
文献阅读：
利用无人机进行移动目标搜
索的运动编码粒子群优化算
法

2022.7.18

摘 要

本文提出了一种名为运动编码粒子群优化 (MPSO) 的新算法, 用于 UAV 寻找移动目标。根据贝叶斯理论, 搜索问题可以转化为目标函数的优化问题, 该函数表示搜索到目标的概率。这里, 所提出的 MPSO 是通过将搜索轨迹编码成一系列在 PSO 算法中的粒子生成过程中的 UAV 运动路径来解决这一问题的。这种运动编码方法允许保留粒子群的重要属性, 包括认知和社会一致性, 从而产生更好的解。进行的与现有方法的仿真结果表明, 与基本 PSO 相比, 所提出的 MPSO 将检测性能提高了 24%, 时间性能提高了 4.71 倍, 此外, 在大多数搜索场景中也优于其他的元启发式优化算法, 包括人工蜂群 (ABC)、蚁群优化 (ACO)、遗传算法 (GA)、差分进化 (DE) 和被囊群优化 (TSA)。我们利用真实的 UAV 在对不同场景下对搜索动态目标进行了实验, 证明了 MPSO 在实际应用中的优势。

关键字: 优化搜索; 粒子群优化; UAV

第一章 引言

无人驾驶飞行器 (UAV) 因其具有能在恶劣的环境下处理不同任务的丰富的工作能力而具有广泛的实际应用, 特别是在监视和救援方面。在使用 UAV 搜索一个丢失目标时, 通常存在一个被称为“黄金时间”的关键时期, 在这个时间段内发现目标的概率最高[1]。随着时间的推移, 由于初始信息的衰减以及天气条件、地形特征和目标动态等外部因素的影响, 这种概率会迅速降低。因此, 使用 UAV 搜索丢失目标的主要目标是寻找一条路径, 该路径可以在给定的目标位置和搜索条件的初始信息下, 使在特定的飞行时间内搜索到目标的概率最大[2,3]。

在文献中, 搜索问题通常被表述为概率函数, 以便可以充分纳入初始假设、搜索条件和传感器模型的不确定性。在[2,4]中, 引入了贝叶斯方法来推导评估 UAV 飞行路径搜索概率的目标函数。初始搜索图被建模为多元正态分布, 其均值和方差是根据关于目标位置的初始信息计算的[5,6]。在[3,6]中, 目标动作由一个随机的马尔可夫过程表示, 该过程可以是确定性的或不确定性的, 这取决于搜索场景。另一方面, 传感器通常被建模为具有“检测到”或“未检测到”两种状态的二元变量[5], 或者被建模为连续的高斯变量[2]。

由于涉及各种概率变量, 搜索问题的复杂性从非确定性多项式时间困难 (NP-hard[7]) 到非确定性指数时间完备性 (NEXP-complete[8]) 的水平不等, 其中可供搜索的解的数量相对于搜索维度和飞行时间呈指数增长。因此, 用微积分等经典方法寻找精确解来求解这个问题变得不切实际, 因此, 经常使用近似方法。许多方法已经被开发出来, 如带有一步 look ahead [2]和 k 步 look ahead [3]的贪婪搜索, 蚁群优化(ACO) [5], 贝叶斯优化方法 (BOA) [4], 遗传算法 (GA) [9,10], 交叉熵优化 (CEO) [11], 分支定界方法[12], 有限深度搜索[13], 和梯度下降法[14,15]。表 1-1 比较了一些算法的主要特征, 其中“多智能体”一栏意味着可以使用多架 UAV 进行搜索的可能性, 而“特设启发式”则是专门为特定的搜索问题开发的。值得注意的是, 大多数方法都是针对移动目标, 并使用检测传感器的二元模型。一些方法 ([4,5,11,13]) 采用多架 UAV 来加速搜索过程, 而其他方法则使用特设启发式来提高搜索概率。

表 1-1 搜索方法之间的比较

方法	工作	目标类型	二元传感器	多智能体
一步 look ahead	[2]	静止和移动	否	否
k 步 look ahead	[3]	移动	是	否
BOA	[4]	移动	是	是
ACO	[5]	移动	是	是
GA	[10]	静止	是	否
CEO	[11]	移动	是	是
有限深度搜索	[13]	静止	是	是
梯度下降法	[14]	静止	否	否

从文献中可以看出, 优化搜索的方法在假设、约束、目标动态和搜索机制方面存在差异。由于其复杂性, 最优搜索, 特别是在有快速移动目标的情况下, 仍然是一个具有挑战性的问题。此外, 传感器、通信和 UAV 技术的最新进展使得新搜索平台的开发成为可能。它们提出了对新方法的需求, 该新方法不仅应该在搜索能力上是鲁棒的, 而且还应该具有诸如计算效率、适应性和最优性等特性。

对于优化, 粒子群优化 (PSO) 是一种潜在的技术, 具有一些关键的优势, 已被成功地应用于各种领域[16,17,18,19,20]。它对初始条件以及目标函数的变化不太敏感, 并且能够通过少量的参数 (包括惯性权重和两个加速因子) 来适应许多搜索场景[21]。与其他随机方法相比, 它一般能以稳定的收敛速度和较短的计算时间找到全局解[22]。更重要的是, PSO 的实现简单, 不仅能够在计算机集群或多处理器上并行运行, 还可以在单个图形卡的图形处理单元 (GPU) 上运行。这可以大大减少运行时间, 而不需要对系统硬件做任何改变[23]。

基于上述分析, 我们将在本研究中采用 PSO 算法来处理复杂场景中快速移动目标的搜索问题, 目的是提高检测概率和执行时间两方面的搜索性能。为此, 我们提出了一种新的运动编码 PSO 算法, 同时考虑到了粒子群的认知和社会一致性。我们的贡献包括: (i) 从搜索问题和概率框架出发, 结合所有假设和约束条件, 为优化制定一个目标函数; (ii) 从改变粒子群的搜索空间以避免陷入局部最大值的想法出发, 开发了一个新的运动编码 PSO (MPSO); (iii) 在实验搜索场景中展示了为 UAV 实施的 MPSO, 以验证其优于其他 PSO 算法的性能, 这是通过广泛的比较分析得到的。结果表明, MPSO 一方面在各种搜索场景中表现出优异的性能, 另一方面在实际实施中仍

然很简单。

本文的其余部分结构如下。第二章概述了制定目标函数的步骤。第三章介绍了所提出的 MPSO 及其解决复杂搜索问题的实现。第四章提供了仿真和实验结果。第五章得出本文的结论。

第二章 问题描述

搜索问题是通过目标、传感器和置信图进行建模来表述的，其细节如下：

2.1 目标模型

在搜索问题中，目标由一个代表其位置的未知变量 $x \in X$ 来描述。在搜索开始前，一个概率分布函数（PDF）被用来根据现有的信息对目标位置进行建模，例如，目标在失去信号之前的已知的最后位置。这个 PDF 可以是一个以已知的最后位置为中心的正态分布，但如果对目标位置一无所知，也可以是一个均匀的 PDF。在搜索空间中，这个 PDF 由一个被称为置信图 $b(x_0)$ 的网格图表示，其中的每个单元的值对应于目标在该单元的概率。可以通过将搜索空间 S 划分为 $S_r \times S_c$ 个单元的网格，并为每个单元设定一个概率来创建该地图。假设目标出现在搜索空间中，我们有 $\sum_{x_0 \in S} b(x_0) = 1$ 。

在搜索过程中，目标可能不是静止的，而是按照一定的模式导航的。这种模式可以用一个随机过程来模拟，我们假设这个过程是一个马尔可夫过程。在本研究中考虑的具有条件确定的目标的特殊情况下，该模式仅取决于目标的初始位置 x_0 。在这种情况下，转移函数 $p(x|x_t-1)$ 表示目标从单元 x_t-1 到 x_t 的概率，对于所有单元 $x_t \in S$ 来说是已知的。因此，如果目标的初始位置已知，则它的路径将是完全已知的。这个假设经常被用在海上幸存者的搜索中[24]，也被用于一般的搜索问题[5]。

2.2 传感器模型

为了寻找和发现目标，在 UAV 上安装了一个传感器，在每个时刻 t 进行一次观测 z_t 。每次观测是独立的，因此一次观测的发生不提供关于另一次观测的发生的消息。实现一个检测算法来得到每次观测的结果，假设只有两种可能的输出，即检测到目标 $z_t = D_t$ ，和未检测到目标 $z_t = \bar{D}_t$ ，其中 D_t 代表时刻 t 的“检测”事件。由于传感器和检测算法的不完善，检测到目标的观测 $z_t = D_t$ ，仍然不能确保目标存在于 x_t 中。这反映在观测概率 $p(z_t|x_t)$ 上，给定传感器模型的知识。在目标位置 x_t 的情况下，没有检测到的概率通过以下方式计算：

$$p(\bar{D}_t | x_t) = 1 - p(D_t | x_t) \quad (2-1)$$

2.3 更新置信图

一旦初始分布 $b(x_0)$ 被初始化，就可以根据贝叶斯方法和传感器的观测序列 $z_{1:t} = \{z_1, \dots, z_t\}$ 来建立目标在时刻 t 的置信图 $b(x_t)$ 。这种方法是通过预测和更新两个阶段递归进行的。在预测阶段，置信图根据目标运动模型随时间递归。假设在时刻 t ，以前的置信图 $b(x_{t-1})$ 是可用的，则预测的置信图的计算如下：

$$\hat{b}(x_t) = \sum_{x_{t-1} \in S} p(x_t | x_{t-1}) b(x_{t-1}) \quad (2-2)$$

从 (2-2) 中可以看出，置信图 $b(x_{t-1})$ 实际上是目标在 x_{t-1} 处的条件概率，给定直到 $t-1$ 时刻的观测值， $b(x_{t-1}) = p(x_{t-1} | z_{1:t-1})$ 。当观测值 z_t 可用时，更新只需将预测的置信图乘以新的条件观测概率，如下所示：

$$b(x_t) = \eta_t p(z_t | x_t) \hat{b}(x_t) \quad (2-3)$$

其中 η_t 是归一化系数。

$$\eta_t = 1 / \sum_{x_t \in S} p(z_t | x_t) \hat{b}(x_t) \quad (2-4)$$

η_t 将目标出现在搜索区域内的概率扩展为 1，即 $\sum_{x_t \in S} b(x_t) = 1$ 。

2.4 搜索目标函数

根据贝叶斯理论，在一次观测中，目标在时刻 t 未被探测到的概率 $r_t = p(\bar{D}_t | z_{1:t-1})$ ，依赖于两个因素：(i) 来自预测阶段的最新置信图 (2-2)，(ii) 未检测到的概率 (2-1)。在整个搜索区域内，该概率由以下因素给出：

$$r_t = \sum_{x_t \in S} p(\bar{D}_t | x_t) \hat{b}(x_t) \quad (2-5)$$

注意，对于“未检测到”事件 $z_t = \bar{D}_t$ ， r_t 正好是 (2-4) 中归一化系数 η_t 的倒数，即 $r_t = 1 / \eta_t$ ，因此小于 1。通过将未检测到的概率 r_t 随时间相乘，然后得到从时刻 1 到 t 未能检测到目标的联合概率 $R_t = p(\bar{D}_{1:t})$ ：

$$R_t = \prod_{k=1}^t r_k = R_{t-1} r_t \quad (2-6)$$

因此，目标在时刻 t 第一次被检测到的概率计算如下：

$$p_t = \prod_{k=1}^{t-1} r_k (1 - r_t) = R_{t-1} (1 - r_t) \quad (2-7)$$

将 p_t 在 t 个时刻上相加，就得到了在 t 个时刻内检测到目标的概率：

$$P_t = \sum_{k=1}^t p_k = P_{t-1} + p_t \quad (2-8)$$

因此， P_t 通常被称为“累积”概率，以区别于 p_t 。注意：

$$P_t = 1 - R_t \quad (2-9)$$

而随着 t 的增大，第一次检测的概率 p_t 会变小，因为在以前的步骤中检测到目标的概率增大。因此，累积概率 P_t 是有界的，并且随着 t 的增加而向1靠近。

现在可以根据(2-8)给定的有限搜索时间来制定搜索问题的目标函数。让搜索时间段为 $\{1, \dots, N\}$ ，搜索策略的目标是确定一条搜索路径 $O = (o_1, \dots, o_N)$ ，可以使累积概率 P_t 最大化。因此，目标函数最终被表述为：

$$J = \sum_{t=1}^N p_t \quad (2-10)$$

第三章 运动编码粒子群优化算法

由于（3-10）中定义的搜索问题是 NP-hard[7, 8]，计算所有可能的路径以找到最优解所需的时间将大大增加，变得难以解决。因此，像 PSO 这样的启发式方法是解决本研究中的最优搜索问题的一个很好的选择。

3.1 粒子群算法

PSO 是一种基于种群的随机算法，受到鸟群社会行为的启发，旨在求解优化问题[16,25]。在 PSO 中，最初生成一个具有随机位置和速度的粒子群。然后每个粒子以认知的方式与其他粒子一起移动和进化，以寻求全局最优。这些运动由个体最优位置 L_k 和粒子群的最优位置 G_k 驱动。令 x_k 和 v_k 分别是一个粒子在第 k 代的位置和速度。该粒子的下一代的运动由以下公式给出：

$$v_{k+1} \leftarrow wv_k + \varphi_1 r_1 (L_k - x_k) + \varphi_2 r_2 (G_k - x_k) \quad (3-1)$$

$$x_{k+1} \leftarrow x_k + v_{k+1} \quad (3-2)$$

其中， w 是惯性权重， φ_1 是认知系数， φ_2 是社会系数， r_1 、 r_2 是从[0,1]范围内的均匀概率分布中采样的随机序列。从（3-1）和（3-2）中可以看出，一个粒子的运动是由三个因素引导的，即遵循自己的方式，向自己的最优位置运动，或者向粒子群的最优位置运动。这些因素之间的比例由 w 、 φ_1 和 φ_2 的值决定。

3.2 最优搜索的 MPSO

根据应用的不同，对 PSO 算法进行了一些修改和改进。然而，在复杂环境下，特别是在有限的时间窗口内，实现 PSO 算法在线搜索动态目标仍然是一项具有挑战性的任务。对于搜索问题，人们希望对粒子的位置进行编码，以使粒子能够逐渐向全局最优运动。一种常见的方法是将位置定义为代表可能的搜索路径的多维矢量：

$$x_k \sim O_k = (o_{k,1}, \dots, o_{k,N}) \quad (3-3)$$

其中 $o_{k,i}$ 对应于搜索地图的一个节点[26,27]。这种方法的缺点是没有覆盖路径节点中的相邻动态行为,因此在搜索过程中可能会导致无效路径。离散 PSO 可以用来克服这个问题,但粒子的动量没有被保留,造成局部最大值[28]。间接方法,如角度编码的 PSO[29]和基于优先级的编码 PSO[30],可以作为一个很好的选择来处理它,并产生更好的结果。然而,他们的映射函数要求相位角在 $[-\pi/2, \pi/2]$ 范围内,这限制了搜索能力,特别是在大维度上。

在这里,我们提出了使用 UAV 运动来编码粒子位置的想法。我们不使用节点,而是将每个搜索路径看作是一组 UAV 运动段,每个运动段都对应于 UAV 从当前单元到飞行平面上的另一个单元的运动。通过将时刻 t 的运动幅值和方向分别定义为 ρ_t 和 α_t ,该运动可以完全由矢量 $u_t = (\rho_t, \alpha_t)$ 来描述。然后,一个搜索路径由 N 个运动段的矢量描述, $U_k = (u_{k,1}, \dots, u_{k,N})$ 。使用 U_k 作为每个粒子的位置, MPSO 的方程可以写成:

$$\Delta U_{k+1} \leftarrow wU_k + \varphi_1 r_1 (L_k - U_k) + \varphi_2 r_2 (G_k - U_k) \quad (3-4)$$

$$U_{k+1} \leftarrow U_k + \Delta U_{k+1} \quad (3-5)$$

图 3-1 展示了一条有三段的路径, $U_k = ((1,0), (1, 3\pi/2), (\sqrt{2}, 7\pi/4))$, 其中置信图以颜色编码,右侧标明了概率值。

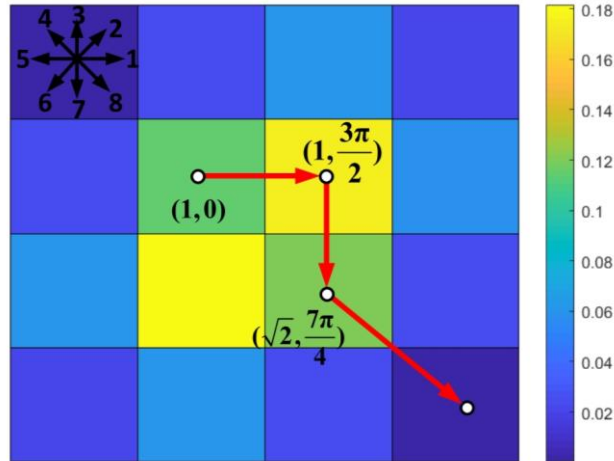


图 3-1 有三段路径的运动编码图, $U_k = ((1,0), (1, 3\pi/2), (\sqrt{2}, 7\pi/4))$

在搜索过程中,还需要将 U_k 映射为直接路径 O_k , 以便评估与 U_k 相关的目标函数。如图 3-1 所示,映射过程可以通过在每个时刻中首先将 UAV 的运动限制

在其八个邻居之一来进行。然后，运动幅值 ρ_t 可以被归一化，运动角度 α_t 可以被量化为：

$$\rho_t^* = 1 \quad (3-6)$$

$$\alpha_t^* = 45^\circ \langle \alpha_t / 45^\circ \rangle \quad (3-7)$$

其中， $\langle \rangle$ 代表四舍五入运算符，即为最近的整数。然后，对应于 UAV 在笛卡尔空间中的位置的节点 $o_{k,t+1}$ 由以下公式给出：

$$o_{k,t+1} = o_{k,t} + u_{k,t}^* \quad (3-8)$$

$$u_{k,t}^* = (\langle \cos \alpha_t^* \rangle, \langle \sin \alpha_t^* \rangle) \quad (3-9)$$

从解码后的路径 O_k 中，可以通过目标函数公式 (3-10) 来评估目标函数，然后可以计算出局部和全局最优，具体如下：

$$L_k = \begin{cases} U_k & \text{if } J(O_k) > J(L_{k-1}^*) \\ L_{k-1} & \text{otherwise} \end{cases} \quad (3-10)$$

$$G_k = \arg \max_{L_k} J(O_k) \quad (3-11)$$

其中 L_k^* 是 L_k 的解码路径。从映射过程可以看出，(3-7) 将运动分为八个可能的方向之一，(3-9) 将运动方向转换为笛卡尔坐标系下的增量，(3-8) 结合增量形成路径的下一个节点。

类似于信号分析中时域和频域之间的交换，MPSO 的映射过程允许粒子在运动空间而不是笛卡尔空间中搜索。这带来了以下优点：

- 运动空间连续地保持着节点的位置，使得每一代进化后的结果路径总是有效的，而笛卡尔空间则不是这样的；
- 在运动空间中，粒子的动量和包括搜索和开发在内的粒子群行为被保留下来，从而保持搜索性能，并使粒子群能够应对不同的目标动态。
- 由于 (3-6) 和 (3-7) 中 ρ_t 的归一化和 α_t 的量化只是为了目标函数评估而进行的，它们的连续值仍被用于速度和位置的更新，如 (3-4) - (3-5) 所示。这一特性对于避免 PSO 的离散化效应非常重要，这样搜索分辨率就不会受到影响。

最后，我们还注意到，MPSO 通过其更新方程 (3-4) - (3-5) 保留了 PSO 的搜索机制从而可以保持 PSO 的优点，如稳定收敛、初始条件无关性和实现的可行性。

3.3 算法实现

图 3-2 显示了 MPSO 的流程图，以说明所提出的算法 1 中的实现。它的结构是基于核心 PSO，但加入了运动编码和解码的过程。当目标为非静态时，需要在计算适应度时进行 (2-2) 和 (2-3) 中的置信图更新。值得注意的是，[23]中提出的并行技术可用于加快 MPSO 的计算过程。

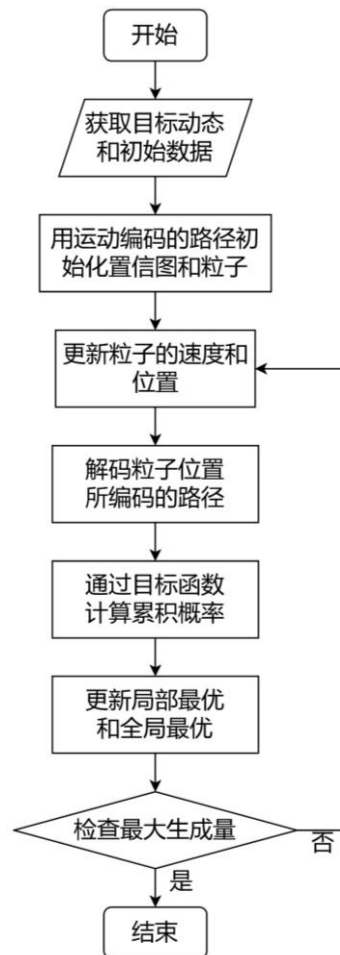


图 3-2 MPSO 算法的流程图

```

/* 初始化:                                     */
获取目标动态和初始数据;
创建置信图;
设置粒子群参数  $w, \phi_1, \phi_2, swarm\_size$ ;

foreach 粒子群中的粒子 do
    创建随机运动编码的路径  $U_k$ ;

    将  $U_k$  分配给粒子位置;

    计算每个粒子的适应度;
    将每个粒子的局部最优值设置为自身;
    将每个粒子的速度设为零;

end
将全局最优设置为最优适应度粒子;

/* 演化:                                     */
for  $k \leftarrow 1$  to 最大迭代次数 do
    foreach 粒子群中的粒子 do
        计算运动速度  $\Delta U_{k+1}$ ; /* Eq.14 */

        计算新的位置  $U_{k+1}$ ; /* Eq.15 */

        解码  $U_{k+1}$  到  $O_{k+1}$ ; /* Eq.19 - 18 */

        更新  $O_{k+1}$  的适应度; /* Eq.10 */

        更新局部最优  $L_{k+1}$ ; /* Eq.20 */

    end

    更新全局最优  $G_{k+1}$ ; /* Eq.21 */

end

```

算法 1: MPSO 的伪代码

第四章 场景设置

为了评估 MPSO 的性能，我们进行了广泛的仿真、比较和实验，详细情况如下：

4.1 算法实现

为了覆盖面广，我们使用了六种不同的搜索场景来分析 MPSO 的最优搜索性能（其中一些场景来自[5]）。这些场景被定义为具有相同的地图大小（ $S_r = S_c = 40$ ），但在 UAV 的初始位置、目标运动模型 $P(x_t | x_{t-1})$ 和初始置信图 $b(x_0)$ 方面有所不同。如图 4-1 所示，概率图用彩色编码，目标动态用白色箭头表示，UAV 的初始位置用白色圆圈表示。这些场景代表了不同的搜索情况，如下所示：

场景 1 有两个相邻的高概率区域。它们的位置和概率值略有不同，这可能导致难以找到一个更好的区域来搜索目标。

场景 2 包括两个分离的高概率区域，它们位于 UAV 位置的相反方向。由于目标正在向西南方向移动，该算法必须迅速确定搜索和跟踪的高概率区域。

场景 3 有一个小而密集区域迅速向东南方向移动。因此，它测试了该算法的探索和适应能力。

场景 4 与场景 3 相似，只是目标正朝着 UAV 的初始位置移动。它进一步评估了搜索算法的适应性。

场景 5 包括两个和起始位置相对的概率区域，其中右侧区域的概率略高。由于目标正在向北移动，该算法需要识别正确的目标区域。

场景 6 与场景 5 类似，但起始位置在潜在区域的下方，目标向东北方向移动。因此，它评估了沿对角线方向搜索的能力。

在我们的评估中，MPSO 在阻尼比为 0.98、 $\varphi_1 = 2.5$ 、 $\varphi_2 = 2.5$ 、 $w=1$ 的情况下实现。粒子群规模设为 1000。最大迭代次数为 100 次，搜索路径的大小为 20 个节点。由于 PSO 的随机性，该算法被执行了 10 次，以找到每个场景的均值和

标准差。

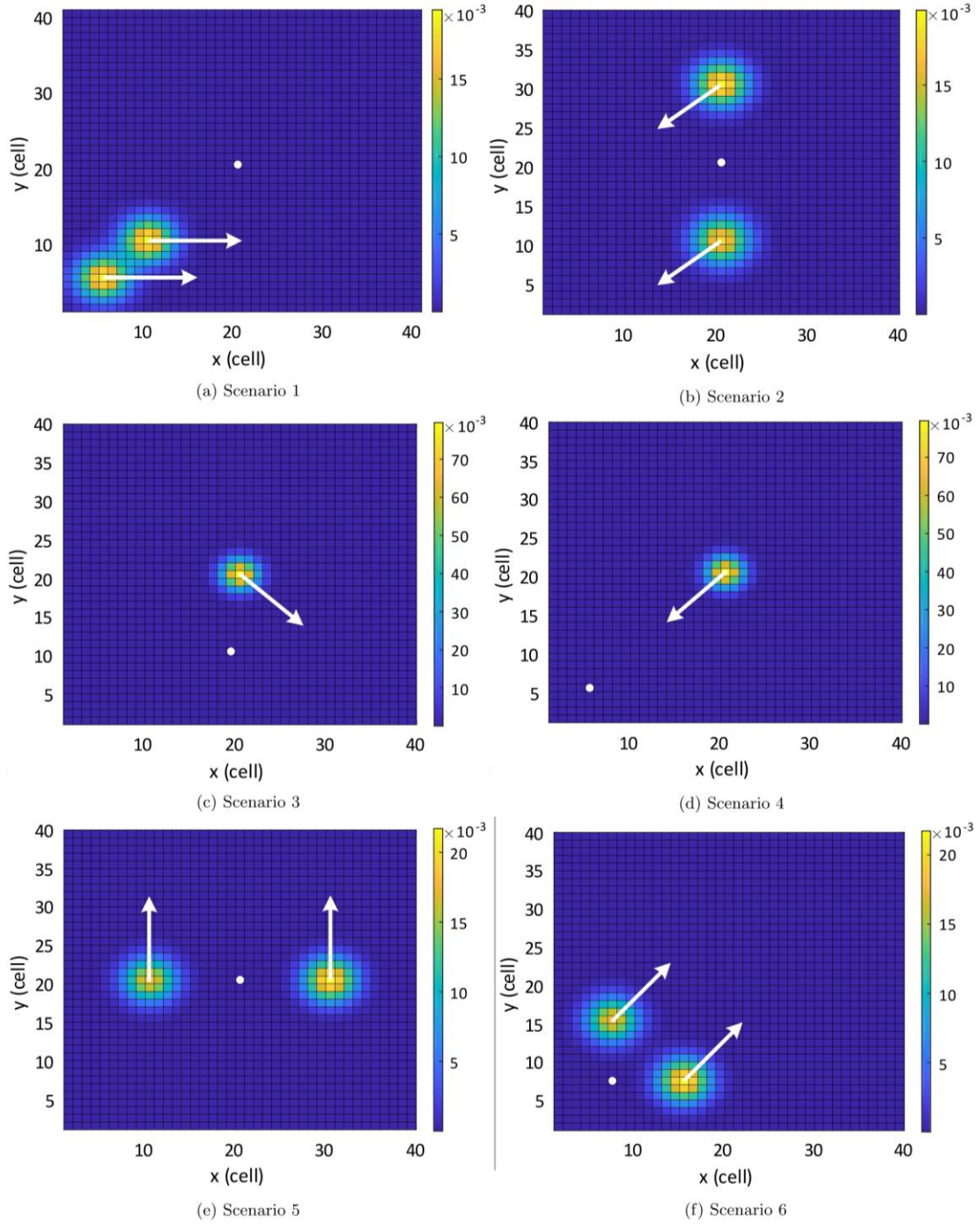


图 4-1 用于评估搜索算法的场景

4.2 搜索路径

图 4-2 展示了 MPSO 在每个场景下的搜索路径以及累积概率值。在所有场景中，MPSO 都能找到最高概率区域，并为 UAV 飞行生成相关路径。对于只有

一个高概率区域的场景，如场景 3 和 4，累积概率很高，因为找到目标的机会没有分散到其他区域。从图 4-2 中还可以看出，概率图只反映了最后一步的目标置信，而搜索路径则代表了随着时间推移对高概率区域的跟踪。通过与图 4-1 的比较，我们可以看出搜索路径适应目标动态以使检测概率最大化。

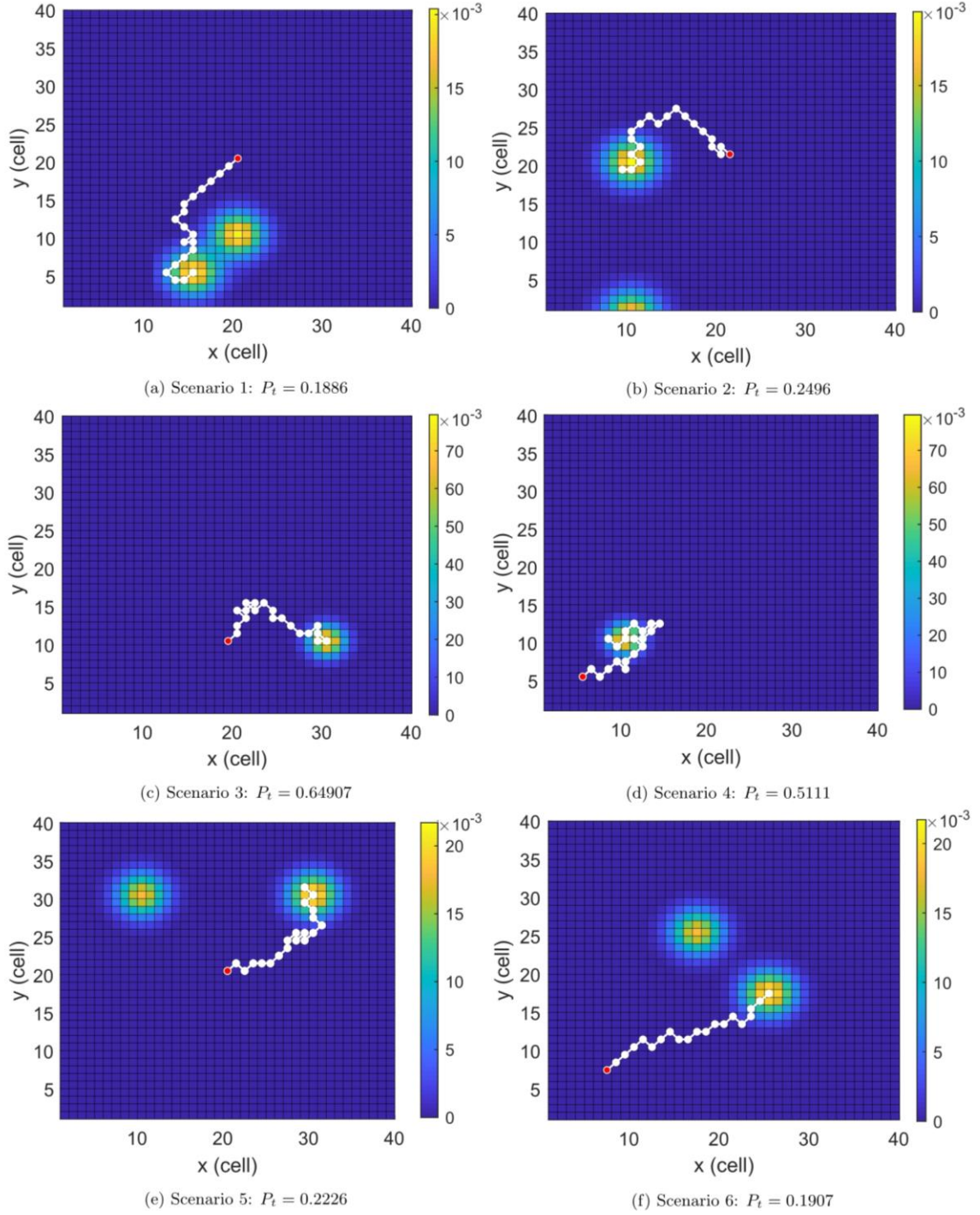


图 4-2 由 MPSO 生成的每个场景的搜索路径