

社区服务管理系统中服务网点配置算法的研究

竺亚^{1,2} 胡克瑾² 唐坤²

(上海市民政局信息中心 上海 200002)¹ (同济大学 上海 200092)²

摘要 服务网点配置问题是在社区服务管理系统设计中需要解决的一个重要问题。由于社区服务具有海量、并发、不确定的特性,使得服务网点配置问题成为一个 NP-hard 问题,使用传统优化方法很难得到最优解或满意解。本文使用改进的遗传算法和绝对中心点算法,并用 VB 完成算法的编制,用来解决这个问题。试验证明获得了良好的效果。

关键词 社区服务管理系统,服务网点配置,遗传算法,绝对中心点算法

Research on Configuration Algorithm of Network Units' Position in Community Service Information System

ZHU Ya^{1,2} HU Ke-Jin² TANG Kun²

(Shanghai Civil Administration Bureau Information Center, Shanghai 200002)¹ (Tonji University, Shanghai 200092)²

Abstract Network units' position configuration is an important problem in community information system. Community service business's large number, simultaneity and uncertainty make the community service become a NP-hard problem, it is hard for traditional algorithm to draw an optimized or satisfied solution. This paper bases on improved genetic algorithm and absolute center algorithm, solves this problem with a VB program. Software experiment proves the algorithm designed by this paper has a good result.

Keywords Community service information system, Network units configuration, Genetic algorithm, Absolute center algorithm

1 问题描述

1.1 研究目的

社区服务管理系统需要根据市民需求的特点与分布,在适当的地点选择优质的服务单位加盟,并进行管理。那么如何针对当前的需求总量和分布情况,应当配置多少家服务网点?它们各自的位置如何?这是管理系统设计中需要解决的一个重要问题。

这个问题可以拿一个例子描述。如果社区服务中心每天收到来自居民的 1000 个家政服务需求,中心需要签约多少家政服务单位?它们的地理位置又如何?

1.2 基本条件

(1)模型研究以家政服务为例,本文使用的一切数据均基于实地调研。

(2)服务人员和居民家庭的比例关系为 1:3,即一个服务人员每天可以服务 3 户居民。

(3)对居民的服务存在粘性,也就是居民在成为社区服务的客户后,会持续使用服务一段时间。本文假设粘性时间为

6 个月,即 180 天。

(4)当前家政市场服务企业以平均每家拥有 90 名服务人员的企业最为普遍。

(5)居民对服务的需求间隔为 2 天,即成为客户的居民平均每 3 天需求一次服务。

(6)根据之前的假设,一个拥有 90 名服务人员的服务单位,最多能够为 $90 \times 3 \times 3 = 810$ 家居民客户提供服务。

(7)为了应对突发需求高峰,并保障每日服务的有序提供,每家服务单位应当保留一定服务资源空闲。本文将保留率设置为 10%,也就是说当服务单位的空闲服务人员少于 9 人,即服务居民的户数超过 729 家时,中心将停止为该服务单位继续指派任务。

(8)本文以街道为基础来研究服务网点的配置。 $729 \div 180 = 4.05$,也就是如果来自某个街道居民的需求平均每天超过 4.05 个的话,就需要为该街道添加一个服务网点。

2 服务网点的配置问题

可以通过两个阶段的计算解决服务网点配置问题。

竺亚 经济师,博士生,研究方向为公共服务、社区服务、电子政务。

子系统,是其他功能子系统进行数据交换和集成的重要工具。本文从电子政务资源整合的实际需要出发,提出一种基于 Web services 技术的数据交换中心的设计方案,并着重描述了其模型、结构以及工作过程,并在此基础上介绍了数据交换中心的实现。基于本方案的数据交换中心已经在某市电子政务平台系统中运行,得到了用户的认可,取得了较好的应用效果。

参考文献

1 王宁,王延章,叶鑫.一种基于数据中心的政府信息资源整合系统

架构设计. 计算机应用研究,2005,22(9):67~68,71

2 中科院中国电子政务研究中心. 电子政务红皮书 2002-07

3 高旻. Web services 的构件库的研究:[硕士论文]. 重庆大学,2005-05

4 张繁,蔡家楣. 电子政务系统中的数据交换和共享服务平台设计. 计算机工程与应用,2003,39(7):226~229

5 Tzvetkov V, Wang Xiong. DBXML - Connecting XML with Relational Databases. In: The Fifth International Conference on Computer and Information Technology (CIT'05),2005

第一阶段:

①如果来自某个街道的需求每天超过 4.05 个,则可以为街道单独设立服务单位,按照每天每 4.05 个需求为标准设立一家。

②在将可以独立设立服务网点的街道独立出来以后,剩余的街道需要联合起来配置服务网点。

③按照启发式算法来计算如何联合起多家街道来设立一家服务网点,本文使用遗传算法来进行联合的研究。

④根据计算结果,一个区的若干街道被分解成了若干个单独为街道设置的网点和若干由多个街道联合起来共同设置的网点。

第二阶段:

①如果网点是为某个街道单独设置的,那么网点位置选择在街道的中心位置上。

②如果多个街道联合设置一家网点,那么按照图论中绝对中心点问题的解决方法,算出多家街道的中心点位置。

3 服务网点配置建模

根据上面的研究,服务网点配置问题需要两个数学模型。

3.1 遗传算法的数学模型

联合街道问题的描述如下:有 k 个街道(1,2,...,k),第 i 个街道的需求为 g_i ($i=1,2,\dots,k$),每家服务网点的服务能力为 q 。已知 $g_i < q$,求能够满足服务需求并且各家街道中心围成的所有区域的总服务距离最短。

为了联合街道,必须对需要的服务网点数有一个估计,本文使用下面的公式^[1]来确定需要的服务网点数:

$$m = \lceil \sum g_i / aq \rceil + 1$$

m 为所需网点数, $\lceil \quad \rceil$ 表示取整,本文将 a 的值设为 0.85。

c_{ij} 表示点 i 到点 j 的服务成本,如时间、路程、花费等。社区服务中心的编号为 0,各分仓库的编号为 i ($i=1,2,\dots,k$),定义变量如下:

$$x_{ij,s} = \begin{cases} 1, & \text{服务人员 } s \text{ 由 } i \text{ 向 } j \\ 0, & \end{cases}$$

$$y_{is} = \begin{cases} 1, & \text{点 } i \text{ 的服务任务由服务人员 } s \text{ 完成} \\ 0, & \end{cases}$$

因此得到的数学模型表示如下:

$$\min Z = \sum_{i=0}^k \sum_{j=0}^k \sum_{s=1}^m c_{ij} x_{ij,s} \quad (1)$$

$$\sum_{i=0}^k g_i y_{is} \leq q, s=1,2,\dots,m \quad (2)$$

$$\sum_{s=1}^m y_{is} = \begin{cases} 1, & i=1,2,\dots,k \\ m, & i=0 \end{cases} \quad (3)$$

$$\sum_{i=0}^k x_{ij,s} = y_{j,s}, j=1,\dots,k; s=1,2,\dots,m \quad (4)$$

$$\sum_{j=1}^k x_{ij,s} = y_{i,s}, i=0,1,\dots,k; s=1,2,\dots,m \quad (5)$$

$$x_{ij,s} = 0 \text{ 或 } 1, j=0,1,\dots,k; s=1,2,\dots,m \quad (6)$$

上述模型中,式(2)为服务网点服务能力约束;式(3)保证了每个街道的服务任务仅由 1 家网点来完成,而所有街道服务任务则由 m 家服务网点协同完成;式(4)和式(5)限制了每个街道有且仅有 1 个服务网点提供服务。

3.2 绝对中心点问题的数学模型^[2,3]

定义 1 在网络图中,弧 $e_j = (v_p, v_q)$ 上某点 x 到顶点 v_i 的最短距离,称为点到顶点的距离。

根据定义,可以推导弧 $e_j = (v_p, v_q)$ 上某点 x 到顶点 v_i 的距离的计算公式:

$$x \in (v_p, v_q),$$

$$d(v_i, x) = \min \{ d(v_p, v_i) + d(x, v_p), d(v_q, v_i) + d(x, v_q) \} \quad (7)$$

令 $d(v_p, v_q) = b_j, d(v_p, x_j) = x$, 则 $d(v_q, x) = b_j - x$, 那么式(7)转化为

$$d(v_i, x) = \min \{ d(v_p, v_i) + x, d(v_q, v_i) + b_j - x \} \quad (8)$$

定义 2 在网络图 G 中,对于弧 e_j 上的点 x_j ,如果满足: $\min_{x \in e_j} \max_{1 \leq i \leq n} d(v_i, x) = \max_{1 \leq i \leq n} d(v_i, x_j) = r(x_j)$, 则称 x_j 为弧 e_j 上局部中心点, $r(x_j)$ 为弧 e_j 上的局部半径。

定义 3 对于网络图 G 中的点 x_0 ,如果满足: $\min_{x \in G} \max_{1 \leq i \leq n} d(v_i, x) = \max_{1 \leq i \leq n} d(v_i, x_0) = r(x_0)$, 则称 x_0 为网络 G 的绝对中心点, $r(x_0)$ 为绝对半径。

根据定义 2 和 3,可以得到求解绝对中心点和局部中心点之间的关系:

$$\min_{1 \leq j \leq n} [\min_{x \in e_j} \max_{1 \leq i \leq n} d(v_i, x_j)] = r(x_0)$$

因此绝对中心点的求解可以化为对 m 条弧 e_j ($j=1,2,\dots,m$) 上的局部中心点的求解^[4~6]。

4 算法设计

4.1 遗传算法设计^[7]

1)染色体结构。本文使用自然数为染色体编码。上节数学模型的解向量可编成一条长度为 $k+m+1$ 的染色体(0, $i_1, i_2, \dots, i_s, 0, i_j, \dots, i_k, 0, \dots, 0, i_p, \dots, i_q, 0$),在整条染色体中,自然数 i_j 表示第 j 个街道,0 的数目为 $m+1$ 个,代表社区服务中心,并把自然数编码分为 m 段,形成 m 个子路径,表示由 m 家服务网点完成所有服务任务。

2)遗传群体初始化。本文规定群体规模取值在 10 到 100 之间。

在初始化染色体时,先生成 k 个街道的一个全排列,再将 $m+1$ 个 0 随机插入排列中。需要注意的是,必须有 2 个 0 被安排在排列的头部和尾部,并且在排列中不能有连续的 2 个 0。这样就构成一条满足问题需要的染色体。重复这一过程,直至生成满足群体规模数的染色体。

3)计算适应度函数。本文将容量约束式(2)转为路程成本的一部分,公式变为

$$Z = \sum_{i=0}^k \sum_{j=0}^k \sum_{s=0}^m c_{ij} x_{ij,s} + M \sum_{s=0}^m \max(\sum_{i=0}^k g_i y_{is} - q, 0)$$

M 为一大正数,表示当一家服务网点的服务量超过它每天最大服务能力时的惩罚系数。在本文的软件实验中, M 取值为 1,000,000。

将服务路程成本转换至适应度函数: $f_i = z' / z$, 其中 f_i 为第 i 条染色体的适应度, z' 为当前群体中最优染色体的路程成本, z 为第 i 条染色体的运输成本。

最后计算染色体适应度的概率分布: $p_i = f_i / \sum f_i$, 其中 p_i 表示第 i 条染色体的适应度概率分布, f_i 表示第 i 条染色体的适应度。

4)交叉算子。将每代种群的染色体中适应度最大的染色体直接复制,进入下一代。种群中其它染色体按其适应度的概率分布,按照随机的方法产生子代。这样既保证了最优者可生存至下一代,又保证了其余染色体可按生存竞争的方法生成子代,使得算法可收敛到全局最优。选中的染色体按一个既定概率——交叉率,产生子代。本文规定交叉率为 0.6。

由于服务网点配置问题的条件约束,如果仍使用简单交叉算子,将有可能产生大量的不可行解。比如,父代 1:0 |

387|04021650 和父代 2: 0|248|03170560, | 内为进行交叉的基因段,交叉位随机选定。经过简单交叉运算后,将产生子代 1: 024804021650 和子代 2: 038703170560。由于初始的问题约束中,在任何一条路径里,服务人员必须并只能访问同一街道 1 次。但在产生的子代中,子代 1 访问了街道 2、4 两次,未访问到街道 3、7;子代 2 访问了街道 3、7 两次,未访问到街道 2、4,因此它们均不是可行解。

因此,这里设计了经过改进的交叉算子,避免非可行解的产生。具体做法如下:

依然随机产生交叉位,如果在交叉位的外侧两端的父代基因都是 0 的话,则对父代进行简单交叉;如果交叉位的外侧两端的父代基因不为 0 的话,则将左交叉位向左移,直至移动到左交叉位的左端基因为 0 时停止。以左交叉位为起点,继续向右移动右交叉位,直至右交叉位的右端基因为 0 时停止。这样就保证了父代染色体都用 1 条子路径来进行交叉。

经过第 1 步的操作,子代中仍然会产生访问同一街道 2 次的情况,需要对子代进行整理。在子代中选择那些重复的自然数,如果该自然数并非在交叉位内的话,则删除之。子代中如存在未访问到某一街道的情况,则在染色体的交叉位外补上该街道对应的自然数。

经过第 2 步的操作,如果产生了某一子路径为空的情况,即染色体中含有 2 个连续的 0,须继续进行第 3 步操作,将其中的 1 个 0 与同染色体其他位上的街道自然码进行交换。对该街道自然码的要求是该码的前一位与后一位均不能为 0。本步操作可放在对子代变异之后进行。

5)变异算子。本文规定变异率在 0.01~0.2 之间,默认变异率为 0.02。变异的策略是随机交换 2 个基因的位置。

在变异后,需用交叉算子的第 3 步操作整理子代,以保证子代的可行性。

6)自然选择。计算子代的适应度,父代和子代中性能最优的 2 个染色体进入种群。

7)结束条件。当算法的当前进化代数小于预先设定的 N 时,返回 3,算法继续。当进化代数= N 时,计算完成。

4.2 绝对中心点算法设计^[8]

(1)先求出网络图 G 的顶点-顶点的距离矩阵 D ;

(2)用 Hakimi 法,找出网络图 G 中的所有无向弧,逐条建立起该弧的点与各顶点距离的关系;

(3)将每条无向弧到所有顶点的距离的图形画在同一坐标上,取曲线的最上面部分,求出最小值,从而得到了局部中心点;

(4)比较所有无向弧的局部中心点,具有最小局部半径的弧上的局部中心点就为绝对中心点。

5 实验分析

本文使用 VB 语言编写了一个程序,用来验证论文算法的有效性。规定各街道坐标在 $[0,100] * [0,100]$ 间随机生成,单个服务节点的最大服务能力为 8。各街道的需求量在 $[0,4]$ 间随机生成。为了使单个服务节点能承担更多的任务,满载系数 α 取 0.85,默认交叉率为 0.6,默认变异率为 0.02。各街道间的服务距离成本 c_{ij} ,由各街道间的直线距离决定,即

$$c_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

实验 1 随机生成 1 个有 8 个街道的服务网点配置问题,得初始数据如表 1 所示。

表 1 实验 1 的初始数据

	区中心	街道 1	街道 2	街道 3	街道 4	街道 5	街道 6	街道 7	街道 8
位置	(31,9)	(76,38)	(77,16)	(90,82)	(60,74)	(76,86)	(11,31)	(25,90)	(10,60)
需求量	0	2.46	0.41	2.16	2.27	1.83	3.76	2.54	2.39

根据各街道的需求量计算出需要的服务网点数:

$$m = \left\lceil \frac{17.82}{0.85 * 8} \right\rceil + 1 = 3$$

取群体规模 $pop=20$,进化代数 $N=50$,使用第 2 节所述算法,在 P4 1.6G 的机器上计算,耗时 0.75s,得到下列结果。

子路径 1:0→8→7→4→0;子路径 2:0→6→0;子路径 3:0→5→3→1→2→0;总服务距离:476.29。软件运行结果如图 1 所示。

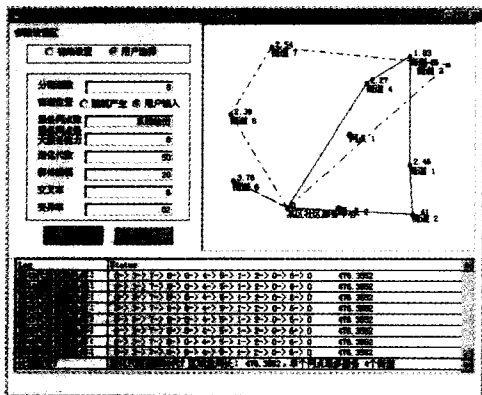


图 1 实验 1 的软件运行结果

实验 2 随机生成 1 个有 20 个街道的服务网点配置问

题,得初始问题,如表 2。

表 2 实验 2 的初始数据

区中心	街道 1	街道 2	街道 3	街道 4	街道 5	街道 6
(52,4)	(15,49)	(0,61)	(51,15)	(25,71)	(38,62)	(35,45)
0	1.64	1.31	0.43	3.38	1.13	3.77
街道 7	街道 8	街道 9	街道 10	街道 11	街道 12	街道 13
(100,41)	(10,52)	(26,79)	(87,7)	(24,89)	(19,25)	(20,99)
3.48	0.39	0.24	1.03	2.35	2.60	1.00
街道 14	街道 15	街道 16	街道 17	街道 18	街道 19	街道 20
(73,91)	(100,95)	(7,73)	(69,86)	(24,3)	(66,14)	(9,30)
0.65	0.58	2.56	1.27	2.69	3.26	2.97

取群体规模 $pop=100$,进化代数 $N=50$,在 P4 1.6G 的机器上计算,耗时 1.5s,得到下列结果。

需要网点数:6

最优路径:0→6→1→0→18→0→12→20→8→0→2→16→11→9→5→0→19→7→10→0→3→13→4→17→14→15→0

总服务距离成本:964.48

从实验 1 和实验 2 的运行结果来看,实验 2 的问题规模是实验 1 的 2.5 倍,但实验 2 的运行时间与实验 1 相比并没有呈指数型增长,仅是实验 1 的 2 倍,可以认为是呈线性变化的方式。可以看出,本文设计的算法对较大规模的服务网点问题的运算效率还是令人满意的。

实验 3 使用实验 2 的初始数据,但进化代数分别设置

为 20,50 和 100 代。得到的运行结果如表 3 所示。

表 3 实验 3 的运行结果

实验	实验 3(1)	实验 3(2)	实验 3(3)
进化代数	20	50	100
总服务距离成本	1153.12	964.48	908.44

从实验 3 的结果可以看出,本文算法的求解结果随着进化代数的增加而逐渐优化,但这种解的优化趋势是在不断变缓的。如果在实际应用中需要兼顾运行结果和运行速度的话,进化代数可取在 50 到 100 之间。

在计算结束后,如果区域总周长大于 M,可认为算法产生了不可行解,这是因为指定的服务网点数过少的缘故。

结束语 从试验结果上看,本文设计的服务网点配置问题的模型和算法具有以下优点:

①运算速度快。服务网点配置问题是一个 NP-hard 问题,本文设计的算法在保证能够获得一个较优化结果的同时,

又保证了计算的速度。

②服务区域可任意扩大或缩小。即使对整个城市或者更大的区域进行配置时,也能维持一个较高的计算速度。

参考文献

- 1 李军,谢秉磊,郭耀煌.非满载车辆调度问题的遗传算法.系统工程理论与实践,2000,20(3):235~239
- 2 方磊,何建敏.给定限期条件下的应急系统优化选址模型及算法[J].管理工程学报,2004,18(1):48~51
- 3 Harche F,Thompson G L. The column subtraction algorithm: An exact method for solving weighted set covering, packing and partitioning problems[J]. Computers Ops Res, 1994,21(6):689~705
- 4 汪寿阳,赵秋红,夏国平.集成物流管理系统中定位——运输路线安排问题的研究[J].管理科学学报,2000,3(6):69~75
- 5 Douglas M I, Papayanopoulos L. Facility location on a tree with maximum distance constraints. Computers and Operations Research, 1995,22(9):905~914
- 6 唐坤.车辆路径问题中的遗传算法设计[J].东华大学学报(自然科学版),2002,28(1):66~70
- 7 Tamir A, Perez -Brito D, Moreno-Perez J A. Polynomial algorithm for the p-centdian problem on a tree networks, 1998,32(14):255~262

(上接第 88 页)

```

RtDataServe. getRtData (OnGetRtData );
}
//设置定时器,每 2 秒获取一次实时数据
function run(){
    window.setTimeout("run()", 2000);
    callServer ();
}
</script>
</head>
<body onload="run()">
    <embed id="svgobj" src="zone.svg" width="800"
height="600" type="image/svg+xml">
    .....
</body>
</html>

```

3.2 性能分析

服务器配置为 2 个 P4 Xeon 3.0GHz 的 CPU,1GB 的内存,安装 Windows 2000 Server 操作系统,tomcat5.0。客户端配置为 P4 2.8GHz 的 CPU,512M 的内存,安装 Windows XP Professional 操作系统。数据库安装在单独的机器上。网络环境为局域网。我们设定客户端电子地图更新间隔是 2 秒。用程序模拟车辆、吊车、箱位的变化来产生数据,每 2 秒有一定比例的堆场元素(指箱位、车辆和吊车)的位置或状态信息被更新,客户端响应时间指从发送异步调用到结果返回之间的平均时间间隔。测试结果见表 1。

表 1 客户端负载随堆场元素更新比例的变化

程序 每 2 秒钟元素更新比例(%)	客户端		
	CPU (%)	内存 (K)	响应时间 (ms)
5	11	82,104	<500
10	11	82,344	<500
20	13	82,696	<500
30	14	83,012	<520
50	15	84,976	<530
70	16	85,121	<550

随着堆场元素更新比例的增加,监控客户端程序的 CPU 使用率和内存使用量稍有增加,因为单位时间内所需处理的数据量增加了,但变化不大。

随着客户端数的增加,服务器的 CPU 使用率和内存用量均有少量增加,因为服务器端要开辟更多的线程来与客户端交互。

结论 本文主要讨论了如何构建基于 SVG 的实时 Web GIS 的技术方法,并在一个实例系统上取得了较好的效果。

GIS 在我国经历了 20 多年的发展,已广泛应用于城市规划、电信网、电力网、土地管理、交通控制、房地产、旅游、公安、消防、资源、环境和国土规划等领域。SVG 作为 XML 在图形显示方面的一个应用,也越来越受到人们的重视。由于 XML 有诸多的优点,完全可以将 XML 扩展应用到 GIS 的多个方面。信息技术的进步,网络技术的迅速发展,使得 GIS 与网络的结合成为必然。而 GIS 与 XML 的紧密结合,也必将深入到信息社会的各个领域,在网络信息时代具有广阔的发展前景。

参考文献

- 1 白玉琪,杨崇俊,刘冬林,等.基于 OpenGIS WMS 的空间信息搜索引擎系统原型.中国图象图形学报,2004,9(1):105~111
- 2 Bowler J, Brown C, Capsimalis M, et al. Scalable Vector Graphics(SVG) 1.0 Specification. W3C, 2001
- 3 张丽静,杨继家,商李彪. SVG 技术在实时信息发布系统中的应用[J].计算机仿真,2006,1(1):255~256
- 4 W3C, Scalable Vector Graphics (SVG). http://www.w3.org/tr/svg
- 5 W3C, Document Object Model (DOM). http://www.w3.org/TR/REC-DOM-Level-1
- 6 杨昆,许泉立. SVG 在 Web GIS 中的应用研究.测绘与空间地理信息,2005,28(3):2~3
- 7 高峰,谈俊忠. JavaScript 在基于 SVG 的网络地图中的应用.江西师范大学学报(自然科学版),2004(3):263~265
- 8 梁民,汪伟.基于 AJAX 技术开发 Web 应用.电脑知识与技术,2006,(5):119
- 9 杨国瑞,张思博.基于 AJAX 的 Web 应用架构设计.现代电子技术,2006(15):95~97