



2017 湖南省研究生数学建模竞赛参赛承诺书

我们仔细阅读了湖南省研究生数学建模竞赛的竞赛规则。

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与队外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛规则的，如果引用别人的成果或其他公开的资料（包括网上查到的资料），必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺，严格遵守竞赛规则，以保证竞赛的公正、公平性。如有违反竞赛规则的行为，我们将受到严肃处理。

我们授权湖南省研究生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

我们参赛选择的题号是（从组委会提供的试题中选择一项填写）：B 题

我们的参赛报名号为（如果组委会设置报名号的话）：201718001004

所属学校（请填写完整的全名）：国防科学技术大学

参赛队员（打印并签名）：

1. 张佩俊
2. 马文凤
3. 孟德利

指导教师或指导教师组负责人(打印并签名)：

日期： 2017 年 4 月 25 日

评阅编号（由组委会评阅前进行编号）：

湖南省第二届研究生数学建模竞赛

题目

智慧考古探测

摘

要：

本文从考古探测车探测过程中的各种约束条件出发，以时间最优为目标，首先确定探测车初始布局设计的总体原则和优化方案，然后把确定探测车的初始布局、行走路线及探测次序这样一个多起点多约束路径规划问题转化为考虑干涉的多背包问题

(Knapsack Problem, KP)，并针对具体问题建立相应的模型，通过基于协同进化 (Coevolutionary Algorithm, CEA) 的粒子群算法 (Particle Swarm Algorithm, PSA) 解决该问题，得到了最优的探测方案，并对该模型的优缺点进行了评价，提出了进一步的修改意见和研究方向。

题目中给出了一个区域的探测点分布图，要求探测车能够在尽量短的时间内完成所有探测点的探测。经过分析，可以得到以下路径规划的三个总体原则：(1) 若总体探测车数目给定，且不考虑探测车成本和使用成本，只要求完成探测的总体时间最短，则希望探测车分布区域尽量相差不大，以使得所有探测车完成探测所用的时间相近，从而在尽量短的时间内完成探测，这是总体原则一；(2) 考虑到水平移动一个节点所用的时间是垂直移动一个节点时间的两倍，相对来说垂直移动的时间较水平移动的时间更容易被其他探测车利用进行探测工作，因此在探测车行进过程中应保证水平移动浪费的时间尽量少，这里浪费的时间是指在这段时间内其他探测车没有执行探测任务，这是总体原则二；(3) 考虑到各个探测车之间的相互干涉，在探测车初始布局和探测次序的选择上应该保证正在执行探测任务的探测车对将要执行探测任务的探测车的影响尽可能地小，从而减少干涉的延迟效应，这是总体原则三。

我们希望能够寻找到一种路径规划方案能够同时满足上述三个原则最优，然而，事实上上述三个原则之间存在冲突，这就使得该问题的解决不能单纯依靠猜想来确定哪一种方案更优。因此，我们根据该问题的数学本质将其转化为一个多背包问题。

首先，考虑探测车之间的干涉作用，建立多台探测车之间的防干涉数学模型，它在后面的路径规划过程中始终作为一种约束，同时提出一种探测车防干涉策略：考虑探测车干涉的距离约束的前提下保证探测车初始布局较为均匀。该策略为探测车初始布局的设计提供一种比较接近最优布局的初始方案，使得后续的优化算法在较小的范围内进行寻优，这将大大减少优化过程的计算量和计算时间。

其次，针对探测车的路径规划问题，考虑垂直移动和水平移动所需时间的差异，提出如下的行走路径及次序优化策略：对于探测车垂直移动的时间利用问题，可以在其行进的过程中进行其他探测车的探测任务，而对于探测车水平移动的时间利用问题，可以利用探测车在垂直方向上移动和执行探测任务的时间跨度来完成其他探测车在水平方向上的移动，将水平移动所浪费的时间降到最低。采用上述的行进策略能够大大节省完成探测所消耗的时间，从而有利于后续的算法快速寻找到最优解。

再次，同时考虑上述两种策略，将探测车的布局问题、探测路径及次序问题转化成多背包问题来求解，即一个典型的 NP 问题。首先建立探测车多背包数学模型，给出探测车初始任务分配的约束条件，同时结合前面的防干涉策略，给出相对较小范围的探测车初始布局方案，采用基于协同进化粒子群算法的方法求解该问题，在路径规划的过程

中考虑前面已经建立的防干涉模型并结合前面给出的探测车行进策略，进行优化，最终得到最优的探测车初始布局、探测路径及扫描次序。

最后，通过 Matlab 编程实现上述优化过程以及探测车执行任务的过程并验证上述方案的正确性，所编写的程序在给定探测车个数以及探测车初始布局、行走路线以及探测次序的前提下，能够得到任意时刻所有探测车的位置、所处的状态及其下一时刻所要执行的命令，从而有利于计算机无线远程协调控制的实现。

关键字：路径规划问题，多背包模型，防干涉模型，协同进化，粒子群算法

一、问题重述

1.1 问题背景

考古探测通常采用探洞、探沟、探方等实验性挖掘，其专业性强、效率低、有破坏性。现有的“探测车”技术已经能够实现非接触获知地下建筑的位置、大小、形状。

探测车的工作原理像海水中的声呐设备。探测车向地下一定范围发射一种特殊的“波”，通过接受并分析反射信号来获知地下建筑物情况或地质结构。用于浅表探测的探测车的成本越来越小，可实现无人操作，由计算机远程控制，未来甚至可以卫星遥感。

由于探测车的作用范围是有限的，当探测区域较大时，需要采用多辆探测车协同进行探测。同时，考虑到探测车的工作原理，在探测过程中若探测车之间的距离太近，会产生干扰，影响探测效果，因此，需要尽量避免或减小探测车之间的影响，以提高工作效率。故研究在给定条件下，如给定探测车数量，如何合理地给出探测车初始布局、行走路线及探测顺序，从而在尽量短的时间内完成探测任务显得十分重要。本文就如何智慧地完成考古探测进行了初步的探索与研究，试图提出一种较为高效的考古探测方案，为考古探测提供一种参考策略。

1.2 问题提出

探测车的探测方式是将地面上的工作区域划分为若干个矩形区域，探测车行走到一个节点停稳后再进行探测，完成本节点探测后行走至下一节点，且只能沿水平或垂直方向到达下一节点。探测车在每个探测点探测的时间相同，探测车的行进是匀速的，且探测和行进两种状态不能同时进行，同时考虑到探测车在执行探测任务时会产生干扰，距离小于一定范围就会导致两次探测时间必须有一定的间隔，这将会降低探测的总体效率。题目中给出了一个区域的探测点分布图以及其他约束条件，利用探测车进行探测，要求建立合理数学模型求解以下三个问题：

一，分别计算 12 台、20 台、64 台、80 台探测车的初始布局，行走路线，扫描次序，完成任务所花的最少时间。

二，需要投入多少台的探测车可以使得完成任务的效率最高？给出其初始布局，行走路线，扫描次序，完成任务所花的最少时间。

三，为了方便计算机进行无线远程协调控制，请进一步将问题 1、问题 2 的结果表达成计算机能够处理的数据组织方式，即：任意时刻探测车在何位置？何时计算机向何探测车发布何指令？

二、问题分析

针对问题一：

我们将其视为给定探测车数目条件下的初始布局、行走路线以及扫描次序确定问题，可将其视为多背包问题进行求解。首先，建立探测车防干涉模型，给出防干涉策略，同时根据垂直和水平移动时间有差异的特点给出充分利用垂直移动和水平移动时间的探测车行进策略；其次，根据探测车布局设计的总体原则，给出探测车初始布局的大致方案，即保证各个探测车负责区域的探测点个数相差不超过某个界限；最后，建立该多背包问题的数学模型，充分考虑上面的两种策略，将水平方向划分区域的长度 L_i 作为待优化参数，其中 $i=1,2,\dots,\frac{m}{2}$ ， m 为探测车个数，采用基于协同进化的粒子群算法进行求解，从而完成给定探测车数目下的初始布局、行走路线以及扫描次序规划任务。如图 1 所示为探测车数目为 12 时的初始布局方案示意图，此时 $i=1,2,\dots$ 。图中， L 表示探测区域水平距离，实心原点表示探测点，空心圆表示探测车。

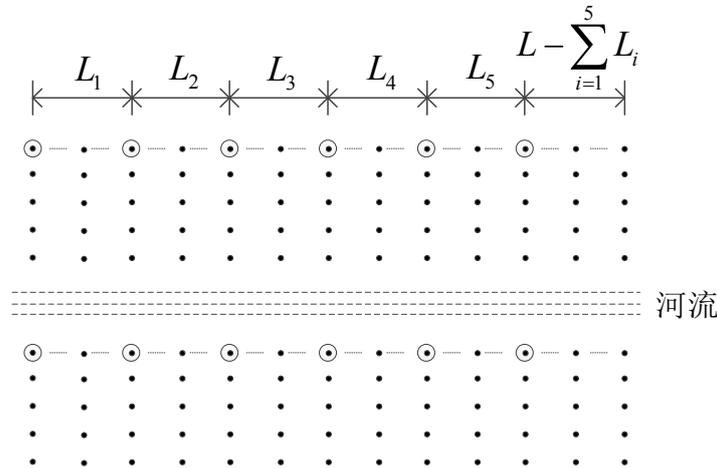


图 1 探测车初始布局方案图

针对问题二：

其实该问题是对问题一的拓展，问题一是在给定探测车数目下的路径规划问题，而问题二则是确定需要多少台探测车能够达到最优，该问题可以视为比问题一更高一维的路径规划问题。把探测车数目也作为优化变量，当给定探测车数目后可以直接采用问题一基于协同进化的粒子群算法进行求解，于是可以直接比较不同探测车数目下的优化结果，即完成探测所需的时间，从而得到多少台探测车可以最高效地完成探测任务。

针对问题三：

该问题是问题一和问题二的实现问题，实际上就是在利用前面的方法确定探测车的初始布局、行走路线以及扫描次序后，采用具体的程序实现探测车的探测任务执行过程，即要求所编写的程序能够表示出在任意时刻 t ，所有的探测车所处的位置及其任务状态以及将要执行的状态（行走或者探测的状态，以及行走或探测的方向）。

三、问题假设

1. 假设不考虑探测车的成本和使用成本，题目中已给出；
2. 假设探测车的个数是偶数，这是为了保证结果的对称性；
3. 假设每个探测车在探测过程中只能沿水平或垂直方向移动，不走重复路线，即不能经过已经探测过的探测点；
4. 假设每台探测车只需要将分配给其负责的区域内的探测点探测完即停止工作，各负其责，效率更高；
5. 假设在给定探测车初始数目和布局的前提下，如果能保证每一时刻都有尽量多的探测车处于扫描状态，则能够保证该探测过程路径最优，时间最短。

四、符号说明

符号	定义	单位
T	探测完所有点需要的总时间	秒(s)
m	探测车总数	个
n	探测点总数	个
$C_i = (x_i, y_i), i = 1, 2, \dots$	探测车坐标位置	千米(km)
$L_i, i = 1, 2, \dots$	各区域水平长度	千米(km)

五、模型的建立与求解

5.1 问题一的分析与解答

由于考探测区域太过巨大，需要探测的探测点数目众多，单台探测车很难在短时间内将所有探测点探测完。为了提高工作效率，实际工程中需要使用两台或两台以上的探测车来完成探测任务。而多台探测车在进行路径规划时，如何协同合作寻求多台探测车系统的最优路径（即所用时间最少）将是该问题的重点研究内容。本文选取了基于协同进化的粒子群算法，通过协调多台探测车的运动行为，使得系统的各台探测车干涉尽量少，得到最优的行走路径，这属于路径规划问题^[1-2]。

5.1.1 多台探测车路径规划分析

在探测车执行大范围探测任务时，一般采用多台探测车同时进行工作，同时要求一部分探测车行进过程中，另一部分探测车最好也处于工作过程，这样可以提高工作效率，保证在所有时刻都有尽量多的探测车在工作，同时还要考虑其路径最短问题。因此，多台探测车路径规划问题描述为：在一个二维平面区域中，探测点数量为 n ，探测车数量为 m ，按照约束条件的要求将探测点分配给每台探测车，在所有探测车之间的相互干涉尽可能少的前提下使得完成探测任务所需要的路径最优、时间最短。

因此，多台探测车路径规划过程中需要考虑的原则与约束条件有以下几个方面：

(1) 探测顺序约束：当探测车数目过多时，不可能令所有的探测车同时工作，这时就需要合理规划所有探测车的探测顺序，即哪些探测车为同一批？同一批探测车何时开始工作？

(2) 探测空间约束：由于要求完成探测所用的总时间最少，因此总希望所有的探测车完成探测工作的时间相差不多，这就要求每台探测车所负责的区域面积相差不多，最好能在所有时刻保持同一批次探测车的构型不变。

(3) 避免干涉的情况：探测车的探测过程随探测车之间的距离不同受到的影响不同，距离越近，探测受到的影响越大，探测延迟时间越长；距离越远，探测受到的影响越小，探测延迟时间越短。故希望探测车行进过程中尽量减少干涉情况的发生，从而减少探测延迟时间，缩减总体探测时间。

下面给出规划多台探测车路径的主要流程：

(1) 初始布局规划及任务分配：为执行探测任务的多台探测车进行初始布局规划，确定其相对位置，并制定各个探测车负责的区域大小，应使所有分配给探测车的探测点数目尽量均匀，以确保总体时间更少。布局规划及任务分配情况影响到后续的路径规划结果与探测车之间的干涉情况。

(2) 多台探测车路径规划：采用合适的算法为每台探测车规划出最优的探测路径，使得各个探测车能够相互协作且干涉情况尽量少。

(3) 仿真验证：使用仿真软件建立探测车探测过程模型^[3]，仿真运行上述初始布局设计、任务分配以及路径规划的结果，验证算法的合理性。

5.1.2 多台探测车布局及探测点分配问题

多台探测车平面路径规划的首要问题是解决如何将众多探测点分配给每台探测车及探测车的布局。每台探测车在各自的负责区域内探测时，有的会受到其他探测车较大

的影响,有的受到其他探测车的影响较小,有的甚至不会受到其他探测车的影响。因此,多台探测车的探测点分配问题还需要考虑到探测车的运动范围、探测车时间的干涉等,其分配原则主要为:

(1) 探测车分配到的探测点尽量集中,不能跨度太大,尽量减少完成其负责区域的时间;

(2) 分配到各台探测车的探测点数量要相差不大;

(3) 探测车之间的距离不宜太小,以免导致与其他探测车发生干涉。

根据以上的分析可以将探测点分配问题转化成多背包问题来求解^[4]。多背包问题是典型的 NP 完全问题,由 Dantzig 在 50 年代末期首次提出。它广泛应用于项目选址、车辆调度、货物装载等。为了简化问题,将探测点和探测车抽象为平面内的点,探测车抽象为背包,探测点抽象为待分配的物体。

探测点多背包数学模型为:在 m 个探测车为 $C=(C_1, C_2, \dots, C_m)$ 和 n 个探测点为 $S=(S_1, S_2, \dots, S_n)$ 的情况下,每个探测点所需的探测时间都是固定的,且探测车移动的速度是固定的,最终使得探测车完成探测任务,且所有探测车完成任务的最终用时最短。采用 $m \times n$ 矩阵表示,由于探测点在分配过程中只能选择分配或者不分配,如果 $X_{ij}=1$ 表示探测点 S_i 由探测车 C_j 探测, $X_{ij}=0$ 则不是。其探测点分配多背包问题模型的目标函数为:

$$f_1 = \min \left\{ \max \left\{ \sum_{i=1}^n V_{ij} X_{ij} \right\}, j=1, 2, \dots, m \right\} \quad (1)$$

其中, V_{ij} 为探测点 S_i 到探测车 C_j 的距离 (即价值), $V_{ij} \geq 0$ 。

根据探测点和探测车之间的关系可知,其约束条件为:

$$s.t. \begin{cases} \sum_{j=1}^m X_{ij} = 1 & \forall i \in \{1, 2, \dots, n\} \\ \left(Y_j - \frac{n}{m} \right) \leq \varepsilon & \forall j \in \{1, 2, \dots, m\} \end{cases} \quad (2)$$

其中, Y_j 表示第 j 个探测车分配得到的探测点数, ε 为给定的探测车分配到的探测点数与理论上每台探测车平均分配到的探测点数的偏差值。第一个约束条件表示在实际探测过程中,每个探测点只能分配给一台探测车进行探测;第二个约束条件表示每个探测车分配得到的探测点数应在一定的偏差值范围内,以确保探测车探测完成时间同步。

多探测车探测点任务分配属于 0-1 多背包问题。传统的多背包问题解决办法主要有递归法和贪心法等,但在处理复杂问题时会占用很多空间,很难找到最优解。我们采用基于协同进化的粒子群算法解决多背包问题,其可用于处理复杂的问题,具有鲁棒性好、收敛性快等特点。

5.1.3 多台探测车防干涉问题

探测车之间如何实现协同合作防止干涉是多台探测车系统中的一个重要问题。多台探测车系统并不是多个单台探测车的简单相加,而是一个多目标的优化问题。多台探测车系统中各台探测车之间一方面是协同完成复杂任务,另一方面需解决在完成各自任务的过程中所可能发生的干涉。在本文探讨的多台探测车完成大区域探测任务系统中,每台探测车都需要实现以下两个目标:完成指定的探测任务;防止与其它探测车发生碰撞。

多台探测车在执行工作过程中有可能会发生干涉,这时需要采取有效措施来解决,

根据该问题的实际约束条件，我们采用下面的干涉应对策略：每隔一段时间更新探测车的坐标位置，并计算两台探测车之间的空间距离。设定探测车优先级，不同优先级的探测车在干涉区内相遇时，优先级较高的探测车按原规划路径探测；优先级较低的探测车把优先级较高的探测车看作障碍物，并在路径规划过程中执行避障措施。因此，如果两台探测车的距离大于安全距离，则两台探测车规划的路径安全。否则优先级别低的探测车停止本次路径的规划，重新寻找探测路径。

下面建立探测车的防干涉模型：

在探测区域内，将探测车视为在平面内寻找探测点的动态点。设 m 个探测车为 $C=(C_1, C_2, \dots, \dots)$ ， n 个探测点为 $S=(S_1, S_2, \dots, \dots)$ ，探测车的移动速度为 $v = \frac{12}{50} m/s$ 。在平面中，探测车 C_i 在任意时刻的位置坐标为 (x_i, y_i) ，其中 $i=1, 2, \dots$ 。当探测车刚好在探测点上时，其坐标为探测点坐标 (x_j, y_j) ，其中 $j=1, 2, \dots$ ；当探测车在两个探测点之间时，则探测车坐标为 $(x_{\rho i}, y_{\rho i})$ ，其计算方法如下。

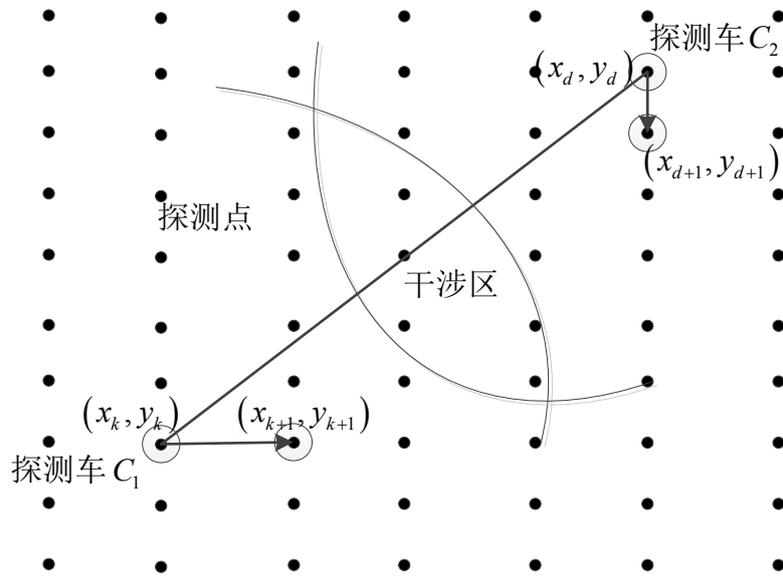


图 2 防干涉数学模型示意图

如图 2 所示，设探测车 C_1 从初始探测点 (x_k, y_k) 运动到最终探测点 (x_{k+1}, y_{k+1}) ，其中 $k=1, 2, \dots$ ；探测车 C_2 从初始探测点 (x_d, y_d) 运动到最终探测点 (x_{d+1}, y_{d+1}) ，其中 $d=1, 2, \dots$ 。如果探测车 C_1 和 C_2 运动时间 t 后，其行走路径为 vt 。因此，探测车 C_1 从初始探测点出发经过长度与初始探测点到最终探测点长度的比例关系为：

$$\lambda_1 = \frac{vt}{\sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2}} \quad (3)$$

同理探测车 C_2 的比例关系为

$$\lambda_2 = \frac{vt}{\sqrt{(x_{d+1} - x_d)^2 + (y_{d+1} - y_d)^2}} \quad (4)$$

则经过时间 t 后，探测车 C_1 的坐标 $(x_{\rho 1}, y_{\rho 1})$ 为

$$(x_k + \lambda_1(x_{k+1} - x_k), y_k + \lambda_1(y_{k+1} - y_k)) \quad (5)$$

探测车 C_2 的坐标 (x_{ρ_2}, y_{ρ_2}) 为

$$(x_d + \lambda_2(x_{d+1} - x_d), y_d + \lambda_2(y_{d+1} - y_d)) \quad (6)$$

探测车在平面内进行探测时，定量衡量发生干涉的影响大小，根据题目设定两个安全距离分别为 $D_1=4.8\text{km}$ 、 $D_2=9.0\text{km}$ 。结合上述表达式可得探测车 C_1 和 C_2 两点的距离为

$$d = \sqrt{(x_{\rho_2} - x_{\rho_1})^2 + (y_{\rho_2} - y_{\rho_1})^2} \quad (7)$$

如果 $d > D_2$ ，则说明探测车 C_1 和 C_2 不会发生干涉，否则认为会发生干涉，且有：当 $D_1 < d \leq D_2$ 时，则探测车 C_1 和 C_2 干涉时间为 6s，即两探测车工作时间间隔至少为 6s；当 $d \leq D_1$ 时，则探测车 C_1 和 C_2 干涉时间为 12s，即两探测车工作时间间隔至少为 12s。

5.1.4 协同进化粒子群算法

协同进化算法通过构造两个或多个种群，建立它们之间的竞争或合作关系，多个种群通过相互作用来提高各自适应性能，适应复杂系统的动态演化环境，以达到种群优化的目的^[5,6]。与协同进化算法相比，遗传算法只采用基于个体自身适应度的演化模式，没有考虑其演化的环境和个体之间的复杂联系对个体演化的影响，它在应用中表现出了易出现未成熟收敛并且收敛的速度较慢等缺陷，而协同进化算法能够利用少数起进化导向作用的个体，减少不必要的计算量，使收敛的速度加快。

根据协同进化算法采用的模型不同，其可以分为以下几类^[7]：

(a) 基于竞争物种的协同进化算法：该思想是由几个相对较小的群体组成一个大群体，几个小群体之间即存在竞争的关系，也存在协同合作的关系。通过个体之间迁移来实现群体之间的信息交流。

(b) 基于捕食者与猎物系统的协同进化算法：猎物为了避免被天敌猎食，需要随时发展进化自己，由此增强自身逃跑能力。而捕食者为了能捕捉到猎物，需要通过进化来使得自己的攻击力更强。这种协同进化算法由此会导致捕食者和猎物都在变强，而使得问题越来越复杂。

(c) 基于共生机制的协同进化算法：该算法的思想是把一个大且复杂的问题分解成若干相对简单的子问题来解决，每个子问题就是一个子种群，通过进化算法来规划一个子种群，并将子种群的规划结果与其他种群进行协调，从而得到整个大问题的解。

本文采用基于共生机制的协同进化算法来解决问题。共生的协同算法就是把一个问题分成几个子问题，对问题的求解相应转化成对应几个子问题的求解，每个子问题都分别建立一个种群并采用进化算法进行求解。由于每个子问题之间存在相互的关联，不同种群的个体只是全局解的一部分，需要参考其它种群并通过适应度函数计算。最后把进化算法收敛得到的结果进行组合，得到一个对应的完整解。共生问题的分解示意图如图 3 所示。

协同进化算法的流程图如图 4 所示。步骤如下：

(a) 初始化种群：种群随机产生 N 个可能解并对每个子种群进行编码；

(b) 适应度评价：在每个子种群内有 N 个适应度值并对其子种群的解进行评价。评价的依据由上一代其他子种群中随机选取合作者与本种群来综合评定；

(c) 检查每个子群体是否满足条件，如果满足要求则输出该群体结果并且停止进化，否则，则转到步骤 (d)；

(d) 进化：对每个子问题进行进化算子操作，并进行适应度的评价。不同进化算法有不同的进化步骤，如遗传算法的变异和交叉、粒子群算法的个体最优位置和群体全局最优位置等；

(e) 根据进化结果更新各种群，返回步骤 (b)，并进行下一次迭代。

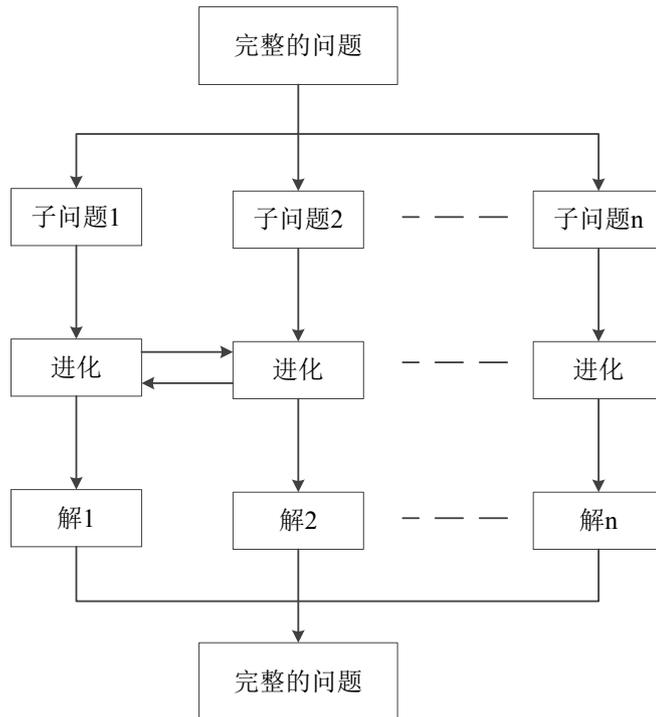


图 3 共生问题的分解示意图

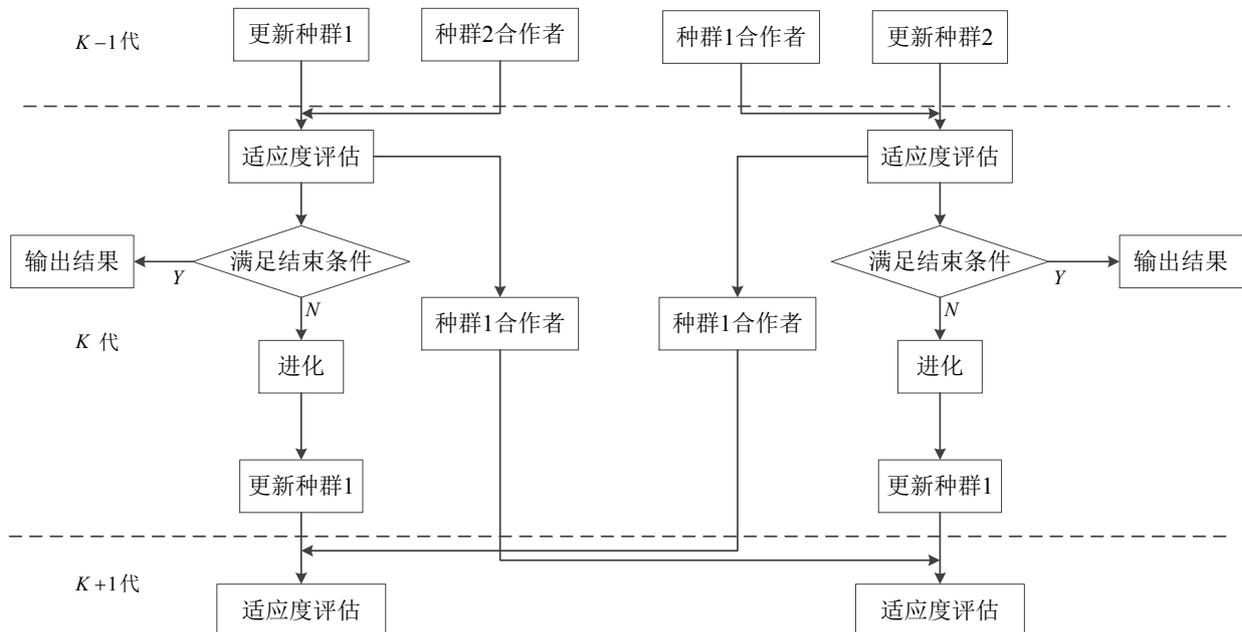


图 4 协同进化算法流程图

5.1.5 多台探测车协同工作流程

如图 1 所示，首先将探测车分布在上下区域的边缘上，忽略探测区域宽度的影响，按照一行排列，上下区域的探测车对齐处理，同一区域各探测车之间的距离未知，按照

探测车的数量 n ，设为距离变量 $L_1, L_2, L_3, L_4, L_5, \dots, L_{n/2}$ ，其值待优化。

探测车的扫描运行方式与待优化变量探测车距离 L 有关，探测车的数量设为偶数，探测车按照分组扫描运行（没有分组的探测车不运行），每组各探测车之间的探测间距要大于 $9km$ ，分组方式扫描的流程如图 5 所示：

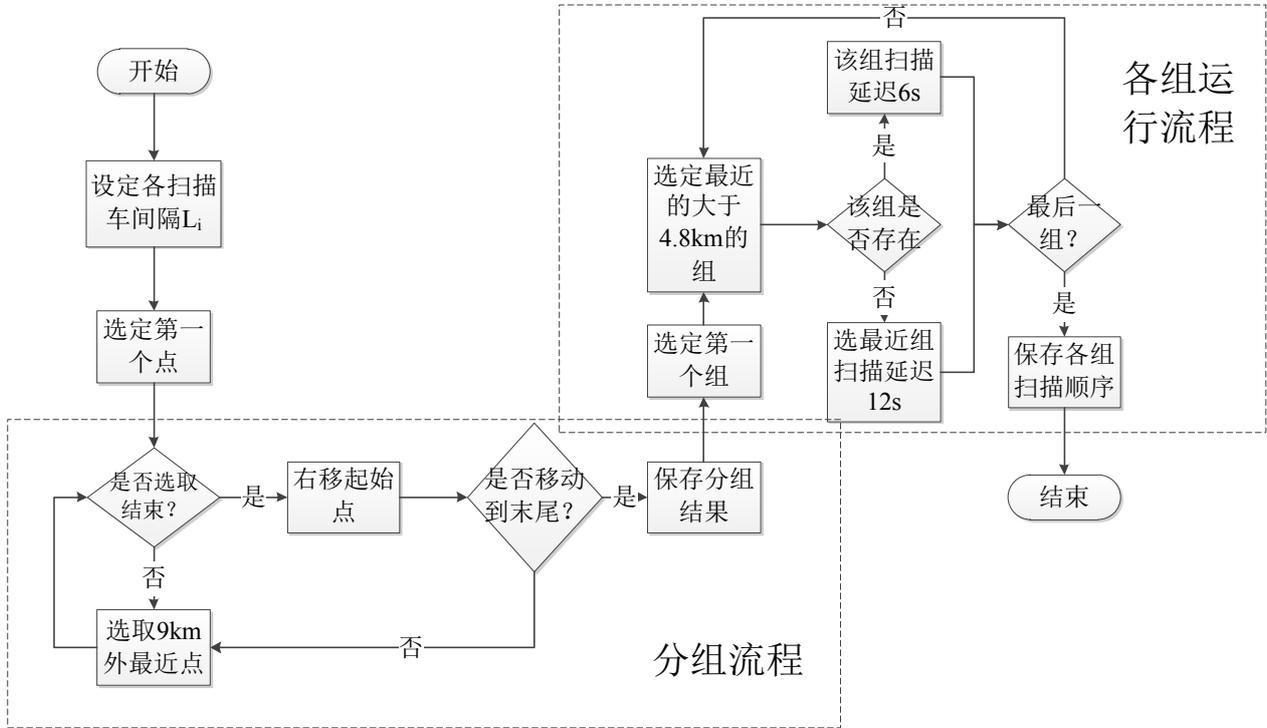


图 5.优化分组运行示意图

每组探测车的选取方法如下：如图 6 所示，从最左侧 M 点开始，以该点为中心， $9km$ 为半径，做一圆形区域①，则该区域之外的探测车满足与该点间距大于 $9km$ 的条件，选取①区域外且离该探测车最近的探测车，如图中 N 点，再以 B 点为圆心做一半径 $9km$ 的圆形区域②，在该区域外选取离 B 最近的点，且该点不在区域①内，如图中 P 点，一直到最右侧搜索结束。这样就组成一组各扫描车之间间距大于 $9km$ 的组，将该组命名为 A ，同样从左侧第二个点继续向右搜索可以形成另一组各扫描车之间间距大于 $9km$ 的组，命名为 B ，同理，根据扫描车的间隔情况可以形成 C, D, \dots 。

5.1.5 基于协同进化粒子群算法的多台探测车路径规划

多台探测车在路径规划时，需要注意两台探测车在干涉区发生干扰，这就涉及多台探测车之间的相互协调和干涉问题。采用基于协同进化的粒子群算法，可以很好地解决多台探测车之间的干涉问题。共生的协同进化模型就是把全局问题分解成若干个子问题来解决，如图 8 所示。在多台探测车的路径规划过程中，全局问题就是多台探测车的路径规划问题。每个单台探测车的规划目标是路径最短，同时规划的路径对于多台探测车来说需要满足相互之间尽可能少的发生干涉，相互协调的要求。根据多台探测车路径规划的要求，将多台探测车的路径规划问题分解成单台探测车的路径规划问题，每台探测车分配到的探测点采用标准粒子群算法求解得出最优规划路径。子问题之间的关联性主要表现在各台探测车之间的干涉问题。在协同进化模型下，多台探测车之间的协调就是通过多个单台探测车之间提供适当的合作者与适应度的计算来实现的。最后根据多个单台探测车的解收敛得到全部子问题的解，将全部子问题的解组合后便得全局解，由此得到多台探测车的路径规划结果^[8]。

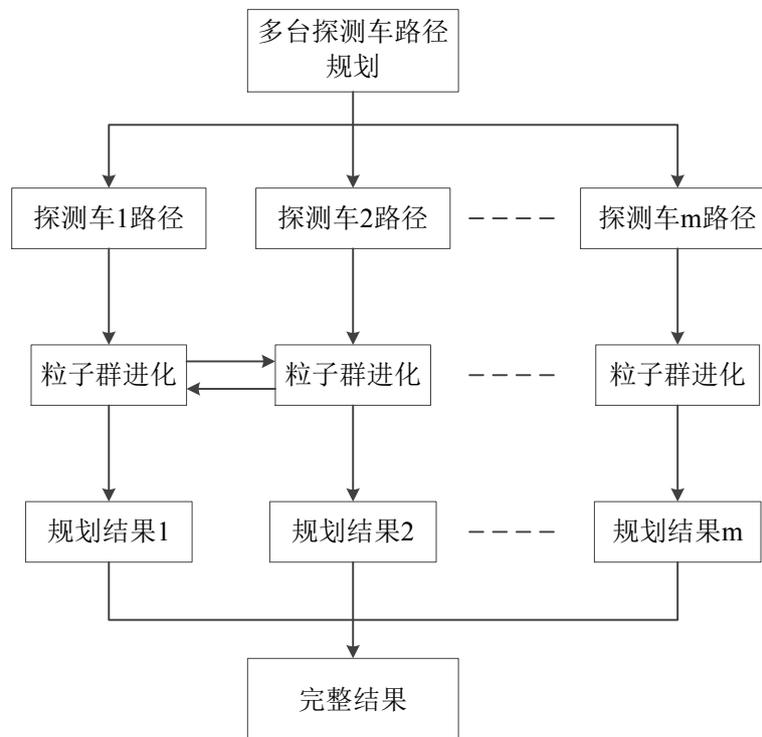


图 8 多台探测车路径规划问题分解示意图

下面给出采用协同进化粒子群算法求解多台探测车路径规划问题的流程，如图 9 所示。其步骤如下：

(a) 初始化：初始化探测车的种群（群体规模为 M ），包括初始化所有粒子的速度、位置和交换序，并对每台探测车的探测点进行编码；

(b) 评价和选择：不同探测车路径之间的解由全局适应度函数评估，判断是否两台探测车之间在运动过程发生干涉和探测点分配是否合理，如果满足条件则输出结果；

(c) 进化终止条件：满足一定的进化次数或者满足局部适应度值时，则说明找到了合适的路径解；否则各台探测车路径重新进行粒子群算法优化；

(d) 根据进化结果更新各种群，返回 (b) 步，并进行下一次迭代。

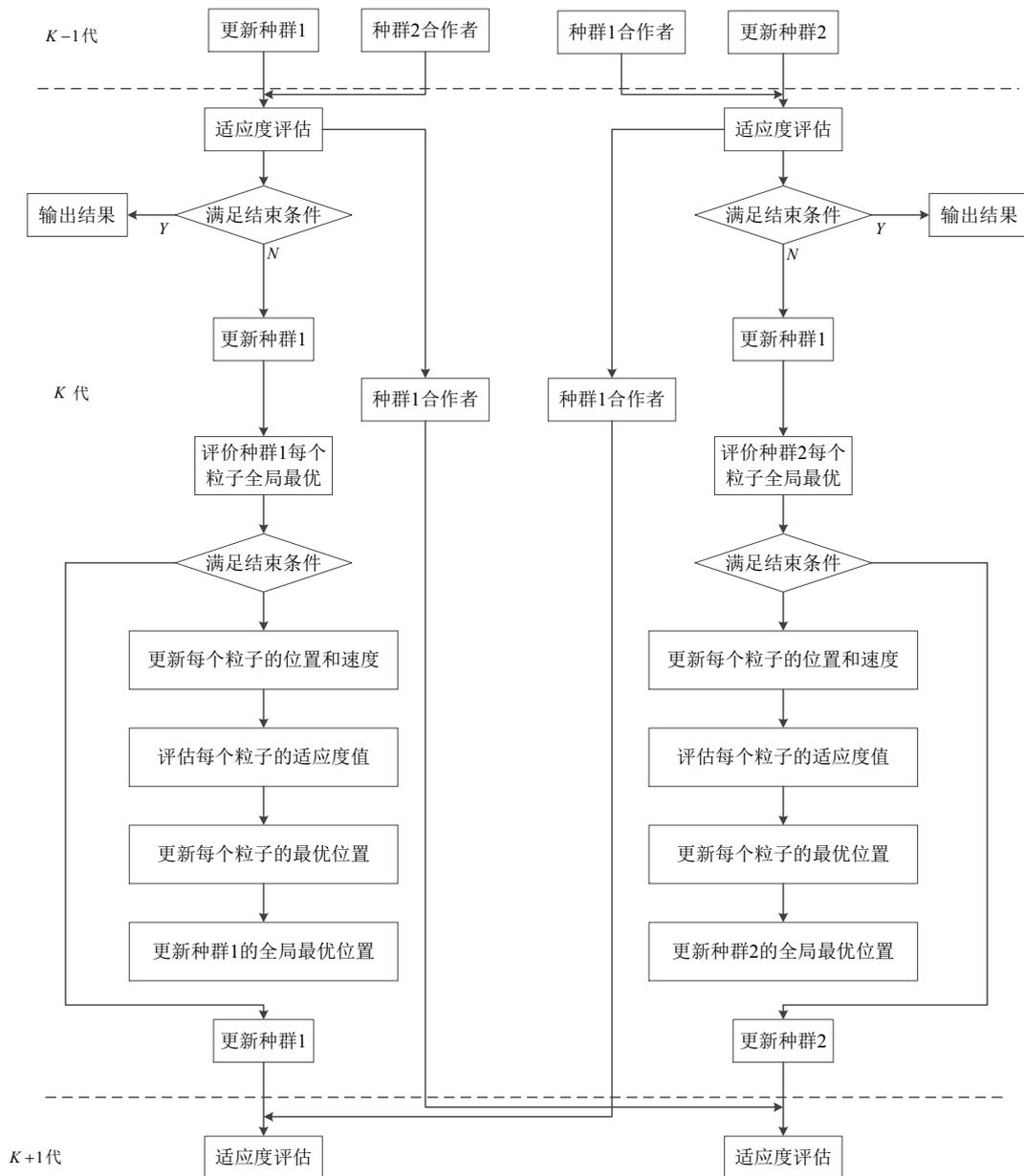


图 9 基于协同进化的粒子群算法流程图

5.2 问题二的分析与解答

问题2是找到完成任务效率最高时的探测车数目,即保证完成任务所花的时间最短,同时没有多余的探测车,指这些探测车不工作或者工作时间很短。由前面的分析可知,问题2是问题1的进一步拓展,只要解决了问题1中给定探测车数目下的路径规划模型建立及求解方法,问题2就可以转化为一个单变量的极值寻优问题。首先,可以将已经解决的问题1简化为这样的一个模型:给出探测车的数目 m 后,采用基于协同进化的粒子群算法就能够找到一种用时最短的探测方案,包括探测车的初始布局、行走路线以及扫描次序。可将其简化为这样的函数形式

$$\begin{cases} Layout_{best} = f_1(m) \\ Route_{best} = f_2(m) \\ Order_{best} = f_3(m) \\ T_{min} = f(Layout_{best}, Route_{best}, Order_{best}) \end{cases} \Rightarrow T_{min} = f(m) \quad (9)$$

式中, $Layout_{best}$ 、 $Route_{best}$ 、 $Order_{best}$ 分别表示最优的初始布局、行走路线以及扫描次序; T_{min} 表示探测车数目为 m 时的最短探测时间。

于是, 根据上面的描述, 问题 2 的本质就是求解函数 $T_{min} = f(m)$ 的极值。由于该函数不是连续的, 不能直接利用微分得到极值。可以采用遍历法求得其极值。注意这里的极值其实不是严格意义上的极值, 而是最小值。考虑到探测车数目越多, 完成任务的时间就越少或者保持不变, 但至少不会增加, 因此最终得到的完成任务的最短时间 T_{min} 随探测车数目 m 的变化曲线趋势为先递减, 后不变, 即存在某个饱和探测车数目, 使得不管以后增加多少探测车, 总探测时间都保持不变。

遍历法求极值: 探测车的数目从 20 开始变化, 依次增加 2 台探测车, 直到 64 (这样取值的目的是根据问题 1 的计算结果可知最优的探测车数目一定在 20-64 之间), 然后计算各自的最短时间, 取最短时间对应的最小的探测车数目即为完成任务效率最高的探测车数目。此方法操作简单, 易于实现。

5.3 问题三的分析与解答

问题 3 的本质是利用计算机语言将探测车的探测过程清晰地表达出来, 即给定了探测车的数目和初始布局, 按照已经优化得到的行走路线和扫描次序进行探测, 易得到任意时刻探测车所处的位置、状态及其将要执行的动作 (行走或者探测)。采用 Matlab 软件来完成此过程的程序化处理。下面就针对实现探测过程的关键问题进行阐述。

(1) 探测车状态更新的周期选择问题

由于探测车完成一次扫描的时间为 12s, 且垂直移动一个节点的时间为 12s, 水平移动一个节点的时间为 24s, 两探测车干涉的作用时间随距离的不同有 6s 和 12s, 因此在探测车执行探测任务中的状态有 4 种: 一是处于探测的前 6s, 二是处于探测的后 6s, 三是处于行走的前半程 (6s 或 12s), 四是处于行走的后半程 (6s 或 12s)。在探测车行进过程中, 每隔 6s, 其状态改变一次或者不改变。因此, 选择 6s 作为探测车状态更新的周期, 这样就能得知任意时刻探测车所处的状态。

(2) 整周期时的探测车位置、状态及将要执行的动作的确定问题

当所给时刻为整周期时, 可以按照下面的思路确定探测车的位置、状态及其将要执行的动作。由于已经确定探测车的行走路径和扫描次序, 其存在对称性和递进性, 因此可以利用这种规律快速计算出所有探测车的位置及其所处的状态。这种规律性首先体现在同一批次每台探测车垂直移动的时间固定, 其行走构型不变, 且不同批次的探测车运行的时间存在一种递进的规律, 依次递减 6s, 如此循环往复, 根据第一批次的探测车状态可以推算以后所有批次探测车的状态及位置。故根据所给的时间 t , 可以推算得到第一批次的探测车所处的位置及其状态, 从而根据上述的规律, 得到同一时刻其他批次探测车所处的位置及状态, 由此得到当前时刻所有探测车的位置、状态及将要执行的动作。

(3) 非整周期时的探测车位置、状态及将要执行的动作的确定问题

当所给时刻为非整周期时, 首先确定紧邻当前时刻的前一个整周期时刻的位置和状态, 然后根据多余的时间以及探测车运行的规律来推算得到当前时刻的位置和状态以及将要执行的动作。

六、结果分析

6.1 探测车台数为 12 时的优化结果与分析

12 个车最优的扫描初始布局为：6 辆车均匀排布在河流上下部分的边线上，如图 10 所示，为了调度运行方便，探测车呈一条直线排布。

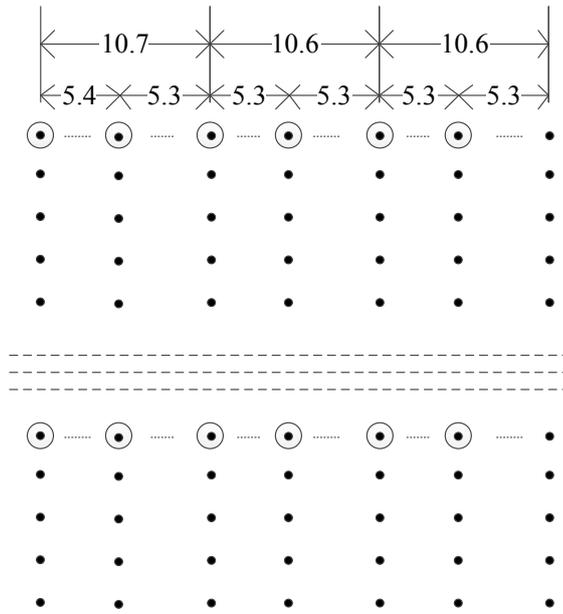


图 10 探测车台数为 12 时的初始布局

探测区域总长度为 31.9km，除最左侧车辆负责 5.4km 宽度外，其余负责 5.3km 宽度，相应的列数分别为 54,53，即单个探测车的最大扫描区域对应 54 列的扫描区域。车辆的分组运行情况如图 11 所示。

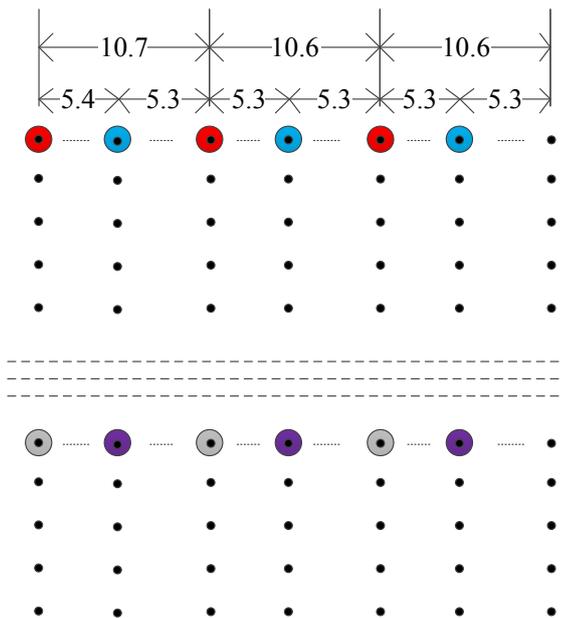


图 11 探测车台数为 12 时的分组情况

12 辆探测车共分为 4 组，每组 3 辆车，按照上图颜色分类，红色代表 A 组，蓝色代表 B 组，灰色代表 C 组，紫色代表 D 组。由于每组的 3 个探测车互相之间至少间隔 10.6km，大于互不影响探测运行的最小距离 9km，所以每组的 3 辆车可以同时执行扫描任务，各组探测车之间的距离不会大于河流的宽度 4.85km，同时小于 9km，所以组与组之间的扫描时间间隔为 6s，按照 A→B→C→D→A...的扫描顺序依次进行扫描，同时为保证尽量减少耗费时间长的横向运动，每组里的每辆扫描车都采用如图 12 所示的迂回运动扫描方式，先竖向扫描一列再横向移动一格扫描再竖向扫描一列直到扫描结束。

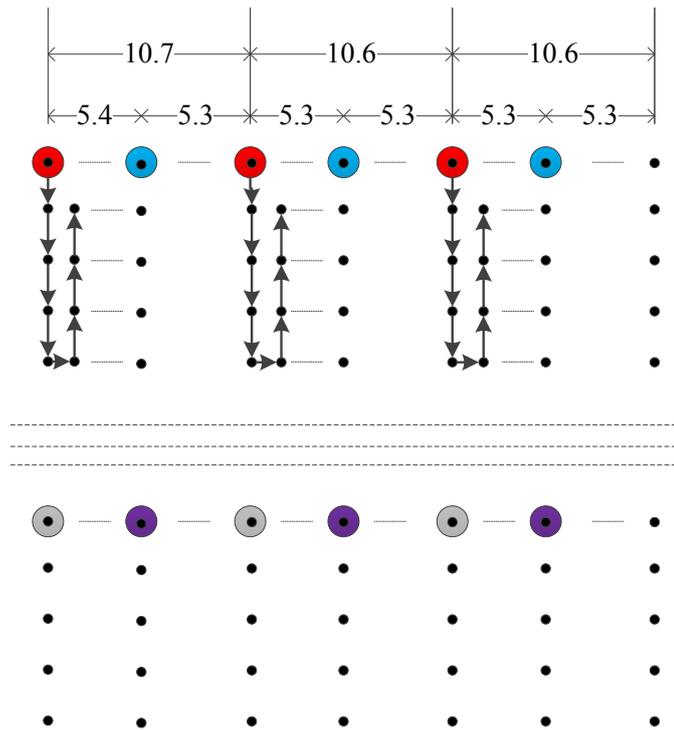


图 12 探测车台数为 12 时的运动方式

A 组探测车从起点开始扫描到 A 组探测车竖直方向行驶一个探测点并扫描结束的各组探测车状态示意图如图 13 所示，其间各组的运动扫描过程如表 1 所示。

表 1 一个竖向扫描间隔过程

运行时间 (s)	A 组状态	B 组状态	C 组状态	D 组状态
12	扫描结束	已扫描 6s	开始扫描	等待
24	开始扫描	已行驶 6s	扫描结束	已扫描 6s
36	扫描结束	已扫描 6s	开始扫描	已行驶 6s

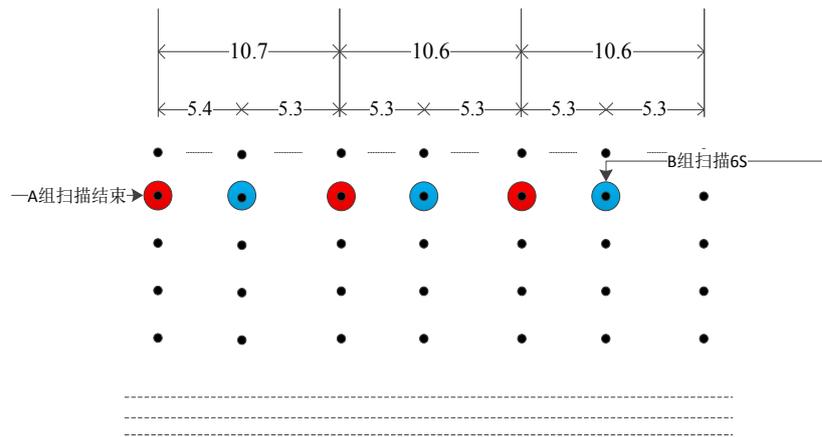


图 13 竖向移动一个扫描点并扫描完成

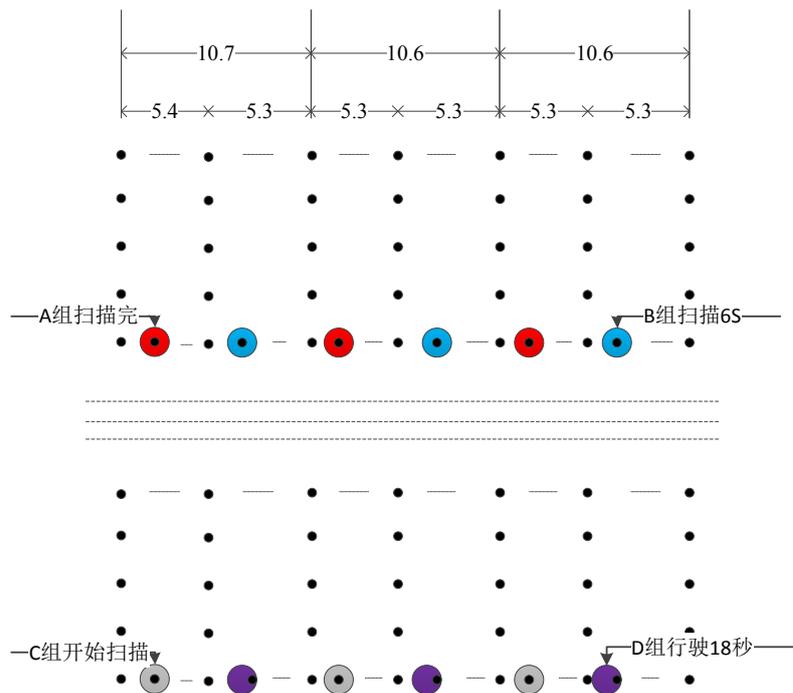
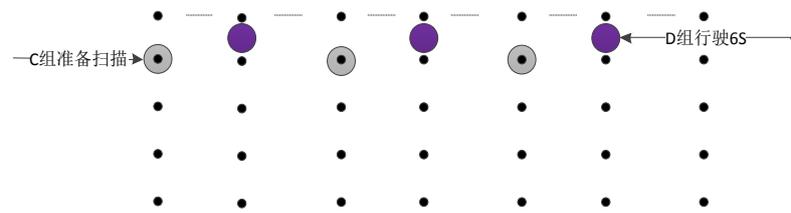


图 14 横向移动一个扫描点并扫描完成

水平横向行走一个探测点扫描结束的示意图如图 14 所示，每一组进行一个竖向间隔的扫描行驶过程需要消耗 24s，当竖向的最后一个点扫描结束后，将进行横向行驶并扫描，横向行驶消耗时间为 24s，长于竖向行驶时间，从 A 组竖向最后一个点扫描结束到横向行驶完成并扫描结束的运动扫描过程如表 2 所示。

表 2 一个竖向扫描间隔过程

运行时间 (s)	A 组状态	B 组状态	C 组状态	D 组状态
12	已行驶 12s	已行驶 6s	扫描结束	已扫描 6s
24	开始扫描	已行驶 18s	已行驶 12s	已行驶 6s
36	扫描结束	已扫描 6s	开始扫描	已行驶 18s

由以上分析可以发现，横向移动一个扫描点并扫描完成需要消耗 36s。

通过以上分析也可以给出 12 台扫描车的最优结果的解释，即 12 辆扫描车的均匀分布将保证所有扫描车的最大扫描区域最小化，每组 3 辆车，分成 4 组，依次间隔 6s 工作的工作方式也保证了每个区域的扫描车始终保持在扫描行驶的交替运行状态而没有等待时间，所以这是对于 12 辆扫描车的最佳分布和运行方式。其总的探测时间为 7122s。

6.2 探测车台数为 20 时的优化结果与分析

采用上述方法优化得到的 20 台探测车的初始布局如图 15 所示，最短耗时为 5904s。

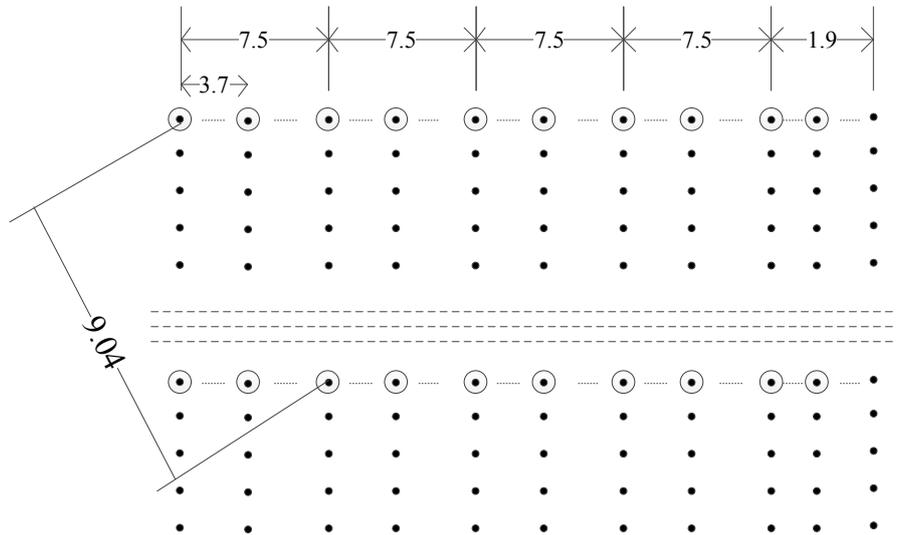


图 15 给定 20 台探测车的最优初始布局

将探测车分为 A,B,C 和 D 组，位于同一组的探测车的运行状态相同，从 A 组到 D 组的起始工作时间恰好相差 6s，各组探测车的布局情况以及运行情况如图 16 所示。各组探测车的行走时序如表 3 所示。

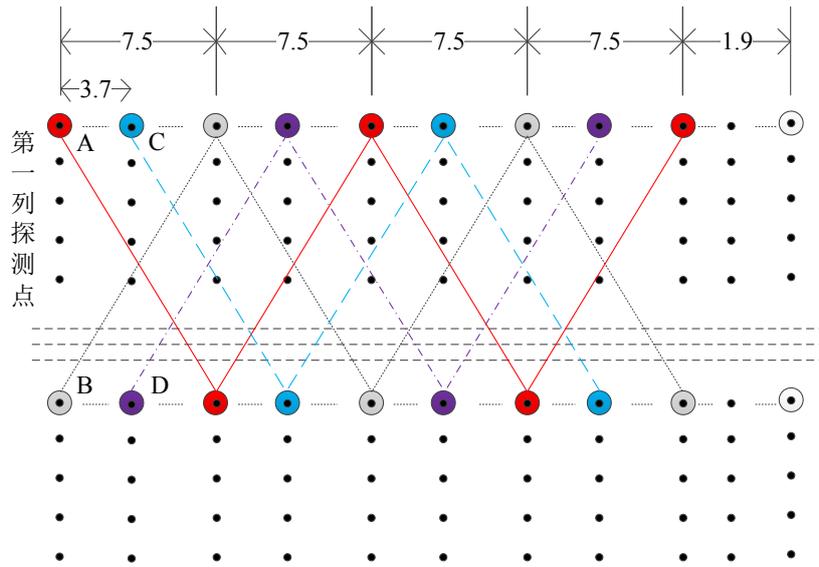


图 16 探测车运行情况

表 3 各组探测车的行走时序

组	探测时间/s	行走时间/s	探测时间/s	行走时间/s	
A	0-12	12-24	30-42	42-54	...
B	6-18	18-30	36-48	48-60	...
C	18-30	30-42	48-60	60-72	...
D	24-36	36-48	54-66	66-78	...

由于组 C 与 B,A 的距离较近, 因此 C 与 B,A 的扫描时间间隔必须大于等于 12; 同理, D 组的与 B,A 的扫描时间间隔也大于等于 12s, 因此 C 组的启动时间相对于 B 组滞后 12s, D 组相对于 C 组滞后 6s。在 A 组沿河流的方向行走至下一个探测点时需等候 6s 再开始探测, 从而防止与距离较近的 C 组的扫描时间相冲突, 此后 B,C,D 组的行走时序重复第一次的行走时序, 此过程不断重复直至完成各列最后一个探测点的扫描过程。在 A 组扫描完每一列的最后一个探测点后向右移动至下一个探测点, 之后再次重复进行扫描、行走、等待, 直至扫描完 3.7km 范围内所有探测点。B,C,D 组的探测行为与 A 组相似, 不同点在于当 A 和 B 组完成 3.7km 范围内所有探测点的探测任务后, 此时还剩余最后一列仍未探测。在探测最后一列探测点时 C 与 D 组的探测过程交替进行, 此时 C 与 D 组探测、行走过程中无需等待时间。

当 A 组刚到达第二列的倒数第二个探测点时, 各个探测车的位置分布情况如图 17 所示。

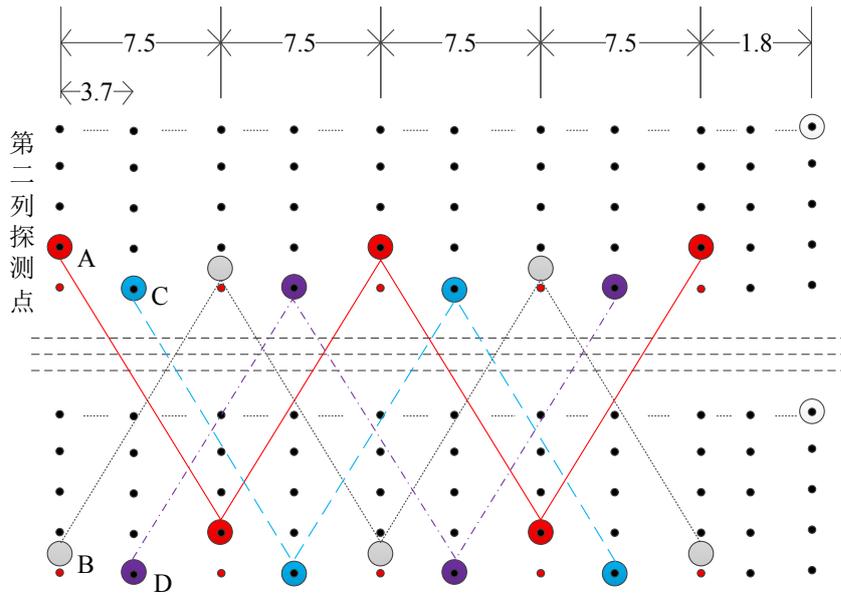


图 17 探测车位置运行过程中的位置分布一

当 A、B 探测车完成 3.7km 范围内所有探测点的探测任务后，此时仅剩余 C、D 完成最后一列探测点的探测过程。当 C 组探测到最后一列探测点 6s 时 D 组探测车刚到达最后一列，此时探测车的分布情况如图 18 所示。

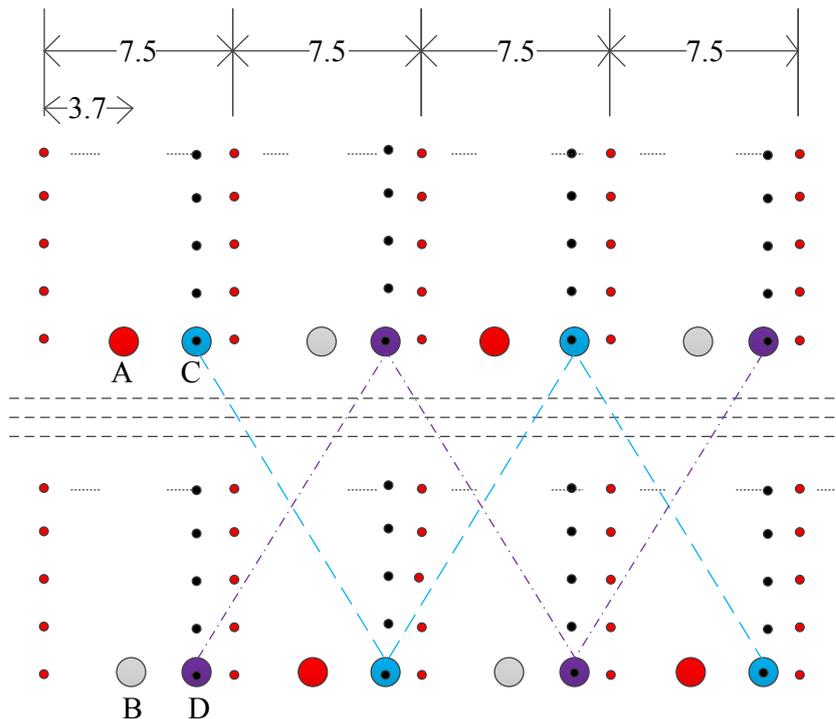


图 18 探测车位置运行过程中的位置分布二

因此在这种模式下的探测总耗时为：

$$18 + 37 \times (108 + 6 \times 4 + 24) + 108 + 6 = 5904s \quad (11)$$

18 代表 C 组相对于 A 组的工作时间滞后了 18s， $108 + 6 \times 4 + 24$ 代表每列再加横向行走时消耗的时间，此时共有 37 列。此外，在完成 37 列探测后仅剩余 C、D 进行探测，此时两者进行探测消耗的总时间为 108s，D 完成探测时相对于 C 滞后 6s，因此得到在

20 辆车工作条件下总需求时间为 5904s，该计算结果与优化算法得到的计算结果保持一致。

6.3 探测车台数为 64 时的优化结果与分析

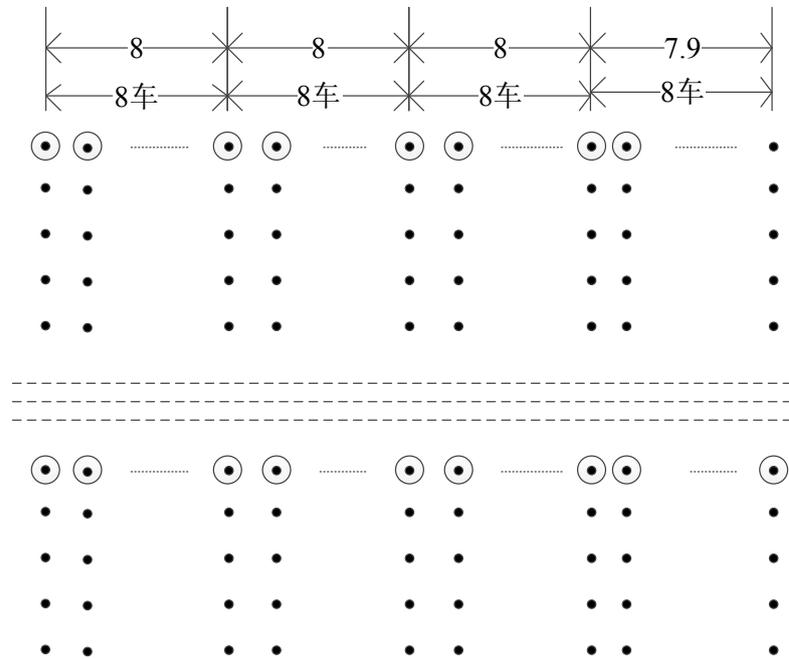


图 19 探测车数目为 64 时的初始布局

如图 19 所示为探测车数目为 64 时的初始布局：在 0-31km 范围内每隔 1km 分布一辆探测车，最后一辆车距右边界的距离为 0.9km，河流左边的分布情况与河流右边的分布情况相同。同样将探测车分为 A₁,B₁,C₁,D₁ 共 4 组，各组探测车的分布情况如图 20 所示。

表 4 A₁,B₁,C₁,D₁ 组探测车的行走时序

组	探测时间/s	行走时间/s	探测时间/s	行走时间/s	
A ₁	0-12	12-24	24-36	36-48	...
B ₁	6-18	18-30	30-42	42-54	...
C ₁	12-24	24-36	36-48	48-60	...
D ₁	18-30	30-42	42-54	54-66	...

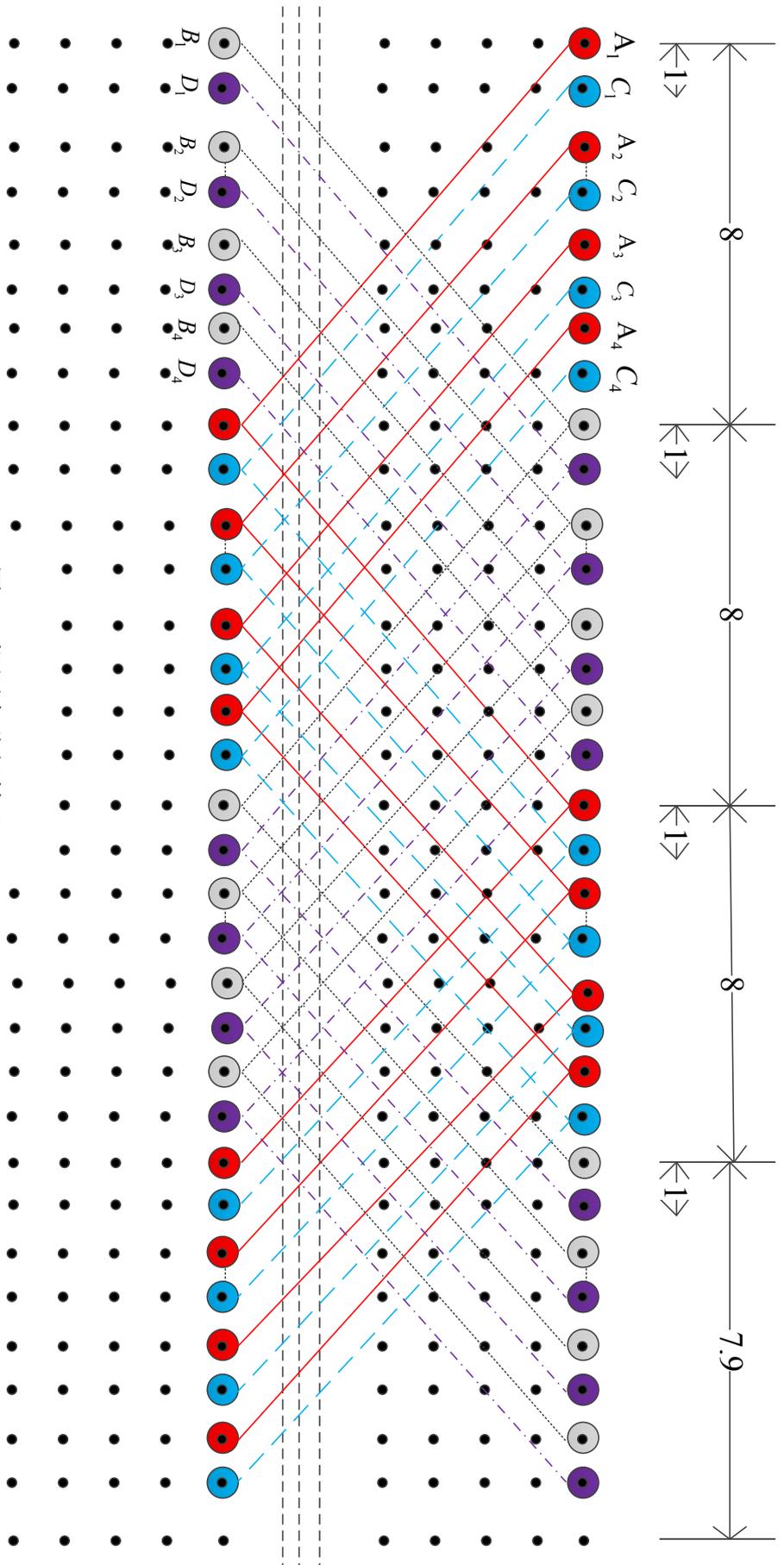


图 17 探测车分组情况

起始时刻 A_1 、 B_1 组共有 5 辆探测车同时工作， C_1 、 D_1 组共有 4 辆探测车同时工作，从 A_1 组到 D_1 组探测车的启动时间依此滞后 6s，即 A_1 开始扫描 6s 后 B_1 开始工作， B_1 扫描 6s 后 C_1 开始工作， C_1 扫描 6s 后 D_1 开始工作，此时 C_1 在扫描时 A_1 处于行走状态， D_1 在扫描时 B_1 处于行走状态。当 A_1 完成第一列最后一个探测点的扫描过程后向右行走至下一列端点位置的探测点上开始等候。当 A_1 向右行走 12s 时距离 A_2 开始工作，其工作过程与 A_1 组探测车的行走过程一致；同理，当 B_1 组探测车完成最后一个探测点的探测后 B_1 组探测车向右行走至紧邻 B_1 组右侧探测点后开始等待。在 B_1 组向右行走 12s 时 B_2 组开始扫描，此过程与 B_1 组探测车的探测过程相同。 C_1 组与 D_1 组的探测过程与 A_1 、 B_1 组相同。

当 A_2 探测车扫描了该列最后一个探测点后向右行走 12s 时 A_1 组探测车开始扫描， A_2 行走至下一列探测点时开始等待，此时 A_1 开始扫描。当 A_1 、 B_1 、 C_1 、 D_1 组与 A_2 、 B_2 、 C_2 、 D_2 组依此扫描 1km 范围内的探测点后， A_3 、 B_3 、 C_3 、 D_3 组与 A_4 、 B_4 、 C_4 、 D_4 组的探测过程与 A_1 、 B_1 、 C_1 、 D_1 组与 A_2 、 B_2 、 C_2 、 D_2 组相同。当 D_1 、 D_2 、 D_3 、 D_4 依此完成 5 次扫描 12s-行走 12s 过程后所有探测车即可完成所有探测点探测过程。优化程序得到当初始布局为 64 辆探测车时完成所有探测点的探测过程共需 4806s。

6.4 探测车台数为 80 时的优化结果与分析

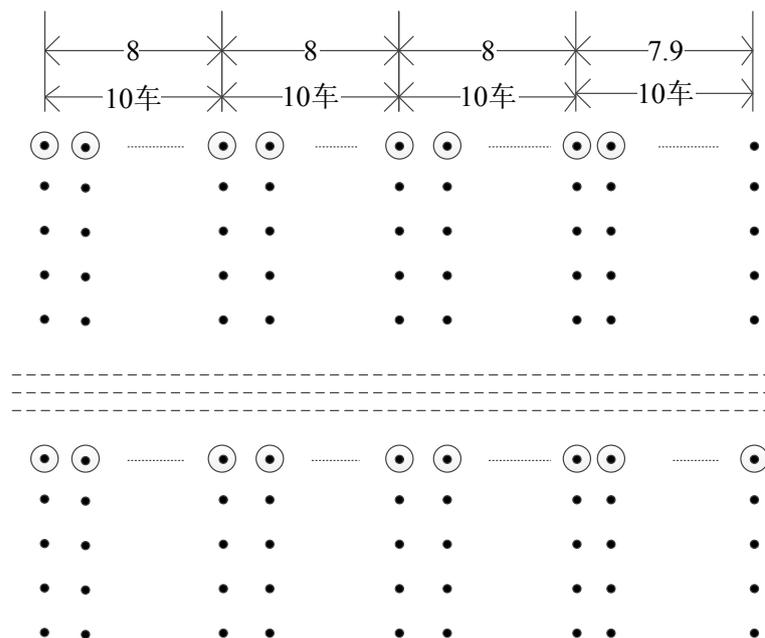


图 21 探测车数量为 80 时的初始布局

如图 21 所示为探测车数量为 80 时的初始布局：在 0-31km 范围内每隔 0.8km 分布一辆探测车，最后一辆车距右边界的距离为 0.7km，河流左边的分布情况与河流右边的分布情况相同。同样将探测车分为 A、B、C、D 共 4 组，各组探测车的分布情况如图 22 所示，其中 a 为 0.8km，b 为 0.7km。

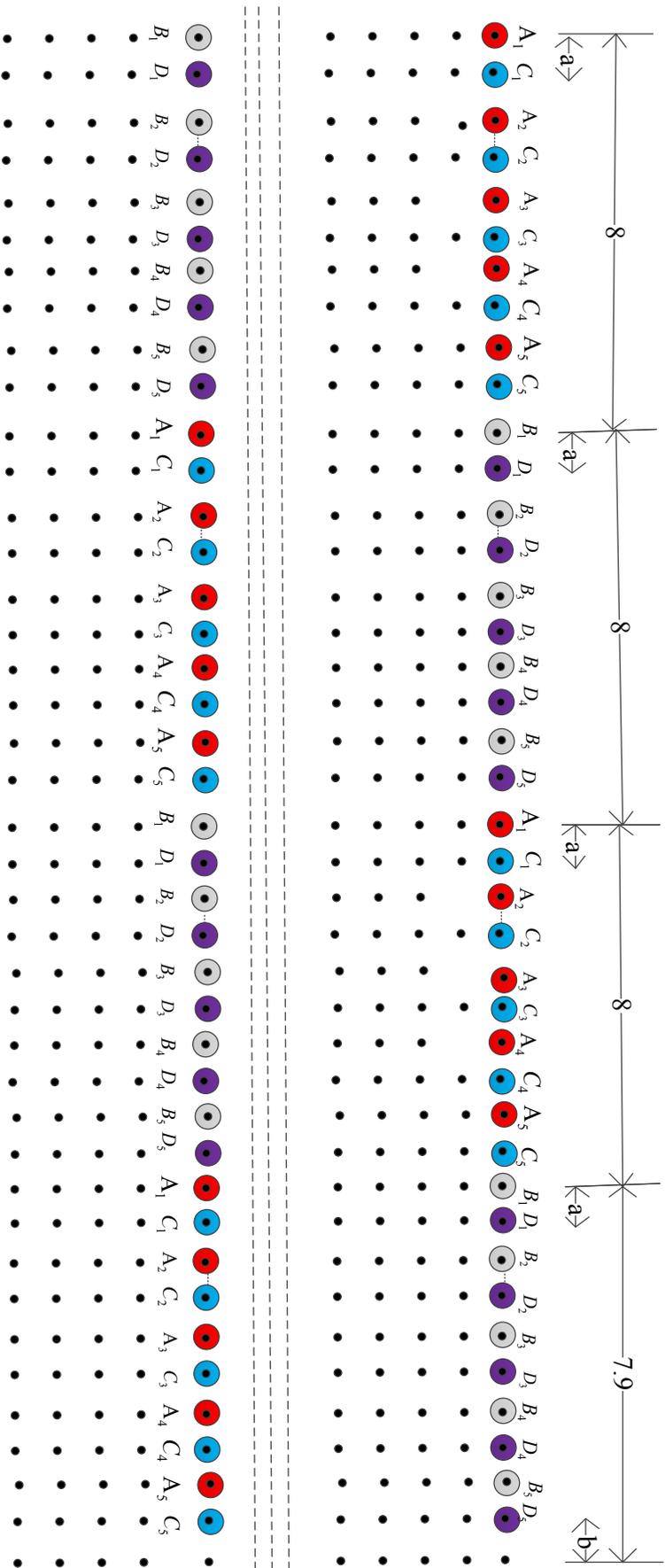


图 22 探测车数目为 80 时的分组情况

表 5 A_1, B_1, C_1, D_1 组探测车的行走时序

组	探测时间/s	行走时间/s	探测时间/s	行走时间/s	
A_1	0-12	12-24	24-36	36-48	...
B_1	6-18	18-30	30-42	42-54	...
C_1	12-24	24-36	36-48	48-60	...
D_1	18-30	30-42	42-54	54-66	...

起始时刻 A_1 、 B_1 组共有 5 辆探测车同时工作， C_1 、 D_1 组共有 4 辆探测车同时工作，从 A_1 组到 D_1 组探测车的启动时间依此滞后 6s，即 A_1 开始扫描 6s 后 B_1 开始工作， B_1 扫描 6s 后 C_1 开始工作， C_1 扫描 6s 后 D_1 开始工作，此时 C_1 在扫描时 A_1 处于行走状态， D_1 在扫描时 B_1 处于行走状态。当 A_1 完成第一列最后一个探测点的扫描过程后向右行走至下一列端点位置的探测点上开始等候。当 A_1 向右行走 12s 时距离 A_2 开始工作，其工作过程与 A_1 组探测车的行走过程一致；同理，当 B_1 组探测车完成最后一个探测点的探测后 B_1 组探测车向右行走至紧邻 B_1 组右侧探测点后开始等待。在 B_1 组向右行走 12s 时 B_2 组开始扫描，此过程与 B_1 组探测车的探测过程相同。 C_1 组与 D_1 组的探测过程与 A_1, B_1 组相同。

当 A_2 探测车扫描了该列最后一个探测点后向右行走 12s 时 A_3 组探测车开始扫描， A_3 组、 A_4 组和 A_5 组的运行过程与 A_1 、 A_2 组的情况相同。同理，当 B_2 探测车扫描了该列最后一个探测点后向右行走 12s 时 A_3 组探测车开始扫描， A_3 组、 A_4 组和 A_5 组的运行过程与 A_1 、 A_2 组的情况相同；当 C_2 探测车扫描了该列最后一个探测点后向右行走 12s 时 C_3 组探测车开始扫描， C_3 组、 C_4 组和 C_5 组的运行过程与 C_1 、 C_2 组的情况相同；当 D_2 探测车扫描了该列最后一个探测点后向右行走 12s 时 D_3 组探测车开始扫描， D_3 组、 D_4 组和 D_5 组的运行过程与 D_1 、 D_2 组的情况相同。

当 A_5 扫描完一列后向右行走 12s 时 A_1 开始扫描其紧邻右侧的探测点， B_5 扫描完一列后向右行走 12s 时 B_1 开始扫描其紧邻右侧的探测点， C_5 扫描完一列后向右行走 12s 时 C_1 开始扫描其紧邻右侧的探测点， D_5 扫描完一列后向右行走 12s 时 D_1 开始扫描其紧邻右侧的探测点。以此扫描过程持续进行直至完成所有探测点的扫描过程，优化程序得到当初初始布局为 80 辆探测车时完成所有探测点的探测过程共需 4806s。

6.5 最优探测车台数及其优化结果与分析

采用遍历法得到的最短时间 T_{\min} 随探测车数目 m 的变化曲线如图 23 所示。从图中可以看出，最短时间随探测车数目增加呈现先减小后不变的规律，当探测车数目 $m = 32$ 时，最短时间 T_{\min} 取到最小值，且此时所用探测车数目也最少，效率最高。

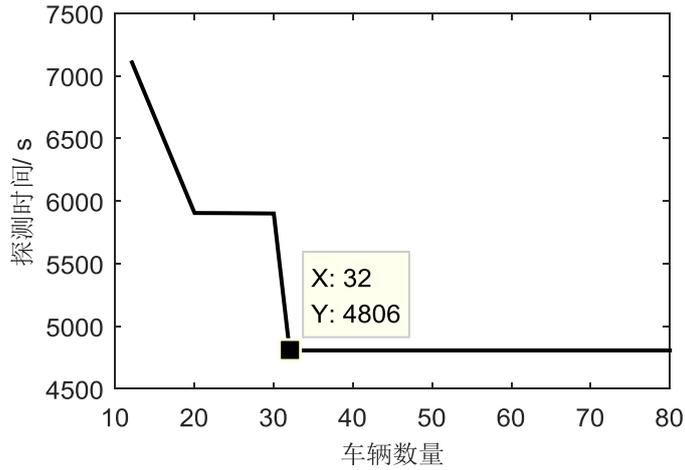


图 23 最短探测时间随探测车数目的变化曲线

下面给出探测车数目为 32 时的探测方案。

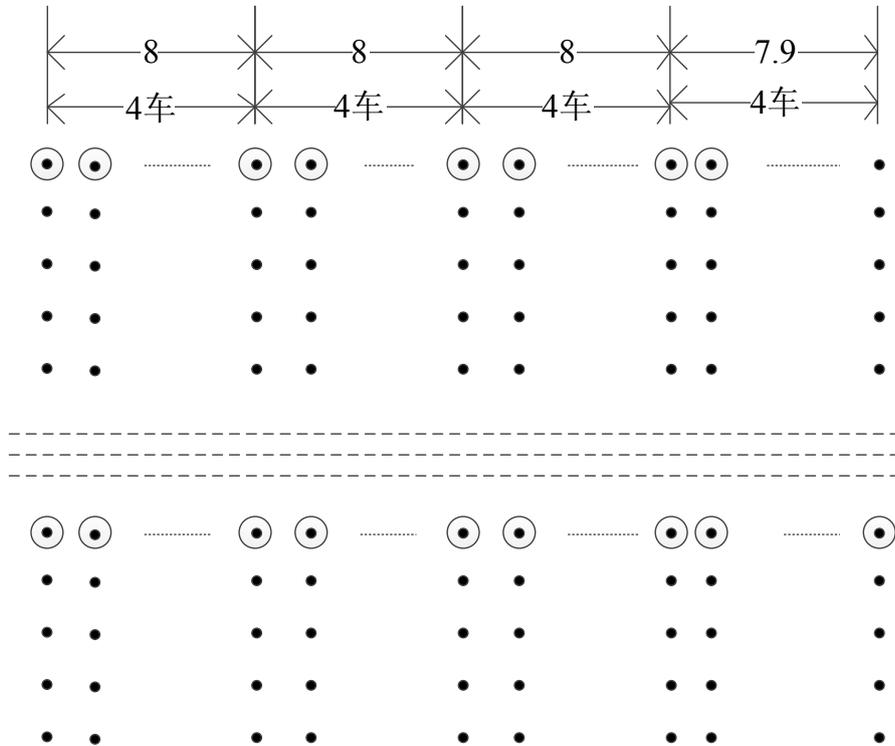


图 24 探测车数量为 32 时的初始布局

如图 24 所示为探测车数量为 32 时的初始布局：在 0-30km 范围内每隔 2km 分布一辆探测车，最后一辆车距右边界的距离为 1.9km，河流左边的分布情况与河流右边的分布情况相同。同样将探测车分为 A_1, B_1, C_1, D_1 和 A_2, B_2, C_2, D_2 ，各组探测车的分布情况如图 25 所示。

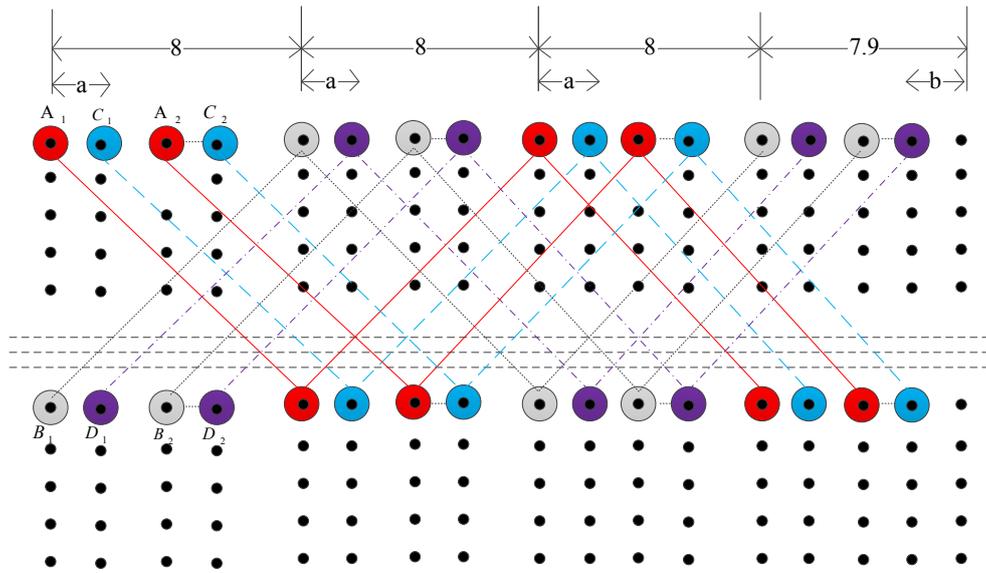


图 25 探测车数量为 32 时的分组情况

其中 a 为 2km, b 为 1.9km。各组探测车的行走时序如表 6 所示。

表 6 A_1, B_1, C_1, D_1 组探测车的行走时序

组	探测时间/s	行走时间/s	探测时间/s	行走时间/s	
A_1	0-12	12-24	24-36	36-48	...
B_1	6-18	18-30	30-42	42-54	...
C_1	12-24	24-36	36-48	48-60	...
D_1	18-30	30-42	42-54	54-66	...

起始时刻 A_1 、 B_1 组共有 5 辆探测车同时工作, C_1 、 D_1 组共有 4 辆探测车同时工作, 从 A_1 组到 D_1 组探测车的启动时间依此滞后 6s, 即 A_1 开始扫描 6s 后 B_1 开始工作, B_1 扫描 6s 后 C_1 开始工作, C_1 扫描 6s 后 D_1 开始工作, 此时 C_1 在扫描时 A_1 处于行走状态, D_1 在扫描时 B_1 处于行走状态。当 A_1 完成第一列最后一个探测点的扫描过程后向右行走至下一列端点位置的探测点上开始等候。当 A_1 向右行走 12s 时距离 A_2 开始工作, 其工作过程与 A_1 组探测车的行走过程一致; 同理, 当 B_1 组探测车完成最后一个探测点的探测后 B_1 组探测车向右行走至紧邻 B_1 组右侧探测点后开始等待。在 B_1 组向右行走 12s 时 B_2 组开始扫描, 此过程与 B_1 组探测车的探测过程相同。 C_1 组与 D_1 组的探测过程与 A_1, B_1 组相同。

当 A_2 探测车扫描了该列最后一个探测点后向右行走 12s 时 A_1 组探测车开始扫描, A_2 行走至下一列探测点时开始等待, 此时 A_1 开始扫描。当 A_1, B_1, C_1, D_1 组与 A_2, B_2, C_2, D_2 组依此扫描 1km 范围内的探测点。优化程序得到当初初始布局为 32 辆探测车时完成所有探测点的探测过程共需 4806s, 此后当初初始布局的探测车数量再增加时探测完所有探测点所需要的时间并不会减小, 因此采用 32 辆探测车即可以用最小的成本实现最短时间内对所有探测点的探测过程, 从而大幅度提高考古探测的经济与时间效益。

6.5 探测车位置与状态的计算机程序实现

本部分结果见程序附件。附录中给出了以初始共有 32 辆探测车为例，时间为 132s 时各个探测车的位置与指令情况。

七、模型的评价

7.1 模型优点

1、采用基于协同进化的粒子群算法可以很好地解决该多起点多约束的探测车路径规划问题，将复杂的路径规划问题交给智能算法来解决，减少了不必要的麻烦。

2、多台扫描探测车采用分组运行的方式，同一组的探测车可以同时运行，这样加快了扫描运行的效率。

3、通过提出探测车路径规划的三个总体原则，减小了待优化变量的优化范围，大大降低了优化算法的计算量，提高了优化效率。

4、采用粒子群算法优化探测车的路径，能够得到全局最优解，且优化效率较高。

7.2 模型缺点

1、模型基于探测车的分组进行模式，而不是对探测车的所有可能运行模式直接进行优化，这种方法在没有对结果进行事后分析验证的情况下可能得不到最优的结果。

2、所提出的总体原则只能在一定程度上定性指导优化过程的进行，不能给出其定量的指导准则，因此，有时对优化的指导缺乏明确的标准，不容易把握。

3、该模型对于探测车数目较少时，易于优化得到结果，但是当探测车数目过多，容易导致计算量膨胀，导致优化效率过低，甚至失败。

八、问题的进一步拓展

1、本模型中假设每辆探测车不计成本，而实际应用中，成本也是工程应用中的一个重要考虑因素，如何在优化过程中同时考虑成本和时间因素将增加模型的优化变量数目，也是值得考虑的问题。

2、本模型中假设探测车的数目为偶数，认为当探测车数目为偶数时，探测时间最短，并没有研究奇数情况下的探测车路径规划问题，可以对其进行相关研究。

3、本模型中研究的探测车都为相同类型，即运行速度和探测时间间隔严格相同，当问题中引入不同的运行速度或时间间隔时，模型的复杂度显然大大增加，如何解决这类问题也值得研究。

4、本模型中假设探测车只能垂直或者水平移动，然而实际工作中探测车可以沿对角线移动，从而减少从一个节点到另一个节点的时间，此种情况下的优化结果必然与本文中的优化结果不同，也可以进行研究。

5、通过本问题的分析发现，竖直方向的运行路线时间消耗要小于水平方向的运行路线，所以可以提前规划好每辆车的运行路线，即让水平方向的移动过程尽可能减少。

九、参考文献

- [1] 吴孟达, 成礼智, 吴翊, 毛紫阳, 王丹, 数学建模教程, 长沙: 国防科学技术大学出版社, 2013
- [2] 姜启源, 谢金星, 叶俊, 数学模型 (第三版), 北京: 高等教育出版社, 2003
- [3] 肖华勇, 实用数学建模与软件应用, 西安: 西北工业大学出版社, 2008
- [4] 马勇, 多移动机器人路径规划研究: [博士学位论文]. 华中科技大学, 2012
- [5] 肖常, 基于免疫进化与协同进化的移动机器人路径规划: [硕士学位论文]. 长沙: 中南大学, 2008
- [6] 许珂, 刘栋, 多粒子群协同进化算法, 计算机工程与应用, 45 (3): 51-54, 2009
- [7] 赵硕, 多机器人协同的焊接路径规划研究: [硕士学位论文]. 合肥: 合肥工业大学, 2015
- [8] 毕晓君, 张永健, 沈继红, 高维多目标多方向协同进化算法, 控制与决策, 10: 1737-1743, 2014

附录

表 7 132s 时 32 辆探测车的位置与状态

编号	X	Y	状态
1	0.1	0.2	等待
2	2	0	停止扫描
3	4	0	开始扫描
4	6	0	等待
5	8.075	0.2	向右
6	10	0	停止扫描
7	12	0	等待
8	14	0	等待
9	16.1	0.2	等待
10	18	0	停止扫描
11	20	0	开始扫描
12	22	0	等待
13	24.075	0.2	向右
14	26	0	等待
15	28	0	等待
16	30	0	等待
17	0.075	5.25	向右
18	2	5.05	停止扫描
19	4	5.05	等待
20	6	5.05	等待
21	8.1	5.25	等待
22	10	5.05	开始扫描
23	12	5.05	等待
24	14	5.05	等待
25	16.075	5.25	向右
26	18	5.05	停止扫描
27	20	5.05	等待
28	22	5.05	等待
29	24.1	5.25	等待
30	26	5.05	等待
31	28	5.05	开始扫描
32	30	5.05	等待