

基于遗传算法的配 送路线优化系统



——测试报告

汇报小组：HADEAN

目录

1 项目概要	2
1.1 项目信息	2
1.2 测试阶段	2
1.3 测试环境	2
1.3.1 硬件信息	2
1.3.2 软件信息	3
1.3.3 测试人员	3
2 测试分析	3
2.1 测试案例与测试结果	3
2.2 测试结果分析	9
2.3 测试结果与建议	10
2.3.1 风险分析及建议	10
2.3.2 测试结论	11

1 项目概要

1.1 项目信息

本项目是基于遗传算法的配送路线优化系统的实现，车辆由单一起点出发，配送多个终点后在回到起点，根据车辆数量，承载限制，不同车辆服务成本、运行里程限制的条件选择最优路径，使成本最小化，配送订单最大化，满载率最大化。

1.2 测试阶段

本测试处于项目收尾阶段，主要进行确认测试，确定系统可以正常运行，并符合需求说明书中的全部功能与性能要求，从用户角度进行确认，保证系统可正常运行，完成配送路线的优化求解。

1.3 测试环境

1.3.1 硬件信息

设备型号	DELL XPS 13 9360
处理器	Intel(R) Core(TM) i7-8550U CPU @ 1.80GHZ 2.00GHZ
内存空间	16.0GB

1.3.2 软件信息

操作系统	Windows 10
开发环境	Visual Studio 2017

1.3.3 测试人员

顾育同 刘强华 梁家硕 郑江浩

2. 测试分析

2.1 测试案例与测试结果

测试案例 1

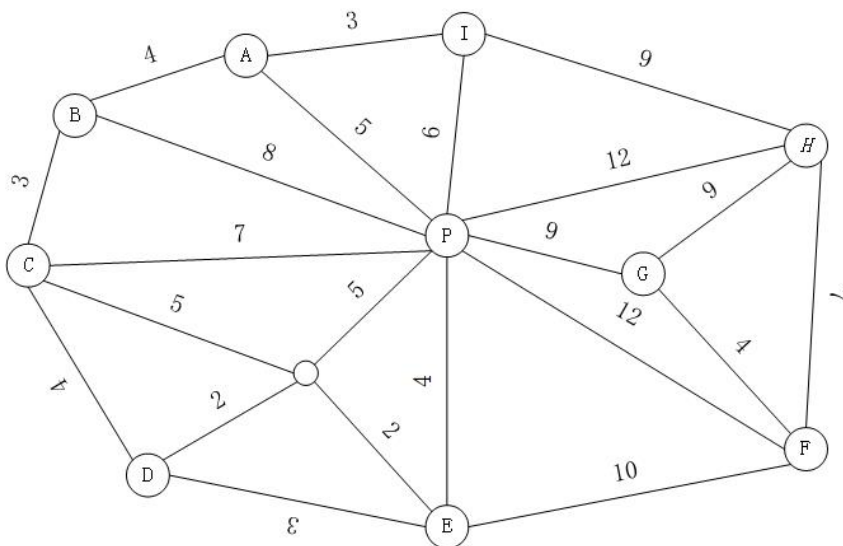


图 1 普通图案例

车辆种类	2
车辆成本	10、18
车辆数量	5、5
车辆载重	2、5
最大里程数	35

车辆速度	10
最大行驶时间	8
单次卸货时间	5

表 1 普通图限制信息

配送点名称	P (配送中心)	A	B	C	D	E	F	G	H	G	I
配送量	0	1.7	0.8	1.3	2.8	1.9	3.5	0.9	0.3	0.9	1.2

表 2 普通图配送点信息

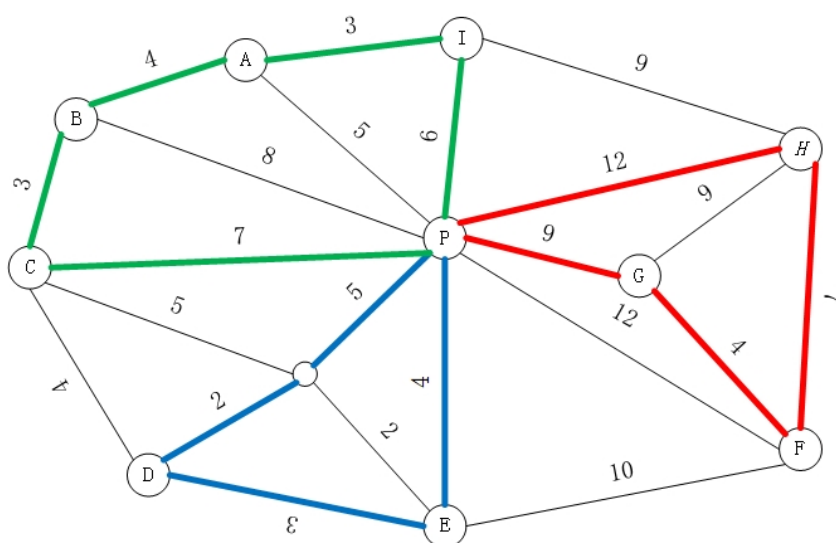


图 2 普通图方案

权重选择	车辆里程最短
(实际载重量/车辆载重)	(车辆路程/单次最大里程)
第一条路径 (5/5)	P, C, B, A, I, P (23/35)
第二条路径 (4.7/5)	P, E, D, P (14/35)
第三条路径 (4.7/5)	P, H, F, G, P (32/35)

表 3 普通图路径方案

测试案例 2

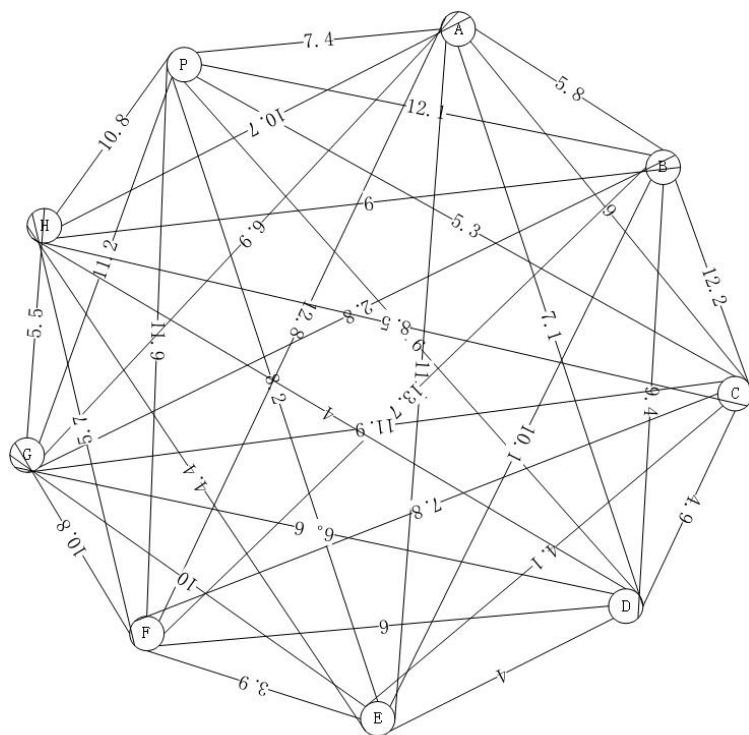


图 3 稠密图案例

车辆种类	1
车辆成本	10
车辆数量	10
车辆载重	8
最大里程数	25
车辆速度	10
最大行驶时间	8
单次卸货信息	5

表 4 稠密图限制信息

配送点名称	P(配送中心)	A	B	C	D	E	F	G	H
配送量	0	2	1.5	4.5	3	1.5	4	2.5	3

表 5 稠密图配送点信息

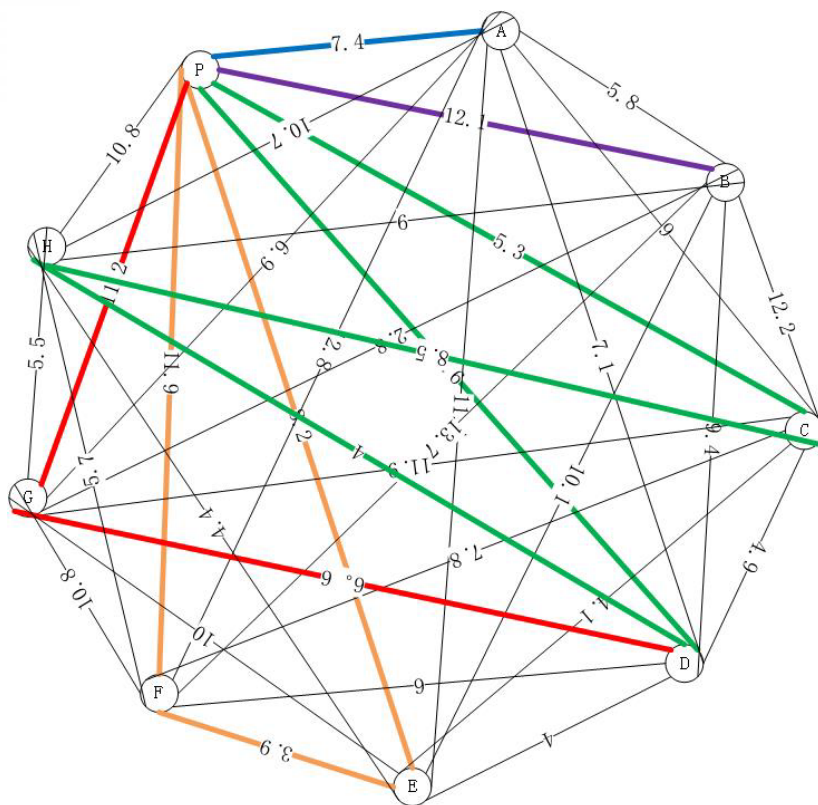


图 4 稠密图方案

权重选择	车辆里程最短
(实际载重/车辆载重)	(车辆路程/单次最大里程)
第一条路径 (7.5/8)	P, C, H, (D), P (24/25)
第二条路径 (5.5/8)	P, E, F, P (24/25)
第三条路径 (5.5/8)	P, G, D, P (24.4/25)
第四条路径 (1.5/8)	P, B, P (24.2/25)
第五条路径 (2/8)	P, A, P (14.8/25)

表 6 稠密图路径方案

测试案例 3

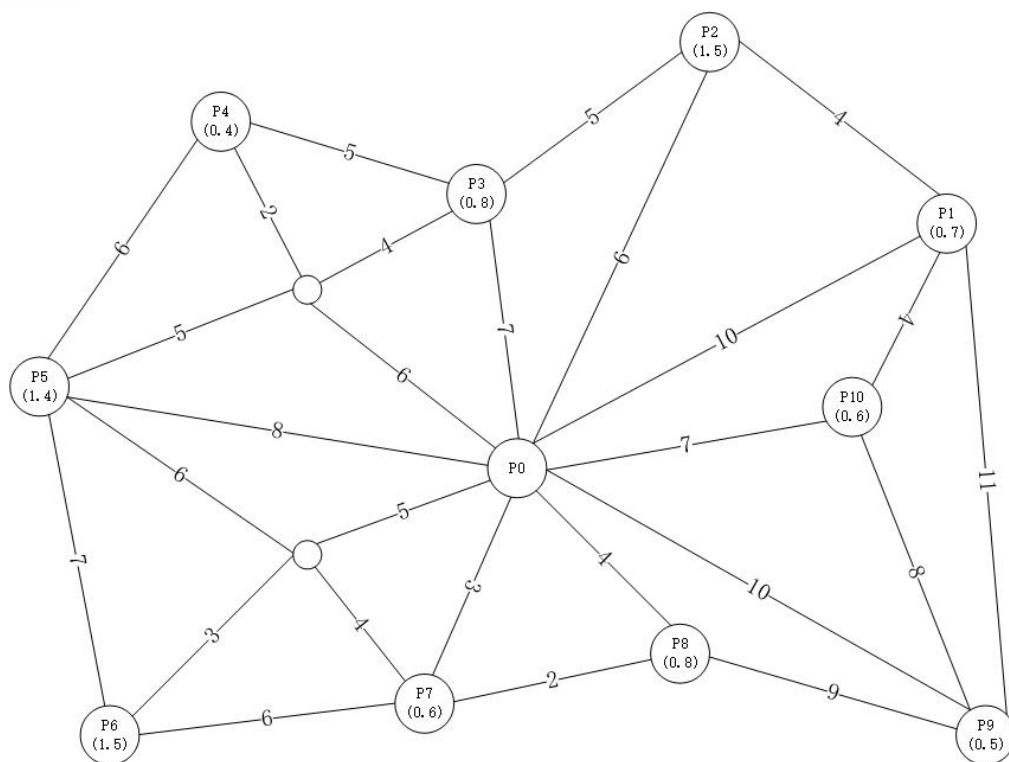


图 5 多节点图案例

车辆种类	2
车辆成本	10, 15
车辆数量	20
车辆载重	2, 4
最大里程数	35
车辆速度	10
最大行驶时间	8
单次卸货信息	5

表 7 多节点图限制信息

配送点名称	P0(配送中心)	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
配送量	0	0.7	1.5	0.8	0.4	1.4	1.5	0.6	0.8	0.5	0.6

表 8 多节点图配送点信息

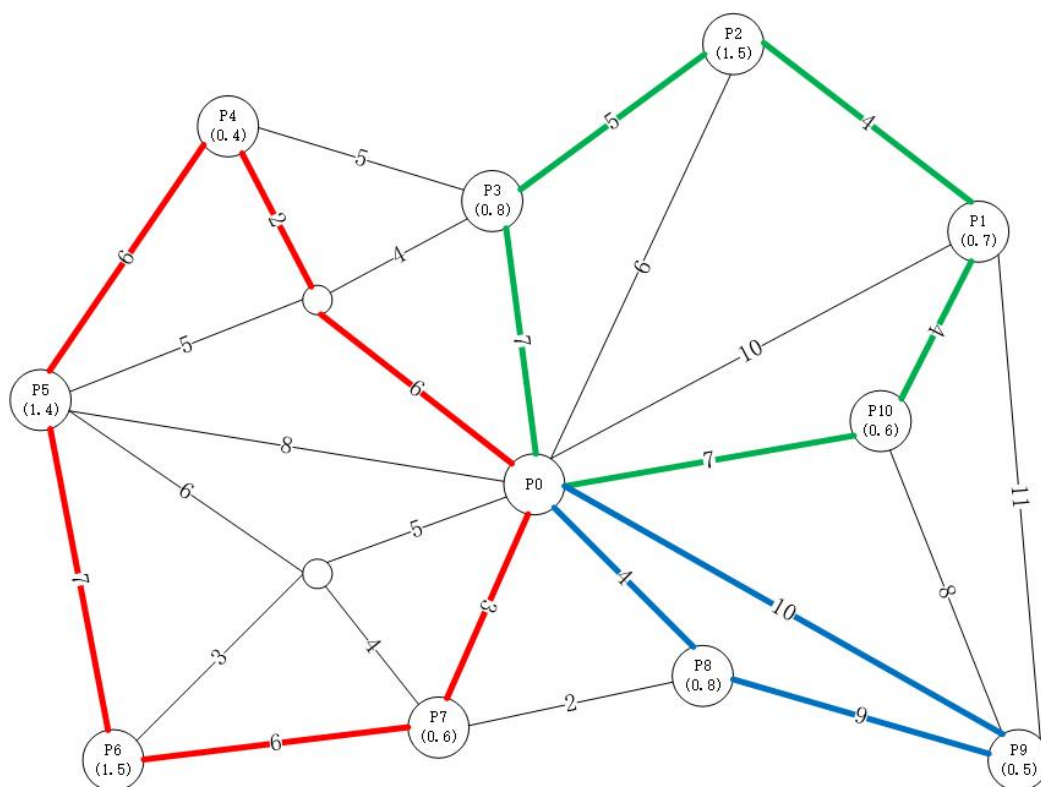


图 6 多节点图方案

权重选择	车辆里程最短
(实际载重/车辆载重)	(车辆路程/单次最大里程)
第一条路径 (3.9/4)	P0, P4, P5, P6, P7, P0 (30/35)
第二条路径 (3.6/4)	P0, P3, P2, P1, P10, P0 (27/35)
第三条路径 (1.3/2)	P0, P9, P8 (23/35)

表 9 多节点图路径方案

2.2 测试结果分析

测试案例 1 是普通图的案例，其中包含多个普通节点和一个路口节点。通过图 2 可以看出，测试所出的方案中，各条路径之间没有交叉，各成一个圆环，对于配送点的遍历没有出现重复。同时从表 3 可以看出每辆配送车的满载率也是十分的高的，最低也是 4.7/5，充分利用了每辆车的载重能力，也保证了每辆车的路程不超过最大限制路程，在满足各限制条件下，得到了最优方案。

测试案例 2 是稠密图的案例，其中包含多个普通节点，每个节点之间进行了全连接的操作，所以拥有较多的边数。通过图 4 可以看出，测试结果中，每条路径之间没有出现交叉，各成一个循环，除了有两条路径在 D 点出现了重复经过，其中一个为配送经过，一个是返回并且非配送经过，经过分析，可以发现在使用 D 点作为跳板进行返回配送中心的路径，是在该情况下的最短选择，所以选择存在合理性。从表 4、表 6 可以看出本案例中单次最大里程设置比较极限，设置为 25，而方案中多数里程为 24 或 24.4，十分逼近限制值。在配送 A 点和 B 点的时候，可以看出存在一个循环 PABP，走出来的总路程更短，但由于该循环的路程达到 25.3，超过了单次最大里程，所以没有出现该循环，而是分成了两次进行了配送。可以看出限制条件对于方案选取有着十分有效的影响，在限制下的出最优方案。

测试案例 3 是多节点图，有着多个普通节点和两个路口节点，通过图 6 可以看出，测试得到方案中，每条路径没有出现交叉且各成一个圆环。同时从表 9 看出，在满载率上，都保持在 50%以上，也都符合单次最大路

程的限制。而两个路口节点中，只有一个路口节点进行了使用，可以看出对路口节点与配送点之间的区分是有效的，并不会把路口节点包括在必须遍历的节点集中，在符合限制下得到了最优结果。

通过所有测试案例来看，所得出的结果中不同循环均没有出现交叉的情况，至少保证了最优方案是在循环未交叉的方案中，进行优化产生的。唯一出现了单边重合情况的案例 2，经过检查发现，确实是在通过重复节点回到配送点，属于该情况下的最优解。案例 3 中有两个路口节点，根据测试结果来看，有一个路口节点是未被经过的，可以证明，路口节点在优化方案中并不是必须经过的，符合最初的设计想法。同时在对三种测试案例进行测试的过程中，发现前两个案例的运行时间都比较接近，但是在对多节点图进行测试时，发现时间明显变长，因此可以看出图的节点数的规模与边数的规模对算法运行的影响中，节点规模的影响要更大一些，因为随着节点数的增加，组合方案的可能性会进行指数级的增长，所以对于运行速度的影响比较明显。总体上通过对三个案例测试，可以看出均可以得到最优解，可以完成所预计的功能。

2.3 测试结果与建议

2.3.1 风险分析及建议

在测试过程中发现在面对多节点图的过程中，遗传算法可能需要更多的时间与迭代才能够得到最优方案，而在面临数据量更大的多节点时，可能因可能性的指数增长，导致需要迭代的种群规模与迭代次数出现大幅上涨，因此建议在处理大型多节点图的时候，最好将多节点图进行拆成更小

规模的几张图，用几张图依次进行路径优化，可以提升求解速度以及减小参数规模。

2.3.2 测试结论

系统达到了最初设计用来物流路线优化的目标，实现了所需要的系统功能，确认可以完成配送路线优化的任务。