

doi: 10.3969/j.issn.1006-1576.2011.04.011

基于蒙特卡罗的无线传感器网络战场目标定位模型

唐克, 谢保军, 卢金星

(解放军炮兵学院 火控教研室, 合肥 230031)

摘要: 针对目前我军使用 GPS 全球定位系统存在的不足, 在对无线传感器网络进行分析的基础上, 建立基于蒙特卡罗 (Monte Carlo, MCL) 的无线传感器网络定位算法模型。分别对 MCL、质心和 Amorphous 3 种算法进行比较, 并以蒙特卡罗定位算法为例, 研究估计误差与网络参数及算法间的关系。结果表明, MCL 算法误差最小。

关键词: 无线传感器; 网络; 蒙特卡罗; 定位

中图分类号: TP301.6; TP393.01 **文献标志码:** A

Wireless Sensor Network Battlefield Target Fixed Position Model Based on MCL

Tang Ke, Xie Baojun, Lu Jinxing

(Staff Room of Firepower Control, Artillery Academy of PLA, Hefei 230031, China)

Abstract: Aiming at the current shortcomings of military use of global positioning system (GPS), based on analyzing wireless sensor network, establish wireless sensor network positioning algorithm based on Monte Carlo (MCL). Compare MCL algorithm, centroid algorithm, and Amorphous algorithm. Take MCL positioning algorithm as example, research and estimate the relationship among error, network parameter and algorithm. The result shows that MCL algorithm's error is smallest.

Keywords: wireless sensor; network; MCL; fixed position

0 引言

GPS 全球定位系统是目前应用最广泛最成熟的定位系统, 通过卫星的授时和测距对用户节点进行定位, 具有定位精度高、实时性好、抗干扰能力强等优点, 但由于 GPS 的控制权掌握在美军手里, 且卫星定位系统易受敌方的电子干扰, 在密林、涵洞和隧道中或受高大建筑物阻挡时, 因接收不到卫星信号而不能工作, 因此, 在我军重要的军事装备中, 不应将 GPS 定位作为唯一的或主要的定位手段。

C⁴ISRT 系统是利用先进的高科技技术, 为未来的信息化战争设计一个集命令、控制、通信、计算、智能、监视、侦察和定位于一体的战场指挥系统, 受到了军事发达国家的普遍重视, 各国均投入了大量的人力和财力进行研究。

无线传感器是具有感知、计算、处理和无线通信能力的微型传感器, 能够感应及侦测环境的目标物及其改变, 处理收集到的数据, 并将处理过后的资料以无线传输的方式送到数据收集中心或基站。一般由传感器模块、处理器模块、无线通信模块和能量供应模块 4 部分组成。

无线传感器网络是由大量无线传感器以自组织方式构成的无线网络, 其目的是协作感知、采集和

处理网络覆盖区域内感知对象的信息, 并发送给观察者。由于无线传感器网络可以安装或撒布在室内或户外, 通过网络中各个节点的协作达到对目标定位和跟踪的目的, 现已成为 C⁴ISRT 系统不可或缺的一部分。

无线传感器网络节点定位算法包括基于距离的定位算法 (三边测量法、三角测量法、极大似然估计法)、与距离无关的定位算法 (质心算法、DV-Hop 算法、Amorphous 定位算法和 APIT 算法) 等。蒙特卡罗移动方法在随机事件研究方面有着重要的作用, 能科学地反映随机事件的变化规律, 在统计物理、仿真模拟、机器人定位以及无线传感器网络等研究领域中有广泛的应用。笔者提出了蒙特卡罗方法的移动节点定位方法, 对基于蒙特卡罗的无线传感器网络战场目标定位模型和算法进行探讨。

1 无线传感器网络节点定位算法

在无线传感器网络节点定位技术中, 根据节点是否已知自身的位置, 把无线传感器节点分为信标节点 (或锚点) 和未知节点。信标节点在网络节点中所占的比例很小, 可以通过携带 GPS 定位设备等手段获得自身的精确位置。信标节点是未知节点定位的参考点。除了信标节点外, 其它无线传感器节

收稿日期: 2010-12-18; 修回日期: 2011-01-12

作者简介: 唐克 (1962—), 男, 安徽人, 工学硕士, 副教授, 硕士生导师, 从事炮兵武器系统分析与仿真研究。

点就是未知节点，它通过信标节点的位置信息来确定自身位置。

评估一个定位算法时，不但要考虑到定位的精确度，还要考虑到网络配置和网络通信的代价。增大锚点的密度以及增加通信的频率都可以提高定位的精确度，但同时也增加了额外的费用，比如能量、金钱、时间等等，而这些在无线传感器网络中很受限制。因此，如何有效地确定合适的配置参数是笔者在算法评估中需要考虑的主要问题。下面，以蒙特卡罗定位算法为例，来研究估计误差与网络参数及算法间的关系。

2 无线传感器网络战场目标定位算法评估

2.1 蒙特卡罗定位算法模型

初始化样本集：开始节点的位置未知，其取样集可表示为

$$L_0 = \{\text{在配置域中随机抽取 } N \text{ 个样本}\}$$

更新样本集：根据先前的样本集 L_{t-1} 和新的观察 O_t 计算新的样本集 L_t

$$L_t = \{\}$$

While(size(L_t)<N)do

$$R = \{l_t^i \mid l_t^i \in p(l_t^i \mid l_{t-1}^i), l_{t-1}^i \in L_{t-1}, 1 \leq i \leq N\} \quad (1)$$

$$R_{filtered} = \{l_t^i \mid l_t^i \in R \ \& \ p(O_t \mid l_t^i) > 0\} \quad (2)$$

$$L_t = \text{choose}(L_t \cup R_{filtered}, N)$$

其中：N 表示样本的最优个数，是一个常数； L_t 表示 t 时刻节点的样本集； l_t^i 表示 t 时刻节点第 i 个样本的位置； $p(l_t^i \mid l_{t-1}^i)$ 表示在 l_{t-1}^i 条件下 l_t^i 的概率分布； $p(O_t \mid l_t^i)$ 表示在 l_t^i 条件下 O_t 的概率分布。

1) 预测阶段的节点取样

式 (1) 描述了节点由先前的样本集 L_{t-1} 中样本的概率分布来取样本，称这个过程为预测阶段。假设节点不知道自己的速度大小和方向，只知道速度不超过指定值 v_{max} 。用 $d(l_1, l_2)$ 表示取样点 l_1 、 l_2 间的距离，如果速度在 $[0, v_{max}]$ 均匀分布，则当前位置的概率分布可表示为：

$$p(l_t^i \mid l_{t-1}^i) = \begin{cases} \frac{1}{\pi v_{max}^2}, d(l_t, l_{t-1}) < v_{max} \\ 0, d(l_t, l_{t-1}) \geq v_{max} \end{cases} \quad (3)$$

2) 预测阶段的节点过滤

式 (2) 描述由当前的观察信息来过滤预测阶段中不符合条件的样本。

可以通过锚点的通信协议来过滤节点预测过程的样本。假设锚点的邻居节点传递锚点的信息，即：

$$S \rightarrow Region \quad HELLO \mid ID_S \mid loc_t$$

$$N \rightarrow Region \quad HELLO \mid ID_N \mid \{(ID_S, loc_t)\}.$$

锚点向通信范围内广播自己的锚点代号和位置信息，同时锚点的邻居节点向通信范围广播自己的节点代号和锚点的信息。

如果用 S 表示能被取样节点感知的锚点集，用 T 表示不能被取样节点感知，但可以被取样节点的邻居节点感知的锚点集，过滤的结果为：

$$filter(l) = \forall s \in S, d(l, s) \leq r \wedge \forall s \in T, r < d(l, s) \leq 2r \quad (4)$$

3) 反复取样

首先在先前的样本集 L_{t-1} 中采用蒙特卡罗方法随机产生 N 个取样节点 l_t^i 并赋概率绝对权值 \tilde{w}_t^i ，通过过滤阶段的过滤，去掉不符合条件的样本，这样反复地取样和过滤，直至符合条件样本个数达到预先规定的常数 N 。

4) 节点的位置估计

样本的绝对权值为 \tilde{w}_t^i ，标准化后的相对权值为

$\frac{\tilde{w}_t^i}{\sum_{k=1}^N \tilde{w}_t^k}$ ，这样节点的位置估计为：

$$l_t = \sum_{i=1}^N l_t^i \frac{\tilde{w}_t^i}{\sum_{k=1}^N \tilde{w}_t^k} \quad (5)$$

为简化算法，假设符合条件的样本权值都为 1。

2.2 无线传感器网络参数的设定

在模拟试验中规定节点随机分布在 $500\text{m} \times 500\text{m}$ 的矩形区域，节点和锚点的通信半径 r 均为 50m 。

网络和节点主要参数：节点和锚点的速度 $(v_{max}, v_{min}, s_{max}, s_{min})$ 。速度的单位为节点每单元时间的移动距离。节点速度随机取于 $[v_{max}, v_{min}]$ ，锚点速度随机取于 $[s_{max}, s_{min}]$ 。节点密度 (n_d) ，即一跳范围内节点的平均个数。锚点密度 (s_d) ，即一跳范围内锚点的平均个数。

节点的移动模型假定：假设节点不知道自己速度的方向和大小，随机选择自己的速度 $v \in [0, v_{max}]$ 和目的地，到达目的地后停止移动并设速度大小为 0，所以平均速度为 $v_{max} / 2$ ，随后准备下一个目的地。除此以外，节点仅可以判断其它的节点是否在自己

的通信范围内, 但不能测量距离。

通信时间间隔(t_u), 试验多数用一个确定的 t_u , 同时, 速度的单位为 r/t_u ; 取样个数确定(N), 取样个数越多, 定位就越精确, 但需要消耗大量的金钱和时间。

2.3 算法测试软件的开发平台

软件开发采用算法执行和表现分离的思想来设计。算法采用 Java 语言设计, 因为 Java 的一个显著优点就是运行时环境提供了平台无关性, 这样算法的执行不受具体开发平台的限制, 即“一次编成到处运行”, 在 Windows 操作系统下开发的 Java 程序可以在 Unix 等操作系统运行, 除此之外, Java 程序还可以运行在服务器上, 可以被远程调用, 达到资源共享的目的。算法的表现采用 Delphi 界面编成, 因为 Delphi 具有友好的界面设计能力, 能在较短的时间内设计出美观实用的程序。

2.4 算法测试软件结果评估

1) 精确度结果评估

MCL 算法的精确度取决于节点和锚点运动的速度, 随着时间的推移, 节点接收的信息增多, 定位的精确度得到提高。分别改变节点和锚点的运动速度(v_{max}, s_{max}), 这里设定三对值($0.2r, 0$)、($r, 0$)、(r, r), 由 MCL 算法得到如图 1。

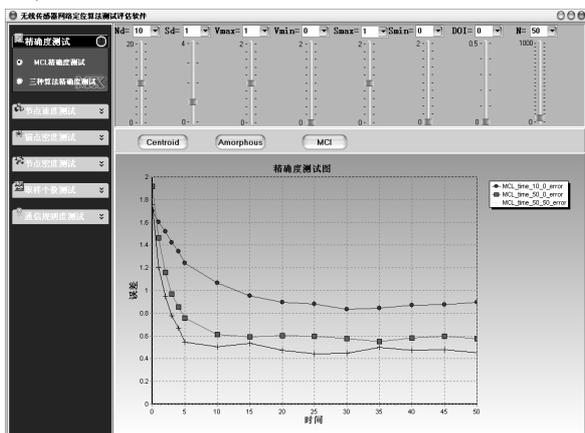


图 1 MCL 精确度测试

由图 1 可知: ① MCL 算法定位分为 2 个阶段: 初始化阶段和稳定阶段。初始化阶段随着节点观察信息的增多, 误差逐渐减少。稳定阶段节点移动带来的不确定达到一个相对的平衡, 误差上下变化的幅度较小。② 节点和锚点的运动速度越快, 误差达到稳定阶段的趋势也越快, 而且误差的值越小。

设定 $n_d=10, s_d=1, v_{max}=s_{max}=r$, 分别测试

MCL、质心和 Amorphous 算法的误差得到如图 2。

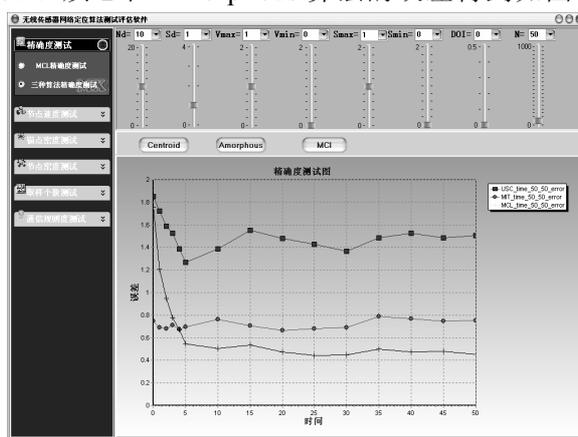


图 2 3 种算法的精确度测试

由图 2 可知: 和 MCL 算法相比, 质心和 Amorphous 算法对时间的敏感度低, 不能随时间的推移误差迅速达到稳定。

2) 节点速度结果评估

改变节点的速度, 分别取($s_d=1, s_{max}=v_{max}$), ($s_d=1, s_{max}=s_{min}=r$), ($s_d=2, s_{max}=v_{max}$), ($s_d=2, s_{max}=s_{min}=r$) 4 组对 MCL 算法进行测试得到如图 3。

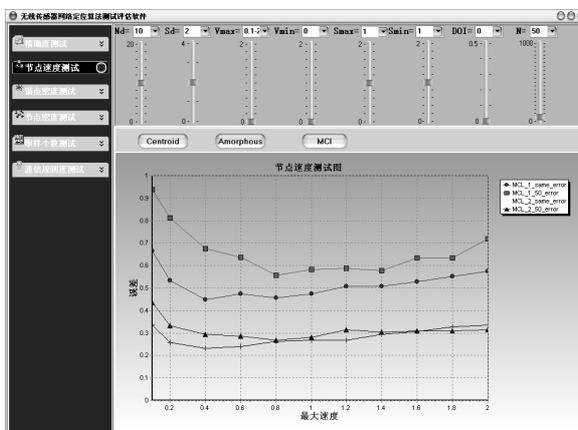


图 3 节点速度测试

当速度增加时, 预测阶段节点的概率分布区域变大降低定位的准确度, 但过滤阶段由于观察信息增多提高定位的精确度。由图 3 可知: ① 当锚点和节点的速度相等, 误差在 $0.1r \sim 0.3r$ 间是下降的, 其后逐步上升。② 当锚点匀速运动时, 误差随节点速度的增加而增加。③ 改变节点的速度相当于改变节点的通信间隔, 速度在 $0.4r \sim r$ 间的增加对误差没有太大影响, 即在速度较低的网络中, 可以增大节点通信间隔从而降低通信代价。

3) 锚点密度结果评估

增加锚点的密度可以提高定位误差, 但提高网络配置的代价。改变锚点的密度 s_d , 并设 $n_d=10$ 、

$v_{\max} = s_{\max} = 0.2r$ ，分别用质心、Amorphous 和 MCL 算法进行误差分析得到如图 4。

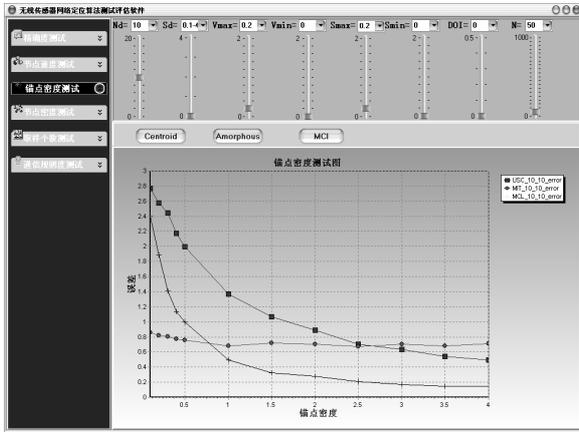


图 4 锚点密度测试

由图 4 可知：① 质心和 MCL 算法对锚点的密度敏感度高，随着锚点密度的增加误差迅速下降，Amorphous 算法对锚点的密度敏感度低。② 当锚点密度大于 1 时，MCL 的误差最小。

4) 节点密度结果评估

改变节点的密度，设 $s_d=1$ 、 $v_{\max} = s_{\max} = 0.2$ ，分别用质心、Amorphous 和 MCL 算法进行误差分析得到如图 5。

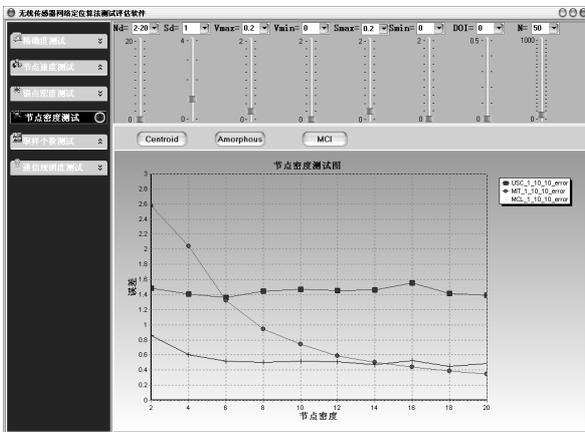


图 5 节点密度测试

由图 5 可知：① 质心和 MCL 算法对节点密度的敏感度低，而 Amorphous 算法对节点的敏感度高。② MCL 算法需要阀门节点密度值就可以使节点从邻居节点接收二跳信息，而且需要较少的邻居节点就足够了。而 Amorphous 算法则需要较高的节点密度，当节点密度小于 6 时，定位误差较差；节点密度大于 15 时，定位较准确，这是因为网络节点的密度对跳数影响较大。

5) 取样个数结果评估

改变取样个数，分别取

$(s_d=1, v_{\max} = s_{\max} = 0.2r)$ ， $(s_d=1, v_{\max} = s_{\max} = r)$ ， $(s_d=2, v_{\max} = s_{\max} = 0.2r)$ ， $(s_d=2, v_{\max} = s_{\max} = r)$ 4 组对 MCL 算法进行测试得到如图 6。

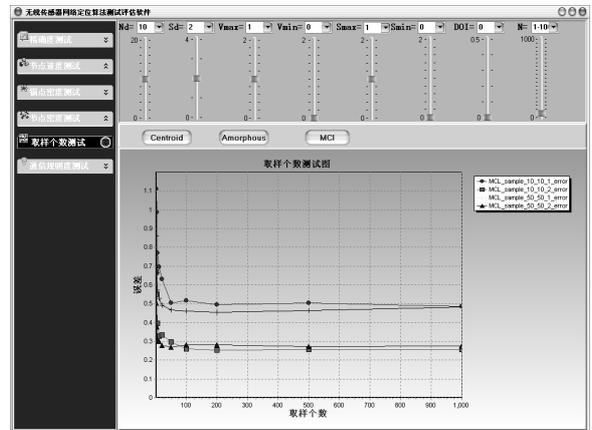


图 6 取样个数测试

由图 6 可知：① 开始误差曲线下落较快，因为较少的取样个数不能很好地反映位置的概率分布，当取样个数达到 50 时，误差曲线达到稳定状态，只有取样个数达到 1000 时定位误差有所减少。② 当锚点密度较大时，较少的取样个数就可以达到较高的准确度。

6) 通信规则度结果评估

实际节点的通信半径因受各种环境的影响不可能是某一定量 r ，而是在一定范围内随机变化的，可以用规则度 DOI 表示半径变化的不规则性。例如 $DOI=0.1$ ，表示通信半径在 $[0.9r, 1.1r]$ 区间随机变化。改变节点的通信规则度 DOI ，设 $n_d=10$ ， $s_d=1$ ， $v_{\max} = s_{\max} = 0.2$ ，分别用质心、Amorphous 和 MCL 算法进行误差分析得到如图 7。

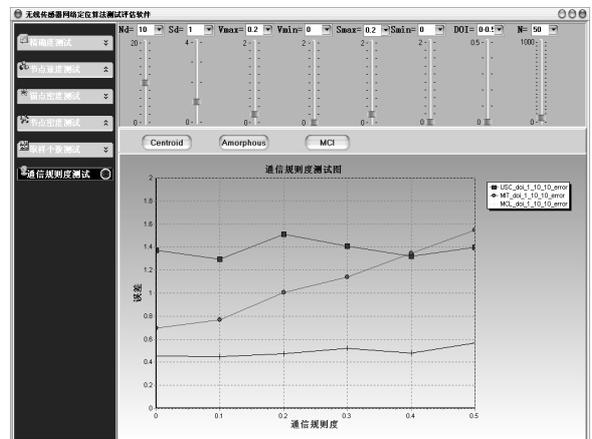


图 7 通信规则度测试

由图 7 可知：质心和 MCL 算法不是严格受规则度影响，但 Amorphous 算法随着规则度的增加定

位误差也随着增大。这是因为节点通信半径的变化对 Amorphous 算法中的跳数影响较大, 通信半径不规则性给 Amorphous 算法带来积累的误差随着跳数的增加而增加。

3 结论

1) 通过对 3 种算法比较, MCL 算法误差最小。2) 在节点速度测试中, 速度在 0.4 ~ 1 间对误差没有太大的影响, 所以在配置速度较慢的网络时, 可以延长节点的通信时段从而降低通信代价节约能量。3) 当锚点密度大于 1 时, MCL 算法的误差小且变化幅度小, 在无线传感器网络配置时可优先选

(上接第 28 页)

表 1 指挥员博弈中的参数估计值

参数	策略博弈			
	无经验		有经验	
	老练的 EWA	EWA	老练的 EWA	EWA
ϕ	0.44	0.00	0.29	0.22
δ	0.78	0.90	0.67	0.99
κ	0.00	0.00	0.04	0.00
α	0.24	0.00	0.77	0.00
α'	0.00	0.00	0.41	0.00
d	0.16	0.13	0.15	0.11
LL (内)	-2 095.32	-2 155.09	-1 908.48	-2 128.88
LL (外)	-968.24	-992.47	-710.28	-925.09
\bar{p} (内)	0.06	0.05	0.07	0.05
\bar{p} (外)	0.07	0.07	0.13	0.09

在表 1 中, 给出了更符合实际战争中出现的“老练”指挥员模型的参数估计, 既包括无经验的指挥员, 也包括经验丰富的指挥员。在适应性 EWA 模型 ($\alpha=0$) 中, δ 和 κ 的估计值分别接近于 1 和 0, 因此, 在指挥员仅依靠一股信念作出决策时进行限制是合理的。当加入经验丰富的“老练”后, 老练参与者的估计比例在无经验和富有经验的参与者中的估计值分别是 24%和 77%。这种由经验而导致的百分比上升, 意味着存在着“对学习的学习”, 即经验丰富的指挥员可以更好的从过去的历史经验中得到学习, 通过了解和熟悉作战对手情况, 可以改善自己的决策行为, 制定出针对对方的更科学、有效的决策。

4 结论

通过构建基于 EWA 的指挥员学习模型, 可以有效解决指挥员因缺乏成功历史经验而影响作战决策质量的问题。

参考文献:

[1] 刘树海. 作战行动学[M]. 北京: 解放军出版社, 2009.

择此密度。4) 当节点密度大于 6 时, MCL 算法的误差小且变化幅度小, 在无线传感器网络配置时可优先选择此密度。5) 当取样个数大于 50 时, MCL 算法的误差小且变化幅度小, 取样个数一般取 50。

参考文献:

[1] 孙利民, 李建中, 等. 无线传感器网络[M]. 北京: 清华大学出版社, 2005.

[2] 郑相全, 等. 无线自组网技术实用教程[M]. 北京: 清华大学出版社, 2004.

[3] 杨毅, 罗德超. 多尺度分析技术在无线传感器网络节点定位中的应用[J]. 四川兵工学报, 2010, 31(4): 93-96.

[2] 刘卫国. 作战指挥行为学[M]. 北京: 解放军出版社, 2005.

[3] 韩志明. 作战决策行为研究[M]. 北京: 国防大学出版社, 2005.

[4] (美) 科林·凯莫勒. 行为博弈—对策略互动的实验研究[M]. 北京: 中国人民大学出版社, 2006: 312-329.

[5] 梁开卷. 决策模式选择的权变性[J]. 领导科学, 1996.

[6] 李超, 刘安惠, 等. 军事行为科学[M]. 北京: 军事谊文出版社, 1998.

[7] (美) 朱·弗登伯格. 博弈学习理论[M]. 北京: 中国人民大学出版社, 2004.

(上接第 30 页)

1) 敌人不会从机场撤离, 敌人将采取坚守的行动策略;

2) 我方应该采取破坏该机场的行动, 而不是夺取该机场, 也不是坐视不管。

3 结语

数字化炮兵作战行动根植于军事行动的复杂系统当中, 具有较大的弹性和不确定性。模糊亚对策理论以量化的思维模式, 综合考虑各种因素对数字化炮兵作战行动的影响, 局势偏好排序合理, 突出己方作战行动受敌方指挥策略的影响, 解决了以往数字化炮兵作战行动多预案间的矛盾性, 客观、全面反映了炮兵作战行动的特征和实际情况, 提高了数字化炮兵作战行动预案科学性, 为做好数字化炮兵行动预案、数字化炮兵行动评估具有借鉴意义。

参考文献:

[1] 陈之宁. 模糊数学及其军事应用[M]. 北京: 海潮出版社, 2003.

[2] 袁宏伟, 张扬. 灰色局势决策理论在炮兵火力计划优选中的运用[J]. 指挥控制与仿真, 2008(6): 20-22.