

A new 2D Hindmarsh-Rose neuron, its circuit implementation, and its application in dynamic flexible job shops problem

Yao Lu(卢尧), Weijie Nie(聂伟杰), Xu Wang(王旭), Xianming Wu(吴先明), and Qingyao Ma(马晴瑶)
Citation: Chin. Phys. B, 2025, 34 (**12**): 120503. DOI: 10.1088/1674-1056/ae0892
http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Chaos of cavity optomechanical system with Coulomb coupling

Yingjia Yang(杨应佳), Liwei Liu(刘利伟), Lianchun Yu(俞连春), Weizheng Kong(孔伟正), Haiyan Jiao(焦海燕), Xiaoyan Deng(邓小燕), and Xiaoyong Li(李小勇)
Chin. Phys. B, 2025, 34 (**8**): 080503. DOI: 10.1088/1674-1056/add509

Generation of multitype, multicavity chaotic attractors via impulse-function-based state variable extension

Xiaoyu Hu(胡晓宇), Siteng Wang(王思腾), Panpan Wu(邬盼盼), Hongbo Cao(曹红博), Tianwei Yang(杨天纬), and Zhongshuo Dong(董忠硕)
Chin. Phys. B, 2025, 34 (**8**): 080502. DOI: 10.1088/1674-1056/add249

Complexity dynamics analysis of a dual-channel green supply chain with government intervention and cap-and-trade regulation

Yuhao Zhang(张玉豪), Lin Huang(黄林), and Man Yang(杨满)
Chin. Phys. B, 2025, 34 (**7**): 070501. DOI: 10.1088/1674-1056/adca9c

Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection

Fei Yu(余飞), Dan Su(苏丹), Shaoqi He(何邵祁), Yiya Wu(吴亦雅), Shankou Zhang(张善扣), and Huige Yin(尹挥戈)
Chin. Phys. B, 2025, 34 (**5**): 050502. DOI: 10.1088/1674-1056/adb8bb

Model-free prediction of chaotic dynamics with parameter-aware reservoir computing

Jianmin Guo(郭建敏), Yao Du(杜瑶), Haibo Luo(罗海波), Xuan Wang(王暄), Yizhen Yu(于一真), and Xingang Wang(王新刚)
Chin. Phys. B, 2025, 34 (**4**): 040505. DOI: 10.1088/1674-1056/adb733

SPECIAL TOPIC — Biophysical circuits: Modeling & applications in neuroscience

A new 2D Hindmarsh–Rose neuron, its circuit implementation, and its application in dynamic flexible job shops problem

Yao Lu(卢尧)[1,2], Weijie Nie(聂伟杰)[1], Xu Wang(王旭)[1], Xianming Wu(吴先明)[1,†], and Qingyao Ma(马晴瑶)[1]

[1] *School of Mechanical and Electrical Engineering, Guizhou Normal University, Guiyang* 550025, *China*

[2] *Technical Engineering Center of Manufacturing Service and Knowledge Engineering, Guizhou Normal University, Guiyang* 550025, *China*

(Received 11 July 2025; revised manuscript received 13 September 2025; accepted manuscript online 18 September 2025)

We propose a simplified version of the classic two-dimensional Hindmarsh–Rose neuron (2DHR), resulting in a new 2DHR that exhibits novel chaotic phenomena. Its dynamic characteristics are analyzed through bifurcation diagrams, Lyapunov exponent spectra, equilibrium points, and phase diagrams. Based on this system, a corresponding circuit is designed and circuit simulations are carried out, yielding results consistent with the numerical simulations. To explore practical applications of chaotic systems, 2DHR is employed to improve the solution of the flexible job-shop scheduling problem with dynamic events. The research results demonstrate that applying 2DHR can significantly enhance the convergence rate of the optimization algorithm and improve the quality of the scheduling solution.

Keywords: chaos, nervous system, meta-heuristic algorithm, dynamic flexible job-shop scheduling problem

PACS: 05.45.Gg, 87.19.ll, 89.75.–k

DOI: 10.1088/1674-1056/ae0892 **CSTR:** 32038.14.CPB.ae0892

1. Introduction

Neurons, as the essential structural and functional components of the nervous system, produce intricate electrical activities that underpin the physiological mechanisms of higher cognitive functions in the brain. To investigate the intrinsic mechanisms and dynamic behaviors that contribute to these complex electrical activities, researchers have developed mathematical models that simulate the dynamics of biological neurons. The Hodgkin–Huxley (HH) model,[1] one of the earliest mathematical representations, was formulated based on experimental observations of neuronal electrical activity. In an effort to achieve a balance between mathematical simplicity and biological accuracy, Hindmarsh and Rose[2] introduced the two-dimensional Hindmarsh–Rose (2DHR) model in 1982. They[3] later expanded this model to a three-dimensional (3D) version in 1984 to incorporate more intricate dynamics, including neuronal bursting phenomena.

Owing to its simple algebraic structure and its ability to exhibit a rich repertoire of dynamical behaviors, the HR model has garnered significant and sustained research attention since its inception, leading to numerous subsequent investigations. Consequently, the model has rapidly become a paradigmatic framework within the fields of computational neuroscience and nonlinear dynamics, and it has since been intensively studied and extended in various directions. In the realm of hardware implementation, overcoming the computational bottlenecks associated with large-scale network simulations has driven research toward physical realizations of the model, resulting in the development of efficient, low-power, multiplierless circuit solutions.[4] Moreover, the construction of dedicated hardware platforms has facilitated the experimental acquisition of the system's dynamic behaviors, providing physical validation that corroborates the theoretical models.[5] Concurrently, researchers have delved deeper into the model's intrinsic dynamical complexity. Through bifurcation analysis[6] and studies of the singular limit,[7] they have systematically revealed the pathways from periodic bursting to chaos, clarifying the central role of homoclinic orbits[8] in organizing these complex transitions. Analysis of the model's discrete-time versions has also uncovered phenomena specific to discrete systems, such as Flip and Neimark–Sacker bifurcations.[9] A particularly cutting-edge research area involves generalizing the model's mathematical framework. The introduction of memristors to simulate electromagnetic induction effects has led to the discovery of novel phenomena such as hidden attractors[10] and superextreme multistability.[11] New analytical tools, including energy computation and control methods, have been developed for these systems,[12,13] and their complex dynamics have shown potential for engineering applications such as image encryption.[14] Furthermore, extensions to fractional-order HR models[15,16] and Filippov systems[17] have greatly enriched the dynamical repertoire of neuron models. At the network level, research has extended to systems of coupled HR neurons, focusing on synchronization phenomena,[18] collective dynamics, and their role in information processing and transmission.[19,20]

†Corresponding author. E-mail: jsdxwxm@126.com

© 2025 Chinese Physical Society and IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

Meta-heuristic algorithms have garnered significant attention from researchers in the field of the flexible job-shop scheduling problem (FJSP) owing to their robust global search capabilities, adaptability, and computational efficiency. However, when the search space of the problem is very large or its topological structure is highly complex, meta-heuristic algorithms are prone to becoming trapped in local optima. To address this issue, researchers have incorporated chaos theory, leveraging the randomness and ergodicity of chaotic sequences to prevent the algorithm from falling into local optima, improve its optimization performance, and find approximate global solutions.[21–23]

Building upon the classical HR model, this paper introduces a novel two-dimensional discrete HR neuron model to explore new dynamical possibilities. By structurally modifying the canonical 2DHR equations, we propose a discrete dynamical system incorporating a novel nonlinear term, designed to uncover unique dynamic behaviors not present in the original model. A systematic investigation is conducted to comprehensively characterize the dynamical properties of this new model. Through theoretical analysis, we identify the system's equilibrium points and establish their stability conditions. We then employ numerical simulation tools, including bifurcation diagrams, Lyapunov exponent spectra, and phase portraits, to meticulously map the model's dynamical evolution paths as a function of key parameters. To corroborate the theoretical analysis and numerical simulations, and to demonstrate the model's physical realizability, a corresponding analog circuit was designed and implemented. Both PSIM software simulations and hardware circuit experiments successfully reproduced the model's unique coexistence and evolution of attractors, with experimental results showing excellent agreement with numerical findings. This research not only introduces a new neuron model with rich dynamical characteristics, but also provides dual validation via simulation and hardware experiments, offering both a theoretical foundation and a practical reference for the design of innovative neuromorphic circuits.

In engineering applications, the FJSP requires that, under the constraints of limited machine resources and processing tasks, the processing sequence of each job and the allocation of machines be reasonably arranged while optimizing multiple objective functions. In this paper, the 2DHR chaotic model is utilized to enhance a meta-heuristic algorithm for solving FJSP. The results demonstrate that 2DHR effectively guides the algorithm in escaping local optima and finding approximate global solutions.

2. HR neuron model

2.1. A simple description of the HR neuron model

Building upon the classical Hodgkin–Huxley model,[1] Hindmarsh and Rose[2] first introduced the 2DHR neuron model in 1982. The model is mathematically characterized by a set of two coupled first-order differential equations, expressed as follows:

$$\begin{cases} \dot{x} = y - ax^3 + bx^2 + I, \\ \dot{y} = c - dx^2 - y, \end{cases} \tag{1}$$

where x represents the membrane potential, y corresponds to the recovery variable associated with ionic currents, and I denotes the externally applied current. The parameters a, b, c, and d are positive constants that govern the neuron's dynamics. In numerous studies of the Hindmarsh–Rose model, these parameters are typically assigned the physiologically relevant values $a = 1$, $b = 2$, $c = 1$, and $d = 5$.

To enable richer electrical activity in the membrane potential and to generate more complex dynamical behaviors, Hindmarsh and Rose[3] proposed a 3D neuron model in 1984. This model builds upon the original 2D formulation by introducing a third differential equation, expressed as

$$\begin{cases} \dot{x} = y - ax^3 + bx^2 + I - z, \\ \dot{y} = c - dx^2 - y, \\ \dot{z} = r(s(x + x_1) - z), \end{cases} \tag{2}$$

where z is the bursting variable acting as a slow outward current that modulates the externally injected current I. The parameter x_1 is a constant related to the bursting behavior. The additional parameters r and s are positive constants, with r typically set to a small value, commonly $r = 0.01$, and $s = 4$.

2.2. Construction of a new Hindmarsh–Rose Neuron Model

Inspired by the introduction of a new variable in the above model, we modify Eq. (1) and propose a 2D discrete dynamical system, formulated as

$$\begin{cases} x_{n+1} = -kx_n^3 + y_n + 1, \\ y_{n+1} = -ux_n^2 - y_n. \end{cases} \tag{3}$$

For simplicity, we denote

$$\begin{cases} G(x) = -kx_n^3 + 1, \\ H(x) = -ux_n^2. \end{cases} \tag{4}$$

By substituting Eq. (4) into Eq. (3), the 2DHR neuron model can be expressed as

$$\begin{cases} x_{n+1} = G(x) + y_n, \\ y_{n+1} = H(x) - y_n. \end{cases} \tag{5}$$

2.3. Equilibrium points and stability analysis of the system

The equilibrium point of the system $S=(x^*,y^*)$ must satisfy $x_{n+1}=x_n=x^*$ and $y_{n+1}=y_n=y^*$. Substituting the equilibrium point into the discrete system, it can be expressed as

$$\begin{cases} x^* = -k(x^*)^3 + y^* + 1, \\ y^* = -\frac{u}{2}(x^*)^2. \end{cases} \tag{6}$$

Simplifying Eq. (6), we obtain

$$k(x^*)^3 + \frac{u}{2}(x^*)^2 + x^* - 1 = 0. \tag{7}$$

The Jacobian matrix of the discrete map at $S=(x^*,y^*)$ is

$$J = \begin{bmatrix} -3k(x^*)^2 & 1 \\ -2ux^* & -1 \end{bmatrix}. \tag{8}$$

For discrete dynamical systems, the stability of a fixed point is determined by the magnitudes of its eigenvalues. If all $|\lambda_i|<1$, the equilibrium is asymptotically stable. If any $|\lambda_i|>1$, it is unstable. When $|\lambda_i|=1$, a bifurcation may occur. It is evident that the location of the equilibrium state is influenced by the parameters k and u.

3. Dynamical behavior

This section primarily investigates the dynamical behavior, bifurcation phenomena, and attractor dynamics of the 2DHR neuron model. The analysis employs bifurcation diagrams, Lyapunov exponents (LEs), and phase portraits.

3.1. Dynamical behavior

For a nonlinear dynamical system, chaotic dynamics are typically characterized using bifurcation diagrams and LEs. With k fixed at 0.15 and u varying within the interval [0.65,0.8], the bifurcation diagram and Lyapunov exponent plot of the discrete dynamical system based on the Hindmarsh–Rose neuron model are presented in Fig. 1. Figure 1(a) displays the bifurcation diagram. For $k=0.15$, when u lies within the interval [0.65,0.654], the system exhibits a stable period-3 orbit. Correspondingly, Fig. 1(b) shows that the LEs in this interval are less than or equal to 0. Additionally, for u within the intervals (0.664,0.665), (0.67,0.671), (0.672,0.676), (0.681,0.687), and (0.694,0.695), the system also remains in periodic states, with LEs similarly ≤ 0 as indicated in Fig. 1(b). Outside these intervals, the system enters chaotic regimes, where the LEs are positive.

3.2. Analysis of system dissipativity

By examining Fig. 1(b), over the entire parameter range $u\in[0.65,0.8]$, LE_2 remains negative, with its absolute value typically exceeding that of LE_1. Consequently, the sum $LE_1+LE_2<0$ holds throughout this interval, indicating that the discrete system is dissipative. When such a dissipative system exhibits chaotic behavior ($LE_1>0$), the attractor is generally a strange attractor with a fractal structure. The fractal dimension of the strange attractor can be estimated using the Kaplan–Yorke dimension,[24] defined as

$$D_{\mathrm{kY}} = j + \frac{1}{|LE_{j+1}|}\sum_{i=1}^{j} LE_i. \tag{9}$$

For this 2D discrete system in the chaotic regime, $j=1$, thus

$$D_{\mathrm{kY}} = 1 + \frac{LE_1}{|LE_2|}. \tag{10}$$

Since $0<LE_1/|LE_2|<1$, it follows that $1<D_{\mathrm{kY}}<2$, indicating that the attractor has a non-integer fractal dimension.

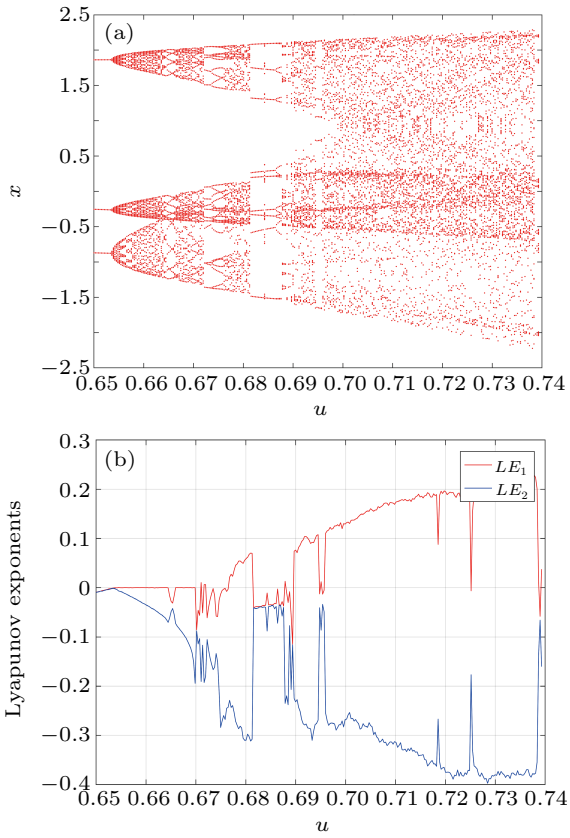


Fig. 1. Dynamical behavior of the 2DHR neuron model as a function of the control parameter u. (a) Bifurcation diagram; (b) Lyapunov exponents.

3.3. Analysis of chaotic attractors

To illustrate the chaotic attractors and their internal crisis phenomena, phase portraits in the x–y plane are presented with $k=0.15$ fixed and the control parameter u varied. The results are shown in Fig. 2. When $u=0.68$, as shown in Fig. 2(a), three distinct chaotic attractors emerge in the phase space based on the HR neuron circuit, each occupying separate, non-overlapping regions. Some isolated transient points are also observed. The corresponding LEs are $LE_1=0.060$ and $LE_2=-0.308$.

When $u=0.70$, as shown in Fig. 2(b), the independent attractors begin to expand but remain spatially separated. The system has $LE_1=0.132$ and $LE_2=-0.272$.

When $u = 0.71$, Fig. 2(c) shows a significant topological transformation: the three previously disjoint chaotic attractors undergo an internal crisis and merge into a single, larger strange attractor. The Lyapunov exponents in this case are $LE_1 = 0.167$ and $LE_2 = -0.314$.

When $u = 0.73$, as illustrated in Fig. 2(d), the merged strange attractor continues to expand, exhibiting increased structural complexity and occupying a larger region of the phase space. This indicates enhanced chaotic behavior, with $LE_1 = 0.207$ and $LE_2 = -0.381$.

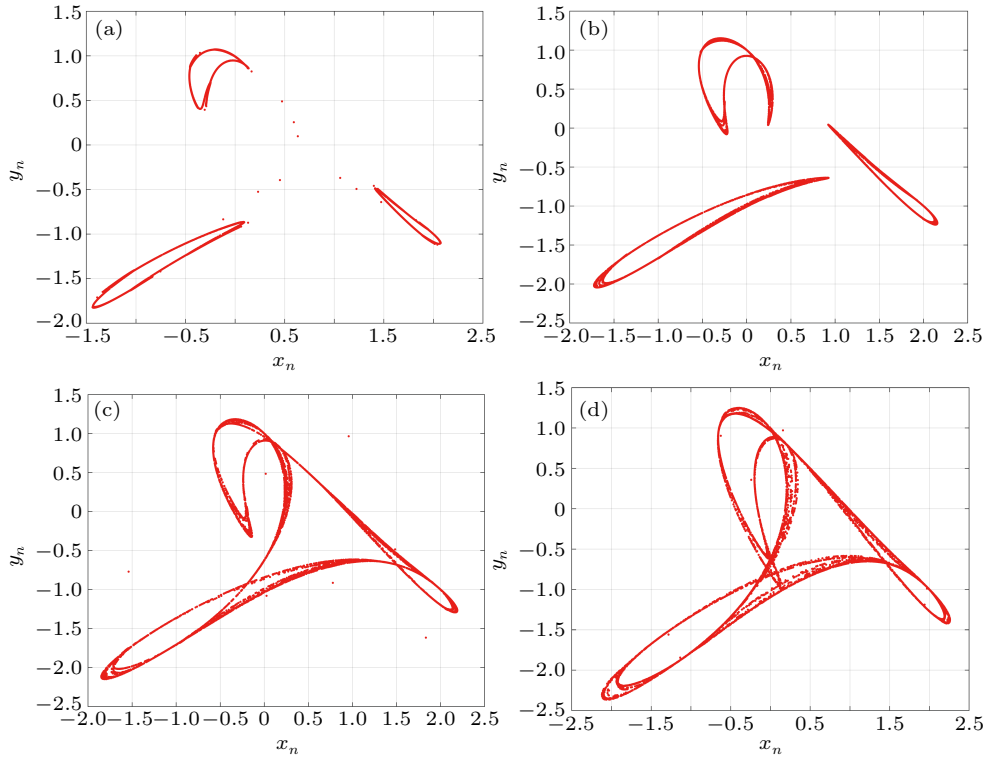


Fig. 2. Chaotic attractors of the 2DHR neuron model under different values of u. Subfigures (a)–(d) show the phase portraits in the x–y plane for $u = 0.68$, 0.70, 0.71, and 0.73.

4. Circuit implementation

The objective of this section is to validate the aforementioned theoretical analysis through the design and implementation of an appropriate circuit. As shown in Fig. 3, all the operational amplifiers used are AD711, powered by a ±15 V dual power supply, and the analog multiplier employed is the AD633. By applying Kirchhoff's laws, the circuit can be analyzed, and the corresponding equations are given as follows:

$$\begin{cases} x_{n+1} = -\dfrac{R_6}{R_4}x_n^3 + \dfrac{R_2R_6}{R_1R_5}y_n + \dfrac{R_6}{R_3}V_0, \\ y_{n+1} = -\dfrac{R_9}{R_7}x_n^2 - \dfrac{R_9}{R_8}y_n, \end{cases} \tag{11}$$

where $R_1 = R_2 = R_8 = R_9 = 10$ kΩ, $R_3 = R_5 = R_6 = 15$ kΩ, $R_4 = 100$ kΩ, and R_7 is a variable resistor.

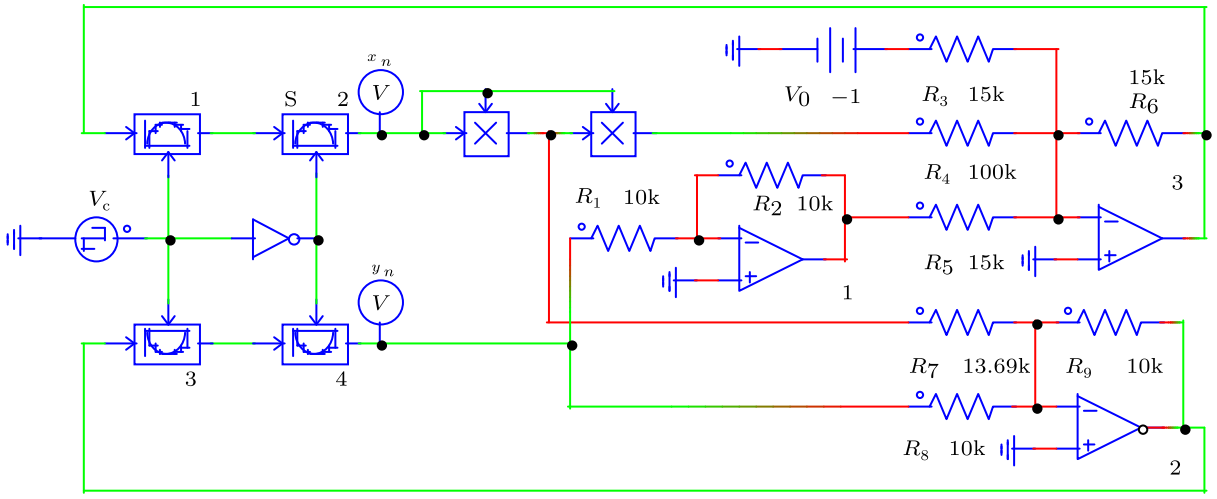


Fig. 3. PSIM simulation model of the HR neuron-based circuit.

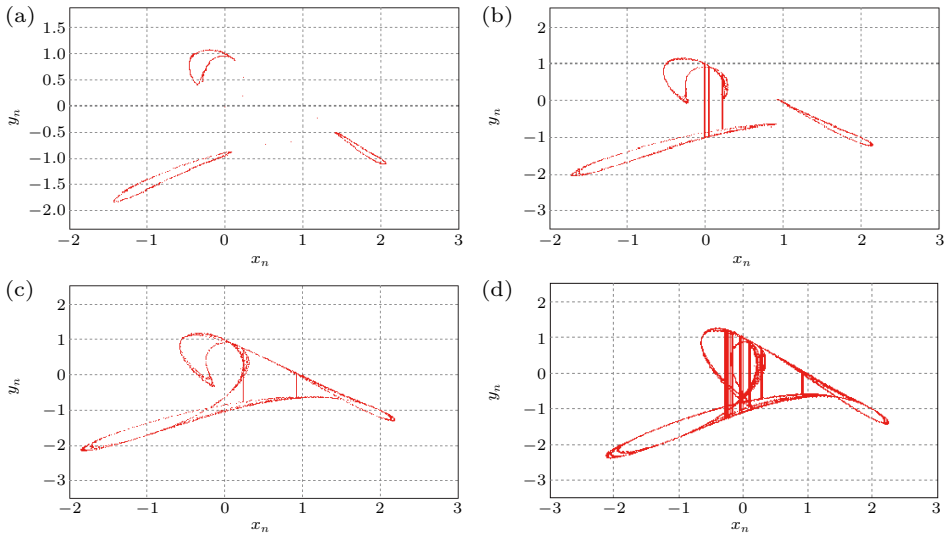


Fig. 4. Phase portraits of attractors on the x_n–y_n plane for different values of R_7. Subfigures (a)–(d) correspond to $R_7 = 14.7$ kΩ, 14.29 kΩ, 14.08 kΩ, and 13.69 kΩ, respectively.

For the four representative values of $u = 0.68$, 0.70, 0.71, and 0.73 shown in Fig. 2, the corresponding variable resistor values are $R_7 = 14.7$ kΩ, 14.29 kΩ, 14.08 kΩ, and 13.69 kΩ, respectively. Based on the PSIM circuit simulation model, the phase portraits on the x_n–y_n plane are illustrated in Fig. 4. Comparison between Figs. 2 and 4 reveals excellent agreement between the PSIM simulation and numerical results. This consistency demonstrates that the HR neuron-based circuit can reproduce various types and scales of attractor dynamics, effectively confirming the outcomes of the numerical simulations.

This section demonstrates the physical feasibility and robustness of the proposed 2DHR model in capturing chaotic dynamics through the design and simulation of analog circuits. The next section explores the application of the 2DHR model to real-world engineering challenges, leveraging its chaotic properties to address the complex dynamic flexible job shop scheduling problem (DFJSP) and utilizing its ergodic, quasi-random sequences to significantly enhance the performance of metaheuristic algorithms. Following the conclusions of this section, the 2DHR model can be implemented in hardware, and the corresponding circuit may serve as a true hardware random number generator or as a chaotic signal source for a dedicated optimized coprocessor. Compared with software-based pseudorandom numbers, hardware-generated random numbers offer potential advantages in speed and power consumption.

5. Flexible job shop design based on 2DHR chaotic grey wolf optimization (GWO)

5.1. Overview of dynamic flexible job shop

As a classic combinatorial optimization problem, job shop scheduling is known to be NP-hard. Its objective is to select the optimal processing machines for each job under multiple constraints, in order to minimize both the makespan and the energy consumption of the schedules.

In the FJSP, each operation can be processed on multiple alternative machines, and the processing sequence on each machine exhibits heightened flexibility. This configuration introduces increased complexity in schedule optimization. This study extends the investigation to dynamic flexible job shops, incorporating chaotic optimization mechanisms to achieve optimal scheduling solutions.

The corresponding mathematical description of the DFJSP is provided in Ref. [25], and the structure chart is shown in Fig. 5. The job set is processed in the workshop according to the initial scheduling scheme. When a dynamic event occurs in the workshop, the scheduling center can update the schedule based on the type of event and return the updated information to the workshop for further processing.

A scheduling instance involving three machines and four jobs is illustrated in Table 1. According to the scheduling method, the specific scheduling scheme is shown in Fig. 6, where the horizontal axis represents the processing time points, the vertical axis represents the allocated machines, and the boxes indicate the start and end times of each job. As this problem is a discrete optimization problem, idle periods of processing machines are inevitable. The goal of scheduling is to determine the optimal matching between jobs and machines to achieve the predetermined objectives.

The dynamic events considered in this paper include new job arrivals,[26] start-up of new machines, and processing time failures of jobs.[27] The makespan reflects the latest time required for a workshop to complete a batch of jobs, while the energy consumption represents the total energy utilized during the production process. A shorter makespan indicates better utilization of resources, such as more efficient use of machinery and human resources. Similarly, lower energy consump-

tion suggests more economical use of power resources and a reduced environmental impact.

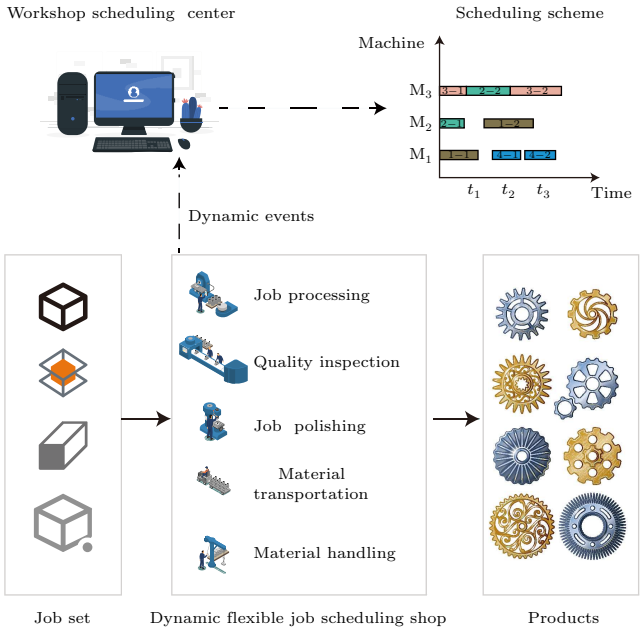


Fig. 5. Schematic of the dynamic flexible job shop.

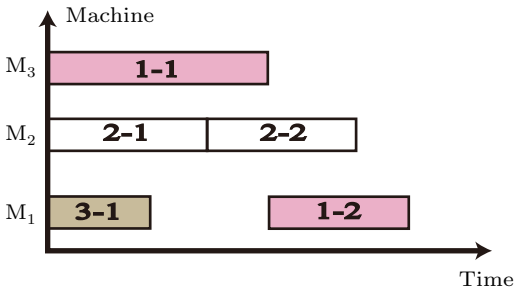


Fig. 6. Gantt chart of a specific scheduling example derived from Table 1.

Table 1. One numerical instance of the DFJSP.

Jobs	Operations	Machine
J_1	$O_{1,1}$,	M_1, M_3
	$O_{1,2}$	M_1, M_2
J_2	$O_{2,1}$	M_1, M_2
	$O_{2,2}$	M_1, M_2, M_3
J_3	$O_{3,1}$	M_1, M_2, M_3

Optimizing the scheduling plan to shorten the makespan can improve production efficiency, reduce costs, and enhance customer satisfaction. Meanwhile, minimizing energy consumption can lower operational expenses, extend equipment lifespan, and contribute to environmental sustainability. Therefore, by considering both makespan and energy consumption as optimization objectives, a dynamic flexible job shop can achieve a balance between production efficiency and resource conservation, thereby enhancing the overall performance and competitiveness of the workshop.

The objective function is denoted as

$$\begin{cases} f_1 = \min(C_{\max}), \\ f_2 = \min\left(\sum_{k=1}^{m} E_k^{\mathrm{p}} + \sum_{j=1}^{N} E_j^{\mathrm{s}} + E_f\right), \end{cases} \tag{12}$$

where f_1 represents the minimum scheduling makespan, $C_{\max}$ is the final makespan of each scheme, and f_2 represents the total energy consumption of the workshop for the corresponding scheduling scheme. E_k^{p} denotes the processing energy consumption of machine k, E_j^{s} denotes the energy consumption of operation j transferred between machines, and E_f represents inherent energy consumption of the workshop, such as lighting and thermal heating. Here, m is the total number of machines in the scheduling scheme, and N is the total number of operations in the scheme.

5.2. Design of an improved chaotic GWO algorithm

5.2.1. GWO algorithm description

As shown in Fig. 7, the GWO algorithm is a swarm intelligence optimization method inspired by the hierarchical system and predatory behavior of grey wolves in nature. In this hierarchy, the alpha (α) wolf is the leader directing the hunting process; the beta (β) wolf is second in rank; the delta (δ) wolf is third; and the omega (Ω) wolf is the lowest rank. During the hunting phase, higher-ranked wolves first approach the prey, gradually form an encirclement, and then coordinate with lower-ranked wolves for cooperative hunting.

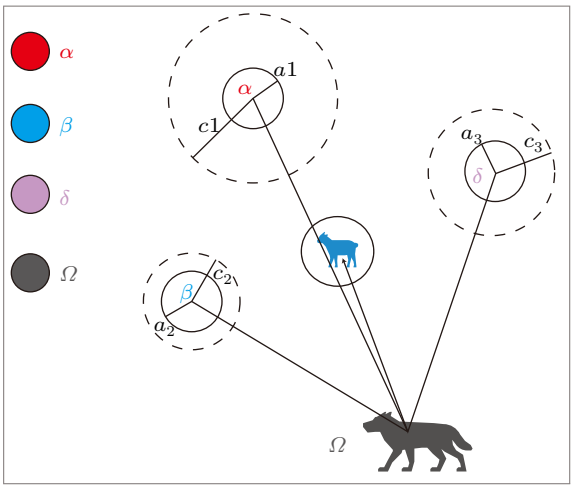


Fig. 7. Hierarchical hunting process of grey wolves in the GWO algorithm.

The GWO algorithm demonstrates excellent performance in balancing local exploitation and global exploration, resulting in high solution accuracy and fast convergence. The main stages of grey wolf hunting are summarized as follows:

(i) Tracking and approaching the prey.

(ii) Chasing, encircling, and harassing the prey until it stops moving.

(iii) Attacking the prey.

The position of an omega wolf (Ω) is updated based on a weighted combination of the positions of the three leading wolves (α, β, δ). The update procedure is described as follows.

1. *Calculate the distance from the leading wolves*: For the current position P_Ω of the omega wolf, the distances to wolves α, β, and δ are computed as

$$\begin{aligned} D_\alpha &= |C_1 \cdot P_\alpha - P_\Omega|, \quad D_\beta = |C_2 \cdot P_\beta - P_\Omega|, \\ D_\delta &= |C_3 \cdot P_\delta - P_\Omega|, \end{aligned} \tag{13}$$

where C_1, C_2, C_3 are random vectors calculated by $C = 2 \cdot r_2$, with $r_2 \in [0,1]$.

2. *Generate candidate positions*: Based on the positions and distances of the leading wolves, three candidate positions are generated

$$\begin{aligned} P_1 &= P_\alpha - A_1 \cdot D_\alpha, \quad P_2 = P_\beta - A_2 \cdot D_\beta, \\ P_3 &= P_\delta - A_3 \cdot D_\delta, \end{aligned} \tag{14}$$

where A_1, A_2, A_3 are coefficient vectors, computed by $A = 2a \cdot r_1 - a$, with $r_1 \in [0,1]$, and a decreases linearly from 2 to 0 over the course of iterations.

3. *Update new position*: The new position of the omega wolf is given by the average of the three candidate positions

$$P_\Omega = \frac{P_1 + P_2 + P_3}{3}. \tag{15}$$

The parameter a controls the balance between exploration and exploitation, decreasing with the number of iterations. The dynamic adjustment of A and C enables the algorithm to switch between global search ($|A| > 1$) and local search ($|A| < 1$). All operations are performed element-wise, making the algorithm suitable for multi-dimensional optimization problems. These equations guide the search toward the optimal solution by simulating the social hierarchy and cooperative hunting behavior of grey wolf groups.

5.2.2. Chaotic improvement on GWO

The proposed 2DHR neuron model, as a discrete dynamical system, exhibits excellent randomness and ergodicity. This property allows candidate solutions to be distributed evenly across the solution space, facilitating the discovery of better global optima. In addition, local chaotic disturbances drive solutions to explore outward, thereby escaping local optima and further enhancing their search capability. Based on these characteristics, chaotic improvements are applied to enhance the performance of the traditional GWO algorithm. These improvements include chaotic population initialization, chaotic crossover, position update with chaotic disturbance, and chaotic adaptive weighting.

(i) Chaotic population initialization

As shown in Fig. 8, the encoding and decoding scheme for DFJSP adopts a three-layer structure.

1. *Generation of chaotic sequences*: Chaotic sequences x_i and y_i are generated using the 2DHR system. Given that the value range of the chaotic sequences is $[-a, b]$, we employ Eq. (16) to normalize x_i and y_i to the range $[0,1]$ for mapping the scheduling scheme.

2. *Job and machine mapping*: Jobs to be processed are determined based on the order information. The numbers appearing in the job sequence indicate the jobs to be processed, and the frequency of each number corresponds to the number of operations required for that job. The machine index is mapped using the chaotic sequence y_i, representing the selected machine from the candidate machine set for each operation. Specifically, the candidate machines are evenly divided, and when the corresponding chaotic sequence y_i falls within a specified range, it is mapped to the machine index for that operation.

3. *Construction of scheduling scheme*: The chaotic sequence x_i is mapped to determine the operation order. The numerical order in x_i represents the scheduling sequence of operations, and the processing machine is determined based on the mapping obtained in step 2 to obtain a complete scheduling scheme. The length of the chaotic sequence is equal to that of the job sequence

$$\begin{cases} x_i = \dfrac{x_i' + a}{a + b}, \\ y_i = \dfrac{y_i' + a}{a + b}. \end{cases} \tag{16}$$

Here, a and b represent the value ranges of the chaotic sequences, and x_i and y_i are the resulting normalized chaotic values.

(ii) Chaotic crossover

As shown in Fig. 9, the crossover process is illustrated. First, the intersection position Z (blue part in Fig. 9) is determined. The elements belonging to Z in parent p_1 are inserted sequentially into the offspring, followed by inserting elements not in Z (grey part in Fig. 9) from parent p_2 all at once to form a complete offspring.

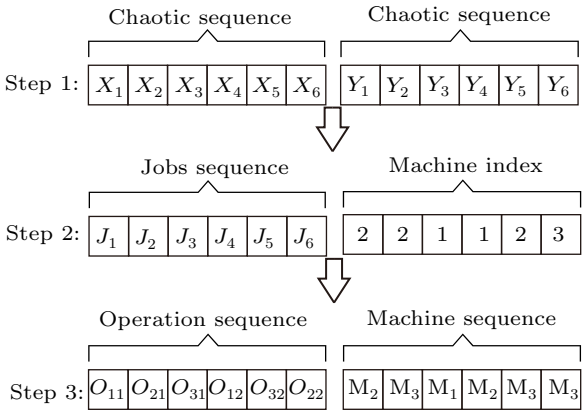


Fig. 8. Encoding and decoding scheme for the dynamic flexible job shop scheduling problem using 2DHR chaotic sequences.

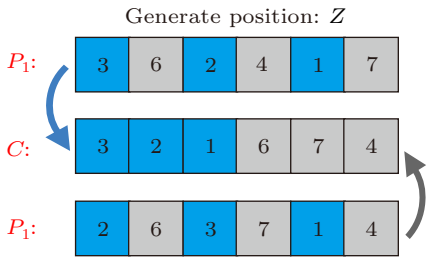


Fig. 9. Chaotic crossover process in the 2DHR-enhanced GWO algorithm.

(iii) Position update with chaotic disturbance

In DFJSP, the introduction of chaotic sequences provides an exploration direction within the solution space for crossover operations. This diversity of search strategies allows the algorithm to explore adjacent regions more effectively, thereby escaping local optima and improving the efficiency and performance of global search. Determining the intersection position using chaotic sequences is a feasible approach. Two individuals, P_1 and P_2, are randomly selected from the population as parents, and the intersection position is determined by Eq. (17) for crossover

$$z = x_i - x_1 \cdot (1-\alpha), \tag{17}$$

where x_i represents the value of the chaotic sequence, x_1 is its initial value, z is the insertion position, and α is a chaotic control parameter set to 0.8 that controls the influence intensity of the chaotic sequence.

The incorporation of chaotic disturbance into the grey wolf algorithm further enhances the global search capability and population diversity. The chaotic sequence prompts the grey wolf to conduct additional local exploration, thereby improving optimization performance. Specifically, hybrid disturbance is achieved using

$$x_{\text{new}} = x_{t+1} + \lambda(x_i - 0.5), \tag{18}$$

where x_{t+1} represents the current grey wolf individual, which is further evolved through chaotic perturbation to generate a new individual x_{new}. λ is a parameter controlling the intensity of chaotic perturbation, set to 0.1, and x_i is the first value generated by the chaotic sequence. Chaotic perturbation effectively helps the population escape local optima, increases population diversity, and enhances the global search ability and optimization performance of the algorithm.

(iv) Chaotic adaptive weight

In the grey wolf algorithm, the control parameter A typically decreases linearly with the number of iterations to balance global and local search capabilities. However, this linear change may cause the algorithm to become trapped in local optima for complex optimization problems. To address this limitation, chaotic sequences are introduced to dynamically adjust parameter A, enhancing both global search capability and population diversity. The dynamic properties of chaotic sequences provide additional flexibility and randomness in adjusting A, preventing the algorithm from prematurely falling into local optima. In the improved grey wolf algorithm, the update formula for parameter A is

$$A = A_{\text{initial}} \cdot (1 - t/T) + \gamma \cdot x_i, \tag{19}$$

where A_{initial} is the initial value of parameter A, typically set to 2, t is the current iteration number, T is the total number of iterations, and γ is the moderating factor controlling the influence of the chaotic sequence on A, set to 0.5 in this study.

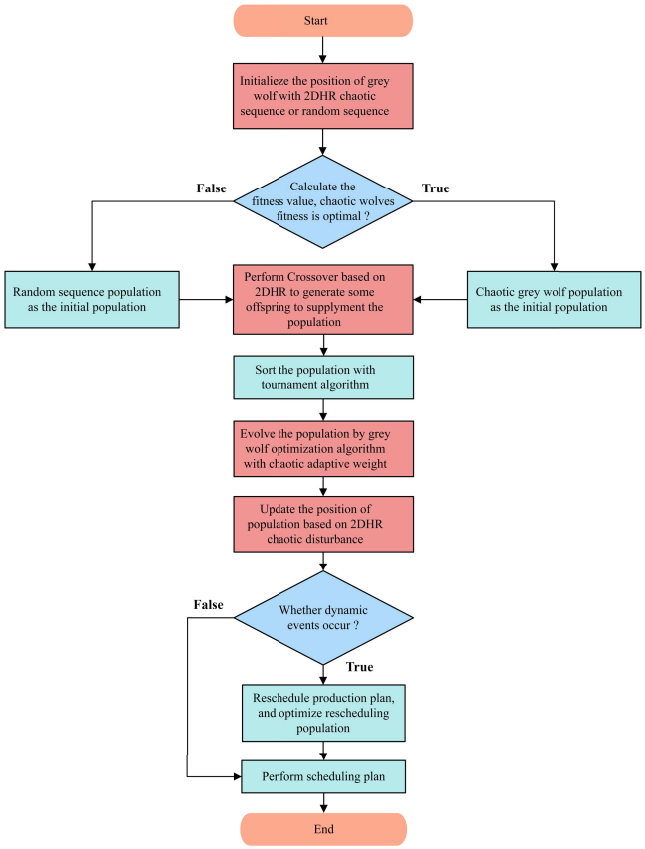


Fig. 10. Flow chart of the CGWO algorithm, highlighting the stages with 2DHR chaotic improvements.

The flowchart of the chaotic grey wolf optimization (CGWO) algorithm is shown in Fig. 10, where the four components incorporating chaotic improvements are highlighted in pink. In the initialization phase, 2DHR chaotic sequences are employed to disperse grey wolf individuals across the global solution space. Subsequently, chaotic crossover is performed to generate offspring and supplement the population. The population is then sorted using a tournament selection algorithm, followed by the application of GWO with chaotic adaptive weighting. The positions of individuals are updated through 2DHR chaotic disturbance. Once dynamic events are detected, the rescheduling mechanism is activated; otherwise, the scheduling plan is executed on the production line.

5.3. Simulation and analysis

5.3.1. Experimental setup

To accurately demonstrate the performance improvements of 2DHR chaotic sequences in enhancing the CGWO optimization algorithm, relevant experiments were conducted on DFJSP. The experimental data in this study are derived from Li *et al.*[27] The results indicate that CGWO achieves the best optimization performance. All algorithms were implemented in MATLAB R2023a on a PC equipped with an AMD Ryzen 3.60 GHz processor and 32.0 GB RAM.

The occurrence times of dynamic events are set as $t_1 = 20$, $t_2 = 40$, and $t_3 = 60$. The processing time failure of jobs is set to 20, the arrival time of new jobs to 40, and the start-up time of new machines to 60. The fixed facility energy consumption of the workshop is 35, with a transfer power of 2 per job. The initial population size is set to 500, the post-dynamic-event population size to 100, the tournament mating pool size to 100, and the number of tournament participants is set to 2.

5.3.2. Experimental result

As shown in Fig. 11, by comparing the convergence curves of different chaotic systems in the FJSP optimization problem, the superiority of 2DHR is clearly demonstrated. Analysis shows that 2DHR is significantly superior to other chaotic systems and non-chaotic optimization in terms of convergence speed and final solution quality. It reaches a stable state after approximately 40 iterations, whereas other algorithms require a greater number of iterations to converge.

The performance improvement from chaotic systems is not guaranteed. For example, the Tent map is significantly inferior to non-chaotic optimization methods on both indicators, and the advantage of the Gauss/mouse map on the energy consumption indicator is minimal. This demonstrates that a poorly chosen chaotic system can even undermine optimization performance.

Regarding the makespan indicator, the performance of 2DHR surpasses the logistic map, Chebyshev map, Gauss/mouse map, Tent map, and non-chaotic optimization by approximately 2.33%, 2.33%, 6.67%, 20.75%, and 8.70%, respectively. For energy consumption, the corresponding improvements are 2.29%, 2.29%, 6.57%, 19.27%, and 8.00%, respectively. In addition, 2DHR provides a better initial population distribution, which allows individuals to approach the optimal solution more quickly, whereas other algorithms require more iterations to gradually improve the initial solution.

In conclusion, when addressing optimization problems with 2DHR chaotic systems, the 2DHR algorithm not only converges rapidly but also achieves superior solutions, with relatively high-quality initial solutions. These characteristics make it an effective approach for solving such problems.

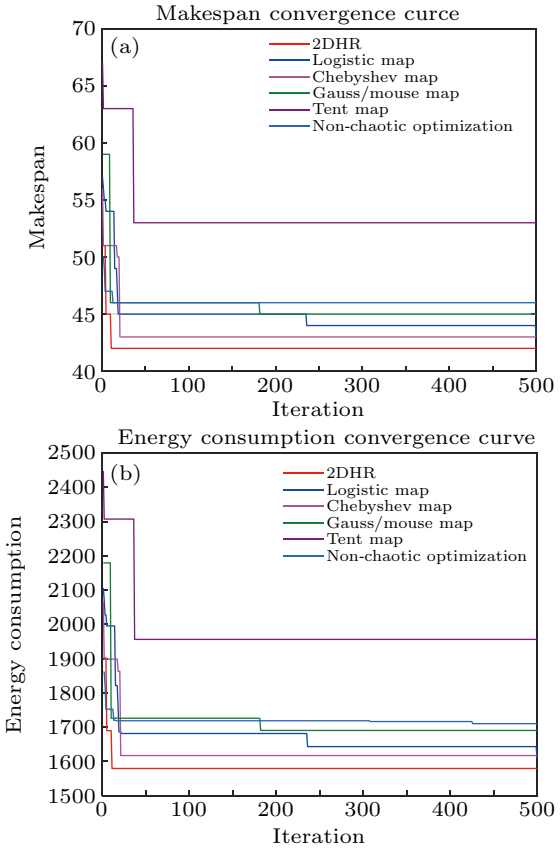


Fig. 11. Convergence curves of initial population fitness for different chaotic systems, illustrating the superior convergence speed and solution quality of the 2DHR-based CGWO algorithm.

As shown in Table 2, the proposed 2DHR chaotic system demonstrates significant advantages in solving the FJSP. Evaluated on ten benchmark instances, the 2DHR algorithm achieves optimal or near-optimal performance in most scenarios. It exhibits superior performance over the Gauss/mouse map and the Tent map in both makespan and energy consumption metrics, while also showing modest but consistent improvements compared to the logistic map and Chebyshev map in most cases. Specifically, in terms of makespan, 2DHR outperforms the Gauss/mouse map by approximately 23.28% and the Tent map by approximately 30.19% on average. Regarding energy consumption, it surpasses the Gauss/mouse map by approximately 22.57% and the Tent map by approximately 29.23% on average. Compared to the logistic map and Chebyshev map, 2DHR maintains comparable or slightly better performance in most tasks, with more noticeable benefits in complex instances (e.g., M8, M9, M10). These results indicate that the 2DHR algorithm is highly effective in optimizing operation sequencing and machine allocation, providing superior scheduling solutions to meet practical production requirements. The performance also reflects the high efficiency of the 2DHR system in conducting global search. Overall, the 2DHR chaotic system not only exhibits theoretical compatibility with FJSP but also demonstrates excellent practical applicability. This robust performance can be attributed to its strong nonlinear fitting capability and adaptive learning abil-

ity, which facilitate the identification of near-global optimum solutions. Therefore, for complex optimization problems — particularly those requiring global search and adaptive learning — the 2DHR chaotic system exhibits unique advantages and promising application potential.

After optimization through non-dominated sorting, tournament selection, and CGWO, neighborhood search via 2DHR chaotic disturbance is applied to elite solutions to obtain higher-quality solutions. In Fig. 12(a), red points denote the optimal positions of CGWO individuals, while the blue regions represent the fitness values of neighboring searches, where lower fitness values indicate superior search performance. The results demonstrate the existence of better solutions in the vicinity of the current optimum, highlighting the efficacy of 2DHR chaotic disturbance. Figure 12(b) illustrates the expanded solution space, confirming the capability of chaotic disturbance to guide grey wolves toward improved solutions. Decoding the optimal solution from Fig. 12(b) yields the initial schedule's Gantt chart shown in Fig. 13(a), with a final makespan of 90 time units.

The dynamic events considered in this study represent common occurrences in DFJSP production environments, closely reflecting real-world manufacturing scenarios, thus possessing significant practical implications. As depicted in Fig. 13(b), the first dynamic event involves the processing time failure of a job. At $t = 20$, due to machine aging or operator non-compliance, operations $O_{7,2}$ require special treatments (e.g., surface grinding, precision calibration), resulting in extended processing times. Before $t = 20$, jobs follow the original schedule. The final makespan after this event reaches 101 time units.

As shown in Fig. 13(c), the arrival of a new job at $t = 40$ would significantly increase the current schedule's makespan if incorporated without optimization. The newly arrived job 9 (containing three operations) disrupts the original schedule, resulting in a final makespan of 101. Prior to $t = 40$, jobs follow the original schedule.

Table 2. Comparison of the fitness of different chaotic systems in the initialization stage.

	Metric	2DHR	Logistic map	Chebyshev map	Gauss/mouse map	Tent map
M1	Makespan	47	52	52	72	48
	Energy	1.755×10^4	1.930×10^4	1.930×10^4	2.632×10^4	1.790×10^4
M2	Makespan	42	43	43	45	48
	Energy	1.579×10^4	1.609×10^4	1.616×10^4	1.690×10^4	1.788×10^4
M3	Makespan	256	251	259	293	359
	Energy	9.253×10^4	9.071×10^4	9.364×10^4	1.051×10^5	1.284×10^5
M4	Makespan	86	88	88	176	200
	Energy	3.192×10^4	3.264×10^4	3.254×10^4	6.346×10^4	7.162×10^4
M5	Makespan	214	216	216	293	307
	Energy	7.695×10^4	7.761×10^4	7.752×10^4	1.047×10^5	1.090×10^5
M6	Makespan	131	138	131	202	272
	Energy	4.882×10^4	5.133×10^4	4.889×10^4	7.360×10^4	9.819×10^4
M7	Makespan	233	236	234	264	366
	Energy	8.357×10^4	8.457×10^4	8.393×10^4	9.442×10^4	1.299×10^5
M8	Makespan	591	596	598	615	724
	Energy	2.118×10^5	2.134×10^5	2.142×10^5	2.201×10^5	2.578×10^5
M9	Makespan	418	428	429	559	616
	Energy	1.512×10^5	1.548×10^5	1.551×10^5	2.006×10^5	2.202×10^5
M10	Makespan	421	428	429	559	616
	Energy	1.523×10^5	1.549×10^5	1.551×10^5	2.006×10^5	2.202×10^5

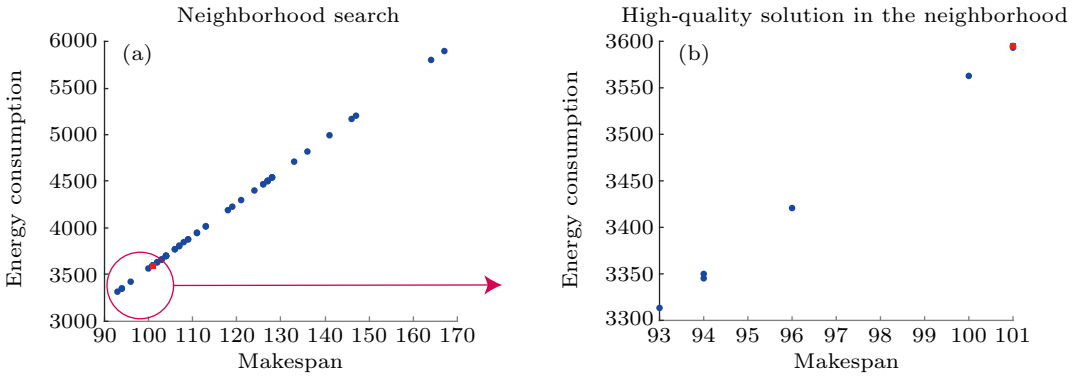


Fig. 12. Neighborhood search performance based on 2DHR chaotic disturbance. Red points indicate the optimal positions of CGWO individuals, while blue regions represent the fitness landscape of neighboring searches, showing the ability of chaotic disturbance to guide solutions toward better optima.

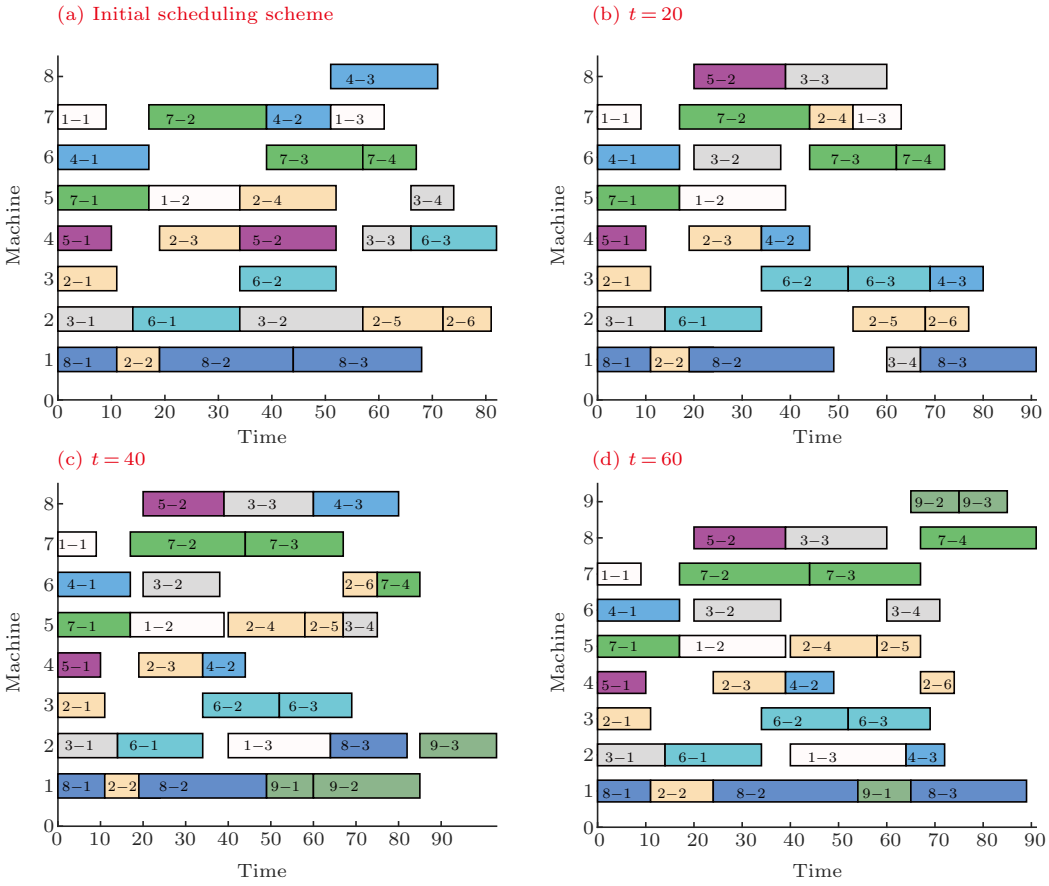


Fig. 13. Processing schemes under multiple dynamic events.

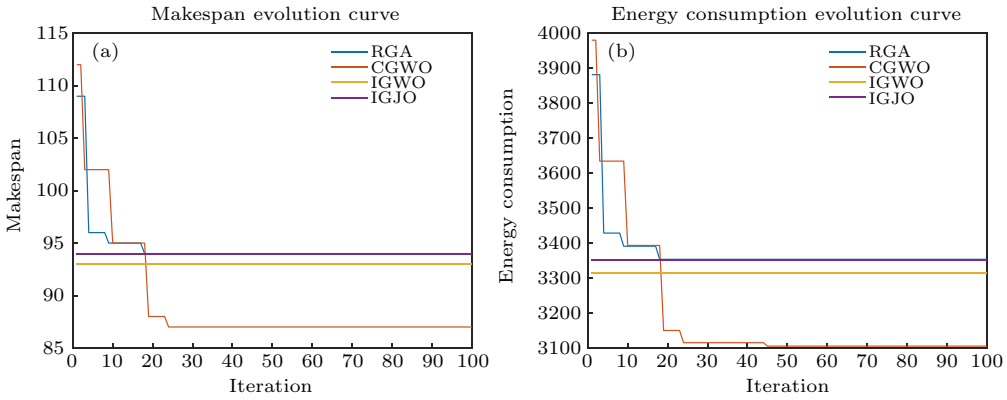


Fig. 14. Evolutionary curves of workshop makespan and total energy consumption.

As shown in Fig. 13(d), machine 9 is started up and integrated into the production line at $t = 60$ after debugging. The addition of this machine reduces the current makespan and accelerates task completion. By rescheduling unprocessed jobs to the newly available machine, the final makespan is reduced to 92, demonstrating that machine start-up effectively decreases production times, consistent with practical manufacturing scenarios. Before $t = 60$, jobs are scheduled according to the original scheme.

Figure 14 illustrates the evolution curves of workshop makespan and total energy consumption. In this study, four different algorithms — random generation strategy (RGS), CGWO, improved grey wolf optimization (IGWO), and chaotic golden jackal optimization (CGJO) — were compared to evaluate their convergence speed and solution quality. The results indicate that the CGWO algorithm, when combined with the 2DHR system, gradually approaches an approximate optimal solution within the first 20 iterations. Subsequently, it continues to explore the solution space more effectively due to the combined effects of chaotic perturbation, chaotic crossover, and chaotic adaptive weighting. In contrast, IGWO and CGJO become trapped in local optima from the outset, and their algorithmic characteristics prevent them from escaping, resulting in relatively flat evolutionary curves. While RGS exhibits some optimization during the initial 20 generations due to its randomness, its purely stochastic nature limits its opti-

mization performance compared to CGWO.

In this study, the performance of the algorithms is evaluated using the hypervolume (HV) and the inverted generational distance (IGD). Maximizing HV and minimizing IGD indicate better overall quality of the solution set. The key to calculating IGD lies in establishing the true Pareto front. Here, the non-dominated solutions obtained from multiple experiments across different instances are used as an approximate true Pareto front for IGD calculation. The HV and IGD evaluation results for RGS, CGWO, IGWO, and CGJO are presented in Table 3, where the black bolded entries indicate superior performance compared to the other algorithms. Clearly, CGWO outperforms the other three algorithms in nearly all cases. This superiority demonstrates that the dynamic characteristics of the 2DHR chaotic system provide CGWO with richer search strategies and enhanced optimization mechanisms. The chaotic initialization process allows the grey wolves to occupy the solution space more uniformly. Subsequently, chaotic perturbations drive the grey wolves to explore the vicinity of optimal solutions further, fully leveraging the dynamic properties of the 2DHR system. Consequently, CGWO can effectively locate global optimal solutions for the DFJSP, achieving excellent multi-objective optimization performance.

Table 3. Comparison of HV and IGD results among the four algorithms.

Indicator	Instance	RGS	CGWO	IGWO	CGJO
HV	MK01	0.5010	**0.5529**	0.4870	0.4970
	MK02	0.4471	**0.5250**	0.4691	0.4750
	MK03	0.4518	**0.5681**	0.5249	0.3987
	MK04	0.4983	**0.6013**	0.5714	0.4950
	MK05	0.3090	**0.4784**	0.4252	0.3821
	MK06	0.3256	**0.5548**	0.5515	0.4352
	MK07	0.2425	**0.3355**	0.2824	0.2990
	MK08	0.1595	**0.4651**	0.4319	0.1761
	MK09	0.1728	**0.4020**	0.3920	0.1993
	MK10	0.1794	**0.4385**	0.3987	0.2625
IGD	MK01	0.0498	**0.0397**	0.0498	0.0502
	MK02	0.1213	**0.1210**	0.1256	0.1231
	MK03	**0.0443**	0.1440	0.0990	0.1817
	MK04	0.1939	**0.0096**	0.0122	0.2936
	MK05	0.1246	**0.0157**	0.0280	0.2545
	MK06	0.1390	**0.0057**	0.0945	0.3320
	MK07	0.0768	**0.0604**	0.1056	0.1735
	MK08	0.1031	**0.0188**	0.0580	0.6422
	MK09	0.0712	**0.0083**	0.0335	0.5007
	MK10	0.1218	**0.0242**	0.0415	0.5016

6. Conclusion

This study proposes a novel 2DHR, derived by simplifying the classical 2DHR model. The 2DHR model exhibits unique chaotic dynamics not observed in traditional models. Through comprehensive dynamic analysis, including bifurcation diagrams, Lyapunov exponent spectra, equilibrium points, and phase portraits, the complex dynamic behaviors and pathways to chaos of the model are thoroughly revealed. To validate the physical feasibility of the 2DHR system, a corresponding analog circuit was designed and implemented, and the simulation results exhibit excellent agreement with numerical simulations.

Furthermore, the proposed 2DHR model is integrated into a CGWO-based DFJSP framework. Compared to IGWO, CGJO, and initial rescheduling populations, CGWO achieves superior optimization performance in DFJSP. Under various dynamic events and constraints, CGWO effectively identifies optimal scheduling schemes, significantly reducing both makespan and energy consumption. Future work will focus on the physical implementation of DFJSP optimization based on the CGWO algorithm, leveraging the chaotic 2DHR system in real-world manufacturing applications.

Acknowledgments

This work was supported by the Graduate Research Fund of Guizhou Province (Grant No. 2024YJSKYJJ165), the National Natural Science Foundation of China (Grant No. 62061008), and the Guizhou Provincial Basic Research Program (Natural Science) (Grant No. Qian Ke He Ji Chu-ZK [2024] General 439).

References

[1] Hodgkin A L and Huxley A F 1952 *J. Physiol.* **117** 500
[2] Hindmarsh J and Rose R 1982 *Nature* **296** 162
[3] Hindmarsh J and Rose R 1984 *Proc. R. Soc. B* **221** 87
[4] Cai J, Bao H, Chen M, Xu Q and Bao B 2022 *IEEE Trans. Circuits Syst. I Regul. Pap.* **69** 2916
[5] Linaro D, Poggi T and Storace M 2010 *Phys. Lett. A* **374** 4589
[6] Innocenti G, Morelli A, Genesio R and Torcini A 2007 *Chaos* **17** 043103
[7] Shilnikov A and Kolomiets M 2008 *Int. J. Bifurcat. Chaos* **18** 2141
[8] Storace M, Linaro D and Lange E 2008 *Chaos* **18** 033128
[9] Li B, Liang H and He Q 2021 *Chaos Solitons Fractals* **146** 110856
[10] Bao B, Hu A, Bao H, Xu Q, Chen M and Wu H 2018 *Complexity* **2018** 3872573
[11] Vijay S D, Thamilmaran K and Ahamed A I 2023 *Nonlinear Dyn.* **111** 789
[12] Wang Y 2022 *Phys. Scr.* **97** 075202
[13] Njitacke Z T, Fozin T F, Muni S S, *et al.* 2022 *AEU Int. J. Electron. Commun.* **155** 154361
[14] Xie Y, Kang Y, Liu Y, *et al.* 2014 *Sci. China Technol. Sci.* **57** 914
[15] Yang Y and Liao X 2018 *IEEE Trans. Neural Netw. Learn. Syst.* **30** 306
[16] Li Z, Xie W, Zeng J and Zeng Y 2023 *Chin. Phys. B* **32** 010503
[17] Rajagopal K, Khalaf A J M, Parastesh F, *et al.* 2019 *Nonlinear Dyn.* **98** 477
[18] Guo Z, Li Z, Wang M and Ma M 2023 *Chin. Phys. B* **32** 038701
[19] Belykh V N, Osipov G V, Petrov V S, *et al.* 2008 *Chaos* **18** 037106
[20] Malik S A and Mir A H 2020 *Neural Netw.* **123** 372
[21] Fan C, Wang W and Tian J 2024 *J. Manuf. Syst.* **74** 180
[22] Zhang Y, Yao X and Xu S 2024 *J. Mech. Sci. Technol.* **38** 5581
[23] Xu Y, Wang D, Zhang M, Yang M and Liang C 2025 *Swarm Evol. Comput.* **93** 101836
[24] Plan J L and Yorke J A 1979 In: *Functional Differential Equations and Approximation of Fixed Points* (Springer)
[25] Baykasoğlu A, Madenoğlu F S and Hamzadayı A 2020 *J. Manuf. Syst.* **56** 425
[26] Duan J and Wang J 2022 *Expert Syst. Appl.* **203** 117489
[27] Li Y, Tao Z, Wang L, Du B, Guo J and Pang S 2023 *Rob. Comput. Integr. Manuf.* **79** 102443