

一种配电网网络差分禁忌线路规划方法

张贵军 夏华栋 周晓根 张贝金

(浙江工业大学信息与工程学院 杭州 310023)

摘 要 针对配电网规划问题,基于差分进化算法(DE)和禁忌搜索算法(TS)协同优化框架,提出了一种差分禁忌混合算法(DETS)。首先,将配电约束条件划分为硬约束和软约束,硬约束用于保证配电网络拓扑结构的合理性;软约束用于提高种群多样性。然后,设计DE及TS两层优化结构,外层利用DE快速收敛特性为内层提供较好的初始个体;内层利用TS贡献全局搜索能力,避免陷入局部最优。其次,设计修复算子来避免DE算法易产生不可行解的问题。最后,采用10个标准测试函数验证了DETS算法的性能,同时利用DETS算法实现了某市的中低压配电网络线路规划。

关键词 配电网络,线路规划,差分进化,禁忌搜索,GIS

中图分类号 TP391.9 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.10.047

Hybrid Differential Evolution Based on Tabu Search Algorithm for Distribution Network Line Planning

ZHANG Gui-jun XIA Hua-dong ZHOU Xiao-gen ZHANG Bei-jin

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract A hybrid differential evolution algorithm (DE) based on tabu search algorithm (TS) was proposed in this paper for the planning of power distribution. Firstly, the distribution constraints are divided into hard constraints and soft constraints. Hard constraints ensure the reasonability of the topological structure of the distribution network, and soft constraints improve the diversity of the population. Secondly, a hierarchical structure is adopted. The outer layer provides excellent initial individual for the inner layer by rapidly convergent DE algorithm, while the inner layer provides a global searching process and avoids being trapped in local optimum by TS algorithm. A repairing operator is also designed to solve the non-feasible solution generated in DE. Finally, the performance of DETS is verified by 10 benchmark functions. In addition, the line planning of medium-low distribution network of a certain city is achieved by the proposed DETS.

Keywords Distribution network, Line planning, Differential evolution, Tabu search, GIS

1 引言

配电网线路规划本质上就是对变电站的位置和容量、馈线的路径和数目、分段开关的位置和数目、网络接线方式的一个优化过程^[1,2]。规划在变电站容量、馈线容量、电压降落、网络结构及可靠性等约束条件下,通过减少新设备、新线路的安装费用来减少已有设备、线路的扩容费用,通过减少运行维护费用和提高系统可靠性来达到优化的目的。

配电网线路规划是一个复杂的大规模组合优化规划问题^[3,4]。在求解策略的选择上,传统的确定性算法如线性/非线性混合整数规划比启发式算法具有更好的收敛性^[5],但是其难以处理规划中的非线性和非凸性特征,而且不可避免地陷入维数灾^[6],因此其只适用于小规模配电网线路规划。启发式算法由于其优秀的多点搜寻能力,能在合理的时间内

完成对大规模、多目标的规划问题求解^[7],因此配电网线路规划的研究多集中在现代启发算法,尤其是在进化算法上^[8]。但是启发式算法在处理组合优化问题时对约束条件难以有效处理且容易出现早熟收敛现象,如何针对上述问题对算法进行改进是研究的热点。

差分进化算法(Differential Evolution, DE)^[9]是基于群体智能理论的优化算法,通过群体内个体间的合作与竞争产生的群体智能指导优化搜索。DE算法原理简单、通用性强、易于实现而且具有快速收敛性和较强的稳健性,已被证明是目前最高效的随机优化算法之一^[10]。DE算法因以上优点在通信、电力、化工及生物信息学等领域得到了广泛应用,但是由于其激烈的个体竞争,导致算法极易趋于早熟收敛而陷入局部最优^[11,12]。

针对以上问题,国内外学者相继提出了一些改进算法。

到稿日期:2015-09-11 返修日期:2016-01-24 本文受国家自然科学基金(61075062, 61573317),浙江省自然科学基金(LY13F030008),浙江省重中之重学科开放基金(20120811),浙江省科技厅公益项目(2014C33088),浙江省大学生“新苗计划”(2015R403077)资助。

张贵军(1974—),男,博士,CCF会员,主要研究方向为智能信息处理、智能交通系统、优化理论及算法设计和生物信息学, E-mail: zgj@zjut.edu.cn; 夏华栋(1989—),男,硕士生,主要研究方向为智能信息处理;周晓根(1987—),男,博士生,CCF学生会员,主要研究方向为智能信息处理和优化理论及算法设计;张贝金(1990—),男,硕士,主要研究方向为信息处理。

Mendes 等^[13]提出一种多种群差分进化算法(DynDE),该算法设置两个种群,当一个种群的最优个体与另一个种群中的最优个体很接近时,则对其中一个种群重新初始化,即保持一个种群的最优个体而对另一个重新初始化,以保持种群的多样性,防止算法早熟收敛。Das 等^[14]将差分进化算法与模拟退火算法混合(AnDE),通过一种退火选择机制(即以一定的概率接受较差个体)和一种中心聚集变异策略(即使得新个体随机向当前种群中个体的平均值靠拢)来避免算法陷入局部最优。Bhattacharya 等^[15]将差分进化算法与生物地理学算法混合(DE/BBO),利用 BBO 算法的迁移和变异操作来提高 DE 算法的全局搜索能力,同时从一定程度上防止算法早熟收敛。Song 等^[16]将差分进化算法与分布式估计算法混合(DE-EDA),使得算法能够快速定位到最优解所在的区域,同时利用一种改进变异策略来提高种群的多样性,防止算法早熟收敛。Qin 等^[17]提出一种自适应差分进化算法(SaDE),通过对进化前期的学习过程来自适应调整变异策略及控制参数,以提高算法性能。Wang 等^[18]提出一种具有复合新个体策略和控制参数的差分进化算法(CoDE),通过将不同的变异策略与不同的控制参数进行组合来竞争产生个体;从而降低算法的计算代价。以上改进算法均取得了一定的效果。

禁忌搜索算法(Tabu Search, TS)^[19,20]是一种全局逐步寻优算法,是对局部搜索算法的一种扩展。TS 算法通过引入一个灵活的存储结构和相应的禁忌准则来避免迂回搜索,并通过藐视准则来赦免一些被禁忌的优良状态,从而保证搜索的多样化以防止陷入早熟,实现全局优化^[21]。本文在 DE 算法中引入 TS 搜索策略,针对配电网线路规划提出一种差分禁忌算法(DETS)。首先,外层由 DE 快速收敛,为内层提供优良初始个体;内层由 TS 贡献全局搜索能力,避免陷入局部最优。其次,针对现有算法在处理配电网线路规划时存在的未能对约束条件有效处理的问题,采取区分软硬约束的方式,同时设计修复算子。最后,针对某市电力管线管理 GIS 系统现有配电网的线路规划需求,开发配电网线路规划模块,达到了求解大规模配电网线路规划的要求。

2 配电网线路规划优化模型

2.1 目标函数

在文献[4]模型的基础上,综合考虑资金的时间价值,按等额分付资本回收计算,以年投资、运行维护费用及缺电损失费用最小为目标采用如下配电网线路规划优化模型:

$$\min F = \sum_{i \in S_1} (\omega + H) l_i f(D_i) X_i + \sum_{i \in S_2} H l_i f(D_i) + \sum_{i \in S} \tau_{\max} d l_i^2 R_i + \sum_{i \in S} \lambda_i l_i T I_{IEAR_i} \times E_{EENS_{i,r}} \quad (1)$$

其中, S_1 、 S_2 和 S 分别为新建线路集、已建线路集和总线路集; ω 是年等值回收系数; H 为维修和折旧占固定投资费用的百分比; l_i 是线路 i 的长度(km)(每条线路长度不等); $f(D_i)$ 为线径为 D_i 的线路单位长度的造价(元/km); X_i 为第 i 条线路的决策变量(选择该线路作为馈线支路,则 X_i 取1,反之取0); R_i 为线路 i 的电阻(Ω); I_i 为线路 i 的通过电流(A); $\tau_{\max}$ 为年最大损耗时间(h); d 为单位电价(元/kWh); λ_i 为线路 i 的故障率(次/km); T 为故障的平均修复时间(h); I_{IEAR_i} 为线路 i 的缺电损失评价率(元/kWh); $E_{EENS_{i,r}}$ 为线路 i 的电量不足期望值(kw/期间)。其中线路 i 由决策变量 X_i 决定,同时

又决定电阻 R_i 的取值,因此模型(1)在数学上是一个复杂的非线性规划模型。

2.2 约束条件

1)潮流约束

$$A_t P_t = D_t \quad (2)$$

其中, A_t 是 t 时间段的结点有功注入和线路通过有功功率的关联矩阵, P_t 是 t 时间段的线路通过有功功率, D_t 是 t 时间段的结点注入有功功率($t=1,2,\dots,T$), T 是规划的时间范围。

2)容量约束

$$P_i \leq P_{i \max} \quad (3)$$

其中, P_i 是 t 规划时段 i 线路的容量, $P_{i \max}$ 是 i 线路的最大容量($i \in S$), S 为可选的线路集合(包含已存在的线路)。

3)电压降落约束

$$V_{i \min} \leq V_i \leq V_{i \max} \quad (4)$$

其中, $V_{i \min}$ 是结点最低电压, $V_{i \max}$ 是结点最高电压, V_i 是 t 规划时间段 i 结点的电压($i=1,2,\dots,N$), N 为结点总数。

4)辐射性约束

$$M = N - 1 \quad (5)$$

其中, M 是线路规划方案中选择的支路数目, N 是结点总数,规划线路要求不成环状分布。

2.3 约束设计

本文综合考虑上述约束在算法中编码的难易程度及约束性质,将其分为两类:硬约束和软约束。

由于配电网的正常运行必须要满足辐射性结构,因此将辐射性约束4)作为硬约束。算法中产生的不满足辐射结构的解必须经过结构修复才能进入下一代。相对于辐射性约束,潮流约束1)、容量约束2)、电压降落约束3)等电气约束更易于编码,且允许一定程度上的偏差,因此可作为软约束加入到目标函数中。硬约束的限制可以保证配电网的正常运行;软约束的存在可以保持种群多样性,避免算法陷入局部最优。

3 差分禁忌算法的配电网线路规划

DE算法的贪婪选择策略能够加快收敛速度,同时又容易导致早熟收敛而陷入局部最优,本文在DE算法中引入全局搜索能力较强的TS算法。算法分为内外两层,外层执行DE算法为内层提供优良的初始个体,即当算法收敛于局部最优时进入内层,将DE算法所求得的最优解作为TS算法的初始个体,且置禁忌表为空,并将初始解的目标函数值作为“best so far”状态,转入TS算法进行搜索^[19]。若TS算法产生的最优解与DE算法所求得的最优解相等,则表明此解为全局最优解;若TS算法产生的最优解优于外层DE算法所求得的最优解,则表明通过TS算法跳出了局部最优。此外,由于DE算法经过交叉变异操作后,解原本的辐射结构很容易被破坏而成为病态解,如果直接让这些病态解进入下一代,不仅会极大地干扰算法对优秀解个体的搜索,而且会延长搜索时间,本文通过设计结构修复算子来进行修复,并对原始地理信息数据进行处理,使其相互联系,同时在算法中使用前推回代

3.1 结构修复算子设计

在进行选择操作前,需要对病态解的网络结构进行修复,本文通过解闭环连孤边对病态解进行修复,使其恢复辐射结构。修复流程如图1所示。

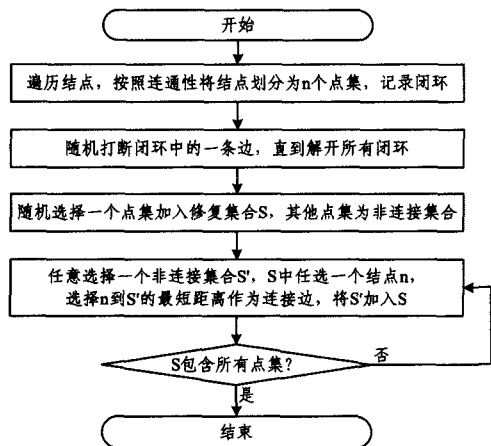


图1 修复流程

修复流程的具体步骤如下。

1) 首先遍历病态解中的结点,按照连通性将结点划分为 n 个点集,每个集合中的结点都是一棵连通树(即集合中的任意一个结点,经过若干条边后,都能到达集合中的任意其他结点)或者孤点、孤边,并记录下连通树中的闭环;

2) 随机打断闭环中的一条边,直到解开所有闭环;选择一个点集加入修复集合 S ,其他点集均为未连接点集;

3) 任意选择一个非连接集合 S' ,在修复集合 S 中任意选择一个结点,在结点与 S' 中结点相连的所有边中选择一条路径最短的边作为 S 和 S' 的连接边,然后将 S' 加入修复集合 S ;

4) 重复步骤3),直到所有点集加入修复集合。

3.2 数据处理和潮流计算

3.2.1 数据处理

原始空间信息数据通常是一系列离散互不相关的地理信息元素,相互之间缺乏关联,而配电网规划关心的是各地理信息元素之间的相互关系,因此需要对原始数据进行处理。

1) 首先由原始数据生成并简化街道段,构造基本地理拓扑;

2) 生成街区;

3) 判别负荷街区归属并计算可行接线位置,生成扩展拓扑。

3.2.2 潮流计算

由于配电网具有闭环结构、开环运行的特性,稳态运行时网络结构多为辐射性;又由于配电网的 R/X 比值较高、网络PQ结点为主、PV结点较少等特点,因此输电网络中常用的回路阻抗法、改进牛顿法、PQ分解法等潮流计算方法并不适用,本文采用树状网络中常用的前推回代法进行潮流的计算^[22]。

3.3 算法设计

算法选择待选支路的状态作为决策变量,因此采用二进制编码,“1”表示支路被选中,“0”表示支路未被选中,将所有支路的二进制状态顺序组成的二进制字符串视为一个解。算法具体描述如下。

步骤1(初始化) 参考Prim算法设计构造配电网的一般生成树:定义两个集合 S 和 S' , S 为已连接结点,初始为空; S' 为未连接结点。将变电站结点放入集合 S ,随机选择一条与该结点相连且边上另一结点位于集合 S' 的边,将边上的另一结点加入集合 S ,重复该操作,逐条将边加在生成树上,直至 S 中包含配电网中的所有结点为止。按照上述方法生成 NP (种群规模)个初始解 $X_i^0 (i=1,2,\dots,NP)$ 并计算它们的评价函数。

步骤2(变异操作) 第 G 代,在 X_i^G 的基础上生成新的变异个体 V_i^{G+1} 。

$$V_i^{G+1} = X_i^G + F \cdot (X_{r1}^G - X_{r2}^G) \quad (6)$$

其中, F 为缩放因子; X_{r1}^G 和 X_{r2}^G 为随机选择的两个个体; $i \neq r1, r2; r1, r2 \in \{1, 2, \dots, NP\}$ 。由于算法采用二进制编码,解 X 均为二进制字符串,因此式(6)中的运算符重载为逻辑运算符:“+”表示逻辑与运算;“-”表示逻辑异或运算;“-”表示逻辑或运算。相应的控制参数 F 也改为随机生成的二进制字符串。

步骤3(交叉操作) 在第 G 代,由变异生成的个体 V_i^{G+1} 和第 G 代个体 X_i^G 根据式(7)产生测试个体。

$$U_{ji}^{G+1} = \begin{cases} V_{ji}^{G+1}, & \text{if } (\text{randb}(j) \leq CR) \text{ or } j = \text{rnbr}(i) \\ X_{ji}^G, & \text{if } (\text{randb}(j) > CR) \text{ and } j \neq \text{rnbr}(i) \end{cases} \quad (7)$$

其中, $i=1,2,\dots,NP, j=1,2,\dots,D; \text{randb}(j)$ 为 $[0,1]$ 之间的均匀分布随机数; CR 为 $[0,1]$ 之间的交叉概率; $\text{rnr}(i)$ 为 $\{1, 2, \dots, D\}$ 之间的随机量,确保新个体中至少有一个扰动量。

步骤4(结构修复算子设计) 根据2.1节通过解闭环连孤边对病态解进行修复,使其恢复辐射结构。

步骤5(选择操作) 根据式(8)对测试个体 U_i^{G+1} 和第 G 代个体 X_i^G 的目标函数值进行比较,选择目标函数值较小的个体作为新种群的个体 X_i^{G+1} 进入下一代。

$$X_i^{G+1} = \begin{cases} U_i^{G+1}, & \text{if } f(U_i^{G+1}) < f(X_i^G) \\ X_i^G, & \text{otherwise} \end{cases} \quad (8)$$

重复步骤2—步骤5,当算法收敛于局部最优解时进入步骤6。

步骤6(执行TS算法) 置禁忌表为空,将当前解作为初始个体,并将其目标函数值作为“best so far”状态,执行TS算法进行搜索^[19]。

步骤7(邻域解生成) 对当前解进行置换操作产生邻域解,即随机交换初始解字符串中的任意两个位置上的二进制编码。按此方法产生 $N_{neighbor}$ 个邻域解作为候选解。

步骤8(禁忌判断与演化) 对候选解判断藐视准则是否满足。若满足,则用满足藐视准则的最佳状态 x 替换当前解,用 x 对应的禁忌对象替换最早进入禁忌表的禁忌对象,同时用 x 的目标函数替换“best so far”状态,转入步骤7;若不满足藐视准则,则在候选解中选出不在禁忌表中的最优候选解 y ,用 y 替换当前解,用 y 对应的禁忌对象替换最早进入禁忌表的禁忌对象。重复步骤7、步骤8,当算法迭代次数超过迭代阈值,并且“best so far”状态连续 $N_{unchanged}$ 代不发生变化时进入步骤9。

步骤9(返回DE算法) 比较Tabu算法产生的“best so

far”状态与进入 Tabu 算法前的局部最优解的目标函数值,若两者相等,则表明该局部最优解即是全局最优解;若“best so far”状态小于局部最优解的目标函数值,则表明跳出了局部最优,并根据式(9)将“best so far”状态的精英个体与上一代种群的个体交叉,从而将优秀基因传入下一代。

$$V_i^{G+1}=X_{best}^G+F\cdot(X_{r1}^G-X_{r2}^G)$$

(9)

重复步骤 2—步骤 9,直至找到全局最优解。

4 数值仿真

为了验证所提 DETS 算法的性能,选取 SaDE^[17] 和 CoDE^[18] 算法进行比较分析。实验中,采用 10 个典型的标准测试函数进行测试,表 1 列出了各测试函数的名称、数学表达式、搜索范围及全局最优解。其中,函数 f_1-f_4 为单模函数, f_5-f_{10} 为多模函数。

表 1 10 个标准测试函数

函数名	数学表达式	搜索范围	全局最优值
Sphere	$f_1(x)=\sum_{i=1}^N x_i^2$	$[-100,100]$	0
Schwefel 2.22	$f_2(x)=\sum_{i=1}^N x_i +\prod_{i=1}^N x_i $	$[-10,10]$	0
Zakharov	$f_3(x)=\sum_{i=1}^N x_i^2+(\sum_{i=1}^N 0.5ix_i)^2+(\sum_{i=1}^N 0.5ix_i)^4$	$[-5,10]$	0
Rosenbrock	$f_4(x)=\sum_{i=1}^{N-1} (100(x_{i+1}-x_i^2)^2+(x_i-1)^2)$	$[-2,2]$	0
Griewank	$f_5(x)=1+\frac{1}{4000}\sum_{i=1}^N x_i^2-\prod_{i=1}^N \cos(\frac{x_i}{\sqrt{i}})$	$[-600,600]$	0
Schaffer 2	$f_6(x)=\sum_{i=1}^{N-1} (x_i^2+x_{i+1}^2)^{0.25}(\sin^2(50(x_i^2+x_{i+1}^2)^{0.1})+1)$	$[-100,100]$	0
Schwefel 2.26	$f_7(x)=-\sum_{i=1}^N (x_i \sin(\sqrt{ x_i }))$	$[-500,500]$	-418.983N
Himmelblau	$f_8(x)=\frac{1}{N}\sum_{i=1}^N (x_i^4-16x_i^2+5x_i)$	$[-100,100]$	-78.3323
Ackley	$f_9(x)=-20\exp(-0.2\sqrt{\frac{1}{N}\sum_{i=1}^N x_i^2})-\exp(\frac{1}{N}\sum_{i=1}^N \cos(2\pi x_i))+20+e$	$[-30,30]$	0
Rastrigin	$f_{10}(x)=10N+\sum_{i=1}^N (x_i^2-10\cos(2\pi x_i))$	$[-5,5]$	0

DETS 算法设置:种群规模 $NP=50$,交叉概率 CR 和缩放因子 F 均设置为 0.5,禁忌表长度为 $2N$,其中 N 为问题维数。SaDE 算法和 CoDE 算法的参数设置均采用文献[16, 17]中的设置。为了公平比较,所有算法均设置相同的运行终止条件,且每种算法对于每个测试函数均独立运行 30 次。实验环境为: Intel(R) Core i5-2410M CPU@2.30 GHz with 8GB RAM, Windows 7, 算法实现代码采用 Visual Studio 2012 C++。

为了验证 DETS 算法的收敛速度和可靠性,选取函数评价次数(FES)和成功率(SR)两个指标进行衡量。FES 为算法在设定的目标函数评价次数 300000 次内,函数值误差 $(f(x)-f(x^*))$ 达到给定的精度 $\delta=0.00001$ 时所需的目标函数评价次数的平均值,其中, $f(x)$ 为算法所求的最优值, $f(x^*)$ 为函数的全局最优值(见表 1)。SR 为算法的成功求解次数和总运行次数之比,其中,规定算法在 300000 次函数评价次数内能达到给定的精度 $\delta=0.00001$ 时为成功。

表 2 给出了各算法的函数评价次数 FES 和成功率 SR,其中“NA”表示算法在设定的目标函数评价次数 300000 次内所求得的最优函数值误差无法达到给定的精度 δ ,最优结果通过加粗标出。从表中可以看出,对于大部分函数,DETS 算法的收敛速度优于 SaDE 算法和 CoDE 算法。具体来说,在 FES 方面,SaDE 算法仅在函数 f_2 和 f_3 上优于 DETS 算法,CoDE 没有在任何函数上优于 DETS 算法;在 SR 方面,SaDE

算法可以对 6 个函数 100%求解成功,CoDE 算法可以对 9 个函数 100%求解成功,DETS 算法可以对所有的函数 100%求解成功。其次,从表中最后一行的平均值可以看出,DETS 算法所需的目标函数评价次数最少,为 45451, SaDE 算法和 CoDE 算法分别为 77512 和 91095,即 DETS 算法相对于 SaDE 和 CoDE 分别节省了 41.4%和 51.0%的目标函数评价次数,而且由于 DETS 算法利用 TS 算法对局部最优禁忌使算法跳出局部最优,因此其可靠性最高,平均成功率为 1.000, SaDE 算法和 CoDE 算法分别为 0.859 和 0.997。

表 2 函数评价次数和成功率

函数	N	SaDE		CoDE		DETS	
		FES	SR	FES	SR	FES	SR
f_1	30	20297	1.00	40155	1.00	18212	1.00
f_2	30	24368	1.00	56421	1.00	26908	1.00
f_3	30	137487	1.00	77979	1.00	59791	1.00
f_4	30	NA	0.00	236957	1.00	118425	1.00
f_5	30	21740	0.73	43878	1.00	17681	1.00
f_6	30	71833	1.00	172776	1.00	79036	1.00
f_7	30	36050	1.00	59982	1.00	33478	1.00
f_8	30	22873	1.00	35565	1.00	19752	1.00
f_9	30	30453	0.93	56826	1.00	23242	1.00
f_{10}	30	62508	0.93	130409	0.97	57987	1.00
平均值		77512	0.859	91095	0.997	45451	1.000

选用平均函数误差值(Mean error)和标准偏差值(Std dev)两个指标来衡量各算法对 10 个测试函数所求得的结果的性能。另外,为了验证所提算法相对于其他比较算法是否具有显著性优势,选用 Wilcoxon Signed Rank Test^[23] 对各算

法 30 次优化得到的结果进行非参数假设检验,显著性水平为 5%,用“+”表示所提算法显著优于所比算法,用“-”表示所提算法显著差于所比算法,用“≈”表示两种算法之间没有显著性差异。

表 3 各函数的 30 维优化结果性能表

函数	N	SaDE	CoDE	DETS
		Mean error(Std dev)	Mean error(Std dev)	Mean error(Std dev)
f_1	30	1.29E-23(3.07E-23) ⁺	2.16E-10(1.73E-10) ⁺	3.56E-30(8.32E-30)
f_2	30	8.70E-16(5.17E-16) ⁻	3.86E-06(1.32E-06) ⁺	1.87E-14(4.57E-14)
f_3	30	9.27E-02(1.30E-01) ⁺	5.69E-04(7.53E-04) ⁺	3.21E-05(2.17E-05)
f_4	30	5.07E+01(3.36E+01) ⁺	1.97E+01(5.80E-01) ⁺	8.93E+00(2.84E+00)
f_5	30	2.22E-03(4.73E-03) ⁺	2.47E-07(7.34E-07) ⁺	0.00E+00(0.00E+00)
f_6	30	7.51E-04(7.06E-04) ⁻	1.97E+00(4.24E-01) ⁺	1.24E-03(1.65E-03)
f_7	30	0.00E+00(0.00E+00) [≈]	1.66E-02(3.13E-02) ⁺	0.00E+00(0.00E+00)
f_8	30	0.00E+00(0.00E+00) [≈]	0.00E+00(0.00E+00) [≈]	0.00E+00(0.00E+00)
f_9	30	2.40E-01(4.45E-01) ⁺	3.75E-06(1.88E-06) ⁺	6.04E-15(1.66E-15)
f_{10}	30	4.11E-02(1.82E-01) ⁺	3.30E+01(5.81E+00) ⁺	1.55E-13(6.14E-02)
+/≈/-		6/2/2	9/1/0	

表 3 列出了各算法对各测试函数 30 维求解所得结果的性能数据,表中最后一行为 Wilcoxon Signed Rank Test 非参数检验的统计结果,最优结果通过加粗标出。从表中数据可以看出,DETS 算法对于大多数函数求解所得结果的性能优于 SaDE 算法和 CoDE 算法。具体来说,DETS 算法在 6 个函数上显著优于 SaDE 算法,其中包括 3 个单模函数(f_1 , f_3 和 f_4)和 3 个多模函数(f_5 , f_9 和 f_{10}),而 SaDE 算法仅在 2 个函数(f_2 和 f_6)上显著优于 DETS 算法,对于其余函数,由于 DETS 算法和 SaDE 算法都找到了全局最优解,从而获得了相同的结果,因此两者之间没有显著性差异。CoDE 算法没有显著优于 DETS 算法任何函数,仅在 1 个函数(f_8)上与 DETS 算法获得了相同的结果,而 DETS 算法在 9 个函数上

显著优于 CoDE 算法。

表 4 列出了各算法对各测试函数 50 维求解所得结果的性能数据。可以明显看出,DETS 算法对于大多数函数的性能优于 SaDE 算法和 CoDE 算法。具体来说,DETS 算法在 7 个函数上显著优于 SaDE 算法,其中包括 3 个单模函数(f_1 , f_3 和 f_4)和 4 个多模函数(f_5 , f_7 , f_9 和 f_{10}),而 SaDE 算法仅在 1 个函数(f_2)上显著优于 DETS 算法;对于函数 f_6 ,虽然 DETS 算法的结果优于 SaDE 算法,但是两者之间没有显著性差异;对于函数 f_8 ,DETS 算法和 SaDE 算法都找到了全局最优解,从而获得了相同的结果,因此两者之间没有显著性差异。CoDE 算法仅在函数 f_7 上显著优于 DETS 算法,而 DETS 算法在其余所有函数上均显著优于 CoDE 算法。

表 4 各函数的 50 维优化结果性能表

函数	N	SaDE	CoDE	DETS
		Mean error(Std dev)	Mean error(Std dev)	Mean error(Std dev)
f_1	50	9.21E-26(9.49E-26) ⁺	1.57E-12(1.40E-12) ⁺	1.32E-33(1.56E-33)
f_2	50	1.64E-16(8.95E-17) ⁻	1.55E-07(7.15E-08) ⁺	2.86E-15(7.84E-16)
f_3	50	9.15E-02(5.88E-02) ⁺	4.32E-02(4.36E-02) ⁺	8.93E-03(2.11E-03)
f_4	50	3.81E+01(2.16E+00) ⁺	3.86E+01(4.03E-01) ⁺	3.01E+01(2.06E+00)
f_5	50	7.61E-03(1.69E-02) ⁺	9.35E-13(4.92E-13) ⁺	0.00E+00(0.00E+00)
f_6	50	6.79E-01(1.73E+00) [≈]	9.79E-01(2.02E-01) ⁺	4.82E-01(3.76E-01)
f_7	50	2.37E+01(4.99E+01) ⁺	1.89E+01(2.53E+01) ⁻	2.17E+02(6.13E+01)
f_8	50	0.00E+00(0.00E+00) [≈]	3.14E-05(4.62E-12) ⁺	0.00E+00(0.00E+00)
f_9	50	8.40E-01(6.12E-01) ⁺	1.74E-07(9.59E-08) ⁺	1.89E-14(5.15E-15)
f_{10}	50	4.99E+00(4.75E+00) ⁺	8.86E+01(1.10E+01) ⁺	2.57E+00(1.67E+00)
+/≈/-		7/2/1	9/0/1	

表 5 结点数据

结点编号	有功负荷	无功负荷	结点编号	有功负荷	无功负荷
1	0.1	0.048	16	0.15	0.073
2	0.25	0.121	17	0.08	0.039
3	0.16	0.077	18	0.04	0.019
4	0.1	0.048	19	0.1	0.048
5	0.1	0.048	20	0	0
6	0.05	0.024	21	0.06	0.029
7	0.1	0.048	22	0.04	0.019
8	0.1	0.048	23	0.08	0.039
9	0.25	0.121	24	0.1	0.048
10	0.16	0.077	25	0.03	0.015
11	0.1	0.048	26	0.1	0.048
12	0.16	0.077	27	0.16	0.077
13	0.1	0.048	28	0.08	0.039
14	0.1	0.048	29	0.1	0.048

从上述结果和分析可以看出,本文所提 DETS 算法对大部分测试函数求解的性能优于所比算法,不仅收敛速度快,可靠性高,而且解的质量也较高。

5 案例分析

基于 OpenGIS 框架,采用 J2EE 规范的 SSH 架构,以城市基础地形数据为系统背景图层、城市电力管线数据为专题图层、动态采集的设备数据为服务图层,开发某市电力管线管理 GIS 系统。

本文基于该平台开发了配电网线路规划模块,并利用该模块对某市一个 10kV 的中低压配电网进行了规划,网络包含 1 个变电站、28 个负荷和 43 条可选支路。结点数据如表 5 所列,其中容量 0 的结点为变电站;支路数据如表 6 所列。

表6 馈线数据

支路号	起点	终点	电阻(Ω)	电抗(Ω)	造价(元)	容量(A)	支路号	起点	终点	电阻(Ω)	电抗(Ω)	造价(元)	容量(A)
1	27	28	2.1	0.4	15000	125	23	19	20	4.41	0.84	31500	90
2	28	29	2.1	0.4	15000	125	24	20	21	4.62	0.88	33000	90
3	29	23	2.1	0.4	15000	125	25	21	22	2.1	0.4	15000	125
4	23	25	2.205	0.42	15750	125	26	27	1	1.68	0.32	12000	125
5	23	24	2.205	0.42	15750	125	27	28	2	1.68	0.32	12000	125
6	24	26	8.925	1.7	63750	90	28	29	3	1.68	0.32	12000	125
7	1	2	2.1	0.4	15000	125	29	23	4	1.68	0.32	12000	125
8	2	3	2.1	0.4	15000	125	30	24	5	1.68	0.32	12000	125
9	3	4	2.1	0.4	15000	125	31	26	8	1.68	0.32	12000	125
10	4	5	2.1	0.4	15000	125	32	1	16	1.68	0.32	12000	125
11	5	6	2.1	0.4	15000	125	33	3	15	1.68	0.32	12000	125
12	6	7	2.1	0.4	15000	125	34	4	14	1.68	0.32	12000	125
13	7	8	2.205	0.42	15750	125	35	5	13	1.68	0.32	12000	125
14	16	15	4.62	0.88	33000	90	36	6	12	1.68	0.32	12000	125
15	15	14	2.205	0.42	15750	125	37	7	11	1.68	0.32	12000	125
16	14	13	2.625	0.5	18750	125	38	8	10	1.89	0.36	13500	125
17	13	12	2.205	0.42	15750	125	39	16	17	1.68	0.32	12000	125
18	12	11	2.1	0.4	15000	125	40	15	19	1.68	0.32	12000	125
19	11	10	2.1	0.4	15000	125	41	13	20	1.68	0.32	12000	125
20	10	9	2.1	0.4	15000	125	42	11	21	1.68	0.32	12000	125
21	17	18	2.1	0.4	15000	125	43	10	22	1.68	0.32	12000	125
22	18	19	2.1	0.4	15000	125							

图2为配电网络初始结构,设置参数如图3所示:资本回收系数为0.1,维护及折旧占固定投资百分比的5%,单位电量费用系数为0.5元/kWh,年最大负荷利用小时为5000h,缺电损失评价率为5元/kWh,缺电修复时间为3h,线路故障率为0.2次/km,最大允许电压降落为10%。如2.3节所述,将辐射结构设置为硬约束,将潮流约束、容量约束和电压降落约束等电气约束设置为软约束,运行基于差分禁忌算法的配电网络线路规划模块后,为了便于人工选择,程序提供了多种不同规划方案。图4为程序所得最优规划方案,其对应的费用(即目标函数值)为1181304.810元,图5和图6为次优规划方案,费用分别为1182252.318元和1186473.100元。

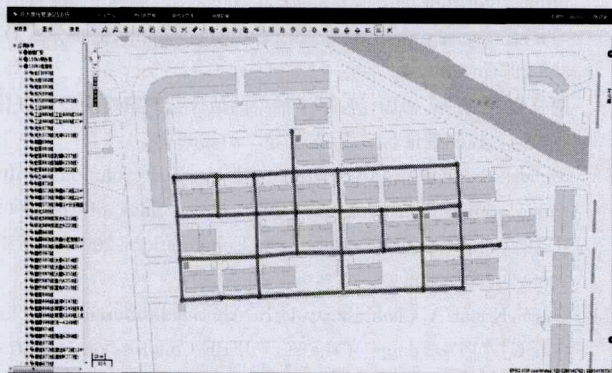


图2 初始配电网络

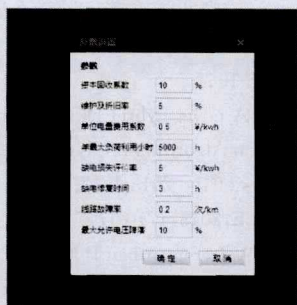


图3 参数设置

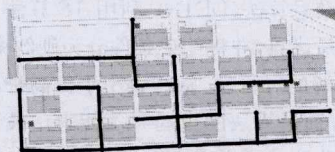


图4 最优规划结果

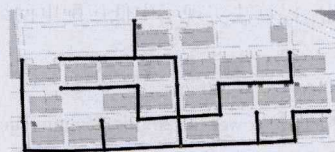


图5 次优规划结果一

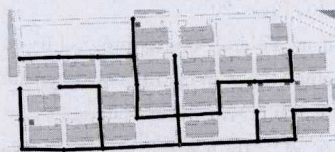


图6 次优规划结果二

为验证所提 DETS 算法的优势,分别选用基本 DE 算法及其改进算法 CoDE 算法和 SaDE 算法对上述配电网络优化后进行比较分析。以迭代次数 100 代为终止条件,每种算法各独立运行 30 次。首先,图7给出了4种算法的平均收敛曲线图,可以看出,DETS 算法收敛速度最快,可很快得到问题的全局最优解,CoDE 算法仅次于 DETS 算法,DE 算法和 SaDE 算法收敛速度较慢,迭代到 100 代时仍未找到全局最优解。

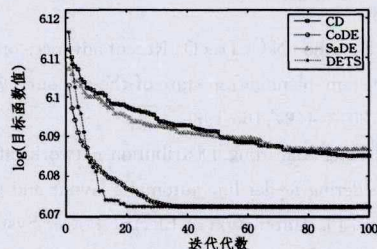


图7 4种算法平均收敛曲线图

从表 7 中优化结果的各项性能数据也可以看出,DE 算法和 SaDE 算法在给定代数内只能得到局部最优解,CoDE 算法和 DETS 算法均能以 100% 的成功率在给定代数内求得全局最优解,但 DETS 算法的平均收敛代数仅为 20.1 代,而 CoDE 算法为 30 代,DE 算法和 SaDE 算法在给定的迭代代数内没有收敛,而且 DETS 算法的平均收敛时间为 4713.2ms,CoDE 为 24388.8ms,即 DETS 算法相对于 CoDE 算法节约了 80.7% 的求解时间。由此说明,DETS 算法具有比 CoDE 算法更快的收敛速度;另外,DETS 算法在给定代数内优化得到的费用的最小值、最大值以及平均值均优于 DE 算法和 SaDE

表 7 算法比较结果

算法	平均收敛代数	平均收敛时间(ms)	最小费用(元)	最大费用(元)	平均费用(元)	标准偏差	成功率	t-test
DETS	20.1	4713.2	1181304.810	1181304.810	1181304.810	0	100%	NA
DE	>100	—	1187472.727	1256335.063	1216758.677	14072.033	0	+
CoDE	30.0	24388.8	1181304.810	1181304.810	1181304.810	0	100%	≈
SaDE	>100	—	1181304.810	1248816.943	1221215.580	15630.861	3.33%	+

结束语 DE 算法具有较强的稳健性和快速收敛性,在配电网络线路规划上具有很大的优势,然而 DE 算法极易出现早熟收敛现象。针对此问题,在 DE 算法中引入 TS 算法,提出一种差分禁忌算法(DETS),利用 TS 算法对局部最优禁忌来跳出局部最优点,并将约束条件分为硬约束和软约束,硬约束用于保障解的可行性,软约束用于提高种群多样性以避免过早收敛,提高了算法的全局寻优能力。最后,采用 10 个标准测试函数验证了所提 DETS 算法在收敛速度、可靠性及解的质量方面的优势;同时,针对某市电力管线配电网络线路规划模块,利用所提算法实现了中低压配电网络的线路规划。

参 考 文 献

[1] Gonen T. Electric power distribution engineering [M]. Boca Raton;CRC Press,2014

[2] Willis H L. Power distribution planning reference book[M]. Boca Raton;CRC Press,2010

[3] Ma Jun, Zhang Yi-bin. Security protocol based on IEC60870-5-104 for communication in distribution automation[J]. Computer Science,2013,40(11):81-84(in Chinese)

马钧,张一斌. 基于 IEC60870-5-104 的配电自动化通信安全协议[J]. 计算机科学,2013,40(11):81-84

[4] Liu Jun,Liu Zi-fa,Huang Wei,et al. An intelligent distribution network planning method based on geographical differential evolution[J]. Automation of Electric Power Systems,2007,31(2):32-35(in Chinese)

刘军,刘自发,黄伟,等. 基于混合微分演化算法的配电网架结构智能规划[J]. 电力系统自动化,2007,31(2):32-35

[5] Floudas C A, Gounaris C E. A review of recent advances in global optimization [J]. Journal of Global Optimization, 2009, 45 (1):3-38

[6] Ganguly S, Sahoo N C, Das D. Recent advances on power distribution system planning; a state-of-the-art survey [J]. Energy Systems,2013,4(2):165-193

[7] Li You,Chang Xian-rong. Distribution network intelligent planning considering feeder line automatic layout and tie-line wiring investment[J]. Automation of Electric Power Systems,2012,36 (14):30-35

[8] Wang D T C,Ochoa L F,Harrison G P. Modified GA and data

算法,而且 DETS 算法标准偏差为 0,说明算法比较稳定。此外,为了验证 DETS 算法相对于所比算法是否具有显著性优势,本文选用 t-test 进行非参数假设检验,结果如表 7 最后一列所列,其中“+”表示所提算法显著优于所比算法,“≈”表示所提算法与所比算法之间没有显著性差异。可见,所提算法显著优于 DE 算法和 SaDE 算法;对于 CoDE 算法,虽然与 DETS 算法相比,两者的结果没有显著性差异,但是 DETS 算法的平均收敛速度及时间均优于 CoDE 算法。总体来说,DETS 算法不仅可以防止 DE 算法早熟收敛,而且可靠性较高,收敛速度较快。

envelopment analysis for multistage distribution network expansion planning under uncertainty[J]. IEEE Transactions on Power Systems,2011,26(2):897-904

[9] 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

[10] Zhou Xiao-gen,Zhang Gui-jun,Hao Xiao-hu. Differential evolution algorithm with local abstract convex region partition[J]. Acta Automatica Sinica,2015,41(7):1315-1327(in Chinese)

周晓根,张贵军,郝小虎. 局部抽象凸区域划分差分进化算法[J]. 自动化学报,2015,41(7):1315-1327

[11] Das S, Suganthan P N. Differential evolution: A survey of the state-of-the-art[J]. IEEE Transactions on Evolutionary Computation,2011,15(1):4-31

[12] Zhou Xiao-gen,Zhang Gui-jun,Mei Shan,et al. Differential evolution algorithm based on abstract convex underestimate selection strategy[J]. Control Theory and Applications,2015,32(3):388-397(in Chinese)

周晓根,张贵军,梅珊,等. 基于抽象凸估计选择策略的差分进化算法[J]. 控制理论与应用,2015,32(3):388-397

[13] Mendes R, Mohais A S. Dynde: a differential evolution for dynamic optimization problems [C] // Proceedings of the 2005 IEEE Congress on Evolutionary Computation, 2005. IEEE, 2005,3:2808-2815

[14] Das S,Konar A,Chakraborty U K. Annealed differential evolution[C]//Proceedings of the 2007 IEEE Congress on Evolutionary Computation. IEEE,2007:1926-1933

[15] Bhattacharya A,Chattopadhyay P K. Hybrid differential evolution with biogeography-based optimization for solution of economic load dispatch[J]. IEEE Transactions on Power Systems, 2010,25(4):1955-1964

[16] Song X,Tang L. A novel hybrid Differential Evolution-Estimation of Distribution Algorithm for dynamic optimization problem [C]//Proceedings of the 2013 IEEE Congress on Evolutionary Computation. IEEE,2013:1710-1717

[17] Qin A K,Huang V L,Suganthan P N. Differential evolution algorithm with strategy adaptation for global numerical optimization [J]. IEEE Transactions on Evolutionary Computation, 2009,13(2):398-417

- [18] Wang Y, Cai Z, Zhang Q. Differential evolution with composite trial vector generation strategies and control parameters[J]. IEEE Transactions on Evolutionary Computation, 2011, 15(1): 55-66
- [19] Glover F, Laguna M. Tabu Search[M]. New York: Springer, 2013
- [20] Cai Yan-guang, Tang Ya-lian, Zhu Jun. Hybrid tabu search algorithm for solving incident vehicle routing problem[J]. Computer Science, 2015, 42(4): 230-234, 273(in Chinese)
蔡延光, 汤雅连, 朱君. 混合禁忌搜索算法求解关联运输调度问题[J]. 计算机科学, 2015, 42(4): 230-234, 273
- [21] Sahoo N C, Ganguly S, Das D. Multi-objective planning of elec-

trical distribution systems incorporating sectionalizing switches and tie-lines using particle swarm optimization[J]. Swarm and Evolutionary Computation, 2012, 3: 15-32

- [22] Yan Wei, Liu Fang, Wang Guan-jie. Phase component power flow calculation of three phase radiation distribution network[J]. Automation of Electric Power Systems, 2002, 26(10): 24-27 (in Chinese)
颜伟, 刘方, 王官洁. 三相辐射型配电网的相分量潮流计算[J]. 电力系统自动化, 2002, 26(10): 24-27
- [23] Corder G W, Foreman D I. Nonparametric Statistics for Non-Statisticians: A Step-By-Step Approach[M]. Hoboken: John Wiley & Sons, 2009

(上接第 213 页)

参 考 文 献

- [1] Shi Kai-quan. Inverse P-sets [J]. Journal of Shandong University(Natural Science), 2012, 47(1): 98-109(in Chinese)
史开泉. 逆 P-集合[J]. 山东大学学报(理学版), 2012, 47(1): 98-109
- [2] Shi Kai-quan. Function inverse P-sets and the hiding information generated by function inverse P-information law fusion [C]// 13th IFIPWG 6. 11 Conference on e-Business, e-Services, and e-Society, 13E 2014 Sanya, China, November, 28-30, 2014 Proceedings Digital Services and Information Intelligence. 2014: 224-237
- [3] Ren Xue-fang, Zhang Ling, Shi Kai-quan. Surplus-deficiency of cardinal number and inverse P-augmented matrices [J]. Journal of Shandong University(Natural Science), 2015, 50(10): 13-18 (in Chinese)
任雪芳, 张凌, 史开泉. 基数余-亏与逆 P-增广矩阵[J]. 山东大学学报(理学版), 2015, 50(10): 13-18
- [4] Lin Hong-kang, Fan Cheng-xian. Embedding camouflage of inverse P-information and applications [J]. International Journal of Convergence Information and Technology, 2012, 7(20): 471-480
- [5] Fan Cheng-xian, Huang Shun-liang. Inverse P-reasoning discovery identification of inverse P-information [J]. International Journal of Digital Content Technology and its Applications, 2012, 6(20): 735-744
- [6] Shi Kai-quan. P-sets, inverse P-sets and the intelligent fusion-filter identification of information [J]. Computer Science, 2012, 39(4): 1-13(in Chinese)
史开泉. P-集合, 逆 P-集合与信息智能融合-过滤辨识[J]. 计算机科学, 2012, 39(4): 1-13
- [7] Shi Kai-quan, Tang Ji-hua, Zhang Ling. Intelligent fusion of inverse P-information and recessive transmission of information intelligent hiding [J]. Systems Engineering and Electronics, 2015, 37(3): 599-605(in Chinese)
史开泉, 汤积华, 张凌. 逆 P-信息智能融合与信息智能隐藏的隐性传递[J]. 系统工程与电子技术, 2015, 37(3): 599-605
- [8] Shi Kai-quan. P-sets[J]. Journal of Shandong University(Natural Science), 2008, 43(11): 77-84(in Chinese)
史开泉. P-集合[J]. 山东大学学报(理学版), 2008, 43(11): 77-84
- [9] Shi Kai-quan. P-sets and its application [J]. Advances in Sys-

tems Science and Applications, 2009, 9(2): 209-219

- [10] Zhang Ling, Tang Ji-hua, Shi Kai-quan. The fusion of internal P-information and its feature of attribute conjunction [J]. Journal of Shandong University(Natural Science), 2014, 49(2): 93-97(in Chinese)
张凌, 汤积华, 史开泉. 内 P-信息融合与它的属性合取特征[J]. 山东大学学报(理学版), 2014, 49(2): 93-97
- [11] Shi Kai-quan, Zhang Li. Internal P-sets and data Outer-recovery [J]. Journal of Shandong University(Natural Science), 2009, 44(4): 8-14(in Chinese)
史开泉, 张丽. 内 P-集合与数据外-恢复[J]. 山东大学学报(理学版), 2009, 44(4): 8-14
- [12] Shi Kai-quan. P-sets and its applied characteristics [J]. Computer Science, 2010, 37(8): 1-8(in Chinese)
史开泉. P-集合与它的应用特性[J]. 计算机科学, 2010, 37(8): 1-8
- [13] Shi Kai-quan. P-reasoning and P-reasoning discovery- identification of information [J]. Computer Science, 2011, 38(7): 1-9(in Chinese)
史开泉. P-推理与信息 P-推理发现-辨识[J]. 计算机科学, 2011, 38(7): 1-9
- [14] Zhang Li, Xiu Ming, Shi Kai-quan. P-sets and Applications of Internal-Outer Data Circle [C]// Proceedings of 2nd International Conference on Quantitative Logic and Soft Computing; Quantitative Logic and Soft Computing. Berlin: Springer-Verlag, 2010: 581-591
- [15] Qiu Yu-feng, Chen Bao-hui. f -Model Generated by P-Set [C]// Proceedings of 2nd International Conference on Quantitative Logic and Soft Computing; Quantitative Logic and Soft Computing. Berlin: Springer-Verlag, 2010: 613-620
- [16] Li Yu-ying, Zhang Ling, Shi Kai-quan. Generation and Recovery of Compressed Data and Redundant Data [C]// Proceedings of 2nd International Conference on Quantitative Logic and Soft Computing; Quantitative Logic and Soft Computing. Berlin: Springer-Verlag, 2010: 661-671
- [17] Zhang Ling, Ren Xue-fang. P-Sets and its $(f, \bar{f})$ -Heredity [C]// Proceedings of 2nd International Conference on Quantitative Logic and Soft Computing; Quantitative Logic and Soft Computing. Berlin: Springer-Verlag, vol. 2, 2010: 735-743
- [18] Xiu Ming, Shi Kai-quan, Zhang Li. P-sets and $\bar{F}$ -data Selection-Discovery[C]// Proceedings of 2nd International Conference on Quantitative Logic and Soft Computing; Quantitative Logic and Soft Computing. Berlin: Springer-Verlag, 2010: 791-799