Hindawi
Mobile Information Systems
Volume 2022, Article ID 2670241, 9 pages
https://doi.org/10.1155/2022/2670241

Research Article

Heterogeneous Group Risk Decision Behavior Simulation Based on Particle Swarm Optimization Algorithm

Na Lu

School of Economics and Management, Xi'an Aeronautical University, Xi'an 710077, Shaanxi, China

Correspondence should be addressed to Na Lu; luna@xaau.edu.cn

Received 26 May 2022; Revised 27 June 2022; Accepted 8 July 2022; Published 29 July 2022

Academic Editor: Shadi Aljawarneh

This paper studies the general SoC design AMBA bus and proposes an automatic generation method of software structure test data based on adaptive data optimization algorithm. By simplifying the basic particle expansion equation and eliminating the particle velocity term, an adaptive scheme based on inertia weight is proposed. In the research process, this article fully considers heterogeneous groups. Investment companies are divided into three types of decision-makers: reciprocal, intelligent, and leveraged, and their investment behavior is modeled. Given that factors such as technical level will affect the future of the project, and swarm simulation software is used to simulate and analyze the impact of the profitability of smart community microgrid construction projects on decision-making and to conduct dynamic research on market modeling risks. This article first outlines the construction of the renewable energy macro- and micro-market risk decision-making behavior model and clarifies the logic process of the market to promote the consumption of renewable energy. Then, it analyzes the causes of market risks based on dynamic models, first examines the relationship between key risks and risk factors, forms risks that affect returns (bilateral random risks and market efficiency risks), and then an analysis framework for the impact of renewable energy. This paper applies it to the analysis of data-based algorithms, thereby promoting the development of data-based algorithms.

1. Introduction

This article briefly summarizes the basis and importance of personal research, the status quo of CNC systems and multicore processors, and the real-time analysis, real-time research, and modification of Linux operating system failures. It also studied the planning and real-time maintenance principles of the Linux operating system, and transplanted the patch to the multi-core ARM platform [1]. On this basis, this paper analyzes the characteristics of the data optimization algorithm and the correlation scheme between real-time tasks and specific CPU cores, and uses the delay monotonic rate to schedule the real-time tasks of the data optimization algorithm system [2]. Using the integrated version of Qt, Qt Embedded has developed a graphical user interface for CNC systems. A point-to-point linear interpolation algorithm is written to perform basic interpolation functions. The experimental test results show that the system constructed in this paper can effectively improve the real-time performance of the task scheduling of the data optimization algorithm system, which creates conditions for further shortening the cycle and improving processing efficiency [3]. This paper deeply analyzes the impact of different project incomes on the group's free use behavior and overall contribution level, and then establishes an overall project income distribution model under a certain project income, and changes the income distribution index as a reward. This paper mainly studies the impact of income heterogeneity groups on major investment behavior [4]. Finally, in response to the actual collective action problem of the self-financing construction of the smart community microgrid residents, it actually provides a scientific and feasible theoretical basis for city managers and project designers, and proposes countermeasures [5]. Build a bilateral random risk decision-making behavior control model to realize effective prediction of key risk factors. Based on the most important risk factors, different attributes of time series, and forecasting needs, this article also proposes a variety of different forecasts and corrections: AdaBoost ELM regression model is designed for REC prices, ARIMA model is used for forecasting, and then learning methods are used for error correction, and it is recommended to make predictions based on LSTM. A deep LSMNet network with convolution, loop, loop jump, and autoregressive components is designed to predict load and line loss. A market risk management and decision simulation system are designed to transform and apply the theoretical results of this work [6]. Considering the subject of quota obligation, design enterprise-level cloud applications and distributed application integration schemes; for enterprise-level cloud applications, design the architecture of risk decision behavior management module, and examine the application of cloud service mode; for light distributed applications, the Tangle distributed ledger is used to simulate P2P energy transactions to verify the application value of blockchain technology in the energy market [7].

2. Related Work

The literature introduces one of the methods of Linux system; real-time conversion is to add real-time boot patches and discusses in detail the main real-time preference patch conversion technology, focusing on interrupt threads, high-precision clocks, key part preferences, and protocol priority inheritance [8]. The literature introduces the inspection and analysis of system performance, and tests the modified real-time Linux system, which usually interrupts the response time, context change time and clock accuracy, as well as the operating effectiveness of the graphical user interface and the operating efficiency of the interpolation program [9]. The literature introduces multi-source risks. The existing four benchmark simulation models examine the impact of various combinations of risk levels on system performance in order to provide effective guidance for the controlling society to make wise decisions about carbon resource allocation. It mainly introduces the experimental design, theory, risk-level modeling, and result statistical analysis to effectively evaluate the impact of risk factors on system performance. Related risk factors are composed of main characteristics, mechanisms, and multiple factors [10]. The literature introduces the decision-making simulation of the risk management of the renewable energy consumption market. Aiming at the risks arising from economic risks, energy risks, power quality risks, etc., a multi-objective risk decision model is constructed [11]. The key parameters of the model are obtained by simulating advanced energy markets at home and abroad to obtain decision-making solutions. It designs a risk management and decision-making simulation system, converts and implements theoretical research results, and simulates P2P energy transactions and risk decisions [12]. The literature introduces the experimental methods of using search technology to solve various software testing problems, especially through the automatic generation of structural test data, and many results have been achieved [13]. Research on automatic routing test data generation method is very important to improve software testing efficiency, ensure software quality, reduce the workload of testers, automate software testing process, and reduce software cost [14]. In the field of software testing, the research on automatic test data generation has developed into a hot spot for research and discussion by scientists at home and abroad.

3. Multi-Core ARM Data Optimization Algorithm and Heterogeneous Population Model

3.1. Multi-Core ARM Processor. The processor is an important part of the SoC system circuit. In the first stage of SoC design, the appropriate processor needs to be selected according to SoC functional requirements and actual application options. Make the performance of the processor meet the design requirements and, at the same time, will not waste too many resources on the chip, and achieve the purpose of optimizing the design. The processor mainly considers the processor type and the number of processor cores according to the requirements of the application scenario.

After data processing, the data are transmitted to the chip between chips, between boards, and between remote controllers through various high-speed peripheral interfaces; compressed, encoded, and decoded internally; and then sent to external advertisements of various specifications for display. Make the selected renderer compatible with optimized video rendering. On the other hand, the designed SoC circuit must meet the requirements that it can be used as an independent application processor, so that the selected processor can support the operation of different operating systems. Through the above analysis and summary, finally select the industry's most classic ARM Cortex-A9 as the processor core of the chip.

Reset mode: The reset signal designed by Cortex-A9 can be divided into independent reset processor. The reset mode of each reset signal is shown in Table 1.

The processor soft reset will initialize most of the logic circuits of the Cortex-A9 processor, except for the debug logic. Breakpoints and watch points are retained when the processor is restarted, and processor reset is usually used to restart a system that has been running for a period of time. Using nCPURESET and nNEONRESET to perform a soft reset corresponds to the reset sequence described in power-on reset. The only difference is that nDBGRESET must remain high during the restart sequence to ensure that all values in the debug log will not change.

3.2. Research on Heterogeneous Groups. A heterogeneous group refers to a group that is not completely homogenous, because in real life, individuals within a group show differences in public goods donations due to different income levels, input contributions, returns, and preferences for the final result. Heterogeneous social preferences in turn have a direct impact on the behavioral decisions of individual donations, showing varying degrees of cooperation. In the real system, there is almost no completely homogenous group, because the differences in factors such as age, wealth,

Table 1: Reset signal mode list.

Mode	nCPURESET	nNEONRESET	nDBGRESET
Power-on reset/cold reset	0	0	0
Processor reset/soft or warm reset	0	0	1
SIMDMPE power-on reset	1	0	1
Debug logic reset	1	1	0
Normal working mode	1	1	1

gender, knowledge, and experience have a certain impact on the behavior of participants. In addition, some studies have examined the heterogeneity of individuals in terms of social class, work industry, and religious beliefs.

By establishing a heterogeneity evaluation index system, this paper studies the relationship between heterogeneity and cooperation stability. The results of the study showed that the level of information determines the number of donations from heterogeneous groups. The investigation of the influence of the heterogeneity of individual social preferences on the supply effect of public goods shows that its contribution level is significantly higher than that of the general population. The demonstration effect of the first mover will also significantly improve the supply of public goods. Szolnoki et al. studied the evolution of cooperation in the space public product game where unconditional collaborators and various conditional collaborators coexist. Frean et al. suggested that there should be reasonable scheduling between cooperation and non-cooperation, and found that this strategy is very different from complete cooperation or complete betrayal.

3.3. Data Optimization Algorithm Model. The DRM algorithm is used for repetitive tasks, and the priority is determined based on the duty cycle. The shorter the time, the higher the priority. When a high-priority task arrives, it will overtake the low-priority CPU. RM is analyzed based on the following assumptions:

(a) All task requests are periodic.

(b) The request period of the b task is a constant.

(c) Tasks exist independently, the execution or end of a task will not depend on other task processes, and there is no communication between each other.

(d) Tasks can be advanced, and high-priority tasks can precede tasks with low CPU priority. For multiple independent periodic tasks, the lower limit of the system processor load rate programmed by the RM algorithm is

$$U = n\left(\sqrt[n]{2} - 1\right)(n > 1). \tag{1}$$

Schedule a set of tasks to meet this condition. Processor load is the sum of the load speed of all tasks:

$$U = \sum_{i=1}^{n} e_i/\mathbf{r}_i. \tag{2}$$

The monotonic rate planning algorithm does not consider the time required for context switching. When there are multiple short-term tasks in the system, because short-term tasks are prioritized, it tends to constantly change the context and involve developers of less important tasks. In this case, frequently switching tasks is time-consuming and even causes low-priority tasks to time out, thereby reducing real-time system performance.

The request with the lowest period and the longest period can directly enter the ready state, and requests for other tasks must be suspended first. The priority of the preparation task is higher than the delayed state. If the task delay time in the delayed state is 0, it will change from the delayed state to the ready state, and the task that has entered the ready state can no longer switch to the delayed state. If no task is in the ready state and the task currently being executed voluntarily exits the CPU, select the delayed state task scheduling execution.

The DRM algorithm defines the processor utilization of some tasks τ_i, τ_{i+1}, $\ldots$, τ_j, $1 \le i \le j \le n$, and the calculation formula is

$$U_{i\ldots j} = \sum_{k=i}^{j} e_k/\mathbf{r}_k. \tag{3}$$

The delay time of task τ_1 is

$$\Delta_1 = r_1 - e_1. \tag{4}$$

The time of the task τ_i, $i = 2, 3, \ldots, n-1$ in the delayed state is

$$\Delta_i = \alpha^* *_{r_i} * U_{i+1\ldots n}. \tag{5}$$

The DRM algorithm simulation test results confirm that the DRM algorithm not only reduces the frequent changes of the RM algorithm context, but also improves the processor utilization. When the system is overloaded, low-priority tasks can be delayed because high-priority tasks are processed first. The DRM algorithm is used as a criterion for evaluating time priority. But in CNC, there are not only real-time periodic tasks, but also sudden real-time tasks.

For each real-time task, calculate its slack time laxity. The current time is t, D_i is the absolute deadline of task τ_i, w_i is the remaining execution time of task τ_i, and then the relaxation time of task τ_i is

$$\text{Laxity}_i = D_i - \left(t + w_i\right). \tag{6}$$

If the downtime of the task is zero and the task cannot be scheduled, the task will exceed the deadline. In order to prevent unplanned tasks from running, tasks that have no free time for scheduling are migrated to other CPU cores, and the CPU is replaced by tasks running on that core.

Next, use the RM algorithm and the extended DRM algorithm to compare the programming results. There are two real-time tasks τ_a and τ_b in the system. The unit time is t. The task parameters are shown in Table 2. The task τ_a has a period of 4 t, the worst-case execution time is 2 t, and the task τ_b has a period of 16 t. The time is 6 t.

The results of scheduling using two algorithms are shown in Figure 1. It can be seen from the figure that these two real-time tasks can be executed under the two scheduling strategies.

Particle swarm optimization is a population-based search technology proposed by Kennedy and Eberhart. With its simple concept, easy parameter setting, and simple implementation, it immediately attracted the attention of scientists in the field of evolutionary computing. For more than ten years, it has achieved rapid development. Theoretical research is becoming more and more perfect, and at the same time it has been applied to many practical engineering fields, and its scale has been continuously expanded, becoming a new entry point for academic research. Each particle updates its position and speed according to four pieces of information: current position and speed, individual guidance, and group world guidance. In the $t+1$ generation, the velocity and position of the particle i are updated as follows:

$$v_{id}(t+1) = w\cdot v_{id} + c_1\cdot r_1\cdot\left(p_{id}(t) - x_{id}(t)\right) + c_2\cdot r_2\cdot\left(p_{\mathrm{ed}}(t) - x_{id}(t)\right), \tag{7}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), i = 1,2,\cdots,n, d = 1,2,\cdots,N. \tag{8}$$

In BPSO, the position of the particle is binary coded, and its value is only 0 or 1, but there is no such speed limit. Specifically, its location update formula is as follows:

$$x_{\mathrm{id}} = \begin{cases} 0 & \mathrm{if} S\left(v_{\mathrm{id}}\right) < r_3 \\ 1 & \mathrm{Other} \end{cases} \tag{9}$$

Among them, r_3 is a random vector in [0,1], and the velocity iv of the particle is still updated by formula (7); S(•) is the sigmoid function, and the formula is as follows:

$$S(v) = \frac{1}{1+\mathrm{e}^{-v}}. \tag{10}$$

Generally speaking, particle swarm optimization is to set the neighboring particle swarm of specific particles in advance. At the same time, the size and number of these adjacent parts have a certain impact on the integration speed of the algorithm. There is evidence that the larger the adjacent nucleus, the faster the algorithm. If the adjacent particle is small, it can prevent the nucleus from approaching prematurely. Although Kennedy has studied the different topological structures between particles and their influence on particle swarm optimization methods, they still have not got a clear conclusion.

Table 2: Real-time task parameter table.

Task	Period()	Exe time(t)
Ta	4	2
Tb	16	6

Inertial weight: Inertial weight was originally proposed by Shi and Eberhart to balance the relationship between global optimization ability and local optimization ability. They can use the inertial weight to control the impact of the previous generation on the current speed. In general, choosing a smaller inertia weight can make the particles maintain a slower speed in the original direction, so that the particle swarm has better developability (local search ability); choosing a larger inertia weight will lead to particle movement and no good development in the original direction. It has a higher speed, so that the particle swarm has a better scanning ability (global search ability). Therefore, adjusting the size of w can effectively balance the exploration and development capabilities of the particle swarm. The following is the time-varying linear fitting method:

$$w(t) = w_{max} - \frac{\left(w_{max} - w_{min}\right)\cdot t}{T_{max}}, \tag{11}$$

In other words, it is impossible to optimize all performance indicators at the same time, but to achieve a balanced result between the various goals, as shown in the following formula:

$$\begin{aligned} &\mathrm{max}y = F(x) = \left(f_1(x), f_2(x), \ldots, f_M(x)\right), \\ &\mathrm{s.t.}\begin{cases} g_i(x) \geq 0, & i = 1,2,\ldots,p, \\ h_j(x) = 0, & j = 1,2,\ldots,q. \end{cases} \end{aligned} \tag{12}$$

The C measure of the solution set A and B, denoted as C(A, B), represents the ratio of the number of elements in B that are dominated by the elements in A to the total number of elements in B, namely,

$$\mathrm{C}(A,B) = \frac{|\{b \mid b\in B, \exists a\in A, \ni : a \prec b\}|}{|B|}. \tag{13}$$

The SP measure is defined as follows:

$$\begin{aligned} &SP(S_1) = \sqrt{\frac{1}{n-1}\sum_{j=1}^{n}\left(d^*(S_1) - d\left(x_j\right)\right)^2}, \\ &\mathrm{s.t.}\, d\left(x_j\right) = \mathrm{min}_{k\in(1,2,\ldots,n)}\left(\sum_{i=1}^{2}\left|f_i\left(x_j\right) - f_i\left(x_k\right)\right|\right), \\ &d^*(S_1) = \frac{1}{n}\sum_{j=1}^{n} d\left(x_j\right), \\ &j = 1,2,\ldots,n. \end{aligned} \tag{14}$$

Figure 2 shows the principle diagram of particles moving in the solution space.

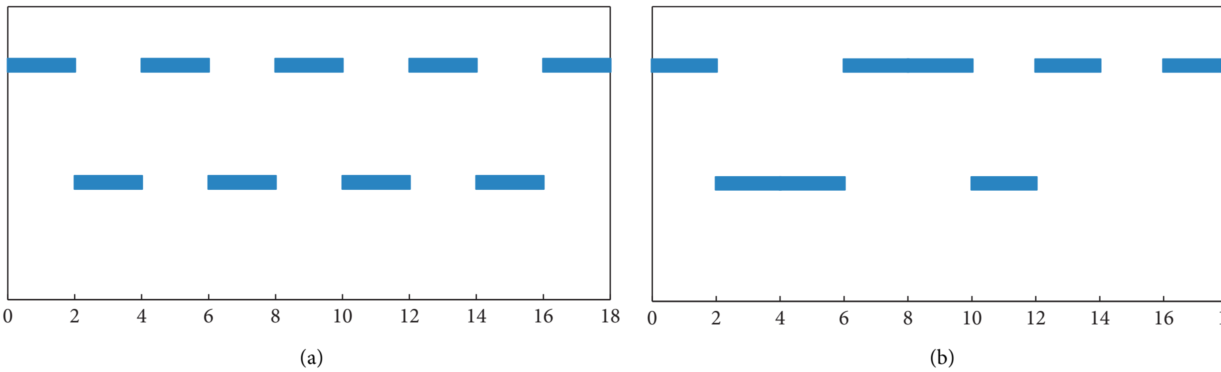

FIGURE 1: Example of scheduling using two algorithms. (a) An example of scheduling using RM. (b) An example of scheduling using DRM.

4. Simulation Research on Heterogeneous Group Risk Decision-Making Behavior

4.1. Risk Decision. Under C&T conditions, manufacturing companies face four decisions when allocating carbon resources: self-purification, carbon acquisition, quota consumption, and fines. Due to many uncertain factors, such as price fluctuations, construction time, and cost wait. Decisions regarding the allocation of carbon dioxide resources are also related to risks from multiple sources. On the basis of the research, combined with the specific circumstances of various decision-making, the research focuses on two types of decision-making risks, "self-cleaning risk" and "carbon purchase risk," and four types of marginal risks, that is, the marginal risks associated with related decisions. Due to the complexity of the model, there is no risk of temporarily controlling the use of quotas. The description and explanation of the related risks are shown in Table 3.

This research evaluates the risk level from the probability of occurrence of risk events and the severity of risk loss. The probability of occurrence of risk events is divided into five levels: low, low, high, high, and extremely high (see Table 4).

For details, see the master's thesis "Design of Multi-Source Risk Decision System for Enterprise Carbon Resource Allocation under Conditions of Total Control and Trading" as shown in Table 5.

The calculation formula is as follows:

$$I_i = L_i/PBT \times 100\%$$
$$PBT(t) = p(t) \times P\ d(t) \times (1 - SCR). \quad (15)$$

4.2. Risk-Level Control Model. The criteria for judging the risk loss level of carbon price fluctuations are as follows: tPre-tax profit = tperiod sales revenue ∗ (1-income cost rate) = tperiod product unit price ∗ tperiod sales volume ∗ (1-income cost rate).

$$PBT(t) = p(t) \times P\ d(t) \times (1 - SCR). \quad (16)$$

Among them, $P(t)$ is profit before tax in period t (without considering income tax); (t) is unit price of the product in period t; (t) is monthly output in period t; and SCR is cost of income ratio.

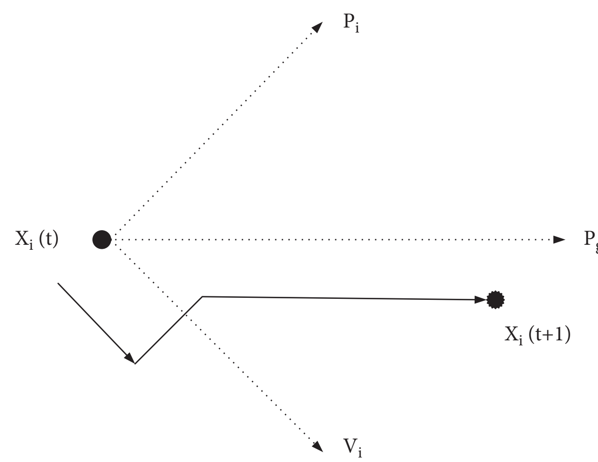

FIGURE 2: Schematic diagram of particle movement.

Actual risk loss due to carbon price fluctuations is

$$L_{21}' = \left(P_{e\,d}'(t) - P_{e\,d}(t)\right) \times q(t) = \left(P_{e\,d}'(t) - P_{e\,d}(t)\right) \times p(t) \times P\ d(t) \times \alpha_1 \times \alpha_2. \quad (17)$$

When the market carbon price risk loss index range is [0, 2],

$$\frac{L_{21}'}{PBT} = \frac{\left(P_{ed}'(t) - P_{ed}(t)\right) \times p(t) \times Pd(t) \times \alpha_1 \times \alpha_2}{p(t) \times Pd(t) \times (1 - SCR)}. \quad (18)$$

Calculate

$$P_{ed}(t) < P_{ed}'(t) \le P_{ed}(t) + \frac{\delta_m \times (1 - SCR)}{\alpha_1 \times \alpha_2}. \quad (19)$$

When the market carbon price risk loss index range is [3, 5],

$$\frac{L_{21}'}{PBT} = \frac{\left(P_{ed}'(t) - P_{ed}(t)\right) \times p(t) \times Pd(t) \times \alpha_1 \times \alpha_2}{p(t) \times Pd(t) \times (1 - SCR)} = \frac{\left(P_{ed}'(t) - P_{ed}(t)\right) \times \alpha_1 \times \alpha_2}{(1 - SCR)}. \quad (20)$$

Calculate

Table 3: Related risks involved in the decision-making process.

Decision type	Risk definition	Explain
Buy carbon	"Carbon price fluctuation risk"	The fluctuation of carbon price brings uncertainty to the expected cost of carbon procurement, which makes the cost of carbon procurement too high.
	"Failure to complete the self-purification target risk"	Failure to purify yourself or extend the time of self-purification increases the risk of not reaching the goal of self-purification.
Self-purification	"Excessive construction period risk"	The uncertainty of the construction period of the self-cleaning project will lead to costs and losses after the construction period is extended.

Table 4: P guidelines.

Probability of risk	Probability of a risk event	Remarks
(85%,100%]	Extremely high	Risk events are almost certain to happen
(60%,85%]	High	Risk events occur in more cases
(15%,60%]	Higher	Risk events occur under certain circumstances
(5%, 15%]	Lower	Risk events rarely occur
(0,5%]	Low	Risk events hardly happen

Table 5: C guidelines.

Risk loss index (severity of consequences)	Severity of loss of risk event (risk loss level)	Remarks (loss/exceeding pretax profit)
4~5	The essential	20% or more
3~4	Serious	1 0%~20%
2~3	Moderate	5%~1 0%
1~2	Small	1%~5%
0~1	Ignorable	1% or less

$$P_{e\,d}'(t) > P_{e\,d}(t) + \frac{\delta_s \times (1 - SCR)}{\alpha_1 \times \alpha_2}. \tag{21}$$

Assuming that the risk loss L22′ of failing to complete the self-purification target has the following nonlinear relationship with the time required for the risk factors to complete the remaining emission reduction target,

$$L_{22}' = max\left\{0, \frac{T_g - (T - t)}{T_g}\beta_1 \times q(t)\right\}. \tag{22}$$

Bring in the L_{22}' and PBT formula to get

$$\mathrm{T - t} < T_g < (T - t)/\left(1 - \frac{\delta_m \times (1 - SCR)}{\beta_{1\times\alpha_1\times\alpha_2}}\right). \tag{23}$$

Bring in the L_{22}' and PBT formula to get

$$(T-t)/\left(1-\frac{\delta_c\times(1-SCR)}{\beta_{1\times\alpha_1\times\alpha_2}}\right)<T_g<(T-t)/\left(1-\frac{\delta_s\times(1-SCR)}{\beta_{1\times\alpha_1\times\alpha_2}}\right). \tag{24}$$

Finally, the following formula is obtained:

$$L_{31}' = max\left\{0, \frac{T_{sp}(t) - (T - t)}{T_{sp}(t)}\beta_1 \times q(t)\right\}. \tag{25}$$

4.3. Model Test Results. In order to test the model, we use the following 8 SPSS line software to check the suitability of the system performance.

The standardized residual of the actual annual carbon purchase cost is shown in Figure 3.

The standardized residual of the actual annual self-purification cost is shown in Figure 4.

The standardized residual of annual self-net risk loss is shown in Figure 5.

The results show that the actual cost of carbon purchases each year determines the total cost of carbon resource allocation, and the income from carbon sales determines the net cost of carbon resources. The carbon resources are distributed obliquely, and the difference meets the normality assumption. Therefore, the performance of the two systems has passed the applicability test of the model.

4.4. Risk Decision-Making Behavior Management Model. The theoretical innovations such as the risk management model established in this paper and the data analysis method system designed in this paper mainly provide knowledge base support for the risk management and decision-making simulation system. Especially enterprise-level cloud applications and distributed applications require big data support from multiple sources and advanced expert systems. The research results of this paper expand and enrich the knowledge base of the expert system, and provide a powerful theoretical tool for users to make use of the system for knowledge mining and buying and selling decisions based on big data. With the development of advanced information technologies such as the "big cloud mobile smart chain," the data sources of risk management and decision-making simulation systems have been further expanded. Distributed energy, smart energy equipment, etc., provide more and

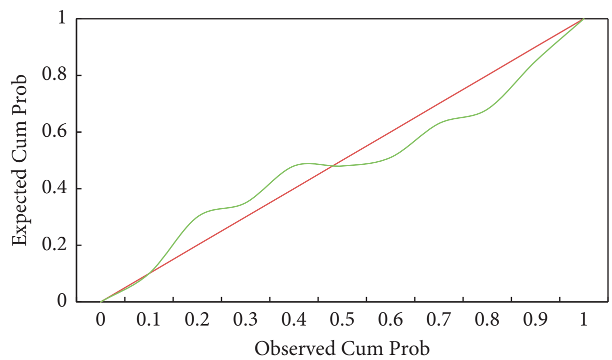

Figure 3: P-P plot of standardized residuals of the actual cost of carbon purchases in years.

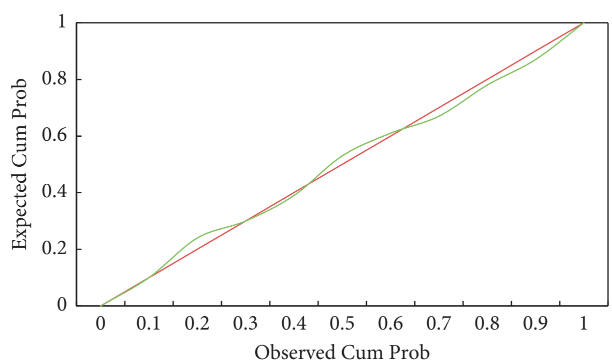

Figure 4: P-P graph of standardized residuals of carbon purchase risk loss in years.

more detailed multi-source data sources for the system. Multi-source data must not only be preprocessed and stored in the enterprise data center, but also distributed to distributed applications. These data also provide support for building a more powerful expert system. This paper is studying mining methods and models based on these data to improve the management and decision-making skills of the expert system.

The system considers the design of enterprise-level cloud applications and distributed applications from the perspective of different types of users. Both types of applications and distributed applications need to be tightly integrated. Especially for distributed applications, the situation of potential users is very complicated, and users can switch between the roles of energy producer and energy consumer repeatedly, which makes the design of system integration solutions more difficult from the user's point of view. Whether it is an enterprise-level cloud application or a distributed application, the objects of its management are the generation, consumption, and behavior of electricity transactions. Therefore, the system designs a system integration scheme from the perspective of the process meeting of both parties.

Renewable energy consumption market risk management and decision-making simulation system integration scheme mainly include the following elements:

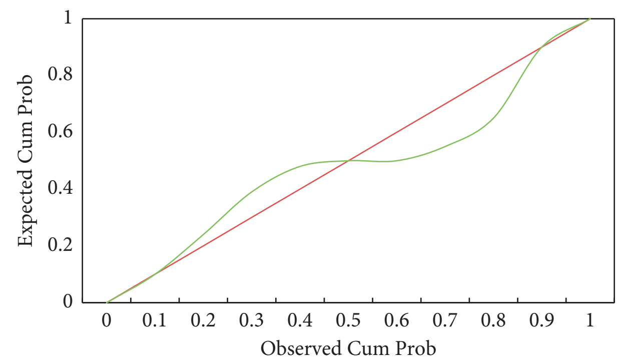

Figure 5: P-P chart of standardized residuals of self-net risk loss.

(1) Under the guidance of the integrated management strategy, combined with the characteristics of the development language and framework, the system structure and functions are designed by analyzing the business needs focusing on power generation, consumption, and commercialization.

(2) According to the application integration process, integration method, and integration rules on the power generation side, an integrated power generation side trend and information flow are formed according to business needs; an integrated energy flow and information flow on the energy side are also formed on the sales side.

(3) Based on market rules, design a two-way matching strategy on the power generation side and the power sales side, and design an integrated service application model based on two-way matching to further form a system operation mechanism.

(4) In the application, through the effectiveness feedback and process monitoring procedures, the comprehensive advantages of the system are evaluated, the integration mode is continuously optimized, the integration is deepened, and the application integration is guaranteed.

Enterprise-level cloud applications are mainly for business users such as e-commerce. The design of this section focuses on risk management modules. Enterprise-level cloud applications can also include visual statistical query and energy trading modules, but they have little to do with the research content of this article. The risk management module of enterprise-level cloud applications is similar to traditional information management systems. This section first designs the system architecture and then discusses the application of cloud models in enterprise-level applications.

Different hierarchical structures complete different functions:

(1) Infrastructure layer provides hardware and software foundation for system implementation. Important infrastructure for enterprise users includes data

centers and virtualized cloud facilities. Considering the interaction between enterprise-level cloud applications and distributed applications, the infrastructure layer also includes distributed infrastructure.

(2) The resource layer is a hierarchical structure built on the infrastructure layer, which mainly integrates the resources of enterprise-level cloud applications and distributed applications. These resources include not only the hardware and software installation provided by the infrastructure layer, but also the various power resources, information resources, and knowledge base resources of the research results of this article.

(3) Service layer management and use of resource layer to provide services for the realization of various applications. The design and implementation of the service layer are essential for business applications to go to the cloud. Generally speaking, providing distributed database services, file services, web services, cluster management, and parallel computing can effectively support cloud applications.

(4) The application layer mainly considers business requirements and implements system functions based on the services provided by the service layer. The enterprise-level cloud application risk management module developed in this section should implement functions such as market risk assessment, multi-objective risk decision-making, random risk management, and effective risk management.

The e-top architecture can build a cloud application risk management module at the enterprise level and further integrate it with the energy information system and energy market operation system to expand the functions of the system. In addition, the cloud application risk management module is subject to company-level guidelines, regulations, market rules, and technical standards, and it shares operation and maintenance management with the portal system.

The structure of each level is interrelated, and its application layer realizes the central function of the module, which is also the key to the transformation of the research results of this article into practical applications.

5. Conclusion

The background of this document is the design of a highly integrated SoC application processing chip based on the ARMA9 core. Through analysis and research on the application scenarios of the multi-core ARM processor architecture, AMBA bus protocol, and project requirements, SoC was designed on the basis of the multi-core ARM processor system architecture, and the simulation verification of the front-end system and FPGA prototype verification were completed. This article uses a simulation of the carbon resource allocation decision-making process of a carbon emission control company to describe the convenience of carbon procurement and the risks faced by carbon emissions. The clean decision-making and analysis of different systems, the output of the system under the combination of structure, and different risk levels are mainly control and emission.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

This study was supported by Soft science research project of Science and Technology Department of Shaanxi Province: Research on the Interaction Mechanism and Evolution of Consistent Group Risk Decision Behavior Based on Computational Experiments in the Digital Era (Project no. 2019KRM093).

References

[1] L. Zheng, Y. Lu, M. Guo, S. Guo, and C.-Z. Xu, “Architecture-based design and optimization of genetic algorithms on multi- and many-core systems,” *Future Generation Computer Systems*, vol. 38, pp. 75–91, 2014.

[2] S. Gao and G. D. Peterson, “GASPRNG: GPU accelerated scalable parallel random number generator library,” *Computer Physics Communications*, vol. 184, no. 4, pp. 1241–1249, 2013.

[3] Q. Y. Duan, V. K. Gupta, and S. Sorooshian, “Shuffled complex evolution approach for effective and efficient global minimization,” *Journal of Optimization Theory and Applications*, vol. 76, no. 3, pp. 501–521, 1993.

[4] L. Zhu and Y. Fan, “A real options-based CCS investment evaluation model: case study of China’s power generation sector,” *Applied Energy*, vol. 88, no. 12, pp. 4320–4333, 2011.

[5] X. Wang and L. Du, “Study on carbon capture and storage (CCS) investment decision-making based on real options for China’s coal-fired power plants,” *Journal of Cleaner Production*, vol. 112, no. 5, pp. 4123–4131, 2016.

[6] B. Suárez-Eiroa, E. Fernández, G. Méndez-Martínez, and D. Soto-Oñate, “Operational principles of circular economy for sustainable development: linking theory and practice,” *Journal of Cleaner Production*, vol. 214, pp. 952–961, 2019.

[7] A. Fujihara, “Proposing distributed traffic information system using blockchain technology,” *IEICE Technical Report*, vol. 117, p. 223, 2018.

[8] K. G. Savvidy, “The MIXMAX random number generator,” *Computer Physics Communications*, vol. 196, pp. 161–165, 2015.

[9] J. Li, J. A. Besada, A. M. Bernardos, P. Tarrío, and J. R. Casar, “A novel system for object pose estimation using fused vision and inertial data,” *Information Fusion*, vol. 33, pp. 15–28, 2017.

[10] A. S. M. S. Azad, V. Fang, and C.-H. Hung, “Linking the interest rate swap markets to the macroeconomic risk: the UK and US evidence,” *International Review of Financial Analysis*, vol. 22, pp. 38–47, 2012.

[11] B. Yalabik and R. J. Fairchild, “Customer, regulatory, and competitive pressure as drivers of environmental innovation,” *International Journal of Production Economics*, vol. 131, no. 2, pp. 519–527, 2011.

[12] S. Markose, S. Giansante, and A. R. Shaghaghi, "'Too interconnected to fail' financial network of us CDS market: topological fragility and systemic risk," *Journal of Economic Behavior & Organization*, vol. 83, no. 3, pp. 627–646, 2012.

[13] J. Wu, Y. Feng, and P. Sun, "Sensor fusion for recognition of activities of daily living," *Sensors*, vol. 18, no. 11, p. 4029, 2018.

[14] C. Melendez-Pastor, R. Ruiz-Gonzalez, and J. Gomez-Gil, "A data fusion system of GNSS data and on-vehicle sensors data for improving car positioning precision in urban environments," *Expert Systems with Applications*, vol. 80, pp. 28–38, 2017.