

基于抽样的因果图仿真推理算法研究

石庆喜

(重庆工商大学计算机学院 重庆 400067)

摘要 由于因果图的经典推理的计算复杂度是 NP 难的,因此其不便于推广和使用。基于因果图理论和 Monte Carlo 思想,提出了基于抽样的 A-R Sampling 和重要性抽样的因果图仿真推理算法。在故障诊断中的有效应用验证了方法是可行的。

关键词 因果图,仿真,重要性抽样

中图分类号 TP391.9 **文献标识码** A

Research of Simulation Reasoning Algorithm in Causality Diagram Based on Sampling

SHI Qing-xi

(College of Computer Science, Chongqing Technology and Business University, Chongqing 400067, China)

Abstract Because the causality diagram's reasoning calculation is a NP problem, it is inconvenient in its popularization and application. This paper presented two simulation reasoning algorithms including A-R sampling and importance sampling based on causality diagram. The effective application in fault diagnosis shows the method is feasible.

Keywords Causality diagram, Simulation, Important sampling

1 因果图介绍

动态因果图理论^[1,2,5,6,8]是在故障诊断的实际应用过程中,根据专家知识的特点以及工业领域故障自身的特殊性,在信度网基础上发展起来的一套方法。如图 1 所示,与信度网相比,其具有一些明显的优势:对网络的拓扑结构没有限制,采用直接因果强度而不是条件概率,避免了在给定知识时知识间的相关性问题,符合客观实际,更便于知识的表达;而且引进了逻辑门,使表达因果关系更加清楚、自然;在结构中引进了动态性,能根据在线情况的变化动态变换网络结构,使之更符合实际。经过近几年的持续研究,因果图理论已经发展成了一个能够处理离散变量和连续变量的混合因果图模型,并且在一些领域进入了实用阶段。

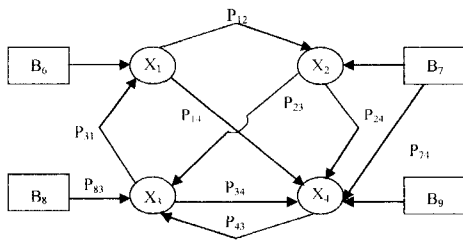


图 1 因果网

但是在经典的精确推理过程中,无法避免 NP-HARD 问题,而且整个推理过程复杂,例如,最终割集计算复杂度就达 $O(M!m)$ (基本事件数为 m , 中间事件节点数目为 M)^[1,5]。于是考虑到因果图模型的特点,提出了基于抽样的仿真近似推理算法。

2 Monte Carlo 法概述

基于仿真的 Monte Carlo 法^[3,4],首先通过采样技术对某概率分布进行采样来得到一组样本,然后对该组样本进行统计计算得到待求概率值。通常采样阶段和统计计算阶段完全分开。采样方法的目的是设计一套算法,以最快的速度产生满足统计计算要求的样本。Monte Carlo 法的主要思想是通过一组样本计算某函数的数学期望。其计算公式如下:

$$E(f(\cdot)) = \frac{\sum_{X_1, X_2, \dots, X_N} f(X_1, X_2, \dots, X_N) * P(X_1, X_2, \dots, X_N)}{\sum_{X_1, X_2, \dots, X_N} P(X_1, X_2, \dots, X_N)} \approx \frac{1}{M} * \sum_{t=0}^{M-1} f(X_1^t, X_2^t, \dots, X_N^t) \quad (1)$$

式(1)中需要计算函数 $f(X_1, \dots, X_N)$ 对于概率分布 $p(X_1, \dots, X_N)$ 的数学期望。如果直接计算,需要对随机变量 X_1, \dots, X_N 的所有可能组合状态进行求和(如式(1)中间部分)。采用 Monte Carlo 方法,只需要将函数 $f(X_1, \dots, X_N)$ 对所有的样本求平均,就可以得到该函数的数学期望。为了便于理解和表述,可以对上面的公式进行改写。在式(1)中假设随机变量 X_1, \dots, X_N 所有可能的组合构成了一个状态空间 Ω , 其中 ω 为该状态空间中的一个状态,定义 ω_t 为第 t 个样本的状态,且 $\omega_t \in \Omega$ 。另假设在 M 个样本中出现的的所有组合状态构成状态空间 Ω_1 , 且 $\Omega_1 \subseteq \Omega$ 。这样式(1)可变为:

$$E(f(\cdot)) = \sum_{\Omega} f(\omega) * P(\omega) \approx \sum_{\Omega_1} f(\omega) * \frac{Num(\omega)}{M} = \frac{1}{M} * \sum_{t=0}^{M-1} f(\omega_t) \quad (2)$$

式中,函数 $Num(\omega)$ 表示状态 ω 在样本中出现的次数, $Num(\omega)/M$

到稿日期:2011-10-10 返修日期:2011-12-30 本文受国家自然科学基金(60643006),重庆市自然科学基金(2010BB2259)资助。

石庆喜(1973-),男,博士,副教授,主要研究方向为人工智能、因果图理论,E-mail:shiqx@sohu.com.

则表示状态 ω 在样本中出现的频率。随着样本量的增加,当样本数趋近于无穷大时, $\Omega_n = \Omega$,且状态出现的频率将等于该状态的概率。可知 Monte Carlo 法在计算过程中主要以频率代替概率,以样本的算术平均代替数学期望。该方法的计算精度将随着样本量的增加而增加,换句话说,要达到越高的精度,对样本量的需求就越大。在概率推理中,边缘概率的计算是所有计算问题的基础。采用 Monte Carlo 法计算边缘概率时必须先定义一个函数 f 。假设先计算单个变量 $X_i = a$ 的边缘概率,其函数 f 的定义如下:

$$f(X_1, \dots, X_n) = \begin{cases} 1, & X_i = a \\ 0, & X_i \neq a \end{cases} \quad (3)$$

该函数表示当节点 X_i 的取值为 a 时函数 f 的值为 1,否则为 0。在该定义下,函数 f 的数学期望将表示边缘概率 $P(X_i = a)$,即:

$$\begin{aligned} E(f(\cdot)) &= \sum_{X_1, X_2, \dots, X_n} f(X_1, X_2, \dots, X_n) * P(X_1, X_2, \dots, X_n) \\ &= \sum_{X_1, X_2, \dots, X_n} P(X_1, \dots, X_i = a, \dots, X_n) = P(X_i = a) \end{aligned} \quad (4)$$

采用 Monte Carlo 法将式(4)近似为:

$$\begin{aligned} E(f(\cdot)) &= P(X_i = a) \approx \frac{1}{M} \sum_{i=1}^{M-1} f(X_1, \dots, X_n) \\ &= \frac{X_i = a \text{ 的样本数}}{M} \end{aligned}$$

可见采用该方法避开了指数式的求和运算量。计算多个变量的边缘概率的计算方式与此类似。

3 因果图仿真推理算法

3.1 A-R(Acceptance-Rejection) Sampling

因果图模型的本质是表达了在一个故障系统中,事件之间如何按照概率进行相互作用,例如图 2 所表达的因果图模型, A 和 B 的发生与否分别由自身的发生概率决定,而 A 和 B 作为 X 的原因节点对 X 节点的发生,分别独立地贡献概率值,且每个贡献都是简单相加的关系,即 $X = A \cup B = A + B$,由此来决定 X 节点的发生情况。模型实质描述了以下过程:在某个或某几个基本事件(对应于故障系统中的故障源)发生的情况下,模型按照专家所设定的因-果影响的概率大小,将故障的发生情况传递到各个中间事件(对应于故障系统中的现象)上,从而导致某些中间事件处于显示有故障的信号状态上。显然以上整个过程能够采用正向仿真的方法进行。

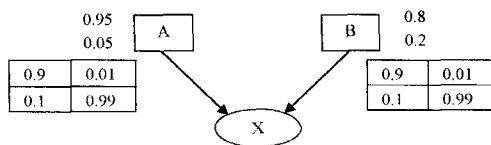


图 2 因果图例

根据因果图理论,可以得到如下定理。

定理 1 如果结点 X_1, \dots, X_k 状态已知, X_{k+1} 是一个结点,且:

$$desc(X_{k+1}) \setminus \{X_1, \dots, X_k\} = \phi$$

则有:

$$p(X_{k+1} | X_1, \dots, X_k) = p(X_{k+1} | pa(X_{k+1}))$$

其中, $desc(X_{k+1})$ 表示 X_{k+1} 子结点集合, $pa(X_{k+1})$ 表示 X_{k+1}

父结点集合。

根据定理 1,可以在给定其父结点状态下得到 X_{k+1} 的发生概率,从而进行抽样,得到如下推理算法。

算法 1

- (1) 首先给所有节点编上序号,并要求父节点的序号必须小于子节点的序号;
- (2) 然后按照序号大小对节点按如下原则进行采样。例如图 2 的序号分别为 A:1; B:2; X:3。
 - 1) 对于基本事件结点(例如 A),直接根据其概率,利用随机发生器确定该节点发生与否(或节点所处的状态);
 - 2) 对于其他节点,由于父节点集合的所有节点的状态都已经确定,因此可以直接确定该子节点的状态值;
 - 3) 对于证据节点,如果随机采样的值与证据不一样,则放弃此次采样,重新从序号最小的节点开始采样。
- (3) 经过以上两步,完成了一次采样,得到了一组与证据一致的样本,然后再重复第(2)步;
- (4) 利用这些样本,通过统计得到需要求取的概率值。

对图 2,如果证据结点是 $X=1$,按照算法 1

1) 确定采样顺序为 A, B, X。

2) 对 $A \sim (0.95, 0.05)$, 采样得到 $A=1$;

对 $B \sim (0.8, 0.2)$, 采样得到 $B=1$;

对 X, 有 $P(X \rightarrow 1) = \eta * (0.99 + 0.9) = \eta * 1.89$, $P(X \rightarrow 0) = \eta * (0.1 + 0.01) = \eta * 0.11$, 所以, $\eta = 0.5$, $P(X \rightarrow 1) = 0.945$, $P(X \rightarrow 0) = 0.055$ 。对 $X \sim (0.945, 0.055)$, 采样得到 $X=1$ 满足证据要求 $X=1$ 。

3) 于是得到一个采样结果(1, 1, 1)。

在步骤 2) 中对 X 采样,若得到 $X=0$, 结果与证据节点不一致,则放弃这次采样。

此种采样方式在采样证据节点时,如果结果与证据节点不一致,则放弃采样,这样会造成采样次数的急剧增加。图 2 中,若证据是 $X=2$,当采样得到 A 处于第 1 个状态($A=1$); $B=1$, 则为了产生和证据一致的样本,对 X 节点至少需要采样 $1/0.01=100$ 次,显然存在很大的效率问题。

但本算法在用于正向推理中,因为没有证据的影响,所以对正向推理还是很有效的。

3.2 重要性抽样算法

一个自然的想法是对证据结点不抽样,直接初始化为其证据值,但对其他结点(包括基本事件结点)抽样则必须基于如下表达式:

$$p(X_{k+1} | X_1 = x_1, \dots, X_k = x_k, E_1 = e_1, \dots, E_m = e_m)$$

这样产生如下问题,如图 2 所示,如果 X 是证据,并且初始化为其证据值, $X=x$, 这样 A 抽样依赖于 C, 即是说要根据 $p(A | X=x)$ 对 A 抽样,再考虑 B; A 依赖于 X 和 A, 即是说要根据 $p(A | X=x, A=a)$ 对 B 抽样,但 $p(A | X=x)$, $p(A | X=x, A=a)$ 都是未知的。

为以后的推导,我们引入: $dom(V)$ 定义为结点集合为 V 的状态空间, $dom(p)$ 定义为密度函数为 p 的概率空间。

作为式(4)的推广和大多数定理的结果,有如下结论:

定理 2

$$\sum_{x \in dom(p)} f(x) * p(x) = E_p(f(x)) \approx \frac{1}{n} \sum_{x \sim p} f(x) \quad (5)$$

定理 3 设 q 为概率密度,并且如果对 $\forall x \in dom(q) = dom(p)$ 有: $p(x) > 0 \Rightarrow q(x) > 0$, 则:

$$\begin{aligned} \sum_{x \in \text{dom}(p)} f(x) * p(x) &= \sum_{x \in \text{dom}(p)} f(x) * \frac{p(x)}{q(x)} * q(x) \\ &= E_p(f(x) * \frac{p(x)}{q(x)}) \\ &\approx \frac{1}{n} \sum_{x \sim q} f(x) * \frac{p(x)}{q(x)} \end{aligned} \quad (6)$$

称 q 为重要性函数, $w = p/q$ 为得分或权重。

令 $w' = c * w \propto p/q$, c 为常数, 如果 $x_1, x_2, \dots, x_n \sim q$, 有:

$$\begin{aligned} E_p(f(x)) &\approx \frac{1}{n} \sum_{i=1}^n f(x_i) * w(x_i) \\ &= \frac{1}{\sum_i w'(x_i)} \sum_{i=1}^n f(x_i) * w'(x_i) \end{aligned} \quad (7)$$

考虑在因果图中:

$$p(x|E) = \frac{p(X)}{p(E)} = \frac{\prod_{v \in V} p(X_v | Pa(X_v))}{p(E)} \quad (8)$$

令 $q_E(x) = \prod_{v \in V \setminus \text{dom}(E)} p(X_v | Pa(X_v))$

得到:

$$w(x) = \frac{p(x|E)}{q_E(x)} = \frac{\prod_{v \in \text{dom}(E)} p(X_v | Pa(X_v))}{p(E)} \quad (9)$$

根据式(7), 设

$$w'(x) = \prod_{v \in \text{dom}(E)} p(X_v | Pa(X_v))$$

可以计算出 $E_p(f(x))$ 的估计, 得到如下算法。

算法 2

- (1) 给所有除去证据的节点编上序号, 并要求父节点的序号必须小于子节点的序号。
- (2) 按照序号大小对节点按如下原则进行采样:
 - 1) 给所有证据节点 X 赋值为相应的证据值;
 - 2) 对于基本事件结点(例如 A), 直接根据其概率, 利用随机发生器确定该节点发生与否(或节点所处的状态);
 - 3) 对于所有非证据中间节点, 由于父节点集合的所有节点的状态已经确定, 因此可以直接确定该子节点的状态;
- 4) 按照

$$w(x) = \prod_{v \in \text{dom}(E)} p(X_v | Pa(X_v))$$

计算本次采样的得分。

- (3) 经过以上两步, 完成了一次采样, 得到了一组与证据一致的样本, 然后再重复第(2)步。
- (4) 利用这些样本, 按照算法 3, 可以得到需要求取的概率值。

算法 3

- (1) 令 q 为考察的目标结点状态的得分和, 初始化为 0。
- (2) 令 w_{total} 为所有抽样数据得分和, 初始化为 0。
- (3) 对所有的 $d \in D$, D 为算法 2 得到的抽样集合:

$$w_{\text{total}} = w_{\text{total}} + w(d)$$

对 d 中所考察结点的所有状态 t , 有

$$q(t) = q(t) + w(d)$$

- (4) 计算 $q(t)/w_{\text{total}}$, 即所考察结点的 t 状态的后验概率。

对图 2, 如果证据结点是 $X=1$, 按照算法 2:

- 1) 确定采样顺序为 A, B ;
- 2) 对 X 初始化为 1;
- 3) 对 $A \sim (0.95, 0.05)$, 采样得到 $A=1$;
- 对 $B \sim (0.8, 0.2)$, 采样得到 $B=1$;
- 4) 计算 $w = p(X=1 | A=1, B=1) = 0.945$, 得到采样数据 $(A, B, X) = (1, 1, 1)$, 其得分为 0.99。

根据算法 3, 如果采样得到 3 组数据: $(1, 1, 1) w = 0.945; (1, 0, 1) w = 0.5; (0, 1, 1) w = 0.5$, 则得到当前 $A=1$ 的后验概率近似为: $(0.945 + 0.5) / (0.945 + 0.5 + 0.5) =$

$0.74, A=0$ 的后验概率近似为 0.26。

4 实验结果

以清华大学北京核电模拟培训中心的全尺寸 950MW 核电模拟机^[7]为故障诊断对象, 建立该模拟机二回路典型故障诊断因果图(以下简称二回路故障诊断系统)。整个模型过大, 因此图 3 仅示出对蒸汽发生器系统故障有影响的部分。

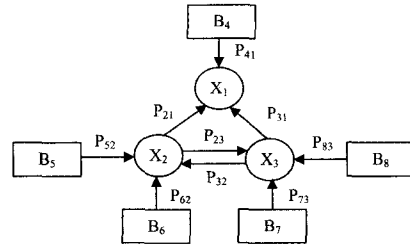


图 3 蒸汽发生器因果图

重要性抽样算法和精确推理算法的诊断效果比较如图 4 所示。当证据为 $X_2 X_3$, M 取值为 1000 时, 其误差为: $E = \sqrt{\sum_{i=1}^K (L_{B_i} - G_{B_i})^2} = 0.1127$, 其中 K 为待考察事件的总数。可以看出, 对于故障的定位(即明确故障源), 算法不会产生错误。而不同的采样次数对精度的影响, 以及对推理所需时间的影响如图 5 所示。

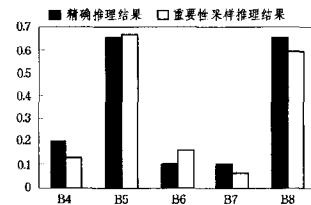


图 4 推理结果比较

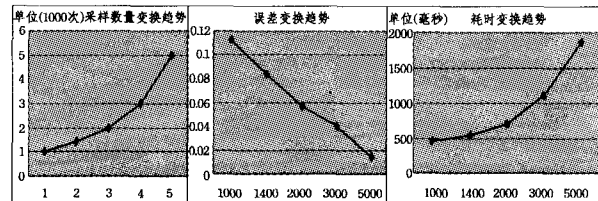


图 5 采样次数对推理精度的影响

结束语 因果图应用于大型复杂环境时, 计算复杂度过高, 通过引入基于重要性抽样的仿真算法保证了时间复杂度不会随网络节点的规模成指数上升, 其精度也能达到较为理想的效果, 该方法的有效性已经在二回路故障诊断系统中得以验证。

参考文献

- [1] Zhang Qin. Probabilistic Reasoning based on Dynamic Causality Tree/Diagrams[J]. Reliability Engineering and System Safety, 1994, 46(3): 209-220
- [2] Zhang Qin. A continuous possibility propagation diagram approach for reasoning under uncertainty [C]//Proceedings of the IEEE International Conference on System, IEEE on Man and Cybernetics, Piscataway, NJ, USA, Vol. 2, 1996: 1426-1429
- [3] Beichl, Sullivan. The importance of Importance Sampling [J]. Computing in Science and Engineering, 1999, 1(2): 71

(下转第 281 页)

从表 5 和表 6 的数据可以看出,本文提出的规则对句法树依赖比较大,例如类型“pro”在标准的句法分析树上本文的规则对其的召回率是 97.31%,在自动的句法分析树上召回率为 59.87%,可见基于规则的中文零指代项识别对于句法树的依赖是比较大的。正是因为规则的方法对句法分析树的依赖大,所以本文提出的规则中最重要的一点就是不针对整个句法分析树进行处理,而是对叶子节点的第一个 IP 节点的字树进行处理。这样,本文提出的规则无论在标准的句法分析树上还是自动句法分析树上,都比别人的方法性能要好,见表 7。

表 7 性能比较

作者	Parser 类型	召回率(%)	正确率(%)	F 值(%)
Yang, Xue ^[9]	标准	70.5	75.3	72.8
Yang, Kong ^[14]	标准	70.2	68.5	69.3
本文	标准	75.11	91.37	82.45
Yang, Xue ^[9]	自动	50.2	57.9	53.8
本文	自动	56.94	79.78	66.45

在标准句法分析树上应用本文提出的规则,从表 7 可以看出,本文提出的规则的性能要比 Yang, Xue^[9] 的高 10% 左右,比 Yang, Kong^[14] 的高 13%。所以可以证明,其对 IP 子树的处理是非常有效的。对自动句法分析的结果也可以看出,本文提出的规则性能要比 Yang, Xue^[9] 的系统性能高将近 13%,这样更加证明了在自动句法分析树进行时,对 IP 子树进行处理时,其可以有效地避免句法分析树上的错误信息。

结束语 在中文零指代研究中,对中文零指代项的识别一直是一个比较难的研究方向,不过近年来,对于缺省识别的研究的重视,使得中文零指代识别研究有了不错的进展。本文是在基于规则的情况下对中文零指代项进行识别的。从实验结果来看,本文提出的规则是可行的;并且在自动的句法树情况下,本文提出的识别方法具有很高的效率,最好的 F 值能达到 66.45%。

本文制定的规则中也漏掉了许多缺省项,在接下来的工作中,我们可以将机器学习的方法和规则相结合,以尽可能地提高识别系统的性能。

参 考 文 献

[1] Soon W M, Ng H T, Lim. A machine learning approach to coreference resolution of noun phrase[J]. Computational Linguistics, 2001, 27(4): 521-544

[2] 王厚峰. 指代消解的基本方法和实现技术[J]. 中文信息学报, 2002(6): 9-17

[3] Wu Dian-song, Liang T. A Case-Based Reasoning Approach to Zero Anaphora Resolution in Chinese Texts[C] // ICCPOL 2006. 2006; 520-531

[4] Wu Dian-song, Liang T. Zero anaphora resolution by case-based reasoning and pattern conceptualization [J]. *ExperSyst. With Appl. : An International Journal*, 2009, 36(4): 7544-7551

[5] Iida R, Poesio M. A Cross-Lingual ILP Solution to Zero Anaphora Resolution[C] // Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics. 2011; 804-813

[6] Iida R, Inui K, Matsumoto Y. Exploiting syntactic patterns as clues in zero-anaphora resolution[C] // Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics COLING-ACL2006. 2006; 625-632

[7] Yeh C-L, Chen Yi-chun. Zero anaphora resolution in Chinese with shallow parsing[J]. *Journal of Chinese Language and Computing*, 2004

[8] Yeh C-L, Chen Yi-jun. An Empirical study of zero Anaphora Resolution in chinese Based on Centering Model[Z]. 2010

[9] Yang Ya-qin, Xue Nian-wen. Chasing the ghost: recovering empty categories in the Chinese Treebank[C] // Coling2010. 2010; 1382-1390

[10] 朱勘宇. 汉语零形回指的句法驱动力[J]. 汉语学习, 2002(4): 73-80

[11] Kong Fang, Zhou Guo-dong. A Tree Kernel-based Unified Framework for Chinese Zero Anaphora Resolution[C] // Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. 2010; 882-891

[12] Zhao Shan-beng, Ng H T. Identification and Resolution of Chinese Zero Pronouns; A Machine Learning Approach[C] // Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. 2007; 541-550

[13] Yeh C-L, Mellish C. An Empirical Study on the Generation of Anaphora in Chinese[J]. *Association for Computational Linguistics*, 1997, 23(1): 171-190

[14] 杨国庆, 孔芳, 朱巧明, 等. 基于规则的中文缺省识别研究[J]. 计算机科学, 2011(12): 255-257

(上接第 253 页)

[4] Cheng Jian, Druzdzel M J. Ais-bn: An adaptive importance sampling algorithm for evidential reasoning in large bayesian networks [J]. *Journal on Artificial Intelligence*, 2000, 13: 155-188

[5] 樊兴华, 张勤, 孙茂松, 等. 多值因果图的推理算法研究[J]. 计算机学报, 2003, 26(3): 310-322

[6] 张勤. DUCG: 一种新的动态不确定因果知识的表达和推理方法 (I): 离散、静态、证据确定和有向无环图情况[J]. 计算机学报, 2010, 33(4): 625-651

[7] Zhang Qin, An Xue-gao, Jin Gu, et al. Application of FBOLES-a prototype expert system for fault diagnosis in nuclear power plants[J]. *Reliability Engineering and System Safety*, 1991, 34: 225-235

[8] Zhang Qin. Dynamic Uncertain Causality Graph for Knowledge Representation and Reasoning; Discrete DAG Cases[J]. *Journal of Computer Science and Technology*, 2012, 27(1): 1-23