ is regarded as a continuous variable, is to consider the distribution F of $X_N = T_{t+sd} - T_{t+ad}$, through the observed value, and then solve the integral equation (9) in this paper, it is calculated according to the former way.

3.4. Assumptions and Rationality Analysis of Defect Matrix Constraints

In solving the MED problem according to the defect matrix constraint, we need to satisfy the following assumptions: the random variables $X_N = T_{t+sd} - T_{t+ad}$ the random variables in each slot $T_{t+sd}$ are independent and identically distributed. The degree of satisfaction of this condition is related to the granularity of time slot and decision node motion model. If the decision node's motion area is homogeneous, the distributions of the random variables $V_n$ on the respective time slots are converged.(there is no similar regional difference between city and suburb) If the decision node's motion range is wide and the granularity of the time slot is large, then the correlation is small in each time slot. For the sake of simplicity, we consider the factors such as the granularity of the time slot, the moving region and so on in setting the operating parameters of the algorithm to satisfy the assumption of independent and identically distributed random variables $V_n$. In practical application, we can improve the satisfaction degree of the hypothesis by sub-region modeling, time-slot granularity selection and so on. We will explain the satisfaction of the hypothesis conditions in the follow-up experimental evaluation.

3.5. Supporting process Analysis Of Decision Information

Based on the decision-making framework and the defect matrix constraint calculation method for solving MED problem, we design and implement an Electronic commerce decisions based on defect matrix constraint (ECDDMC). The operation process of ECDDMC is as follows:
As shown in Figure 2, the initial decision number $H$ of the decision information is 2, which $V_m$ represents the set of decision nodes that holds the decision information. The process of forwarding is: the source decision node $a$ holds the decision information $H = 2$, $V_m = \{a\}$ at the initial time; Then, when the decision node $a$ meets the decision node $b$, according to the judging result of the formula (4), and the decision node $a$ sends the decision information to the decision node $b$. $V_m = \{a, b\}$, $H = 1$, the results shown in Figure 2. (a), then when the decision node $b$ meets the decision node $c$, $b$ is still based on the defect matrix constraint to determine whether the need to support to $c$, the results shown in Figure 2. (b) Note, here, the decision node $b$ is knowledgeable $V_m = \{a, b\}$, can be calculated in accordance with the formula (4).

![Figure 2. The decision information support process of 2 hop](image)

As shown in Fig.3, suppose the initial residual decision number $H$ of decision information is 3. After two support, $H = 1$, $c$ decision node encounters decision node $y$ in a certain time slot, as shown in Figure 3 (a). In this case, it is possible that decision node $c$ knows the set of decision nodes holding decision information $V_m' = \{a, b, c\}$, but in fact $V_m = \{a, b, c, x\}$, since decision node $c$ only has local imprecise knowledge, the stopping rule is calculated according to formula (4) (If $T_{a \rightarrow d} < T_{a \rightarrow d}$, decision node $c$ does not need to support the decision information to decision node $y$), this deviation increases with the number of decisions, as shown in Figure 3 (b), and the number difference of elements $V_m'$ and $V_m$ is two.

In general, the optimal stopping time of observation process can be calculated exactly according to formula (4) under the condition of initial residual decision number $H = 2$. But this also limits the maximum number of copies of decision information in e-business environment. The results of the defect matrix constraint may deviate from the theoretical optimal value due to a lack of global knowledge $V_m$ when $H > 2$. We will evaluate the effect of the $V_m = \{a, b, c, x\}$ approximation by $V_m' = \{a, b, c\}$ simulation experiments.

![Figure 3. The decision information support process of 2 hop](image)
4. SIMULATION IMPLEMENTATION AND PERFORMANCE EVALUATION

4.1. Scene and Parameter Settings

In this paper, simulation implementation and performance evaluation of ECDDMC is carried out under ONE (opportunistic network environment) (Molzahn, Holzer and Lesieutre, 2013), which is an e-commerce environment simulation platform specially developed for DTN e-business environment and e-commerce decision making. In order to better simulate the actual electronic business decision-making, we import the data set of the tracked taxi tracks provided by the Cabspotting project (Beyer and Finck, 2012) in ONE, and then simulate the movement of the decision nodes, which records the trajectory of 500 decision nodes, Decision nodes record their location information every 60 seconds, and the data acquisition time is 30 days. Because it is not meaningful to repeat the full-length experiments on the same dataset, we need to split the entire data set into multiple segments so as to simulate a number of random experiments. Other analog parameters set in Table 1.

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Macro-Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of node</td>
<td>500</td>
</tr>
<tr>
<td>Transmit range (m)</td>
<td>30</td>
</tr>
<tr>
<td>Transmit speed (MBps)</td>
<td>10</td>
</tr>
<tr>
<td>Buffer size (MB)</td>
<td>5–50</td>
</tr>
<tr>
<td>Message size (KB)</td>
<td>500–1024</td>
</tr>
<tr>
<td>Message creation interval (s)</td>
<td>30</td>
</tr>
<tr>
<td>Slot time (s)</td>
<td>60</td>
</tr>
<tr>
<td>Message time to live (h)</td>
<td>333</td>
</tr>
<tr>
<td>Simulation time (h)</td>
<td>333</td>
</tr>
</tbody>
</table>

We compare the Epidemic, Spray and Wait Prophet algorithms implemented in ECDDMC and ONE simulation tools, including transfer success rate, forwarding cost and average decision delay, among which the success rate is the number of successful decision messages; The cost of forwarding is the ratio of the total number of replicas of all decision information to the number of successfully transmitted decision messages and the number of successfully transmitted decision messages; decision delay is the ratio of the total number of decision information from the source decision node The time to send to the destination decision node.

4.2. Validate the Independent and Identically Distributed Assumption of Random Variables over Time Slots

First, we test hypothesis 1 by experiments, that is, through several random experiments to observe the distribution of observations on each time slot to determine the independent assumption of the same distribution of the situation to meet the experimental set time slot size of 60s, and performance evaluation The size of the entire mobile data set is divided into 10000 segments, each section of 20 timeslots, that 10000 repeated experiments. Figure 4 shows the first decision node in the first three time slots on the observation value distribution, can it is seen that the distribution of the three time slots is approximately the same - the three curves are basically similar.

![Figure 4](image1.png)  
**Figure 4.** Distribution of $T_{\text{id}}$ over time slots

![Figure 5](image2.png)  
**Figure 5.** Correlation of $T_{\text{id}}$ over successive time slots

According to the probability theory, we can know that the independence and non-correlation of X and Y are equivalent when the joint distribution of two-dimensional random variables (X, Y) is two-dimensional normal distribution, so we can measure their independence by using two random variable correlations. Figure 5 shows the influence of the length of the slot on the correlation coefficient of the observed sequence of random
variables $T_{r,s}$ on two successive time slots. It can be seen that when the slot length is 60, the correlation coefficient between successive observation $T_{r,s}$ slots is less than 0.1, and the correlation coefficient decreases with the increase of the slot length. When the length of the slot exceeds 120s, the correlation coefficient becomes very low, which is close to zero.

4.3. Comparison of Performance under Global and Local Information Conditions

As mentioned above, decision nodes can only support and store unnecessary information based on local information, which may lead to the formation of sub-optimal path in the e-commerce environment, in addition to the theoretical optimal path, which may result in the storage and transmission of resources. Figure 6 assesses the salience of the problem, where the abscissa represents the initial residual decision limit of the decision information, that is, the number of copies, where the transfer cost is defined as (number of replicas - number of decision messages) / number of decision messages. The results show that there are no significant differences in performance between the three indicators of latency, success rate and transfer cost. The local information will lead to the deviation of the constraint matrix calculation results, but the degree of deviation is restricted by two factors: the actual The number of decision-making information forwarding decisions and the average time of encounter of the decision nodes. Firstly, the number of decision-making is limited by the factors such as life cycle of decision information, decision space and scale of E-business environment etc. Experimental results show that, If the decision number of the decision information exceeds 4 hops, the number of decision information replicas, the transmission and storage load of the whole e-business environment will increase significantly, which leads to the increase of decision information delivery cost and the decrease of success rate. When the total decision number is small, the difference between the local information set and the global information set is finite. Then, due to the near normal distribution $T_{r,s}$, the computation of the defect matrix constraint is to make the value $E\left(\min_{s_{r,s},|T_{r,s}|} \{T_{r,s}\}\right)$ decrease after adding the new decision node. With the increase of the decision number, the probability of encountering a better decision node is reduced, so the number of suboptimal paths is also limited, and the cost of suboptimal path is not completely wasted considering the stochastic nature of the optimization method, so the decision delay can be reduced. Local information approximation of global information on the overall performance of e-commerce environment is relatively small.

![Figure 6. Comparison of performance under global and local information conditions](image)
4.4. Effect of Slot Length on ECDDMC Performance

Figure 7 shows the effect of slot length on the performance of ECDDMC. It can be seen that because the smaller time slot means that the autocorrelation decision is more sensitive, the three indexes of the delay, success rate and overhead are all better with the time slot length reduced. But taking into account the following factors, we select the 60s as the official performance evaluation of the time slot length: the data set acquisition cycle is 60s; smaller time slots lead to auto-decision-making frequency increases, the calculation of cost increases; the frequency of smaller time slots encountering 0 decision nodes is too high.

![Figure 7](image)

(a) delay of sending  (b) sending success rate  
(c) sending cost

**Figure 7.** ECDDMC performance comparison of different slot length

5. CONCLUSION

In this paper, an autocorrelation decision algorithm based on defect matrix constraints is proposed. This algorithm builds a delay-related payoff function according to the distribution of delay from decision node encountered to target decision node at each time slot, and computes the delay threshold corresponding to defect matrix constraint, when the delay from the decision node encountered to the target decision node exceeds the threshold, the support operation is carried out. This algorithm overcomes the blindness of supporting decision information, reduces the delay and improves the transmission success rate, and also controls the cost of e-commerce environment by limiting the number of copies and obtains a relatively good comprehensive performance. Identify the motion patterns of decision nodes on different geographic units, and more accurately fit of the distribution of random variables \( T_w \) is the focus of the follow-up work.

REFERENCES

