

一种通过节点序寻优进行贝叶斯网络结构学习的算法

刘彬 王海羽 孙美婷 刘浩然* 刘永记 张春兰

(燕山大学信息科学与工程学院 秦皇岛 066004)

(燕山大学河北省特种光纤与光纤传感重点实验室 秦皇岛 066004)

摘要: 针对 K2 算法过度依赖节点序, 遗传算法节点序寻优效率差的问题, 该文提出一种直接对节点序进行评分搜索的贝叶斯结构学习算法。该算法以 K2 算法为基础, 首先通过计算支撑树权重矩阵, 构建能够定量评价节点序的适应度函数。然后通过提出混合交叉策略和孤立节点处理机制, 同时利用动态学习因子和倒置变异策略, 提升遗传算法节点序寻优的性能。最后将得到的节点序作为 K2 算法的先验知识得到最优贝叶斯网络结构。仿真结果表明, 该方法解决了 K2 算法依赖先验知识的问题, 相比于其它优化算法, 评分值平均增加了 13.11%。

关键词: 贝叶斯网络结构; 节点序搜索; 节点序适应度函数; K2 算法

中图分类号: TP18

文献标识码: A

文章编号: 1009-5896(2018)05-1234-08

DOI: 10.11999/JEIT170675

Learning Bayesian Network Structure from Node Ordering Searching Optimal

LIU Bin WANG Haiyu SUN Meiting LIU Haoran LIU Yongji ZHANG Chunlan

(School of Information Science and Engineering, Yanshan University, Qinhuangdao 066004, China)

(The Key Laboratory for Special Fiber and Fiber Sensor of Hebei Province, Yanshan University, Qinhuangdao 066004, China)

Abstract: The performance of the K2 algorithm depends on node ordering heavily, and the genetic algorithm can not find the node ordering effectively. For these problems, a new Bayesian structure learning algorithm, named NOK2 (Node Ordering searching for K2 algorithm), is proposed to solve the Bayesian structure learning problem by searching node ordering directly. According to the requirements of K2 algorithm based on prior knowledge and the weight matrix of spanning tree, the fitness function for quantitative evaluation of node ordering is established. The genetic algorithm is redesigned by a new method combines the dynamic learning constants, the hybrid crossover strategy, the inverted mutation strategy and the isolated node processing, so that the algorithm can find the node order of the highest fitness value, and this node sequence is taken as a prior knowledge of the K2 algorithm to obtain the optimal Bayesian network structure. Compared with other optimization algorithms, experimental results indicate that the NOK2 algorithm can significantly increase nearly 13.11% in the scoring metric values.

Key words: Bayesian network structure; Node ordering search; Fitness function of node sequence; K2 algorithm

1 引言

贝叶斯网络(Bayesian Network, BN)是人工智能算法领域重要的理论工具, 通过有向无环图和条件概率表描述变量之间的因果关系以及数据集的特征^[1]。被广泛地应用在自然语言处理、临床医学、机器视觉、数据挖掘、故障诊断等领域^[2-4]。

贝叶斯网络学习主要包括结构学习、参数学习和推理。其中, 贝叶斯网络结构的准确度直接影响参数学习和推理结果从精度, 但是通过数据学习贝叶斯网络结构是 NP-hard^[5]。国内外专家学者提出了

许多从数据中搜索贝叶斯结构的方法, 其中最常见的是打分搜索法。然而这种方法还存在许多问题, 如随着变量数量的增加, 搜索空间会呈指数增长^[6]。如果不同的结构具有相同的独立集会被认为是似然等价, 对于似然等价的结构进行评分, 得到的结果会基本相同, 导致无法推断变量之间准确的因果关系^[7]。

K2 算法由于有节点序的约束, 可以有效避免似然等价的问题, 并且在运行速度和准确度上都优于大部分经典算法。然而大多数情况下节点序是未知的, 通常需要根据专家知识来确定, 专家知识的差异性较大无法保证客观性和准确性, 而且当节点数较多时难以实现。为此, 研究人员提出了许多解决方案, 文献[8]中提出了使用条件独立测试查找节点排序信息的方法, 但该算法时间复杂度较高且无法

收稿日期: 2017-07-07; 改回日期: 2017-11-29; 网络出版: 2018-01-23

*通信作者: 刘浩然 liu.haoran@ysu.edu.cn

基金项目: 国家自然科学基金(51641609)

Foundation Item: The National Natural Science Foundation of China (51641609)

定量评价节点序。文献[9]中提出了使用条件频率的方法确定 K2 算法的节点排序, 该算法不需要复杂的搜索策略, 有效地降低了时间复杂度, 但是对数据集的质量要求较高, 不易得到准确的模型; 文献[10]将最大支撑树和蚁群算法进行结合, 提出 MWST-ACO-K2(MAK)算法可对节点序进行搜索, 但该算法需要将得到的节点序代入 K2 算法得到网络结构之后才能进行评分, 导致其运行时间过长。

本文提出一种直接对节点序进行评分搜索的 NOK2(Node Ordering searching for K2)算法, 解决了 K2 算法无法定量地评价节点序的问题, 可以在不借助专家知识的情况下得到较为精确的贝叶斯网络结构。时间复杂度分析表明 NOK2 算法的时间复杂度优于文献[9]的算法, 仿真结果表明, NOK2 算法对数据量的要求较低, 在小样本情况下依然可以得到较为准确的结构, 而且在运行速度, 结果精度方面优于大部分算法。

2 NOK2 算法构建

NOK2 算法首先对最大支撑树^[11]定向得到支撑树权重矩阵, 结合 K2 算法对先验知识的要求, 建立能对节点序定量评分的适应度函数。之后通过动态学习因子、倒置变异策略以及提出混合交叉策略对遗传算法^[12]易陷入局部最优且运行速度慢等问题进行优化, 用于节点序搜索, 结合本文提出的孤立节点处理机制得到最优节点序, 将其作为先验知识代入 K2 算法, 得到最优贝叶斯网络结构。

2.1 支撑树权重矩阵

设 \mathbf{B} 是一个以集合 \mathbf{X} ($\mathbf{X} = \{X_1, X_2, \dots, X_n\}$) 为节点的贝叶斯网络结构, \mathbf{D} 为训练数据。根据式(1)计算任意两节点之间的互信息^[13], 利用式(2)得到权重矩阵 \mathbf{W} 。

$$I(X_i; X_j) = \sum_{X_i, X_j} P(X_i, X_j) \lg \frac{P(X_i, X_j)}{P(X_i)P(X_j)} \quad (1)$$

$$\mathbf{W}(X_i; X_j) = I(X_i; X_j) \quad (2)$$

从集合 \mathbf{X} 中选择任意节点 X_i 作为起始点, 令集合 $\mathbf{V} = \{X_i\}$, 从集合 $\mathbf{X} - \mathbf{V}$ 中找到另一节点 X_j 使得节点 X_j 到 \mathbf{V} 中任意一个节点的权值 $\mathbf{W}(X_j, v)$ ($v \in \{V\}$) 最大, 用无向边连接 v, X_j 。此时将节点 X_j 也加入集合 \mathbf{V} ; 以此类推, 直到 $\mathbf{V} = \mathbf{X}$ 时, 便可得到权值之和 $W(t)$ 为最大时所对应最大支撑树结构。 $W(t)$ 如式(3)所示。

$$W(t) = \sum_{X_i \in \mathbf{V}, X_j \in \mathbf{X} - \mathbf{V}} \mathbf{W}(X_i; X_j) \quad (3)$$

若在最大支撑树结构中节点 X_i, X_j 之间存在无

向边, 则两节点互为潜在父节点。从而得到最大支撑树中每个节点的初始潜在父节点集合 $\{P(i)\}$ ($i = 1, 2, \dots, n$)。将 $\{P(i)\}$ 中所有元素 $X_{P_{i1}}, X_{P_{i2}}, \dots, X_{P_{ik}}$ 分别作为 X_i 的父节点构建子结构, 通过式(4), 式(5)求得子结构的对数似然值, 判断子结构与训练数据的匹配程度, 得到每个子结构的得分 $\{S_1, S_2, \dots, S_C\}$ 。

$$S = \sum_{i=1}^n \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} m_{ijk} \lg \theta_{ijk} \quad (4)$$

$$\theta_{ijk} = \begin{cases} m_{ijk} / \sum_{k=1}^{r_i} m_{ijk}, & \sum_{k=1}^{r_i} m_{ijk} > 0 \\ 1/r_i, & \text{其它} \end{cases} \quad (5)$$

式中, m_{ij} 表示数据中满足 X_i 父节点组合取值为 j 的样本数; m_{ijk} 表示数据中满足 X_i 父节点组合取值为 j 且 $X_i = k$ 的样本数; r_i 表示节点 X_i 共有 r_i 种取值; q_i 表示节点 X_i 的父节点 $\pi(X_i)$ 取值共有 q_i 种组合, X_i 无父节点时 $q_i = 1$ 。

从 $\{S_1, S_2, \dots, S_C\}$ 中找出得分最高的子结构 $(X_i, X_{P_{ij}})$, 将子结构中新加入的节点 $X_{P_{ij}}$ 添加到 X_i 的父节点集合 $\{P_1(i)\}$ 中, 利用式(6)更新初始潜在父节点集合 $\{P(i)\}$ 。

$$\{P(i)\} = \{P(i)\} - \{P_1(i)\} \quad (6)$$

以更新后 $\{P(i)\}$ 中的节点分别作为 X_i 的父节点, 加入 $(X_i, X_{P_{ij}})$ 中构建新的子结构。以此类推, 按子结构评分增大的原则, 直到子结构的评分不在增加, 得到 X_i 的潜在父节点集合 $\{P_1(i)\}$ 。对每个节点做类似处理, 最终得到所有节点的父节点集合 $\{P_1\}$ 。通过有向边连接 $\{P_1\}$ 中的父节点与子节点, 得到初步定向的最大支撑树结构。其中有可能存在两节点互为父节点, 即双向边的情况。首先去除结构中的双向边, 利用式(4)计算评分。然后将所有被删除掉的双向边以子结构评分依次增大的趋势, 全部转化为单向边, 得到定向后的最大支撑树 \mathbf{DT} , 将有向边的对应位置填入权值, 利用式(7)得到支撑树权重矩阵 \mathbf{T}_{MI} , 其中非零实数为两节点之间互信息, 表示节点 i 是节点 j 的父节点。

$$\mathbf{T}_{\text{MI}}(X_i, X_j) = \begin{cases} \mathbf{W}(X_i, X_j), & \mathbf{DT}(X_i, X_j) = 1 \\ 0, & \text{其它} \end{cases} \quad (7)$$

2.2 适应度函数

K2 算法的基本思想是从空网络开始, 根据节点序及评价网络结构模型优劣的测度函数, 选择使后验结构概率最大的节点作为该节点的父节点。依次遍历所有的节点, 逐步为每个变量添加最佳父亲节

点, 得到最优网络结构^[4]。节点序直接影响 K2 算法的精确度, 如果每个节点的父节点都排在其子节点之前, 则认为该节点序适合作为 K2 算法的先验知识。

针对 K2 算法需要由专家知识给出节点序, 之前的改进算法文献无法定量评价节点序优劣程度的问题, 本节基于支撑树权重矩阵结合 K2 算法所需节点序的特点, 建立节点序的适应度函数。

设 $\{O_1, O_2, \dots, O_n\}$ 是网络节点集合, $(O_{\pi_1}, O_{\pi_2}, \dots, O_{\pi_n})$ 为节点序, 其中 $\pi_1, \pi_2, \dots, \pi_n$ 为 $1, 2, \dots, n$ 的一个置换。利用式(8)得到第 i 个节点 O_{π_i} 与其后所有节点在 \mathbf{T}_{MI} 中的对应关系之和, 将式(8)代入式(9)可得到节点序的适应度函数 $\text{Fit}(\mathbf{O})$:

$$\text{Fit}(O_{\pi_i}) = \sum_{j=i+1}^n \mathbf{T}_{\text{MI}}(O_{\pi_i}, O_{\pi_j}) \quad (8)$$

$$\text{Fit}(\mathbf{O}) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n \mathbf{T}_{\text{MI}}(O_{\pi_i}, O_{\pi_j}) \quad (9)$$

节点序的适应度函数表示在节点序中, 每个节点与其后面的所有节点在矩阵 \mathbf{T}_{MI} 中的对应关系之和, 度函数的值越大表示节点序中父节点在子节点之前的情况越多。故该适应度函数可以定量评价任意节点序是否适用于 K2 算法。

2.3 节点序寻优

节点序寻优问题与旅行商问题 (Traveling Salesman Problem, TSP)^[15]类似, 从任意节点开始, 遍历其余各节点得到节点序, 搜寻适应度函数值最大的节点序, 随着节点个数的增加, 节点序的数量会呈指数增长出现组合爆炸的问题。NOK2 算法通过构建动态学习因子, 利用倒置变异策略以及提出混合交叉策略对遗传算法进行改进, 使算法能够快速对节点序全局搜索。

设 N_{max} 条节点序构成一个种群 $\mathbf{N}(t)$, $\mathbf{N}(t) = \{N^1(t), N^2(t), \dots, N^n(t)\}$ 。 $N^i(t)$ $i = (1, 2, \dots, n)$ 表示第 t 次迭代时, 种群中的第 i 条节点序。种群更新时, 以适应度函数为衡量标准, 通过变异操作, 或与目标节点序进行交叉操作搜索到适应度较高的节点序, 并将其以个体最优 $P_{\text{best}}^i(t)$ $i = (1, 2, \dots, n)$ 和全局最优 $G_{\text{best}}(t)$ 记录。其中 $P_{\text{best}}^i(t)$ 表示第 i 条节点序到第 t 次迭代时, 搜索到的适应度值最高的节点序; $G_{\text{best}}(t)$ 表示到第 t 次迭代时, 种群搜索到的适应度最高的节点序, 分别利用式(10)、式(11)更新。

$$P_{\text{best}}^i(t) = \begin{cases} P_{\text{best}}^i(t-1), & f(N^i(t+1)) < f(P_{\text{best}}^i(t-1)) \\ N^i(t+1), & f(N^i(t+1)) \geq f(P_{\text{best}}^i(t-1)) \end{cases} \quad (10)$$

$$G_{\text{best}}(t) = \left\{ P_{\text{best}}^i(t) \mid f(P_{\text{best}}^i(t)) = \max \left\{ f(P_{\text{best}}^1(t)), f(P_{\text{best}}^2(t)), \dots, f(P_{\text{best}}^n(t)) \right\} \right\} \quad (11)$$

NOK2 算法节点序寻优主要分为种群初始化, 种群迭代, 输出最优节点序 3 个步骤。其中种群迭代包括倒置变异操作, 混合交叉操作以及更新最优值。种群初始化时, 随机生成 n 条节点序 $N^i(0)$ 其中 $i = (1, 2, \dots, n)$, 并将 $N^i(0)$ 作为初始个体最优节点序 $P_{\text{best}}^i(0)$, 并根据式(11)得到初始全局最优节点序 $G_{\text{best}}(0)$ 。

第 t 次迭代时, 对种群中的每一条节点序 $N^i(t)$ 进行倒置变异操作, 设第 i 条节点序为 (o_1, o_2, \dots, o_n) , 从中随机选择一段作为变异区域 $(o_i, o_{i+1}, \dots, o_j)$, 将区域内的元素倒置排列, 得到变异之后的节点序 $N_m^i(t)$, 具体操作如式(12)所示。

$$\begin{cases} N^i(t) = 12345678 \\ (o_i, o_{i+1}, \dots, o_j) = 345 \end{cases} \longrightarrow N_m^i(t) = 12543678 \quad (12)$$

式中, $N^i(t)$ 为原节点序, $N_m^i(t)$ 为倒置变异操作之后的节点序。利用式(13)更新。

$$N^i(t) = \begin{cases} N^i(t), & f(N_m^i(t)) < f(N^i(t)) \\ N_m^i(t), & f(N_m^i(t)) \geq f(N^i(t)) \end{cases} \quad (13)$$

利用式(14)计算动态因子 S , 种群中每条节点序以概率 S 选择 $G_{\text{best}}(t)$ 作为目标节点序 $\text{PG}^i(t)$, 以概率 $1 - S$ 选择 $P_{\text{best}}^i(t)$ 作为目标节点序 $\text{PG}^i(t)$ 。将种群中的每条节点序 $N^i(t)$ 分别与其对应的目标节点序进行混合交叉操作。

$$S = t / T_{\text{max}} \quad (14)$$

式中, T_{max} 为最大迭代次数; t 为当前迭代次数。动态因子增加了算法前期与 $P_{\text{best}}^i(t)$ 交叉的概率, 后期与 $G_{\text{best}}(t)$ 交叉的概率, 即增强算法前期全局搜索能力和后期局部搜索能力。

混合交叉操作时, 从 $\text{PG}^i(t)$ 中随机选择一段作为交叉区域 $(o_i, o_{i+1}, \dots, o_j)$, 删除 $N^i(t)$ 中与交叉区域相重复的元素, 得到删除交叉区域元素的节点序 $N_d^i(t)$ 如式(15)所示。

$$\begin{cases} \text{PG}^i(t) = 28176453 \\ N^i(t) = 12345678 \end{cases} \longrightarrow N_d^i(t) = 12358 \quad (15)$$

找到交叉区域 $(o_i, o_{i+1}, \dots, o_j)$ 两端元素 o_i 和 o_j 在 \mathbf{T}_{MI} 中对应的相邻最大权值 $\mathbf{T}_{\text{MI}}(x, o_i)$ 与 $\mathbf{T}_{\text{MI}}(o_j, y)$ 。当交叉区域元素包含 x, y 中任一元素或 $\mathbf{T}_{\text{MI}}(x, o_i) = \mathbf{T}_{\text{MI}}(o_j, y)$ 时, 区域位置内元素后移, 将 $(o_i, o_{i+1}, \dots, o_j)$ 直接填入 $N_d^i(t)$ 中交叉区域的相应位

置。当 x, y 都不在交叉区域中时，若 $\mathbf{T}_{MI}(x, o_i) > \mathbf{T}_{MI}(o_j, y)$ ，则将交叉区域元素加入到 $N_d^i(t)$ 中的元素 x 之后；若 $\mathbf{T}_{MI}(x, o_i) < \mathbf{T}_{MI}(o_j, y)$ ，则将交叉区域加入到 $N_d^i(t)$ 中的元素 y 之前，具体操作如式(16)所示，约束条件如式(17)所示。

$$\begin{cases} PG^i(t) = 28176453 \\ N^i(t) = 12345678 \longrightarrow \\ N_d^i(t) = 123 \quad 58 \end{cases}$$

$$\begin{cases} N^i(t+1) = 12376458, \mathbf{T}_{MI}(1,7) = \mathbf{T}_{MI}(4,8) \\ N^i(t+1) = 17642358, \mathbf{T}_{MI}(1,7) > \mathbf{T}_{MI}(4,8) \\ N^i(t+1) = 12357648, \mathbf{T}_{MI}(1,7) < \mathbf{T}_{MI}(4,8) \end{cases} \quad (16)$$

$$\text{s.t.} \begin{cases} \max \{ \mathbf{T}_{MI}(x,7) = \mathbf{T}_{MI}(1,7) \} \\ \max \{ \mathbf{T}_{MI}(4,y) = \mathbf{T}_{MI}(4,8) \} \end{cases} \quad (17)$$

交叉操作后，更新 $P_{best}^i(t)$ 和 $G_{best}(t)$ 。当迭代次数达到 T_{max} 时，将 $G_{best}(T_{max})$ 作为最优节点序。在输出最优节点序之前，需要对 \mathbf{T}_{MI} 中是否存在孤立节点进行判断。若节点 X_i 满足式(18)，则 X_i 为孤立节点。

$$\sum_{j=1, j \neq i}^n \mathbf{W}(X_i; X_j) = 0 \quad (18)$$

每个孤立节点会导致出现 n 条得分相同，代入 K2 算法却得到不同结构的节点序。若在节点序中孤立节点排在其子节点之后，会导致 K2 算法无法搜索到此节点，降低算法精确度。若 \mathbf{T}_{MI} 中存在孤立节点，将 $G_{best}(T_{max})$ 中的孤立节点置于节点序前端，作为输出节点序 ρ 。将 ρ 作为 K2 算法的先验知识，得到贝叶斯网络结构。

3 NOK2算法实现

3.1 算法流程

- (1)输入数据，计算支撑树权重矩阵 \mathbf{T}_{MI} ;
- (2)种群初始化，得到 $P_{best}^i(0) \quad i = (1, 2, \dots, n)$ 和 $G_{best}(0)$;
- (3)根据式(12)对每条节点序 $N^i(t)$ 进行倒置变异操作，根据式(13)更新种群;
- (4)利用式(14)计算动态学习因子 S ;

(5)根据动态因子 S 选择目标节点序与目标节点序进行混合交叉操作，更新 $P_{best}^i(t)$ $G_{best}(t)$;

(6)判断是否到达最大迭代次数 T_{max} ，若不成立，迭代次数 $t = t + 1$ ，转到步骤(5)。若成立，输出最优节点序 $G_{best}(T_{max})$;

(7)对节点序中的孤立节点进行处理，得到输出节点序 ρ ;

(8)将输出节点序 ρ 作为先验知识，运行 K2 算法，得到贝叶斯网络结构。

3.2 时间复杂度分析

假设 n 表示节点个数， m 为样本数量， p 为种群规模， It 为迭代次数，则计算权重矩阵的时间复杂度为 $O(mn^2)$ ，构建最大支撑树为 $O(n^2)$ ，对最大支撑树进行定向为 $O(2mn)$ ；K2 算法的时间复杂度为 $O(mn^2)$ ；节点序寻优的过程中，种群初始化的时间复杂度为 $O(np)$ ，适应度函数的复杂度为 $O(n(n-1)/2)$ ，记为 $O(T)$ ，则计算种群适应度值的时间复杂度为 $O(pT)$ ，更新种群极值的复杂度为 $O(p)$ ，变异交叉操作的时间复杂度分别为 $O(pn)$ 和 $O(2pn^2)$ 。因此，节点序寻优过程的时间复杂度为 $O(np) + O(pT) + O(p) + It[O(2pn^2) + 2O(pT) + O(np) + O(2p)] = O(It \times p \times n^2)$ 。NOK2 算法的时间复杂度为 $O(n^2(It \times p + 2m))$ 。

4 仿真实验结果与分析

仿真实验在 MATLABR2014a 软件环境下运行，操作系统为 Windows7, CPU 为 i3-3240 内存为 4 G。通过 3 组实验分别验证适应度函数的可行性，NOK2 算法的节点序寻优能力，以及 NOK2 算法用于贝叶斯网络结构学习的性能。

4.1 不同节点序仿真对比

利用 AISA^[6]网络，随机生成数据量 500, 1000, 3000, 5000 的数据集，构建标准节点序 $A_1 = \{4,5,1,3,6,8,2,7\}$ ，随机节点序 $A_2 = \{3,6,8,7,2,4,1,5\}$, $A_3 = \{1,2,3,8,6,7,5,4\}$ ，理论较差节点序 $A_4 = \{8,7,6,5,3,2,1,4\}$ 。分别作为 K2 算法的先验知识构建贝叶斯网络结构，表 1 为对每组数据集独立计算 BIC^[17]评分及适应度函数值(Fit)50 次

表 1 不同节点序在 ASIA 网络中的性能对比

节点序	500		1000		3000		5000	
	Fit	BIC	Fit	BIC	Fit	BIC	Fit	BIC
A_1	0.6633	-1129.4	0.6927	-2323.5	0.6453	-6769.7	0.6606	-11210
A_2	0.2714	-1134.4	0.3726	-2330.8	0.3412	-6782.8	0.3073	-11220
A_3	0.5163	-1131.6	0.4893	-2327.4	0.4766	-6772.8	0.4829	-11215
A_4	0.0000	-1136.5	0.0000	-2336.2	0.0000	-6784.4	0.0000	-11225

得到的数据。

由表 1 可看出：标准节点序的 BIC 评分明显优于随机节点序，故节点序的准确度直接决定了 K2 算法能否搜索到准确的贝叶斯网络结构；标准节点序的适应度函数值明显高于随机节点序，故本文构建的适应度函数可以有效评价节点序是否适合作为 K2 算法的先验知识。

4.2 算法收敛性仿真对比

利用 ASIA 网络，CAR 网络，ALARM^[18]网络分别生成 3000, 4000 和 5000 个训练样本的数据集，使用 NOK2，遗传算法进行节点序寻优仿真。对每组数据集独立运行算法 100 次，得到最优节点序的适应度值。遗传算法参考文献[19]将参数设置如下：交叉概率为 0.7，变异概率为 0.3。结果如表 2 所示，表中括号内为标准节点序的适应度值， $\mu \pm \sigma$ 为结果的均值和标准差，It 表示最优值出现的迭代次数。

由表 2 中数据可知：NOK2 算法的平均适应度函数值优于 GA 算法，且标准差平均低于 GA 算法 0.83，故在收敛值更加接近标准值的情况下，NOK2 收敛速度更快且不易陷入局部最优解。原因在于混合交叉策略以及动态学习因子降低了算法陷入局部最优的概率。倒置变异策略提升了算法收敛速度。为确保算法能在最大迭代次数内收敛到最优值，种群规模应随着网络节点的数量增加而扩大。

4.3 贝叶斯网络结构学习仿真对比

本节将 NOK2 算法与 Hill Climbing(HC)^[20],

Genetic Algorithm(GA)^[21], Greed Search(GS), MAK 进行对比，根据文献[21]设 GA 中交叉概率为 0.9，变异概率为 0.1，群规模为 50；根据文献[9]设 MAK 中信息素强度系数为 1，信息素蒸发系数为 0.7，启发式因子权重为 2，群规模为 50；根据 4.3 节的分析，NOK2 算法种群规模设为 50，最大迭代次数为 50。

通过 UCI 真实数据集^[22]Auto-Mpg, Nursery 对算法性能对比分析。根据文献[22]，将非二值变量以中位数划分并去除数据集中的缺失数据。对比 NOK2 算法，随机节点序的 K2 算法，MAK 算法的 BIC 得分及运行时间，对每组数据集独立运行 50 次。结果如表 3 所示。通过 ASIA 网络，CAR 网络生成数据集对算法进行仿真，对每组数据集独立运行算法 50 次，运行结果如表 4~表 7 及图 1，图 2 所示，图中错误边数为冗余边、丢失边、反转边之和。

由表 3 可知，NOK2 算法可在真实数据中搜寻到较为准确的结构；由表 4、表 5 可知，在 ASIA 网络中 NOK2 与 K2 算法相比，运行时间增加了 0.917 s，相比于 MAK, GS，运行时间平均缩短了 6.618 s, 12.609 s；CAR 网络中 NOK2 与 GA, MAK, HC 相比，运行时间平均缩短了 21.593 s, 38.882 s, 73.151 s。复杂度分析表明，NOK2 算法的适应度函数复杂度为 $O(n^2)$ ，低于多数算法使用的 BIC 评分函数 $O(mn^2)$ 。

由表 6、表 7 可知：由 NOK2 算法得到贝叶斯

表 2 算法测试结果

网络	种群规模	NOK2		GA	
		$\mu \pm \sigma$	It(s)	$\mu \pm \sigma$	It(s)
ASIA(0.6998)	30	0.6983 ± 0.0062	11.84	0.6601 ± 0.0468	36.55
	50	0.6997 ± 0.0000	7.82	0.6659 ± 0.0431	28.15
	100	0.6999 ± 0.0000	6.44	0.6896 ± 0.0209	18.14
CAR(0.2768)	30	0.2767 ± 0.0003	22.63	0.2678 ± 0.0157	34.22
	50	0.2768 ± 0.0000	15.00	0.2738 ± 0.0079	19.81
	100	0.2768 ± 0.0000	15.76	0.2750 ± 0.0099	15.00
ALARM(10.1083)	100	9.8868 ± 0.4219	134.21	8.5900 ± 0.5239	96.14
	200	9.9399 ± 0.3671	138.16	8.6968 ± 0.5021	110.26
	500	10.1067 ± 0.0507	124.93	9.0487 ± 0.4987	119.05

表 3 真实数据集对比

算法	时间(s)		评分	
	AutoMpg n=8	Abalone n=9	AutoMpg n=8	Abalone n=9
NOK2	0.71	1.24	-1361.14 ± 3.39	-13562.02 ± 6.45
K2	0.08	0.16	-1374.55 ± 0.00	-13575.71 ± 0.00
MAK	7.81	23.53	-1380.60 ± 11.42	-13587.77 ± 16.41

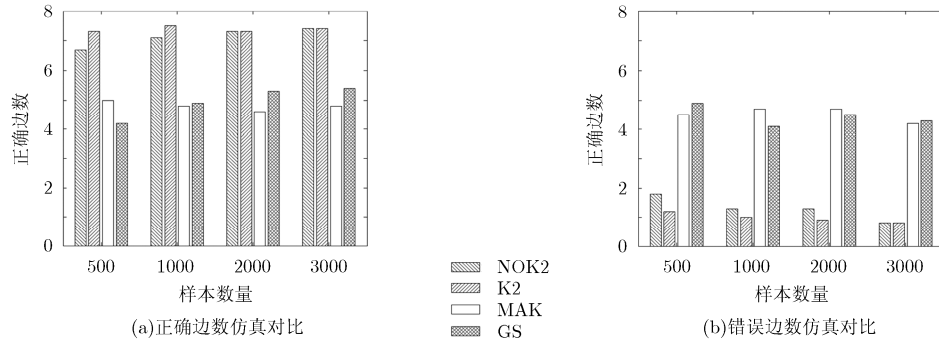


图 1 ASIA 网络中各算法精度对比

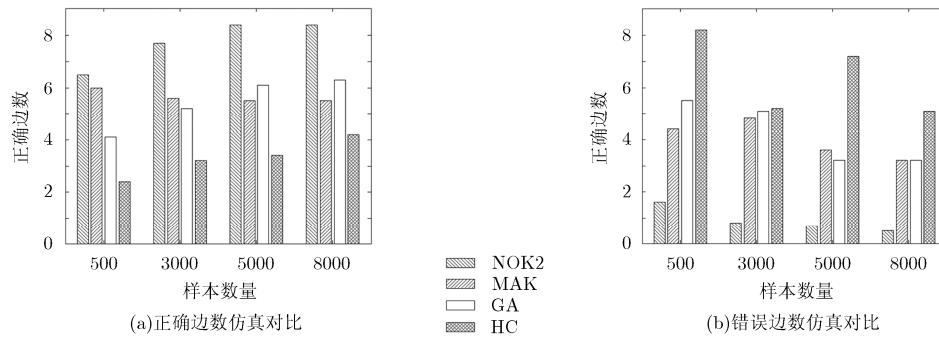


图 2 CAR 网络中各算法精度对比

表 4 不同算法在 ASIA 网络中的运行时间对比(s)

算法	样本数量			
	500	1000	2000	3000
NOK2	0.903	0.911	1.009	1.177
K2	0.066	0.072	0.098	0.096
MAK	7.102	7.318	7.729	8.322
GS	10.400	11.466	15.430	17.142

表 5 不同算法在 CAR 网络中的运行时间对比(s)

算法	样本数量			
	500	3000	5000	8000
NOK2	0.941	1.218	1.301	1.567
MAK	21.550	33.677	44.581	60.744
GA	17.102	21.512	25.204	27.584
HC	55.499	67.498	81.985	92.648

表 6 不同算法在 ASIA 网络中的 BIC 评分对比

算法	样本数量			
	500	1000	2000	3000
NOK2	-1163.1±0.05	-2265.9±0.03	-4481.2±1.41	-6854.1±0.34
K2	-1160.5±0.00	-2259.5±0.00	-4477.9±0.00	-6854.1±0.00
MAK	-2486.8±5.96	-2895.6±2.08	-5036.8±6.94	-7069.5±5.96
GS	-2284.1±0.16	-2930.5±0.67	-5304.9±6.98	-7101.3±8.51

网络结构的 BIC 评分接近 K2 算法且高于其他算法，标准差的平均值明显低于 MAK 等算法，BIC 得分相比于 MAK, GS, GA, HC 算法分别提升了 17.9%, 22.69%, 10.09%, 14.99%，相比于 K2 算法仅降低了 0.14%，基本可以达到 K2 算法的精度。

由图 1，图 2 可知，在没有先验知识的情况下，NOK2 算法可以达到 K2 算法的精度，且高于 GA, HC, MAK, GS 算法，表明算法有良好的贝叶斯网络结构学习性能。当数据量较小时，NOK2 算法的准确度低于 K2 算法，主要原因是较低的数据量会导致算法对最大支撑树定向的精度降低，影响支撑树权重矩阵。相比于文献[9]所提出的算法，NOK2 在小样本的情况下可以得到较为精确的网络结构。原因在于：NOK2 算法可以快速搜索到较为精确的节点序。孤立节点处理机制弥补了最大支撑树定向

表7 不同算法在CAR网络中的BIC评分对比

算法	样本数量			
	500	3000	5000	8000
NOK2	-3445.3±5.21	-21517±0.02	-33288±0.34	-53700±0.19
MAK	-3599.3±8.64	-23410±5.61	-35461±8.63	-69671±16.17
GA	-3775.4±16.31	-23713±9.80	-34752±18.51	-65613±66.91
HC	-4148.7±38.58	-23716±22.05	-36725±34.15	-71014±66.04

时会删除部分有效信息的缺陷,增加了最优节点序的可信度。

5 结束语

本文提出了一种通过对节点序进行评分搜索学习贝叶斯网络结构的算法(NOK2)。该算法通过构建节点序适应度函数实现对节点序定量评分,在遗传算法的基础上通过提出混合交叉策略和孤立节点处理机制,同时利用动态学习因子和倒置变异策略提升算法节点序寻优的性能。将最优节点序作为先验知识代入K2算法,得到最优贝叶斯网络结构。仿真实验表明,NOK2在不依赖任何先验知识的条件下,可以得到较为准确的节点序,与同类结构学习算法相比,该算法具有时间复杂度上的明显优势,且网络结构评分值平均增加了13.11%,有效解决了K2算法依赖专家知识获得节点序的问题。该算法将贝叶斯网络结构学习问题转化为与旅行商问题类似的节点序寻优问题,从而可以将解决旅行商问题的优化算法引入到贝叶斯网络结构学习问题中,为贝叶斯结构学习提供了一种新的思路。

参考文献

- [1] TIEN I and KIUREGHIAN A D. Algorithms for Bayesian network modeling and reliability assessment of infrastructure systems[J]. *Reliability Engineering & System Safety*, 2016, 156: 134-147. doi: 10.1016/j.res.2016.07.022.
- [2] 刘广怡,李鸥,宋涛,等.基于贝叶斯网络的无线传感网高效数据传输方法[J].*电子与信息学报*, 2016, 38(6): 1362-1367. doi: 10.11999/JEIT151027.
LIU Guangyi, LI Ou, SONG Tao, et al. Energy-efficiency data transmission method in WSN based on Bayesian network[J]. *Journal of Electronics & Information Technology*, 2016, 38(6): 1362-1367. doi: 10.11999/JEIT151027.
- [3] QIU H, WEI Z, LIU Y, et al. A Bayesian network meta-analysis of three different surgical procedures for the treatment of humeral shaft fractures[J]. *Medicine*, 2016, 95(51): e5464. doi: 10.1097/MD.0000000000005464.
- [4] 邓歆,孟洛明.基于贝叶斯网络的通信网告警相关性和故障诊断模型[J].*电子与信息学报*, 2007, 29(5): 1182-1186. doi: 10.3724/SP.J.1146.2005.01290.
- [5] DENG Xin and MENG Luoming. Bayesian networks based alarm correlation and fault diagnosis in communication networks[J]. *Journal of Electronics & Information Technology*, 2007, 29(5): 1182-1186. doi: 10.3724 /SP.J.1146.2005.01290.
- [5] CHICKERING D M. Learning Bayesian networks is NP-complete[J]. *Networks*, 1996, 112(2): 121-130. doi: 10.1007/978-1-4612-2404-4_12.
- [6] CARRIGER J F, MARTIN T M, and BARRON M G. A Bayesian network model for predicting aquatic toxicity mode of action using two dimensional theoretical molecular descriptors[J]. *Aquatic Toxicology*, 2016, 180: 11-24. doi: 10.1016/j.aquatox.2016.09.006.
- [7] ROH M C and LEE S W. Human gesture recognition using a simplified dynamic Bayesian network[J]. *Multimedia Systems*, 2015, 21(6): 557-568. doi: 10.1007/s00530-014-0414-9.
- [8] CHEN X W, ANANTHA G, and LIN X. Improving Bayesian network structure learning with mutual information based node ordering in the K2 algorithm[J]. *IEEE Transactions on Knowledge & Data Engineering*, 2007, 20(5): 628-640. doi: 10.1109/TKDE.2007.190732.
- [9] SONG K and KIM D W. An efficient node ordering method using the conditional frequency for the K2 algorithm[J]. *Pattern Recognition Letters*, 2014, 40(4): 80-87. doi: 10.1016/j.patrec.2013.12.021.
- [10] 刘浩然,孙美婷,李雷,等.基于蚁群节点寻优的贝叶斯网络结构算法研究[J].*仪器仪表报*, 2017, 38(1): 143-150. doi: 10.3969/j.issn.0254-3087.2017.01.019.
LIU Haoran, SUN Meiting, LI Lei, et al. Bayesian network structure learning algorithm based on ant colony optimization search optimal node ordering[J]. *Chinese Journal of Scientific Instrument*, 2017, 38(1): 143-150. doi: 10.3969/j.issn.0254-3087.2017.01.019.
- [11] KRUSKAL J B. On the shortest spanning subtree of a graph and the traveling salesman problem[J]. *Proceedings of the American Mathematical Society*, 1956, 7(1): 48-50. doi: 10.2307/2033241.
- [12] HU R S. A hybrid PSO-GA algorithm for job shop scheduling in machine tool production[J]. *International Journal of Production Research*, 2015, 53(19): 1-27. doi: 10.1080/00207543.2014.994714.
- [13] KPPPMAN R and WANG S. Mutual information based labelling and comparing clusters[J]. *Scientometrics*, 2017,

- 111(2): 1157–1167. doi: 10.1007/s11192-017-2305-2.
- [14] COOPER G F and HERSKOVITS E. A Bayesian method for the induction of probabilistic networks from data[J]. *Machine Learning*, 1992, 9(4): 309–347. doi: 10.1007/BF00994110.
- [15] LIN S and KERNIGHAN B W. An effective heuristic algorithm for the TSP[J]. *Operations Research*, 1973, 21(2): 498–516. doi: 10.1287/opre.21.2.498.
- [16] 刘广怡, 李鸥, 张大龙. 一种通过结构边界进行贝叶斯网络学习的算法[J]. 电子与信息学报, 2015, 37(4): 894–899. doi: 10.11999/JEIT140786.
- LIU Guangyi, LI Ou, and ZHANG Dalong. Learning Bayesian network from structure boundaries[J]. *Journal of Electronics & Information Technology*, 2015, 37(4): 894–899. doi: 10.11999/JEIT140786.
- [17] SCHWARZ G. Estimating dimension of a model[J]. *Annals of Statistics*, 1978, 6(2): 461–464. doi: 10.1214/aos/1176344136.
- [18] BEINLICH I A, SUERMONDT H J, CHAVEZ R M, *et al.* The ALARM monitoring mystem: A case study with two probabilistic inference techniques for belief networks[J]. *Lecture Notes in Medical Informatics*, 1989, 38: 247–256. doi: 10.1007/978-3-642-93437-7_28.
- [19] MAJUMDAR J and BHUNIA A K. Genetic algorithm for asymmetric traveling salesman problem with imprecise travel times[J]. *Journal of Computational & Applied Mathematics*, 2011, 235(9): 3063–3078. doi: 10.1016 /j.carm.2010.12.027.
- [20] TSAMARDINOS I, BROWN L E, and ALIFERIS C F. The max-min hill-climbing Bayesian network structure learning algorithm[J]. *Machine Learning*, 2006, 65(1): 31–78. doi: 10.1007/s10994-006-6889-7.
- [21] LARRAÑAGAL P, POZA M, YURRAMENDI Y, *et al.* Structure learning of bayesian networks by genetic algorithms: A performance analysis of control parameters[J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 1996, 18(9): 912–926. doi: 10.1109/34.537345.
- [22] NIE S, CAMPOS C P D, and JI Q. Efficient learning of Bayesian networks with bounded tree-width[J]. *International Journal of Approximate Reasoning*, 2016, 80: 412–427. doi: 10.1016/j.ijar.2016.07.002.
- 刘 彬: 男, 1953 年生, 教授, 博士生导师, 研究方向为无线传感器网络、工业故障检测及预测.
- 王海羽: 男, 1993 年生, 硕士生, 研究方向为群智能算法、贝叶斯网络、工业故障检测及预测.
- 孙美婷: 女, 1992 年生, 硕士生, 研究方向为贝叶斯网络故障及预测.
- 刘浩然: 男, 1980 年生, 教授, 博士生导师, 研究方向为无线传感器网络、工业故障检测及预测.
- 刘永记: 男, 1991 年生, 硕士生, 研究方向为贝叶斯网络故障及预测.
- 张春兰: 女, 1992 年生, 硕士生, 研究方向为工业故障检测及预测.