基于导向差分进化算法的党务活动调度优化方法

孙佩铭[1]，王 喆[2]

（1. 吉林大学 通信工程学院，长春 130012；2. 吉林大学 大数据和网络管理中心，长春 130015）

摘 要：为解决项目活动时间安排不合理、经费开销大的问题，提出了一种基于导向交叉机制的改进差分进化算法（DirDE）。导向交叉机制通过引导种群的全局搜索和局部开发方向提升算法收敛速度。同时，该机制基于父母个体的基因导向交叉帮助算法跳出局部搜索，避免陷入局部最优。实验部分设计了基准函数实验验证DirDE算法的寻优能力。实验分析结果展示，DirDE表现出更好的收敛性、精度及避免陷入局部最优的能力。最后，本文方法在真实的多项目党务活动调度优化线性规划模型上进行模拟实验，算法展现出竞争力，可作为现实党务活动调度问题求解的有效工具。

关键词：人工智能技术；进化算法；差分进化算法；党务活动调度；导向交叉机制

中图分类号：TP391.7 **文献标志码**：A **文章编号**：1671-5497(2025)08-2761-10

DOI：10.13229/j.cnki.jdxbgxb.20240737

Optimization method of party affairs activity scheduling based on directional differential evolution algorithm

SUN Pei-ming[1], WANG Zhe[2]

(1. *College of Communication Engineering*, *Jilin University*, *Changchun* 130012, *China*; 2. *Big Data and Network Management Center*, *Jilin University*, *Changchun* 130015, *China*)

Abstract: To address the issues of unreasonable project activity scheduling and high expenditure, this paper proposes an improved differential evolution algorithm based on directional crossover mechanism (DirDE). The mechanism enhances the algorithm′s convergence speed by guiding the global exploration and local exploitation directions of the population. Meanwhile, this mechanism helps the algorithm jump out of the local search and avoid falling into the local optimum based on the gene guidance of the parent individuals. In the experimental section, benchmark function experiments are designed to verify the optimization ability of the DirDE. The experimental analysis results demonstrate that DirDE exhibits better convergence, accuracy, and the ability to avoid falling into local optima. Finally, the method proposed in this paper was tested through simulation experiments on a real-world linear programming model for multi-project party affairs activity scheduling optimization. The algorithm demonstrated its competitiveness and can serve as an

收稿日期：2024-07-04.

基金项目：吉林省重点研发项目（20190302027GX）.

作者简介：孙佩铭（1985-），女，助理研究员，博士研究生. 研究方向：党务工作和党务系统. E-mail：sunpm@jlu.edu.cn

通信作者：王喆（1979-），男，高级工程师，硕士. 研究方向：计算机网络与大数据分析. E-mail：wangzhe@jlu.edu.cn

effective tool for solving real-world party affairs activity scheduling problems.

Key words: artificial intelligence technology; evolutionary algorithm; differential evolution algorithm; party affairs activity scheduling; directional crossover mechanism

0 引 言

差分进化算法(Differential evolution algorithm, DE)[1]是用于解决连续型问题的群智能优化算法,被广泛应用在医学图像分割[2]、路径规划[3]、资源调度[4]和机器学习等领域[5]。Cuevas等[6]提出了一种基于DE的新颖自动图像多阈值方法,该方法利用DE计算高斯函数的系数来填充图像的一维直方图,实现自动阈值选择;Ayala等[7]提出了一种用于确定图像分割所需的阈值组合的BDE算法,使用Otsu准则成功改善了解决方案的质量和收敛性;Liu等[8]根据黏液菌群的觅食行为提出了一种改进的DE,即MDE,并基于MDE开发了一个多层级图像分割模型,用于乳腺癌分割,该方法的性能通过实验证明了有效性;Tarkhaneh等[9]提出了ALDE,通过将自适应方法和新的变分策略融入DE中,以解决MRI脑图像分割问题;Xu等[10]提出了一种基于记忆算法的蜻蜓算法(DA)和DE,利用Otsu和最小交叉熵(MCE)确定彩色图像分割的最佳阈值;Chen[11]提出了一种改进的DE,称为AGDE,它结合了混沌游戏优化(CGO)的种子更新方法和关联策略(AS)。在CGO中提出了4种种子更新方法,使得不同位置的个体具有多次位置变化的机会,具有一定的随机性和接近最优解的特性,并将这4种种子更新方法融入DE中,增加了算法的种群多样性,并增加了算法跳出局部最优解的机会。Yang等提出将基于遗传算法应用于党务管理中的会议日程优化研究。该论文研究了如何利用遗传算法来优化党务管理中的会议日程安排,提高会议效率和资源利用率。但是,在多项目党务活动调度优化问题中,目前研究算法的收敛速度、精度和避免陷入局部最优的能力还具有提升的空间。

为解决多项目党务活动的调度复杂和困难的问题,本文提出了基于导向交叉机制的差分进化算法(DirDE)应用于多项目党务活动调度优化问题。其中,导向交叉机制在DirDE中增强了算法的开发和探索的平衡能力、种群的多样性以及算法跳出局部最优能力。进一步,本文设计了基于IEEE Congress on Evolutionary Computation (CEC)基准函数集的性能测试实验验证DirDE的性能。实验结果经过标准差、均值、威尔逊排名检验[12]和弗瑞德曼检测[13]等统计性方法分析。无论在哪种统计性方法的分析下,DirDE都展示了出色的优化能力。同时,本文将DirDE应用到多项目党务活动调度优化模拟实验。

1 基于导向交叉机制的差分进化算法

1.1 导向交叉机制的原理

参考文献[14]发现,根据最优个体的信息,导向交叉策略(DX)可以获得强大的搜索能力。受该文献的启发,将DX策略引入原始的差分进化算法可以提高收敛速度和准确性。DX策略的原理如下。

DX策略中的关键参数包括交叉概率(p_{cv})、方向概率(p_d)和乘法因子(α)。p_1和p_2表示两个随机个体。p^j_{mean}和p^j_{best}是第j个随机个体的平均值和最佳值。DX的公式如下所示。

$$\mathrm{val}=\begin{cases}1-0.5^{e^{\left[\frac{\left|p^j_1-p^j_2\right|}{\left(y^j_u-y^j_l\right)}\right]}}, & p^j_1\neq p^j_2\\ 1-0.5^{e^{\left[\frac{\left|p^j_{best}-p^j_{mean}\right|}{\left(y^j_u-y^j_l\right)}\right]}}, & p^j_1=p^j_2, p^j_{best}\neq p^j_{mean}\end{cases} \tag{1}$$

$$\beta=\frac{r}{\alpha^2} \tag{2}$$

式中:val和β为随迭代而变化的两个变量;r为一个随机变量,且$r\in(0,1)$;y^j_u和y^j_l分别为第j个维度中目标函数的上界和下界。

$$c^j_1=\mathrm{val}\times\left(p^j_1\pm p^j_2\right)\pm \alpha^{r_1}e^{(1-\beta)}(1-\mathrm{val})\times\left|p^j_1-p^j_2\right| \tag{3}$$

$$c^j_2=(1-\mathrm{val})\times\left(p^j_1\pm p^j_2\right)\pm \alpha^{(1-r_1)}e^{(-\beta)}\mathrm{val}\times\left|p^j_1-p^j_2\right| \tag{4}$$

$$c^j_1=\mathrm{val}\times\left(p^j_{best}+p^j_{mean}\right)\pm \alpha^{r_1}e^{(1-\beta)}(1-\mathrm{val})\times\left(p^j_{best}-p^j_{mean}\right) \tag{5}$$

$$c^j_2=(1-\mathrm{val})\times\left(p^j_{best}+p^j_{mean}\right)\pm \alpha^{(1-r_1)}e^{(-\beta)}\mathrm{val}\times\left(p^j_{best}-p^j_{mean}\right) \tag{6}$$

方程式(3)~(6)将根据p_d确定式中的±。c_1

和c_2是由DX策略生成的新个体,其中$\alpha=0.95$,$r_1\in(0,1)$。假设个体的维度为d。

1.2　DirDE算法及复杂性分析

DirDE的结构主要包括种群初始化、基因变异、基因交叉和基于导向交叉机制的种群更新策略。其中,基因交叉策略是随机性地对两个体位置进行交流。导向交叉机制则利用种群最优个体作为父母个体,基于父母个体的位置信息结合缩放因子进行导向交叉,引导算法收敛到最优值。随后,DirDE利用贪婪选择机制对种群的个体进行筛选,提高算法的收敛速度。其中,图1展示了DirDE算法的优化流程,导向交叉机制作为算法的关键策略,帮助算法更智能地在搜索空间进行搜索。本文提出结合导向交叉机制的DirDE,通过引导种群进行基因的交叉和变异操作提升了种群多样性、搜索能力以及避免陷入局部最优的能力。

附录表A1中的算法1是DirDE的伪代码,详细解释了本文提出的基于导向交叉机制的差分进化算法的改进结构。假设算法的种群大小为N,评估次数为Maxfes,个体维度为D。原始DE算法的算法复杂度是$O(N+\text{Maxfes}(N{\cdot}D))$。其中,DirDE算法的复杂度由种群初始化策略、差分进化策略和导向交叉机制的复杂度组成。因此,DirDE的总体复杂度为$O(N+\text{Maxfes}(N{\cdot}2D))$,与DE算法的算法复杂度相似,可近似为$O(\text{Maxfes}(N{\cdot}2D))$。

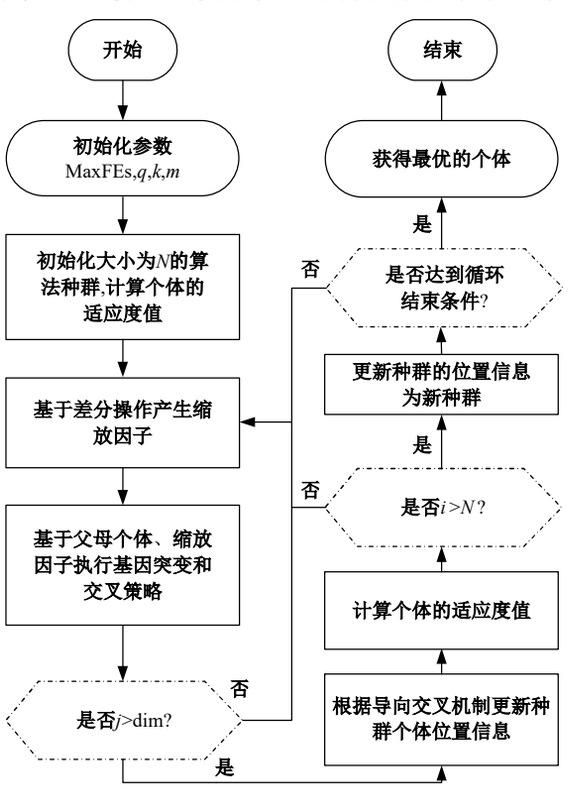

图1　基于导向交叉机制的DirDE流程图

Fig. 1　Flowchart of DirDE

2　实验仿真及分析

2.1　实验设置

基于IEEE Congress on evolutionary computation(CEC)的基准函数集,本文设计了DirDE与其他优秀的群智能优化算法进行了性能的验证和对比。函数实验中的对比算法包括DE[1]、SCA、PSO[15]、CS、MFO、ACOR等优秀的群智能优化算法。实验分析结果通过标准差(STD)、平均值(mean)、威尔逊排名检验(WSRT)和弗瑞德曼检测(FT)等统计分析方法评估。

为确保实验公平,所有算法在相同条件下测试。对比算法使用评估原理(FEs)限制循环次数,在计算个体适应度时,评估次数增加。评估原则保证实验的有效性和确定性,MaxFEs为30万。算法种群规模为30,为降低随机性,将随机测试30次。

2.2　基准函数优化分析及平衡性分析

表1为本次实验的威尔逊排名检验结果,DirDE威尔逊排名第一,"Mean"是弗瑞德曼检测结果,指示DirDE在弗瑞德曼检测中表现最优秀(1.588)。相对其他算法,DirDE具有明显的求解优势。

实验结果显示,DirDE相对差分进化算法(DE),在6个函数中获得更好的优化结果,且在4个函数上表现类似。在整个函数集中,DirDE的表现均优于SCA和PSO。综合分析,DirDE在函数集上收敛精度更高。附录表A2为对比实验结果,含标准差(STD)和平均值(AVG)。STD代表算法在函数求解中的稳定性,AVG表现准确性。STD越小,算法性能越稳定;AVG越小,算

表1　WSRT和FT的分析结果

Table 1　Analysis result of WSRT and FT

	+/−/=	Mean	Rank
DirDE	—	1.588	1
DE	6/0/4	2.460	2
SCA	10/0/0	4.466	5
PSO	9/0/1	4.756	6
CS	8/0/2	4.063	4
MFO	8/0/2	3.468	3

法性能越准确。表中指出每个函数上表现最优秀的算法。在F1至F10上,DirDE均获得最佳或并列最佳STD和AVG。因而,DirDE是表现最精确、稳定的函数集算法。

图2中展示了对比算法的收敛曲线。本文给出F1和F3上的算法收敛曲线。在F1的收敛曲线中,DirDE展示了跳出局部最优的能力,而且基于导向交叉机制的DirDE能够在搜索前期两次跳出局部最优,获得最好的结果。在F3的收敛曲线中,DirDE能够更快速地找到基准函数的最优值。同时,算法展现出更好的收敛速度和精度。其中,DirDE在搜索初段的收敛速度最快。在DE、MFO、PSO和SCA等算法收敛前,DirDE已经收敛。总之,收敛曲线展示了DirDE在函数集上更好的收敛速度、精度和跳出局部最优的能力。

进一步,本部分为了测试DirDE的稳定性和平衡性,测试了DirDE和原算法DE的算法平衡性。图3展示了DirDE在基准函数集的F1和F3函数上算法的全局探索和局部开发的平衡结果。

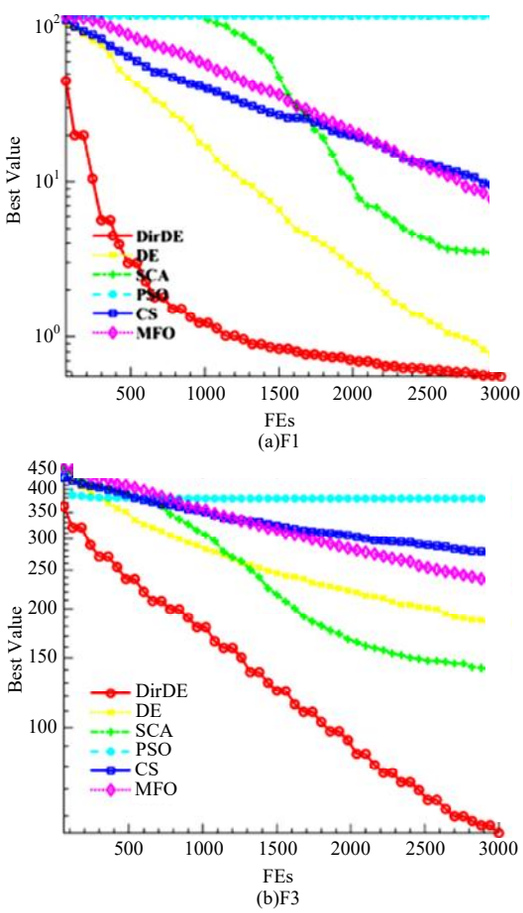

图2 函数收敛曲线

Fig. 2 Convergence curve of this experiment

图3中展示在函数F1上原始算法的全局探索能力是10.815 2%,局部开发能力是89.184 8%;DieDE法的全局探索能力是7.485 8%,局部开发能力是92.514 2%。DirDE通过提升算法的局部开发能力提升了算法跳出局部最优的能力。综合函数实验的评估结果,DirDE结合导向交叉机制提升了种群多样性,拥有更好的函数优化能力。DirDE是有效的函数优化工具。

2.3 DirDE的参数敏感性实验

DirDE的关键参数是变异率(p_{cr}),算法的性能受p_{cr}值的影响。在本节中,设计了关于不同变异率值对DirDE性能影响的参数敏感性实验。理论上,参数p_{cr}的范围是(0,1]。在实际实验中,p_{cr}的取值分别设为0.2、0.4、0.6、0.8和1。表2显示了不同变异率值下DirDE的性能表现。表2中的值为排名。当$p_{cr}=0.8$时,DirDE在10个函数(如F3、F5和F7)上均排名第一,整个函数集中平均排名也是第一。综上所述,当变异率为0.8时,DirDE在优化函数时表现更好。为确保算法的稳定性,本工作中DirDE的p_{cr}取值为0.8。

2.4 基于IEEECEC 2017实验分析

为测试DirDE算法在函数表现中的迁移能力,本部分在IEEE CEC2017的多类型函数集上进行了DirDE和其他优秀群智能算法之间的比较实验。表3展示了多类型函数集中的函数,包含多模态函数、单模态函数、组合函数和混合函数。

表4为本次实验的威尔逊排名检验结果,DirDE威尔逊排名第一,"Mean"是1.922。相对其他算法,DirDE具有明显的求解优势。其中,DirDE算法比ACOR算法在函数集上有7个函数表现更优秀。

附录表A3展示了在IEEE CEC2017函数集对比实验结果的AVG和STD分析。在F1至F9上,DirDE均获得最佳或并列最佳AVG和STD,表现出最精确、稳定的优化能力。进一步,表5显示了WSRT得到的P-value值。DirDE在IEEE CEC2017的大多数函数上获得的P-value值小于0.05,这表明该消融实验结果在统计意义上具有显著的可靠性。

图4展示了对比算法的收敛曲线。根据本文给出在IEEE CEC 2017函数集F2和F8上的算法收敛曲线,DirDE都展示了比SCA、PSO、BA等算法更好的全局收敛能力。DirDE能够在搜索前

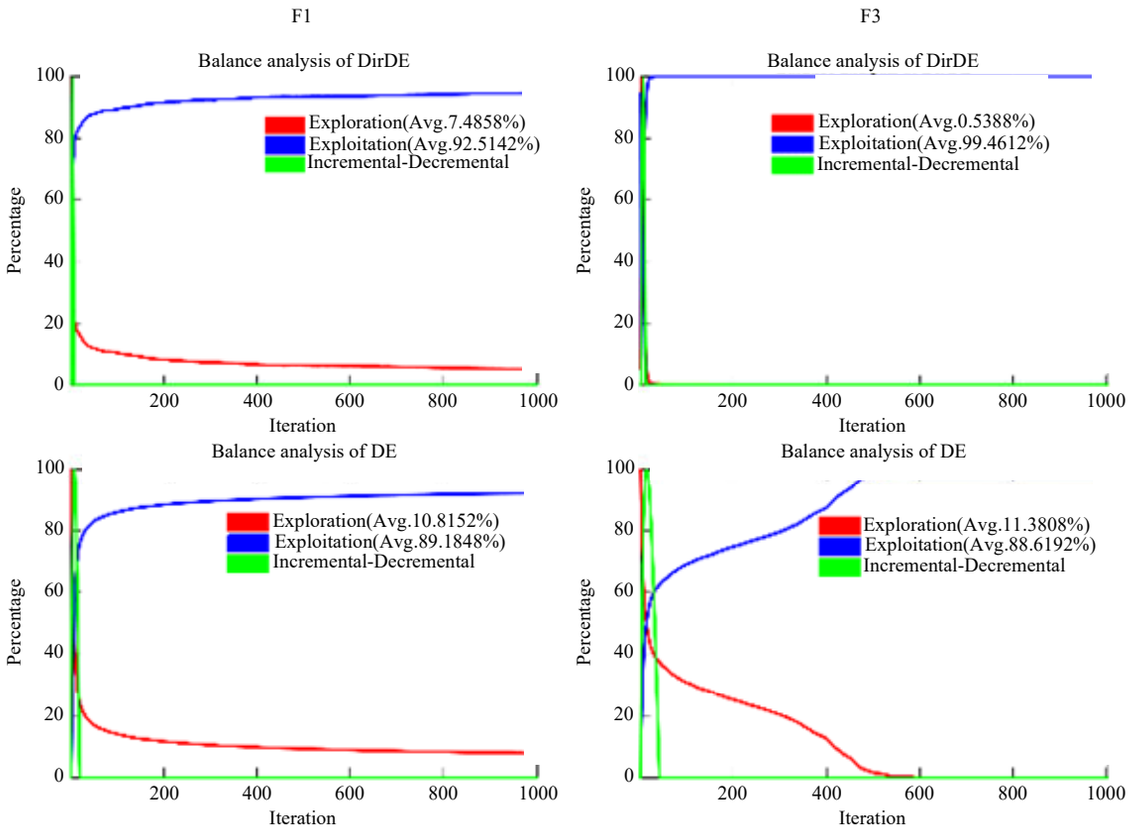

图 3　平衡性分析实验

Fig. 3　Balance analysis experiment

表 2　WSRT 的排名结果

Table 2　Rank result of WSRT

	DirDE	DirDE1	DirDE2	DirDE3	DirDE4
p_{cr}	0.8	0.2	0.4	0.6	1
F1	3	5	4	1	2
F2	2	3	1	4	5
F3	1	5	3	2	4
F4	4	5	3	1	2
F5	1	5	2	3	4
F6	1	1	5	1	1
F7	2	4	1	5	3
F8	1	2	3	5	4
F9	5	1	3	4	2
F10	4	2	5	3	1
AVG	2.4	3.3	3	2.9	2.8
rank	1	5	4	3	2

期迅速地搜索到搜索空间的最优值区域,进而展开局部开发,而且算法始终保持最小的目标值。

基于IEEE CEC 2017函数集的DirDE比较实验结果分析,DirDE算法在面对多类型函数时,如多模态函数、单模态函数和组合函数等,都表现出更加稳定的优秀优化能力。最后,实验结果也通过P-value的统计性分析得到可靠性的验证。

2.5　党务活动调度优化应用

随着党员增加和党务工作发展,党务活动优化问题已成研究重点。调度优化需同时考虑约束条件和目标函数,其中约束条件包含控制变量、等式/不等式以及线性/非线性形式。

传统解决多项目党务调度优化问题的方法包括梯度信息和线性规划模型,但存在精度低和效率问题。近年来,群智能算法逐渐成为热门优化方法,可处理不可微分和大数据问题。因此,本文采用DirDE解决党务活动调度优化问题,并进行实验。党务活动调度的优化问题的优化目标是最小化成本问题。总活动时长假设是早9点到晚16点(x_1)、总活动项目数量(x_2)、各项目花费(x_3)和各项目时长(x_4)是党务活动调度优化问题中的4个关系变量。

本实验基于真实的多项目党务活动调度线性规划模型,其中,式(7)给出了党务活动线性规划模型中的4个优化变量;式(8)展示了求解多项目党务活动优化最小活动经费的目标函数。进一

表3 IEEE CEC 2017函数集

Table 3 Details of IEEE CEC 2017

		Functions	f_{min}
Unimodal Functions	F1	Shifted and Rotated Bent Cigar Function	100
Multimodal Functions	F2	Shifted and Rotated Rastrigins Function	500
	F3	Shifted and Rotated Lunacek Bi_Rastrigin Function	700
	F4	Shifted and Rotated Non-Continuous Rastrigins Function	800
Hybrid Functions	F5	Hybrid Function 2 (N=3)	1 200
	F6	Hybrid Function 5 (N=4)	1 500
	F7	Hybrid Function 6 (N=5)	1 900
Composition Functions	F8	Composition Function 2 (N=3)	2 200
	F9	Composition Function 10 (N=3)	3 000

表4 WSRT和FT的结果CEC 2017

Table 4 Result of WSRT and FT on CEC 2017

	+/−/=	Mean	Rank
DirDE	—	1.922	1
SCA	9/0/0	6.389	7
PSO	8/0/1	4.626	5
MFO	9/0/0	4.485	4
BA	7/1/1	4.719	6
ACOR	7/1/1	2.330	2
GWO	7/0/2	3.530	3

表5 P-value的分析结果

Table 5 P-value result of WSRT

	SCA	PSO	MFO
F1	1.734398E-06	5.446250E-02	1.734398E-06
F2	1.734398E-06	1.734398E-06	1.734398E-06
F3	1.734398E-06	9.842142E-03	2.126636E-06
F4	1.734398E-06	1.734398E-06	1.734398E-06
F5	1.734398E-06	1.734398E-06	1.493564E-05
F6	1.734398E-06	1.734398E-06	3.882182E-06
F7	1.734398E-06	1.734398E-06	1.798848E-05
F8	1.238080E-05	8.216736E-03	3.317258E-04
F9	1.734398E-06	1.734398E-06	1.734398E-06
	BA	ACDR	CWO
F1	1.360111E-05	2.584559E-03	1.734398E-06
F2	1.734398E-06	1.382036E-03	5.709650E-02
F3	1.734398E-06	4.276669E-02	4.284300E-01
F4	1.734398E-06	1.024633E-05	3.000989E-02
F5	4.528065E-01	2.126636E-06	5.216493E-06
F6	1.734398E-06	6.156406E-04	1.734398E-06
F7	1.734398E-06	2.957462E-03	1.734398E-06
F8	1.126540E-05	3.161765E-03	4.070231E-02
F9	1.734398E-06	9.753872E-01	1.734398E-06

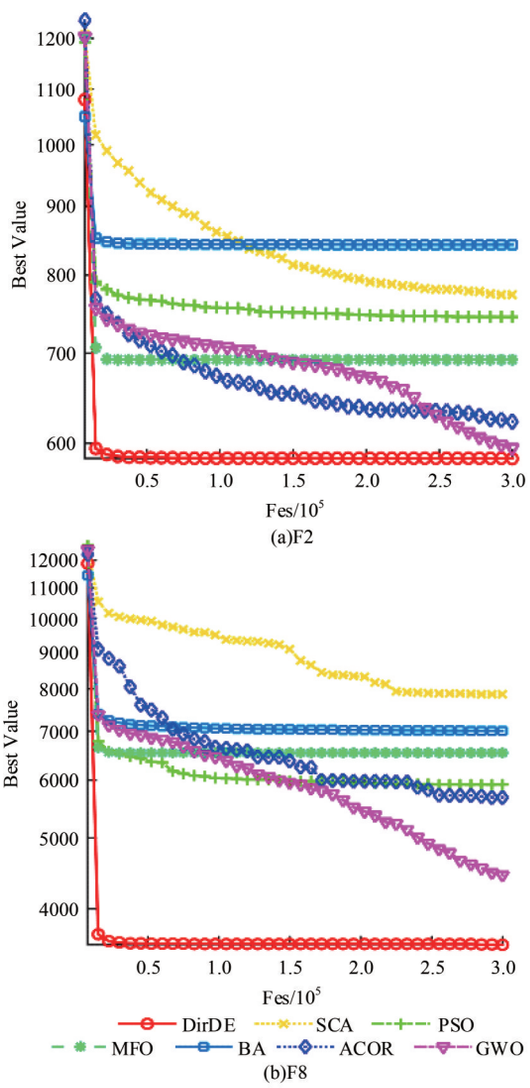

图4 函数收敛曲线

Fig. 4 Convergence curve of this experiment

步,在该线性规划模型中,还根据真实多项目党务活动约束条件分别建立的约束模型。如$g_1(\vec{x})=$

$x_1-\sum_{j=0}^{6}x_j\leqslant 0$ 代表所有活动的总用时要小于党务活动的时间限制。

为了验证DirDE的党务活动调度能力,表6给出了多项目党务活动的相关数据。式(7)~(9)展示了该优化问题的线性关系模型。

$$\vec{x}=[x_1,x_2,x_3,x_4] \tag{7}$$

$$\min f(\vec{x})=ic\times x_1+\sum_{j=0}^{x_2}x_{3,j}\times x_4 \tag{8}$$

$$\text{s.t.}\begin{cases} g_1(\vec{x})=x_1-\sum_{j=0}^{6}x_j\leqslant 0 \\ g_2(\vec{x})=\left[\frac{\max(x_i)}{20}\right]\leqslant 0 \\ g_3(\vec{x})=-\left[30\times t+\left(\sum_{j=0}^{6}x_j-20\times t\right)\right]\leqslant 0 \\ 0\leqslant x_{i,(1\cdots 6)} \end{cases} \tag{9}$$

表7展示了DirDE算法在党务活动调度优化模拟实验的5个优化求解结果。其中,实验设置的算法种群大小为30,算法的优化迭代次数为300,模拟的党务活动项目数为6个。在5种实验结果中,DirDE在方案二和方案五分别获得的最小成本是4 020元,是用了最少的时间完成所有项目并取得最小的成本,党务活动的项目执行顺序是CFADEB。实验结果证明本文提出的方法获得优异的党务活动调度优化结果。

表6　党务活动相关模拟数据

Table 6　Parameter of the problem

项目活动	时长/min	费用/(元·min^{-1})	费用时段/时
A	60	2	11:00−12:00
		10	其他
B	51	18	10:00−13:00
		9	15:00−16:00
C	60	10	9:00−10:00
		30	其他
D	80	15	9:00−13:00
		5	13:00−14:00
E	60	3	14:00−15:00
		6	其他
F	49	20	10:00−11:00
		50	其他

表7　多项目党务活动调度优化问题的算法优化结果

Table 7　Result of this problem

优化结果	项目执行顺序及时间						最优费用(元)
	9:00−10:00	10:00−11:00	11:00−12:00	13:00−14:00	14:00−15:00	15:00−16:00	
方案一	C	F	B	A	E	D	4 560
方案二	C	F	A	D	E	B	4 020
方案三	C	F	B	E	A	F	4 740
方案四	C	B	A	D	E	F	5 280
方案五	C	F	A	D	E	B	4 020

3　结束语

本文研究了多项目党务活动调度问题优化方法,提出一种基于导向交叉机制的差分进化算法,并命名为DirDE。DirDE通过增强种群的多样性和算法的开发及探索的平衡性提升了收敛速度和跳出局部最优的能力,随后,基于IEEE CEC的基准函数集验证了DirDE的性能。同时,DirDE进行了党务活动调度优化模拟实验,并获得优秀的经济和时间计算成本。因此,DirDE可作为基准函数优化和党务活动调度优化的有效求解工具。

4 附 录

表A1 DirDE的伪代码

Table A1 Pseudo-code of DirDE

算法1 DirDE的伪代码
Input MaxFEs, N, D;
Output 最优个体;
初始化算法种群;
计算个体的适应度值和权重;
While FEs<MaxFEs
执行越界调整机制;
计算个体的适应度值;
FEs=FEs+1;
For i=1:N
For j=1:d
基于基因突变更新个体位置;
基于基因交叉更新个体位置;
End For
基于导向交叉机制更新个体位置;
End For
计算适应度值并更新种群;
End While

表A2 DirDE函数实验结果的平均值和方差

Table A2 Analysis result of this experiment

Function	F1		F2	
Item	AVG	STD	AVG	STD
DirDE	0.557 657 598	0.278 898 964	−7 407.076 354	770.578 328 1
DE	0.728 432 366	0.195 771 252	−6 352.436 737	335.908 586 9
SCA	3.463 383 256	2.883 883 188	−3 463.091 399	377.177 083 1
PSO	118.259 697 3	25.314 616 37	−4 252.494 79	529.673 131 6
CS	9.300 416 552	3.962 614 375	−5 096.836 48	277.075 886
MFO	7.125 845 781	2.879 406 697	−7 668.898 381	615.144 348 2
Function	F3		F4	
Item	AVG	STD	AVG	STD
DirDE	54.319 193 1	14.030 728 52	7.999 909 691	2.009 980 328
DE	183.135 391	14.863 933 61	10.483 920 27	0.959 329 059
SCA	141.113 082 3	53.772 903	15.996 416 12	5.074 727 384
PSO	378.796 243 4	23.869 223 94	9.376 909 119	0.431 917 684
CS	274.151 584 8	19.584 280 04	18.299 116 37	0.982 505 883
MFO	231.936 127	30.214 606 56	18.673 543 76	1.611 933 587
Function	F5		F6	
Item	AVG	STD	AVG	STD
DirDE	10.494 579 16	5.563 283 586	−3.862 782 148	2.323 71E−15
DE	15.131 119 95	3.546 735 529	−3.862 782 148	3.449 56E−14
SCA	27.105 499 3	20.321 418 31	−3.843 345 565	0.011 715 935
PSO	11.305 354 2	4.217 776 349	−3.827 018 26	0.040 709 919
CS	139.046 820 8	30.881 430 44	−3.862 007 15	0.000 704 96
MFO	105.320 426 4	43.043 307 31	−3.862 782 148	3.875 8E−13

续表

Function	F7		F8	
Item	AVG	STD	AVG	STD
DirDE	−3.297 780 614	0.049 304 41	−7.991 175 804	3.384 584 52
DE	−3.315 432 869	0.009 676 412	−7.628 655 816	2.280 483 846
SCA	−2.775 652 413	0.357 714 401	−2.093 512 214	1.561 533 941
PSO	−2.699 410 49	0.357 650 702	−2.720 826 086	1.110 185 585
CS	−3.200 962 531	0.042 173 591	−6.581 676 53	1.221 222 586
MFO	−3.219 685 512	0.055 676 612	−5.799 441 3	3.074 278 948
Function	F9		F10	
Item	AVG	STD	AVG	STD
DirDE	−8.983 008 464	2.756 342 06	−8.116 673 19	3.358 198 539
DE	−8.851 974 963	1.639 037 234	−8.914 817 564	1.526 005 655
SCA	−2.440 552 585	1.136 421 079	−2.489 287 59	1.241 180 177
PSO	−2.921 969 928	1.239 688 471	−2.691 428 576	0.924 368 652
CS	−6.807 560 375	1.105 261 999	−6.714 860 461	1.665 301 313
MFO	−7.197 995 68	3.542 479 549	−7.961 080 025	3.487 683 105

表 A3　基于 IEEE CEC 2017 的 DirDE 函数实验结果

Table A3　Analysis result of this experiment

Function	F1		F2		F3	
Item	AVG	STD	AVG	STD	AVG	STD
DirDE	1.092 931E+08	2.212 688E+08	5.846 401E+02	2.561 572E+01	8.926 684E+02	7.053 235E+01
SCA	1.289 989E+10	2.457 938E+09	7.737 518E+02	2.371 079E+01	1.135 049E+03	4.023 382E+01
PSO	1.352 446E+08	1.367 686E+07	7.445 544E+02	3.356 733E+01	9.222 105E+02	1.684 050E+01
MFO	1.217 337E+10	8.461 701E+09	6.922 898E+02	4.647 947E+01	1.205 677E+03	2.140 630E+02
BA	5.619 193E+05	2.695 110E+05	8.427 445E+02	5.995 086E+01	1.640 864E+03	1.887 981E+02
ACOR	1.279 941E+08	4.157 953E+08	6.228 275E+02	5.413 305E+01	9.239 613E+02	4.667 633E+01
GWO	2.059 945E+09	1.476 777E+09	5.952 539E+02	1.603 076E+01	8.757 640E+02	4.969 671E+01
Function	F4		F5		F6	
Item	AVG	STD	AVG	STD	AVG	STD
DirDE	8.764 083E+02	1.842 384E+01	2.643 164E+06	3.032 367E+06	6.000 447E+03	4.180 609E+03
SCA	1.044 130E+03	1.855 794E+01	1.123 463E+09	2.351 720E+08	1.442 318E+07	1.064 585E+07
PSO	9.923 751E+02	2.074 594E+01	2.459 612E+07	1.160 091E+07	4.403 589E+05	2.020 156E+05
MFO	1.011 942E+03	5.117 548E+01	1.688 046E+08	3.085 008E+08	8.719 485E+04	9.509 899E+04
BA	1.054 805E+03	6.167 749E+01	1.899 769E+06	1.474 694E+06	1.206 038E+05	9.515 084E+04
ACOR	9.661 319E+02	6.441 827E+01	1.009 128E+05	1.353 330E+05	1.884 394E+04	1.384 608E+04
GWO	8.869 313E+02	2.092 020E+01	6.627 266E+07	8.700 020E+07	5.720 841E+05	1.253 549E+06
Function	F7		F8		F9	
Item	AVG	STD	AVG	STD	AVG	STD
DirDE	8.458 941E+03	7.325 525E+03	3.572 617E+03	2.551 103E+03	1.465 001E+04	1.618 847E+04
SCA	2.467 860E+07	1.369 098E+07	7.859 300E+03	2.651 573E+03	7.135 765E+07	3.288 423E+07
PSO	1.458 378E+06	6.445 970E+05	5.915 285E+03	2.617 408E+03	3.448 082E+06	1.292 263E+06
MFO	6.537 615E+06	1.967 174E+07	6.541 296E+03	9.862 322E+02	6.724 310E+05	9.925 926E+05
BA	5.745 009E+05	2.344 434E+05	7.008 006E+03	1.133 923E+03	1.198 056E+06	7.352 027E+05
ACOR	2.386 343E+04	2.081 451E+04	5.684 520E+03	1.987 487E+03	1.209 415E+04	4.639 550E+03
GWO	5.152 403E+05	5.453 098E+05	4.449 929E+03	1.477 685E+03	5.379 915E+06	4.706 951E+06

参考文献:

[1] Storn R, Price K. Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization, 1997, 11(4): 341-359.

[2] Zhao D. Ant colony optimization with horizontal and vertical crossover search: fundamental visions for multi-threshold image segmentation[J]. Expert Systems with Applications ,2021, 167: 114122.

[3] 赵鑫, 杨雄飞, 钱育蓉. 改进的蚁群优化算法求解旅行商问题[J]. 计算机工程与设计, 2022, 43(4): 962-968.
Zhao Xin, Yang Xiong-fei, Qian Yu-rong. Improved ant colony optimization algorithm for TSP[J]. Computer Engineering and Design, 2022, 43(4): 962-968.

[4] 肖耀涛. 基于改进蚁群优化算法的云计算资源调度[J]. 微型电脑应用, 2022, 38(2): 160-163.
Xiao Yao-tao. Cloud computing resource scheduling based on improved ant colony optimization algorithm[J]. Microcomputer Applications, 2022, 38(2): 160-163.

[5] 朱显辉, 于越, 师楠, 等. BP神经网络的分层优化研究及其在风电功率预测中的应用[J]. 高压电器, 2022, 58(2): 158-163.
Zhu Xian-hui, Yu Yue, Shi Nan, et al. Research on hierarchical optimization of BP neural network and its application in wind power prediction[J]. High Voltage Apparatus, 2022, 58(2): 158-163.

[6] Cuevas E, Zaldivar D, Pérez C M. A novel multi-threshold segmentation approach based on differential evolution optimization[J]. Expert Systems with Applications, 2010, 37(7): 5265-5271.

[7] Ayala H V H, Santos F M, Mariani V C, et al. Image thresholding segmentation based on a novel beta differential evolution approach[J]. Expert Systems with Applications, 2015, 42(4): 2136-2142.

[8] Liu L. Performance optimization of differential evolution with slime mould algorithm for multilevel breast cancer image segmentation[J]. Computers in Biology and Medicine, 2021, 138: 104910.

[9] Tarkhaneh O, Shen H. An adaptive differential evolution algorithm to optimal multi-level thresholding for MRI brain image segmentation[J]. Expert Systems with Applications, 2019,138: 112820.

[10] Xu L, Jia H, Lang C, et al. A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution[J]. IEEE Access, 2019, 7: 19502-19538.

[11] Chen J. Multi-threshold image segmentation based on an improved differential evolution: case study of thyroid papillary carcinoma[J]. Biomedical Signal Processing and Control, 2023, 85: 104893.

[12] García S, Fernández A, Luengo J, et al. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power[J]. Information Sciences, 2010, 180(10): 2044-2064.

[13] Derrac J, García S, Molina D, et al. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms[J]. Swarm and Evolutionary Computation, 2021, 1(1): 3-18.

[14] Das A K, Pratihar D K. Solving engineering optimization problems using an improved real-coded genetic algorithm (IRGA) with directional mutation and crossover[J]. Soft Computing, 2021, 25(7): 5455-5481.

[15] Kennedy J, Eberhart R. Particle swarm optimization [C]//ICNN'95-international conference on neural networks, Perth, Australia, 1995: 1942-1948.