A method for layout planning of product web UI visual interface based on clone selection algorithm

Yanrong Wang*

Continuous Education College,
Jiaozuo University,
Jiaozuo 454000, Henan, China
Email: yanrongw@mls.sinanet.com
*Corresponding author

JinJin Chao

School of Information Engineering,
Jiaozuo University,
Jiaozuo 454000, Henan, China
Email: 372410479@qq.com

Xuemin Wang

Continuous Education College,
Jiaozuo University,
Jiaozuo 454000, Henan, China
Email: 407042469@qq.com

Abstract: In order to overcome the problems of long time and low recall rate in traditional product webpage UI visual interface layout planning methods, the paper proposes a product webpage UI visual interface layout planning method based on clone selection algorithm. Define the clone selection algorithm and determine the population evolution process of the interface layout planning. Based on this, determine the operation operator of the visual interface layout planning clone selection algorithm, and design the web UI visual interface layout planning algorithm flow to realise the visual interface layout planning. The experimental results show that when the number of product interfaces is 100, the number of extreme points of the planning function is 103, the interface layout planning time is only 7 s, and the data recall rate is as high as 96.5%. It can be shown that the method in this paper has a better visual interface layout planning effect.

Keywords: clonal selection algorithm; layout optimisation; local optimisation; visual interface.

Reference to this paper should be made as follows: Wang, Y., Chao, J. and Wang, X. (2023) 'A method for layout planning of product web UI visual interface based on clone selection algorithm', *Int. J. Product Development*, Vol. 27, Nos. 1/2, pp.1–14.

Biographical notes: Yanrong Wang graduated from Henan University of Technology majoring in computer science and technology. She is currently an experimentalist at Jiaozuo University. Her research interests include Internet of Things Technology and information security.

JinJin Chao received his MS degree in Signal and information processing from ZhengZhou University of Light Industry 2017. He is currently a teaching assistant at Jiaozuo University. His research interests include Computer and Communication Engineering.

Xuemin Wang received her MS degree in Computer Technology from East China University of Science and Technology in 2008. She is currently an associate Professor at Jiaozuo University. Her research interests include computer technology, information security and big data analysis.

1 Introduction

The product web site can effectively display a full range of products. The better the product web UI visual interface layout design effect, the better it can attract customers to purchase. How to enable users to obtain more product information in an effective time and attract more users Attention, web page layout planning technology came into being (Hung and Wang, 2020; Huang et al., 2019; Zhao and Lu, 2019). The core issue of product interface layout planning is page location design. The development of society has promoted the advancement of web page layout planning technology. The page layout planning technology at home and abroad evaluates the effect of page layout planning by product information search time and network data search recall rate (Oppenlaender et al., 2020; Wang et al., 2020). However, the following problems will arise under this evaluation standard: First, for most users who do not rely on product content, the retrieval efficiency of the visual interface has been improved, but most product web pages will be relatively single (Rantung et al., 2020). Second, different interfaces will have unique application scenarios (Ali and Kurniawan, 2019). So, what kind of visual parameters need to be designed for these different interfaces, and how will the planning effect be affected? Related scholars have conducted in-depth research on the layout planning of the product web UI visual interface, and have achieved certain research results.

In related research, Gao and Huang (2019) proposed a visual cognition human-computer interaction interface layout optimisation method, through the visual cognition theory to analyse the cognitive characteristics of the web interface, design page layout ideas according to different user behaviour habits, and extract at the same time The layout-related factors of the human-computer interaction interface use the user's cognitive load to realise the visual cognition of the interface layout and realise the layout planning method of the human-computer interaction interface. This method can improve the optimisation time of the interface layout, but there is a problem of not considering how to avoid the local optimal solution. Ji et al. (2020) proposed a method of interface information layout planning based on cognitive rules, matching the position of each interface information by geometric position, and realising the construction of interface information layout task model through cognitive factors, obtaining reasonable interface layout task information elements, and using particle swarm algorithm Solve the interface layout planning function, and find the optimal layout plan according to the inertial weight to realise the geometric position of the interface. This method can effectively improve the efficiency of interface layout planning, but the optimised web interface has the problem of low data recall rate. Wang and Zhang (2019) proposes a spatial interface planning method based on the perspective of metaphorical cognition, which uses metaphorical cognition methods to map the creation rules of spatial interfaces, and uses symbolic and philosophical features to realise the layout planning of product web UI visual interfaces. This method can effectively improve. After optimisation, the web interface data recall rate, but there is still a problem that the interface planning takes a long time.

In order to solve the problems of the above methods, a layout planning method of product web UI visual interface based on clone selection algorithm is proposed. The overall plan of this article is as follows:

1 In order to improve the data recall rate of the web interface after layout planning, define the visual interface layout planning problem, determine the population evolution process of the web UI visual interface layout plan, and complete the optimisation of the optimal position variable of the interface layout.

2 Based on the population evolution process, design the operation operator of the visual interface layout planning clone selection algorithm to obtain the optimal position node function, and reduce the number of optimal points of the web UI visual interface.

3 According to the results of the optimal planning location node function of the product page UI visual interface, the product page UI visual interface layout planning algorithm process based on the clone selection algorithm is proposed, and the optimal location point clone selection of the interface layout planning is obtained, which effectively solves the UI Visual interface layout planning takes a long time.

4 Design three experiments of web UI visual interface optimisation extreme points, layout optimisation time, web interface data recall rate, and compare and analyse the method of this article and the method of Gao and Huang (2019), Ji et al. (2020), and Wang and Zhang (2019) Interface layout planning effect.

2 Product web UI visual interface layout planning

In order to improve the actual effect of the product webpage UI visual interface layout planning, the clone selection algorithm is introduced. This algorithm can generate a new generation by mutation and de-cloning according to the generation group during the planning and development process, thereby expanding the search range and speeding up the convergence speed through clone selection It can be seen that the use of the clone selection algorithm can transform the UI visual interface layout planning problem into a clone selection problem, so as to improve the convergence of the interface layout planning (Choi et al., 2020; Hikmah et al., 2020).

2.1 Definition of UI visual interface layout planning problem

In the UI visual interface layout planning, the optimisation problem of choosing $X = \{x_1, x_2, \ldots, x_m\}$ as the optimal position variable of the interface layout is expressed by formula (1):

$$P = \max\left\{f\left(e^{-1}(A)\right) : A \int I\right\} \tag{1}$$

In the above formula, the string with a certain length limit for the interface layout is $A = a_1, a_2, \ldots, a_l$, which is the antibody code of the clone selection algorithm of the optimal position variable *X* in the layout (Maimunah et al., 2020; Rahi et al., 2020; Zheng and Ling, 2019; Oswal and Palmer, 2020), represented by $A = e(X)$; at the same time, *X* is called Is the decoding of antibody *A*, at this time $X = e^{-1}(A)$; the position data set *I* in the UI visual interface is also called antibody space (Biskjr et al., 2019), *f* is expressed as the real-valued position change function on the position data set *I* in the UI visual interface, and *f*. It is called the antibody-antigen affinity function.

For antibody *A* of the UI visual interface, the allele of the target planning position is denoted as a_j. In order to improve the effectiveness of the interface layout, the allele can be divided into *m* gene antibody paragraphs of length l_i ($l = \sum_{i=1}^{m} l_i$). Each gene antibody paragraph is represented by x_i, and $x_i \in [d_i, u_i]$, $i = 1, 2, \ldots, m$.

At this time, the binary code is used to express the interface position layout decoding (Sutabri et al., 2019) as:

$$x_i = d_i + \frac{u_i - d_i}{2^{l_i}}\left(\sum_{j=1}^{l_i} a_j 2^{j-1}\right) \tag{2}$$

In the above formula, d_i represents the bit weight in the binary code; u_i represents the bit weight in the interface layout position.

Obtain the antibody population space of the UI visual interface layout plan at this time, you can get:

$$I^n = \left\{A : A = [A_1 A_2 \ldots A_n], A_k \in I, 1 \le k \le n\right\} \tag{3}$$

In the above formula, *n* represents the size of the antibody population planned by the interface, and *n* is *A* positive integer. In this case, antibody group $A = [A_1 A_2 \ldots A_n]$ represents the N-tuple of antibody *A*, and I^n can be represented as a point in the antibody population space. At this time, the global optimal solution set of the interface layout plan is (Lower, 2020):

$$B^* = \left\{A \in I : f(A) = f^* \equiv \max\left(f(A') : A' \in I\right)\right\} \tag{4}$$

The neighbourhood set of the sub-optimal solution λ for the interface layout planning of antibody A_i can be defined as:

$$\xi^\lambda(A_i) = \left\{A_j \middle| 0 \le \left|A_i - A_j\right| \le \lambda, A_j \in I\right\} \tag{5}$$

In the above formula, $\overline{\xi^{\lambda}(A_i)}$ represents the number of $\xi^{\lambda}(A_i)$ elements; $|A_i - A_j|$ represents the optimal distance of the interface layout planning (Kemal and Nursetyo, 2020). Define the non-neighbourhood set of the suboptimal solution λ of the interface layout planning position as:

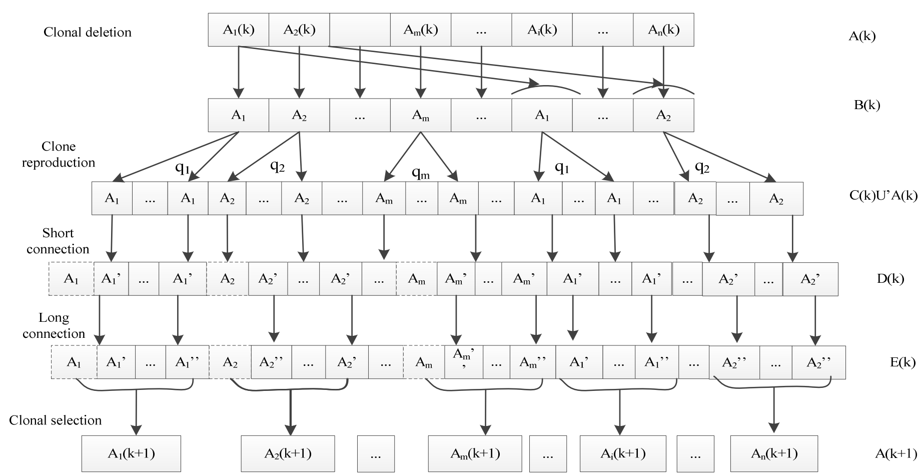
$$\overline{\xi^{\lambda}(A_i)} = \left\{ A_j \left| 0 < \lambda < \left| A_i - A_j \right|, A_j \in I \right. \right\} \tag{6}$$

Interface layout planning location population Under the influence of the clone selection algorithm, the population evolution process of UI visual interface layout planning is shown in formula (7):

$$A(k) \xrightarrow{T_d^c} B(k) \xrightarrow{T_c^c} C(k) \tilde{U} A(k) \xrightarrow{T_{sc}^c} D(k) \xrightarrow{T_{se}^c} E(k) \xrightarrow{T_s^c} A(k+1) \tag{7}$$

According to the above process, Figure 1 shows the population evolution process of UI visual interface layout planning.

Figure 1 Population evolution process of interface layout planning



2.2 *Confirmation of Operation Operator of Clonal Selection Algorithm*

In order to reduce the number of extreme points of the function and solve the problem of falling into the local optimal solution, the population evolution process of the interface layout planning is obtained according to Section 2.1, and the process of determining the interface layout operation operator is designed.

There will be multiple extreme points in the interface layout planning, that is, multiple points can meet the optimal position constraint requirements of the product interface layout planning, and it is easy to fall into the local optimisation problem. For this reason, it is necessary to delete redundant locations in the UI visual interface layout plan through clone delete (T_d^c) to improve the best effect of the layout plan. In this process, the concept of objective function is introduced. The principle of objective function is to express the required objective form through variables. In this article, the objective function is designed to obtain the cloned value-added antibody to obtain the optimal solution (King et al., 2020).

Now express the function set during the clone deletion operation as:

$$B(k)=T_d^c\left(A(k)\right)=\left\{T_d^c A_1(k),\ldots,T_d^c A_n(k)\right\}=\left\{\beta_1 A_1,\beta_2 A_2,\ldots,\beta_n A_n\right\} \tag{8}$$

In formula (8), clone deletion can plan the optimal position of the proliferation antibody and redundant antibody of the page layout. At this time:

$$\beta_1+\beta_2+\ldots+\beta_n=n \tag{9}$$

And the clone multiplication (T_c^c) in the interface layout plan can extract the sub-optimal position of the best position in the page layout rules, at this time:

$$C(k)=\mathrm{T}_c^c\left(B(k)\right)=\left[\mathrm{T}_c^c\left(B_1(k)\right),\ldots,\mathrm{T}_c^c\left(B_n(k)\right)\right]^T \tag{10}$$

$$C_i(k)=\mathrm{T}_c^c\left(B_i(k)\right)=I_i B_i(k),\quad i=1,2,\ldots,n \tag{11}$$

In the above formula, I_i represents the q_i dimensional row vector with element 1, which is the q_i clone of antibody B_i. Generally take:

$$q_i(k)=Int\left[N_c\times f\left(B_i(k)\right)/\sum_{j=1}^{n}f\left(B_j(k)\right)\right]\ i=1,2,\ldots,n \tag{12}$$

In the above formula, $N_c>n$ represents the set value related to the clone scale; $Int(\cdot)$ represents the round-up function. After cloning, the population becomes:

$$C(k)=\left\{C_1(k),C_2(k),\ldots,C_n(k)\right\} \tag{13}$$

In formula (13):

$$C_i(k)=\left[C_{ij}(k)\right]=B_{i1}(k),B_{i2}(k),\ldots,B_{in}(k) \tag{14}$$

$$C_{ij}(k)=B_{ij}(k)=B_j(k),j=1,2,..,q_i \tag{15}$$

The short connection T_{sc}^c is to clone and multiply q_i antibodies according to a point in the $f(\cdot)$ space of the objective function:

$$C_{ij}(k)=B_{ij}(k)=B_j(k),j=1,2,\ldots,q_i,C_i(k)\in C(k) \tag{16}$$

Among them, $\xi^\lambda(C_i)$ produces a new antibody $D_i(k)\in D(k)$, and transmits the information from $C_i(k)$ to $D_i(k)$, the node closest to the target in $\xi^\lambda(C_i)$, and λ is relatively small. Which is:

$$D(k)=T_{sc}^c\left(C(k)\right)=\left[T_{sc}^c\left(\left(C_1 k\right)\right),\ldots,T_{sc}^c\left(\left(C_n k\right)\right)\right]^T \tag{17}$$

In formula (17):

$$T_{sc}^{c}\left(\left(C_1 k\right)\right)=C_i \propto \left\{q_{c1}^{\lambda}, q_{c2}^{\lambda}, \ldots, q_{cm}^{\lambda}\right\} \tag{18}$$

In the above formula, $q_{cm}^{\lambda}=\{1,0\}^{n}$ represents the column vector with the same dimension as C_i, and its element value is $\left|q_{cm}^{\lambda}\right| \leq \lambda$ with the number of 1 to ensure that D_i is in the C_i neighbourhood, and $\propto$ represents the "exclusive OR" bit operation of matrix elements, and the short connection is planned through the interface Later, the antibody population is:

$$D(k)=\left\{A(k), D_1(k), D_2(k), \ldots, D_n(k)\right\}, D_i(k)=A_i^{'}=\left\{A_{i1}^{'}, A_{i2}^{'}, \ldots, A_{iq_i}^{'}\right\} \tag{19}$$

The long connection T_{se}^{c} is to select a point in the non-neighbourhood $\overline{\xi^{\lambda}\left(D_i(K)\right)}$ of $D_i(k)$ according to the preset probability q_{Si} to generate the antibody $E_i(k) \in E(k)$:

$$E(k)=T_{se}^{c}\left(D(k)\right)=[T_{se}^{c}(D_1(k), T_{se}^{c}(D_2(k), \ldots, T_{se}^{c}\left(D_n(k)\right]^{T} \tag{20}$$

Clonal selection operation T_s^{c} is to select outstanding individuals from the progeny after the respective clones of antibodies are proliferated, short-linked, and mutated to produce new antibody groups. At this time, the interface layout operation operator of the antibody group is:

$$E(k)=\left\{\mathrm{A}(\mathrm{k}), E_1(k), \ldots, E_n(k)\right\} \tag{21}$$

At this point, the optimal location node function for the clone selection planning of the product page UI visual interface layout is:

$$A(k+1)=T_s^{c}\left(E(k)\right)=\max\left\{E_i(k)\right\}=\{E_{ij}(k) \middle| \max f\left(e^{-1}\left(E_{ij}\right)\right), j=1,2,\ldots,q_i\} \tag{22}$$

As a result, the optimal planning location node function of the UI visual interface of the product page is obtained, and the layout planning of the UI visual interface is realised.

2.3 The layout planning process of the product page UI visual interface based on the clone selection algorithm

In order to improve the efficiency of interface layout planning, the process of designing the product page UI visual interface layout planning algorithm of the clone selection algorithm is as follows:

In the first step, the product interface layout planning antibody group is initialised, the size of the interface planning population is *N*, and the maximum number of iterations in the product page UI visual interface planning process is $K_{\max}, q_{si}, \lambda, k=0$.

The second step is to initialise the optimal position of the product interface layout plan:

$$A(0)=\left\{A_1(0), A_2(0), \ldots, A_N(0) \in I^{N}\right\} \tag{23}$$

The third step is to calculate the affinity between the optimal location of the page layout plan and the location range:

$$A(0):\left\{f(e^{-1}\left(A_1(0)\right)),\ldots,fe^{-1}\left(A_N(0)\right)\right\} \tag{24}$$

The fourth step is to design and judge the loop condition. When $k = K_{\max}$ or $\left|f(e^{-1}\left(A^*\right)) - f^*\right| \leq \varepsilon$ is satisfied, the loop is ended; A^* is the optimal antibody position of the interface layout, and f^* is the optimal value of the objective function of the interface layout planning. When the function f^* satisfies the loop end condition, get the best position of the interface layout plan; otherwise, continue the loop;

The fifth step is to clone and delete multiple extreme points in the interface layout planning:

$$\begin{aligned} B(k) &= T_d^c\left(A(k)\right) = \left\{T_d^c\left(A_1(k)\right), T_d^c\left(A_2(k)\right),\ldots,T_d^c\left(A_n(k)\right)\right\} \\ &= \left\{\beta_1 A_1, \beta_2 A_2,\ldots,\beta_n A_n\right\} \end{aligned} \tag{25}$$

The sixth step is to clone and multiply the target location node in the interface layout plan:

$$C(k) = T_c^c\left(B(k)\right) = \left[T_c^c\left(B_1(k)\right), T_c^c\left(B_2(k)\right),\ldots,T_c^c\left(B_n(k)\right)\right]^T \tag{26}$$

The seventh step, the mutation planning operator in the interface layout space:

1 The short connection operator in the interface layout space is expressed as:

$$D(k) = T_{sc}^c\left(C(k)\right) = \left[T_{sc}^c\left(C_1(k)\right),\ldots,T_{sc}^c\left(C_n(k)\right)\right]^T \tag{27}$$

2 The long link operator in the interface layout space is expressed as:

$$E(k) = T_{se}^c\left(D(k)\right) = \left[T_{se}^c\left(D_1(k)\right),\ldots,T_{se}^c\left(D_n(k)\right)\right]^T \tag{28}$$

The eighth step, the product page UI visual interface layout plan the best location point clone selection:

$$A(k+1) = T_s^c\left(E(k)\right) = \max\left\{E_i(k)\right\} = \{E_{ij}(k) \mid \max f\left(e^{-1}\left(Eij\right)\right),\ j = 1,2,\ldots,\ qi\} \tag{29}$$

In the ninth step, the number of iterations is increased, $k = k+1$; continue the algorithm calculation, and go to step 4.

Based on the above steps, the design of the product page UI visual interface layout planning method based on the clone selection algorithm is completed. For the actual application effect of the method in this paper, it is necessary to design experiments to further verify.

3 Experimental research

In order to more comprehensively evaluate the product webpage UI visual interface layout planning effect of the method in this paper, the product webpage UI visual interface of a certain company is used as the experimental data, and the product webpage UI visual interface layout planning method based on the clone selection algorithm (the method in this article) is used as the experimental method. , Optimise the layout of human-computer interaction information interface design based on the theory of visual cognition (Gao and Huang (2019) method), the user interface information layout design method based on the cognitive law (Ji et al. (2020) method), and the space based on the perspective of metaphorical cognition The interface design research (the method of Wang and Zhang (2019)) is used as a comparison method for experimental verification. At the same time, in order to ensure the accuracy of the experimental results, experiments were carried out many times and the average value of the values was taken as the result, and the experimental environment under different methods was kept consistent.

3.1 Experimental indicators

This article designs three experimental indicators to verify the effects of different methods on the layout planning of the product web UI visual interface. The experimental indicators are:

1 *The number of extreme points for the optimisation of the planning function*

The smaller the number of optimal points in the planning function, the better the layout planning effect of the product webpage UI visual interface, effectively avoiding the problem of local optimisation, and improving the effectiveness of the product webpage layout planning. Conversely, the more extreme points in the planning function, the worse the layout planning effect of the product web UI visual interface, and the easier it is to fall into a local optimum.

2 *Time spent on layout planning*

This article uses the product webpage UI visual interface layout planning time to reflect the layout planning efficiency. The longer the product webpage UI visual interface layout planning time, the higher the interface layout planning efficiency, and the shorter the interface layout planning time, the lower the interface layout planning efficiency.

3 *Web page data recall rate*

This indicator can reflect the planned web application effect. The higher the recall rate of the product webpage data, the faster the response speed of the network data. Conversely, the lower the recall rate of the product webpage data indicates the slower the response speed of the network data.

The paper uses the above-mentioned experimental indicators as the test indicators for the layout planning effect of the product web UI visual interface, so as to analyse the comparative experimental results.

3.2 Analysis of the results of optimisation of the number of extreme points in the planning function

In order to verify that different methods are easy to fall into the local optimal solution problem for the product web UI visual interface planning function, the multi-extremum point function is used to optimise the number of extremum points to measure the Gao and

Huang (2019) method, Ji et al. (2020) method, Wang and Zhang (2019) method. The method and the interface planning effect of this method, the specific results are shown in Table 1.

Table 1 Number of optimisation extreme points of UI visual interface planning function

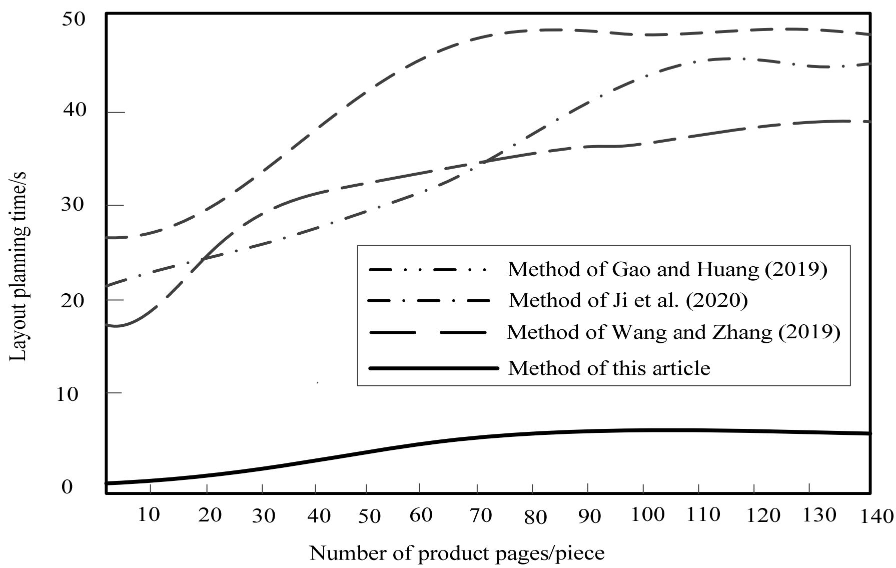
Number of pages/piece	*UI visual interface planning function to optimise the number of extreme points/a*			
	Methods in Gao and Huang (2019)	*Methods in Ji et al. (2020)*	*Methods in Wang and Zhang (2019)*	*Method of this article*
10	35	42	58	12
20	56	65	65	21
30	68	76	86	30
40	83	98	90	42
50	123	143	152	51
60	154	176	254	67
70	178	214	325	72
80	189	264	432	86
90	228	325	542	91
100	264	356	688	103
110	289	389	697	119
120	328	425	743	131
130	376	479	785	136
140	412	547	789	148

Analysing Table 1 shows that the number of optimisation extreme points of the UI visual interface planning function is different under different methods. When the number of web pages is 50, the number of extreme points for the optimisation of the UI visual interface planning function of Gao and Huang (2019) method is 123, and the number of extreme points of the visual interface planning function of Ji et al. (2020) method is 143. The number of extreme points of the visual interface planning function of the method is 152, and the number of extreme points of the visual interface planning function of the method in this paper is 51. When the number of web pages is 100, the number of extreme points for the optimisation of the UI visual interface planning function of the Gao and Huang (2019) method is 264, and the number of extreme points of the visual interface planning function of the Ji et al. (2020) method is 356. The number of extreme points of the visual interface planning function of the method is 688, and the number of extreme points of the visual interface planning function of the method in this paper is 103. According to the data in the above table, the number of extreme points in each interface of the method in this paper is much lower than that of the other three methods, generally one. However, the number of extreme points in each interface under other methods is obviously more. The reason for this result is that the method in this paper effectively solves the problem of easily falling into local optimum through clone proliferation and clone deletion, which reduces the number of optimisation extreme points to a certain extent.

3.3 Analysis of time-consuming results of layout planning

In order to verify the efficiency of the product webpage UI visual interface layout planning, Gao and Huang (2019) method, Ji et al. (2020) method, Wang and Zhang (2019) method, and the method of this paper are used to verify the time used for the page layout planning, and the results are shown in Figure 2.

Figure 2 Time change curve of product webpage UI visual interface layout planning



Analysing Figure 2 shows that the more the number of product web UI visual interfaces, the longer the interface layout planning will take. When the number of product interfaces is 40, the web UI visual interface layout planning of Gao and Huang (2019) method takes 39s, the web UI visual interface layout planning of Ji et al. (2020) method takes 28s, and the web UI visual of Wang and Zhang (2019) method The interface layout planning takes 30s, and the web UI visual interface layout planning of the method in this paper takes only 3s. When the number of product interfaces is 100, the web UI visual interface layout planning of Gao and Huang (2019) method takes 39s, the web UI visual interface layout planning of Ji et al. (2020) method takes 44s, and the web UI visual of Wang and Zhang (2019) method The interface layout planning takes 48s, and the web UI visual interface layout planning of the method in this paper takes only 7s. The interface layout planning of the method in this paper takes significantly less time than the other three methods, and the interface layout planning efficiency is higher. The reason for this result is that the method in this paper calculates the affinity between the optimal position of the page layout plan and the position range, and optimises the interface layout planning algorithm process through the clone deletion of multiple extreme points and the clone proliferation of the target position node. To a certain extent, the efficiency of UI visual interface layout planning has been improved.

3.4 Result analysis of web page data recall rate

In order to verify the effect of web page layout planning under the method of Gao and Huang (2019), method of Ji et al. (2020), method of Wang and Zhang (2019), and the method of this article, the results of obtaining the recall rate of UI visual interface webpage data are shown in Table 2.

Table 2 UI visual interface webpage data recall rate

Number of pages/piece	*Recall rate of web page data in UI visual interface/%*			
	Methods in Gao and Huang (2019)	*Methods in Ji et al. (2020)*	*Methods in Wang and Zhang (2019)*	*Method of this article*
10	88	66	78	97.9
20	86	69	76	98
30	75	84	81	99.3
40	79	76	80	96.5
50	73	72	69	99.6
60	75	80	64	97.9
70	81	82	72	93.6
80	80	58	77	98.8
90	69	67	68	98.3
100	73	66	75	96.5
110	76	65	83	97.4
120	70	68	86	98.2
130	82	73	88	97.4
140	85	79	84	96.8
mean value	78	71.8	77.2	97.6

Analysing Table 2 shows that the UI visual interface web page data recall rate is different when the number of web pages is different. When the number of web pages is 10, the UI visual interface web page data recall rate of the Gao and Huang (2019) method is 88%, the UI visual interface web page data recall rate of Ji et al. (2020) method is 88%, and the UI visual interface web page of Wang and Zhang (2019) method The data recall rate is 88%, and the UI visual interface web page data recall rate of this method is 88%. When the number of web pages is 50, the UI visual interface web page data recall rate of Gao and Huang (2019) method is 73%, the UI visual interface web page data recall rate of Ji et al. (2020) method is 72%, and the UI visual interface web page of Wang and Zhang (2019) method The data recall rate is 69%, and the UI visual interface web page data recall rate of this method is 99.6%. When the number of web pages is 120, the UI visual interface webpage data recall rate of Gao and Huang (2019) method is 70%, the UI visual interface web page data recall rate of Ji et al. (2020) method is 68%, and the UI visual interface web page of Wang and Zhang (2019) method The data recall rate is 86%, and the UI visual interface web page data recall rate of this method is as high as 98.2%. The web page data recall rate after the method planning in this paper is much higher than other methods, which shows that the method in this paper can effectively improve the web page data recall rate. The reason for this result is that the method in this paper redefines the layout planning problem of the visual interface, and converts the layout planning problem of the UI visual interface into a clone selection problem, thereby optimising the optimal position variable of the interface layout, and improving the recall of the UI visual interface webpage data rate.

4 Conclusion

In order to improve the planning effect of the product webpage UI visual interface layout, this paper proposes a product webpage UI visual interface layout planning method based on the clone selection algorithm.

1 In order to reduce the number of extreme points of the function, design the operation operator of the visual interface layout planning clone selection algorithm, and solve the problem of falling into the local optimal solution through clone proliferation and clone deletion. When the number of web pages is 100, the visual interface planning of the method in this paper The number of extreme points of the function is only 103.

2 According to the results of the best planning location node function, the layout planning algorithm process is proposed to realise the clone selection of the best location point. The experimental results show that when the number of product interfaces is 100, the web UI visual interface layout planning of the method in this paper takes time It is only 7s, which shows that the interface layout planning efficiency of the method in this paper is relatively high.

3 The article optimises the design of the optimal location variables of the interface layout by defining the layout planning of the visual interface. When the number of web pages is 120, the UI visual interface web page data recall rate of this method is as high as 98.2%, which shows that the method can effectively improve the recall rate of web page data.

References

Ali, M.H. and Kurniawan, D. (2019) 'Design of Information Systems Web-Based Car Parking Place Mall', *IOP Conference Series: Materials Science and Engineering*, Vol. 662, No. 2, pp.22–31.

Biskjr, M.M., Dalsgaard, P. and Halskov, K. (2019) 'The Same, but Better: Understanding the Practice of Designing for Incremental Innovation in Web Design', *International Journal of Design*, Vol. 13, No. 21, pp.50–67.

Choi, J., Choi, M., Shin, Y. and Lee, I.W. (2020) 'Design of Web-based Monitoring System for Solar Photovoltaic Power Plants', *International Conference on Information Networking (ICOIN)*.

Gao, H. and Huang, W. (2019) 'Optimal design of automobile human-computer interaction information interface layout based on visual cognition theory', *West Leather*, Vol. 41, No. 21, pp.87, 89.

Hikmah, F., Farlinda, S., Roziqin, M.C. and Faris, Z. (2020) 'Information System Design of Web-Based Integrated Surveillance of Ari Disease in The Health Office', *Journal of Physics: Conference Series*, Vol. 1569, No. 2, pp.34–45.

Huang, W.Y., Lee, H.L., Chou, T.Y., Liu, T.L. and Mambretti, J. (2019) 'Implementation of a transnational testbed and web UI system with Layer3 SDX', *Primate Life Histories, Sex Roles, and Adaptability*, Vol. 15, No. 22, pp.17–25.

Hung, J.C. and Wang, C.C. (2020) 'Exploring the website object layout of responsive web design: results of eye tracking evaluations', *The Journal of Supercomputing*, Vol. 13, No. 12, pp.50–56.

Ji, W., Lyu, J., Liu, X., Xu, X. and Zhao, Z. (2020) 'Design method of interface task information layout based on cognitive law', *Computer Engineering and Design*, Vol. 41, No. 5, pp.1358–1366.

Kemal, L. and Nursetyo, K.I. (2020) 'Development Interactive videos as Learning Objects on Web Design Courses in the Education Technology at the State University of Jakarta', *Jurnal Pembelajaran Inovatif*, Vol. 3, No. 1, pp.17–23.

King, A.J., Lazard, A.J. and White, S.R. (2020) 'The influence of visual complexity on initial user impressions: testing the persuasive model of web design', *Behaviour and Information Technology*, Vol. 32, No. 17, pp.43–53.

Lower, E. (2020) 'Sharing Feedback, Sharing Screens: Videoconferencing as a Tool for Stakeholder-Driven Web Design', *Journal of Extension*, Vol. 58, No. 3, pp.58–67.

Maimunah, M., Haris, H. and Priliasari, N. (2020) 'The design of web-based training management information systems at Pt. Sintech Berkah Abadi', *ADI Journal on Recent Innovation*, Vol. 2, No. 2, pp.269–274.

Oppenlaender, J., Tiropanis, T. and Hosio, S. (2020) 'CrowdUI: Supporting Web Design with the Crowd', *Proceedings of the ACM on Human-Computer Interaction*, Vol. 4, No. 21, pp.1–28.

Oswal, S.K. and Palmer, Z.B. (2020) 'Can Diversity be Intersectional? Inclusive Business Planning and Accessible Web Design Internationally on Two Continents and Three Campuses', *Conference of the Association for Business Communication*.

Rahi, S., Ghani, M.A. and Ngah, A.H. (2020) 'Factors propelling the adoption of internet banking: the role of e-customer service, website design, brand image and customer satisfaction', *International Journal of Business Information Systems*, Vol. 42, No. 21, pp.16–33.

Rantung, V.P., Munaiscehe, C.P.C., Rorimpandey, G.C., Sangkop, F.I., Pardanus, R.H.W. and Hoppenbrouwers, S. (2020) 'Web-based application design for agile stakeholder communication', *Journal of Physics: Conference Series*, Vol. 1469, No. 1, pp.123–142.

Sutabri, T., Rian, H., Hendradi, P. and Febrianto, F. (2019) 'Designing the autogate pass dashboard application with android based responsive web design technology', *Proceedings of the First International Conference of Science, Engineering and Technology, ICSET 2019*, 23 November 2019, Jakarta, Indonesia.

Wang, D.F., Zheng, L., Yang, X., Fu, Y. and Suzuki, Y. (2020) 'A Spider-web Design for Decreasing Eigen-frequency with Increasing Amplitude in A PE/ME Composite Energy Convertor', *IEEE Transactions on Industrial Electronics*, Vol. 12, No. 8, pp.15–22.

Wang, M. and Zhang, H. (2019) 'Research on spatial interface design from the perspective of metaphor cognition', *Journal of HeiLongJiang Vocational Institute of Ecological Engineering*, Vol. 32, No. 1, pp.29–31.

Zhao, Y.J. and Lu, M. (2019) 'Research on the application of grid system method in web interface design', *Packaging Engineering*, Vol. 408, No. 18, pp.107–112+119.

Zheng, T. and Ling, C.J. (2019) 'Design and Implementation of Simulated Web APP for Computer Level One Examination Based on HTML5', *Office automation (office equipment and consumables)*, Vol. 24, No. 15, pp.60–62.