

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

张 勤

(北京航空航天大学计算机学院 北京 100191)

摘 要 贝叶斯网络(BN)是国际上流行的处理不确定因果知识的表达和推理模型. 文中指出:即使所有变量均为二状态, BN 中的子变量也有单赋值和多赋值之分. 在单赋值情况下适用的知识的简洁表达和推理方法在多赋值情况下不适用. 为克服 BN 的上述及其它缺陷, 文中提出了 DUCG(Dynamical Uncertainty Causality Graph)理论模型, 以图形方式简洁表达任何情况下的不确定因果关系, 并基于证据化简图形和展开事件, 以得到所关注假设事件及其状态概率表达式. 此外, DUCG 允许知识表达不完备, 使其超越了 BN 理论框架. 一个入侵者报警系统被用来解释 DUCG 理论.

关键词 智能系统; 知识表达; 概率推理; 因果关系; 不确定性

中图法分类号 TP301 **DOI 号**: 10.3724/SP.J.1016.2010.00625

A New Methodology to Deal with Dynamical Uncertain Causalities (I): The Static Discrete DAG Case

ZHANG Qin

(School of Computer Science, Beihang University, Beijing 100191)

Abstract Bayesian Network (BN) is a prominent model to deal with knowledge representation and inference in case of uncertain causalities. This paper discusses the essential difference between single-valued and multi-valued cases. It is pointed out that even a binary child variable can be either single-valued or multi-valued, while the existing compact representations and the corresponding inference algorithms applicable in single-valued cases cannot be simply applied in multi-valued cases. To overcome this problem and others, a new model named as DUCG (Dynamical Uncertain Causality Graph) is presented. By introducing a set of new concepts, DUCG is able to compactly and graphically represent complex conditional probability distributions (CPDs) in different modules, irrespective of whether the cases of the modules are single-valued or multi-valued. The simple connection among separately constructed modules composes a final DUCG. Once the evidence is observed, the first inference step is to simplify the DUCG regardless of queries by applying the 10 rules presented in this paper. The second step is to apply the event outspread algorithm presented in this paper to calculate the updated probabilities of the queries still in concern based on the simplified DUCG, regardless of whether the variables are singly or multiply connected. Sometimes, the qualitative solution can be found by only simplifying DUCG. Correspondingly, the accuracy of parameters is less important in DUCG. Moreover, DUCG enables people to represent the knowledge only in concern but not enough to represent CPDs. In other words,

DUCG does not have to represent the joint probability distribution over a set of variables, although it is able to. This incompleteness of representation and flexible conditional causalities represented in DUCG, in addition to that DUCG is able to deal with the directed cyclic graph (DCG) to be addressed in next paper, etc, makes DUCG beyond BN. The example of an alarm system detecting intruder illustrates the DUCG methodology.

Keywords intelligent system; knowledge representation; probabilistic reasoning; causality; uncertainty

1 引 言

处理不确定因果关系的知识表达方式和推理方法是人工智能系统必须解决的关键问题之一。迄今已提出了许多理论模型。如 Certainty Factor^[1]、Evidence Reasoning^[2]、PROSPECTOR^[3]、Fuzzy Logic^[4]、BN^[5-20] 等等。其中, BN 正日益流行^[12,16]。简洁的知识表达方式和高效的推理算法是 BN 研究的核心问题之一。

众所周知, BN 中条件概率分布 CPD (Conditional Probability Distribution) 的典型表达方式是条件概率表 CPT (Conditional Probability Table)。然而, CPT 需要说明太多参数。例如, 1 个子变量和 5 个父变量且所有变量均有 5 个状态, 则 CPT 中的条件概率参数为 $5^6 = 15625$ 个。针对这一问题, 学术界提出了许多简洁 (compact) 表达方式和推理算法。例如 noisy-OR^[6]、CSI (Context Specific Independence)^[21]、DCD (Dynamic Causality Diagram)^[22] 等等。然而, 这些模型均以二状态变量为例提出和说明, 实际上讨论的均为单赋值情况而非多赋值情况。本文指出: 单赋值情况和多赋值情况有着本质区别, 适用于单赋值情况的简洁表达方式和推理算法通常不适用于多赋值情况。

所谓单赋值 (single-valued) 情况或多赋值 (multi-valued) 情况, 是指所涉及的子变量是单赋值的或多赋值的。单赋值变量和多赋值变量的定义如下。

定义 1. 说明变量状态的原因被称为赋值。单赋值变量是指子变量的赋值状态只有一个 (通常称为 true 状态); 多赋值变量是指子变量的赋值状态多于一个。

例如, 在学术界熟知的盗窃、地震和报警系统例子中^[6], 警报既可被盗窃引发, 也可被地震引发。警报变量有两个状态: “警报响” (true 状态)、“警报不

响” (false 状态)。通常, 仅“警报响”状态的原因被说明 (即盗窃或地震引发警报), 而“警报不响”状态的原因不被说明, 因为“警报不响”状态的原因已被缺省地说明为“警报响”状态的补集。根据上述定义, 作为子变量的报警变量是单赋值的。

又如冰箱温度变量, 该变量有 3 个状态: 温度“高”、“低”和“正常”。“高”状态的原因可能是冰箱压缩机故障; “低”状态的原因可能是冰箱温度探测器故障。这里, 温度变量的两个状态的原因被分别说明。根据上述定义, 作为子变量的温度变量是多赋值的。

需要指出的是: 多赋值变量的含义与多状态变量的含义完全不同。多状态 (multi-state) 变量是指该变量的状态多于两个, 否则即为二状态 (binary) 变量。二状态变量有且仅有两个状态。例如响与不响、雄与雌, 等等。需要注意的是, 二状态变量既可以是单赋值的, 也可以是多赋值的。上例中的报警变量是单赋值的二状态变量; 而性别变量却通常是多赋值的二状态变量, 因为性别变量的两个状态“雄”与“雌”的生物学原因通常被分别说明。

多状态变量通常是多赋值变量。上述温度变量既是多状态变量, 又是多赋值变量。这是因为单赋值多状态变量是无意义的。作为单赋值变量, 多状态变量除 true 状态外的所有其它状态可被压缩为一个状态: false, 是 true 状态的补集。换句话说, 单赋值多状态变量实际上是单赋值二状态变量。

在许多文献中 (例如文献 [16]), 单赋值变量常常被混淆为二状态变量, 而多赋值变量常常被混淆为多状态变量。这种混淆的结果是: 仅适用于单赋值情况的简洁表达方式和推理方法被简单地进行状态概率强制归一而被不适当地推广于多赋值情况。本文下一节将指出, 这样的推广理论上无依据, 实践上是不合理的、不自洽的、不精确的、甚至不具有可操作性。鉴于此, 本文提出了 DUCG 模型, 以解决上述问题和其它问题。

第2节将详细讨论单赋值情况和多赋值情况的本质区别, 进一步阐明提出 DUCG 的3个主要动机之一. 其它两个是处理有向循环图 DCG(将在下一篇文章中讨论)和不完备的知识表达和推理. 第3节简单介绍了笔者在文献[22]中提出的 DCD 模型. 这种介绍伴随着与 noisy-OR 和 CSI 的比较, 因为尽管 DCD 比 CSI 早两年提出, 但学术界并不熟悉. 而 noisy-OR 和 CSI 却已为学术界广泛知晓. 介绍 DCD 的原因在于 DCUG 是从 DCD 发展而来的. 本节除介绍了 DCD 外, 还进一步将其扩展为单赋值 DUCG(S-DUCG). 应当说明, 该节所讨论的实际上仅限于单赋值情况. 第4节讨论多赋值情况的简洁表达, 提出了多赋值 DUCG 模型(M-DUCG). 第5节将 S-DUCG 和 M-DUCG 结合为 DUCG 模型, 既适用于单赋值情况, 又适用于多赋值情况, 实现了文献[23]所提出的简洁表达和推理所需要的充分性(sufficiency)和可分离性(separability). 第6节讨论如何和为什么 DUCG 的表达可以是不完备的. 这种表达的不完备性已经超越了 BN 的理论框架, 是 DUCG 的一个重要新特性. 该特性能够大大减少建造知识库和推理的工作量和难度. 第7节提出基于观测到的证据对 DUCG 进行有效化简的方法. 通过化简, 通常无需任何数值计算即可得到定性的推理结果. 这种化简与待推理问题无关, 所采用的证据包括事件发生的时序. 基于化简的 DUCG, 第8节提出后续的推理算法. 该算法提出了一种处理绑定了 DUCG 所定义的权重系数的事件运算的新算法. 基于应用该算法所展开的事件表达式, 将对应的独立事件的概率参数代入即可算得所关注的假设事件的条件状态概率和条件排序概率. 其中条件排序概率是本文新定义的. 第9节对本文进行小结, 并对后续工作进行简要介绍.

限于篇幅, 本文仅讨论静态、离散、证据确定和有向无环图(Directed Acyclic Graph, DAG)情况.

2 单赋值情况和多赋值情况的本质区别

在许多二状态变量情况中, 人们只需要说明子变量 X_n 的 true 状态的原因即可, 因为 false 状态的原因已经被隐含地说明为 true 状态的补集. 其中, 下标 n 用于标识子变量. 设第2个下标为“1”代表 true 状态, 第2个下标为“2”代表 false 状态. 对于文献[6]中所提出的学术界熟知的盗窃(X_1)、地震

(X_2)和警报(X_3)例子来说, $X_3 = x_{31}$ (警报响)可被 $X_1 = x_{11}$ (盗窃发生)和 $X_2 = x_{21}$ (地震发生)分别引起. 对于领域工程师来说, 分别给出 $X_1 = x_{11}$ 和 $X_2 = x_{21}$ 引发 $X_3 = x_{31}$ 的特殊条件概率是比较容易的, 而直接给出 CPT 则比较困难. 这是因为盗窃和地震分属完全不同的领域. 然而, 需要指出的是, 本例中只有 $X_3 = x_{31}$ 的原因被说明, 而 $X_3 = x_{32}$ (警报不响)的原因不得被单独说明, 因为 $X_3 = x_{32}$ 已被隐含地说明为 $X_3 = x_{31}$ 的补集. 图1解释了这种二状态单赋值情况. 其中, \rightarrow 代表状态层的因果关系, 用文字记为 $P_{nk;ij}$. 下标“ $nk;ij$ ”表明事件 $X_n = x_{nk}$ 由事件 $X_i = x_{ij}$ 引起. 图中的有向弧采用特别的箭头形状是为了区别于 BN 中 CPT 类型的因果关系有向弧 \rightarrow . 此外, 本文中“;”被用来分隔子变量和父变量, 其前的标识是子变量的, 其后的标识是父变量的.

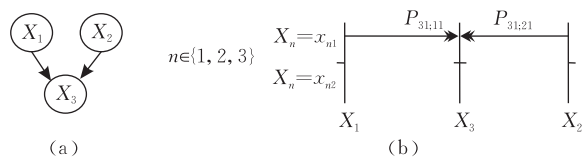


图1 二状态单赋值情况示例

然而, 现实世界并非总是如此简单. 例如, 一个简单的数字记忆电路具有“01”和“10”两个状态或输出. 这是一个典型的二状态变量. 类似于警报变量, 我们可以将其记为 X_3 , 且 $X_3 = x_{31}$ 代表状态“01”、 $X_3 = x_{32}$ 代表状态“10”. 不同的是, $X_3 = x_{31}$ 和 $X_3 = x_{32}$ 可分别由不同的事件引起. 例如, $X_3 = x_{31}$ 由 $X_1 = x_{11}$ 引起; $X_3 = x_{32}$ 由 $X_2 = x_{21}$ 引起; 否则 X_3 的状态随机确定. 上述两个原因分别导致各自的结果的“特殊”条件概率分别为 $p_{31;11}$ 和 $p_{32;21}$. 使用“特殊”条件概率一词的原因在于通常 $p_{nk;ij} \neq Pr\{X_n = x_{nk} | X_i = x_{ij}\}$. 事实上, $p_{nk;ij}$ 是 DCD 中的连接事件 $P_{nk;ij}$ 的发生概率, $p_{nk;ij} \equiv Pr\{P_{nk;ij}\}$. $p_{nk;ij}$ 也是 noisy-OR 中的禁事件(Inhibitor)的补集的发生概率(参见文献[6]或3.1节). 类似的符号在文献[24]中也被采用. 其中 $p_{31;11}$ 和 $p_{32;21}$ 被分别记为 $c_{X11}(X_{31})$ 和 $c_{X21}(X_{32})$. 这些记法的目的在于区分一般的和特殊的条件概率. 为简单起见, 在不引起混淆的情况下, 特殊条件概率在以下将被简称为概率.

第1节提到性别变量(X_3)也是一个二状态多赋值变量. 该变量有“雄”($X_3 = x_{31}$)和“雌”($X_3 = x_{32}$)两个地位相同的状态, 其生物学原因不同, 被分别说明.

抽象地, 一个简单的二状态多赋值情况如图2所示. 图1(b)和图2(b)的本质区别在于: 图2(b)中

$X_3 = x_{31}$ 和 $X_3 = x_{32}$ 的原因被分别说明. 注意: 图 1(a)和图 2(a)是完全相同的. 这意味着单赋值和多赋值两种情况之间的本质差别在变量或节点层面被隐藏了. 这种隐藏在 CPT 表达方式下并无实质意义, 因为这种差别已体现在 CPT 的数据中. 但在简洁表达方式下, 这种隐藏导致了实质性的问题. 这些问题将在稍后讨论.

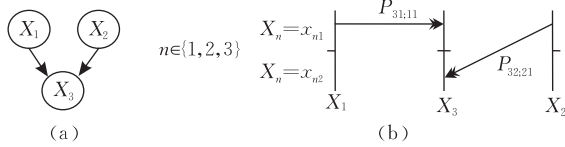


图 2 二状态多赋值情况示例

多赋值情况在子变量为多状态变量时通常遇到. 例如核反应堆温度变量(X_3)可能有 5 个状态: “正常”($X_3 = x_{31}$)、“高”($X_3 = x_{32}$)、“快速升高”($X_3 = x_{33}$)、“低”($X_3 = x_{34}$)和“快速降低”($X_3 = x_{35}$). 这些状态的原因通常不同, 且被分别说明.

众所周知, 在任何情况下, 一个变量的所有状态的发生概率之和应等于 1. 这种性质被称为归一性. 单赋值情况总是满足归一性的, 因为 true 状态的概率不可能超过 1, 即 $Pr\{\text{true} | E\} \leq 1$; 而 false 状态不过是 true 状态的补集, 即 $Pr\{\text{false} | E\} = 1 - Pr\{\text{true} | E\}$. 其中 E 代表任意条件. 然而, 在多赋值情况下, 归一性通常难以满足. 例如, 设图 2 中 $X_3 = x_{31}$ 由 $X_1 = x_{11}$ 引起的概率为 $p_{31,11} = 0.6$; $X_3 = x_{32}$ 由 $X_2 = x_{21}$ 引起的概率为 $p_{32,21} = 0.8$. 如果我们简单地将多赋值情况下的这两个状态分别作为两个单赋值情况对待, 我们有 $X_3 = x_{31}$ 独立于 X_2 ; $X_3 = x_{32}$ 独立于 X_1 . 本质上, 这是因为 $X_3 = x_{31}$ 和 $X_3 = x_{32}$ 的原因被分别说明. 于是, 给定条件 $E = (X_1 = x_{11}) \cap (X_2 = x_{21})$, 如果按照两个单赋值情况来对待, 我们有 $Pr\{X_3 = x_{31} | E\} = Pr\{X_3 = x_{31} | X_1 = x_{11}\} = p_{31,11}$ 和 $Pr\{X_3 = x_{32} | E\} = Pr\{X_3 = x_{32} | X_2 = x_{21}\} = p_{32,21}$. 又由于 $X_3 = x_{31}$ 和 $X_3 = x_{32}$ 互斥, 我们进而有

$$\begin{aligned} Pr\{(X_3 = x_{31}) \cup (X_3 = x_{32}) | E\} &= \\ Pr\{X_3 = x_{31} | E\} + Pr\{X_3 = x_{32} | E\} &= \\ p_{31,11} + p_{32,21} &= 0.6 + 0.8 = 1.4 > 1, \end{aligned}$$

即 X_3 的状态归一性未能满足. 这是由于 $X_3 = x_{31}$ 和 $X_3 = x_{32}$ 因互斥而相关, 同时它们的原因又被作为两个独立的单赋值状态而被分别说明. 解决这一问题的现有方法(例如文献[16])是应用式(1):

$$Pr\{X_n = x_{nk} | E\} = \frac{Pr\{(X_n = x_{nk}) \cap E\}}{\sum_k Pr\{(X_n = x_{nk}) \cap E\}}$$

$$= \frac{Pr\{X_n = x_{nk} | E\}}{\sum_k Pr\{X_n = x_{nk} | E\}} \quad (1)$$

然而, 式(1)仅当 $\sum_k Pr\{X_n = x_{nk} | E\} = 1$ 时成立. 这从式(2)即可看出:

$$\begin{aligned} Pr\{X_n = x_{nk} | E\} &= \frac{Pr\{(X_n = x_{nk}) \cap E\}}{Pr\{E\}} \\ &= \frac{Pr\{(X_n = x_{nk}) \cap E\}}{Pr\{E\} \sum_k Pr\{X_n = x_{nk} | E\}} \\ &= \frac{Pr\{E\} Pr\{X_n = x_{nk} | E\}}{Pr\{E\} \sum_k Pr\{X_n = x_{nk} | E\}} \\ &= \frac{Pr\{X_n = x_{nk} | E\}}{\sum_k Pr\{X_n = x_{nk} | E\}} \quad (2) \end{aligned}$$

式中第 2 个等号成立的条件是 $\sum_k Pr\{X_n = x_{nk} | E\} = 1$. 然而, 按照两个单赋值情况分别算出的 $Pr\{X_n = x_{nk} | E\}$ 通常不满足 $\sum_k Pr\{X_n = x_{nk} | E\} = 1$. 上例已经证明了这一点. 这个问题在 CPT 表达方式下并不存在, 因为 $\sum_k Pr\{X_n = x_{nk} | E\} = 1$ 在 CPT 中总是满足的. 其中, E 代表 CPT 中的父变量的状态组合. 换句话说, 这个问题仅存在于简洁表达方式中.

可能有人认为: 当 $\sum_k Pr\{X_n = x_{nk} | E\} = 1$ 时, 无需应用式(1)实现归一化. 正是因为 $\sum_k Pr\{X_n = x_{nk} | E\} \neq 1$, 才需要应用式(1). 但式(2)表明, 式(1)正是基于 $\sum_k Pr\{X_n = x_{nk} | E\} = 1$ 才成立的. 这就是说, 在式(1)中, 结果成了前提. 可见, 式(1)没有任何理论基础, 是人为强加的.

另一些人可能认为式(1)是处理多赋值情况的近似方法. 然而情况并非如此, 因为许多文献(例如文献[16])认为式(1)是精确解. 此外, 如果式(1)是近似解, 则应有误差估计. 但迄今并未看到关于应用式(1)的误差估计的文献. 最后, 由于本文仅讨论精确解, 近似方法不属于本文的讨论范围.

除上述理论分析外, 下述例子进一步揭示了在多赋值情况下应用式(1)将会产生的 4 个实际问题.

2.1 不合理性

图 3 的示例来自文献[21]中的图 1, 其中所有变量均为二状态变量. 本例采用 CSI 表达方式, 左分支代表 true 状态, 右分支代表 false 状态. 为简单起见, 事件 $X_n = x_{nk}$ 被记为 X_{nk} , 例如 $X_1 = x_{11}$ 被记为 X_{11} ; X_4 的第 j 种父变量状态组合(state combi-

nation of parent variables) 被记为 $SCPV_{4,i}$, 例如 $SCPV_{4,1} = X_{11} X_{21} X_{31}$, 其中事件的相乘代表逻辑与 (AND). 本例中, 如果仅 $X_{41} \equiv (X_4 = x_{41})$ 的原因被

说明, 同时 $X_{42} \equiv (X_4 = x_{42})$ 的原因不被单独说明, 则本例是单赋值情况, 因为 X_{42} 已被隐含地说明为 X_{41} 的补集.

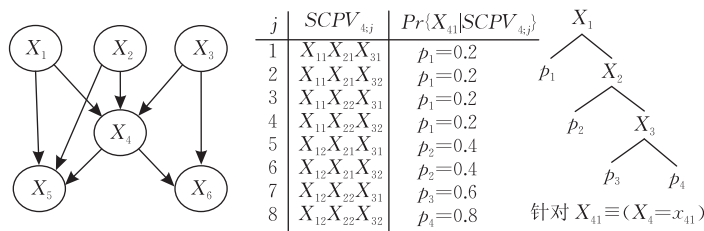


图 3 $X_{41} \equiv (X_4 = x_{41})$ 的原因的 CSI 说明

设我们不将 X_{42} 作为 X_{41} 的补集对待, 而是单独说明其原因如图 4 所示, 则本例变成多赋值情况. 注意, X_1 不是 X_{42} 的父变量, 但却是 X_{41} 的父变量. 一般来说, 在多赋值情况下, 子变量的不同状态可以有不同的父变量.

$$\alpha_{4,4} = 1 / (Pr\{X_{41} | E_4\} + Pr\{X_{42} | E_4\}) = 1 / (0.2 + 0.7) = 1 / 0.9,$$

$$\alpha_{4,5} = 1 / (Pr\{X_{41} | E_5\} + Pr\{X_{42} | E_5\}) = 1 / (0.4 + 0.4) = 1 / 0.8,$$

$$\alpha_{4,6} = 1 / (Pr\{X_{41} | E_6\} + Pr\{X_{42} | E_6\}) = 1 / (0.4 + 0.2) = 1 / 0.6,$$

$$\alpha_{4,7} = 1 / (Pr\{X_{41} | E_7\} + Pr\{X_{42} | E_7\}) = 1 / (0.6 + 0.7) = 1 / 1.3,$$

$$\alpha_{4,8} = 1 / (Pr\{X_{41} | E_8\} + Pr\{X_{42} | E_8\}) = 1 / (0.8 + 0.7) = 1 / 1.5.$$

应用式 (3), 可算出本例的 CPT 如表 1 所示.

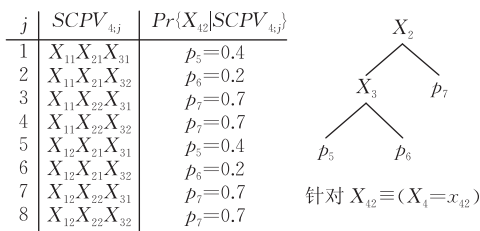


图 4 $X_{42} \equiv (X_4 = x_{42})$ 的原因和参数的独立 CSI 说明

显然, 将图 3 与图 4 相结合, 我们有 $Pr\{X_{41} | SCPV_{4,i}\} + Pr\{X_{42} | SCPV_{4,i}\} \neq 1$. 下面我们来看如果应用式 (1) 强加归一性将会是怎样的情况.

记 $E_j = SCPV_{4,i}$. 式 (1) 可被进一步写成

$$Pr\{X_{nk} | E_j\} = \frac{Pr\{X_{nk} | E_j\}}{\sum_k Pr\{X_{nk} | E_j\}} = \alpha_{n,i,j} Pr\{X_{nk} | E_j\} \quad (3)$$

$$\alpha_{n,i,j} \equiv 1 / \sum_k Pr\{X_{nk} | E_j\} \quad (4)$$

其中, $\alpha_{n,i,j}$ 被称为归一化系数. 在式 (3) 和 (4) 中, 右边的 $Pr\{X_{nk} | E_j\}$ 是归一化前的值, 左边的 $Pr\{X_{nk} | E_j\}$ 是归一化后的值. 注意, 式 (4) 中的归一化系数 $\alpha_{n,i,j}$ 是一个取决于 $E_j = SCPV_{n,i}$ 的变量, 不是常数. 按照式 (4), 图 3 和图 4 所示的多赋值情况的归一化系数为

$$\alpha_{4,1} = 1 / (Pr\{X_{41} | E_1\} + Pr\{X_{42} | E_1\}) = 1 / (0.2 + 0.4) = 1 / 0.6,$$

$$\alpha_{4,2} = 1 / (Pr\{X_{41} | E_2\} + Pr\{X_{42} | E_2\}) = 1 / (0.2 + 0.2) = 1 / 0.4,$$

$$\alpha_{4,3} = 1 / (Pr\{X_{41} | E_3\} + Pr\{X_{42} | E_3\}) = 1 / (0.2 + 0.7) = 1 / 0.9,$$

表 1 由式 (3) 算得的 CPT

j	$E_j = SCPV_{4,i}$	$Pr\{X_{41} E_j\}$	$Pr\{X_{42} E_j\}$	$\alpha_{4,i,j}$
1	$X_{11} X_{21} X_{31}$	1/3	2/3	1/0.6
2	$X_{11} X_{21} X_{32}$	1/2	1/2	1/0.4
3	$X_{11} X_{22} X_{31}$	2/9	7/9	1/0.9
4	$X_{11} X_{22} X_{32}$	2/9	7/9	1/0.9
5	$X_{12} X_{21} X_{31}$	1/2	1/2	1/0.8
6	$X_{12} X_{21} X_{32}$	2/3	1/3	1/0.6
7	$X_{12} X_{22} X_{31}$	6/13	7/13	1/1.3
8	$X_{12} X_{22} X_{32}$	8/15	7/15	1/1.5

从表 1 可见, CPT 是基于很多且不同的归一化系数而算得的. 一般来说, 归一化系数的数量与父变量状态组合 $SCPV_{n,i}$ 的数量相等. 也就是说, 这些不同的归一化系数的数量巨大. 例如, 5 个父变量且每个变量有 5 个状态, 则归一化系数的数量达 $5^5 = 3125$ 个. 这个数量对于领域工程师来说太多了. 在分别说明子变量不同状态的原因和参数时, 他们难以意识到存在如此之多不同的归一化系数. 这里的问题是: 我们为什么需要如此之多隐含的不同的归一化系数? 这些不同的归一化系数是领域工程师在表达知识时意识到的吗? 换句话说, 这些归一化系数是领域工程师想要的吗? 这些隐含的系数是否表达了领域工程师的知识? 显然, 人们在应用式 (1) 或式 (3) 和 (4) 时, 并没有明确地意识到和回答这些问

题. 因此, 简单地将多赋值情况下的不同状态作为单赋值情况下的多个 true 状态分别对待是不合理的.

2.2 不一致性

如果我们将图 3 和图 4 中 p_i 的值(表 2 中的

表 2 两套 p_i 值的比较

j	$E_j = SCPV_{4,j}$	第 1 套 p_i			第 2 套 p_i		
		$Pr\{X_{41} E_j\}$	$Pr\{X_{42} E_j\}$	$\alpha_{4,j}$	$Pr\{X_{41} E_j\}$	$Pr\{X_{42} E_j\}$	$\alpha_{4,j}$
1	$X_{11}X_{21}X_{31}$	$p_1=0.2$	$p_5=0.4$	1/0.6	$p_1=0.1$	$p_5=0.2$	1/0.3
2	$X_{11}X_{21}X_{32}$	$p_1=0.2$	$p_6=0.2$	1/0.4	$p_1=0.1$	$p_6=0.1$	1/0.2
3	$X_{11}X_{22}X_{31}$	$p_1=0.2$	$p_7=0.7$	1/0.9	$p_1=0.1$	$p_7=0.35$	1/0.45
4	$X_{11}X_{22}X_{32}$	$p_1=0.2$	$p_7=0.7$	1/0.9	$p_1=0.1$	$p_7=0.35$	1/0.45
5	$X_{12}X_{21}X_{31}$	$p_2=0.4$	$p_5=0.4$	1/0.8	$p_2=0.2$	$p_5=0.2$	1/0.4
6	$X_{12}X_{21}X_{32}$	$p_2=0.4$	$p_6=0.2$	1/0.6	$p_2=0.2$	$p_6=0.1$	1/0.3
7	$X_{12}X_{22}X_{31}$	$p_3=0.6$	$p_7=0.7$	1/1.3	$p_3=0.3$	$p_7=0.35$	1/0.65
8	$X_{12}X_{22}X_{32}$	$p_4=0.8$	$p_7=0.7$	1/1.5	$p_4=0.4$	$p_7=0.35$	1/0.75

这里出现了一个矛盾: 不同的参数可能对应相同的 CPT, 但却可能产生不同的链式概率推理结果. 为说明这一矛盾, 考虑图 5 中关于冰箱温度的例子.

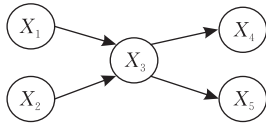


图 5 冰箱温度 BN 示例

其中的变量状态或事件的定义如下:

$X_{11} = \{\text{冷却剂泄漏}\}$; $X_{12} = \{\text{无冷却剂泄漏}\}$;

$X_{21} = \{\text{温度探测器失灵}\}$;

$X_{22} = \{\text{温度探测器正常}\}$;

$X_{31} = \{\text{温度偏高}\}$; $X_{32} = \{\text{温度偏低}\}$;

$X_{33} = \{\text{温度正常}\}$;

$X_{41} = \{\text{食物腐败}\}$; $X_{42} = \{\text{无食物腐败}\}$;

$X_{51} = \{\text{能耗偏高}\}$; $X_{52} = \{\text{无能耗偏高}\}$.

在 CSI 表达式下, X_{31} , X_{32} , X_{33} 和 X_{41} 的原因可被分别说明如图 6 所示. 其中分支自左至右按照变量状态标号顺序排列(1, 2, ...). 基于图 6, X_{31} (温度偏高)可被 X_{11} (冷却剂泄漏)引起, 概率为 0.6; X_{32} (温度偏低)可被 X_{21} (温度探测器失灵)引起, 概率为 0.8; X_{41} (食物腐败)可被 X_{31} 引起, 概率为 0.7; X_{51} (能耗偏高)可被 X_{32} 引起, 概率为 0.5; X_{42} (无食物腐败)是 X_{41} 的补集, 即 $X_{42} = \bar{X}_{41}$; X_{52} (无能耗偏高)是 X_{51} 的补集, 即 $X_{52} = \bar{X}_{51}$.

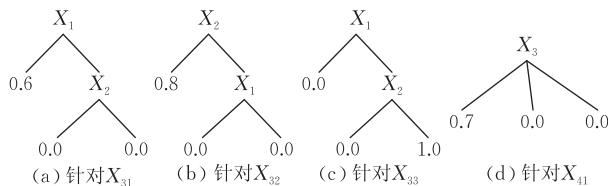


图 6 X_{31} 、 X_{32} 、 X_{33} 和 X_{41} 的原因的 CSI 表达

第 1 套值)改变为表 2 中的第 2 套值, 所算得的 CPT 仍如表 1 所示. 这归因于归一化系数的相应改变.

显然, X_3 是多赋值变量, X_4 和 X_5 是单赋值变量. 这是因为 X_3 的多于 1 个状态的原因被分别说明, X_4 和 X_5 只有 1 个状态的原因被说明.

给定 $E_j = X_{11}X_{21}$, 如果将 X_{31} 、 X_{32} 和 X_{33} 分别作为 3 个单赋值情况下的 true 状态对待, 则根据图 6, 我们有

$$Pr\{X_{31}|X_{11}X_{21}\} = 0.6 \quad (5)$$

$$Pr\{X_{32}|X_{11}X_{21}\} = 0.8 \quad (6)$$

$$Pr\{X_{33}|X_{11}X_{21}\} = 0.0 \quad (7)$$

应用式(1), X_3 在 $X_{11}X_{21}$ 条件下的 CPT 值可算得为

$$Pr\{X_{31}|X_{11}X_{21}\} / \sum_k Pr\{X_{3k}|X_{11}X_{21}\} = 0.6 / (0.6 + 0.8 + 0) = 0.4286 \quad (8)$$

$$Pr\{X_{32}|X_{11}X_{21}\} / \sum_k Pr\{X_{3k}|X_{11}X_{21}\} = 0.8 / (0.6 + 0.8 + 0) = 0.5714 \quad (9)$$

$$Pr\{X_{33}|X_{11}X_{21}\} / \sum_k Pr\{X_{3k}|X_{11}X_{21}\} = 0 / (0.6 + 0.8 + 0) = 0.0 \quad (10)$$

如果我们将图 6 或式(5)~(7)中的参数改变为

$$Pr\{X_{31}|X_{11}X_{21}\} = 0.3 \quad (11)$$

$$Pr\{X_{32}|X_{11}X_{21}\} = 0.4 \quad (12)$$

$$Pr\{X_{33}|X_{11}X_{21}\} = 0.0 \quad (13)$$

根据式(1), 我们仍有

$$Pr\{X_{31}|X_{11}X_{21}\} / \sum_k Pr\{X_{3k}|X_{11}X_{21}\} = 0.3 / (0.3 + 0.4 + 0) = 0.4286 \quad (14)$$

$$Pr\{X_{32}|X_{11}X_{21}\} / \sum_k Pr\{X_{3k}|X_{11}X_{21}\} = 0.4 / (0.3 + 0.4 + 0) = 0.5714 \quad (15)$$

$$Pr\{X_{33}|X_{11}X_{21}\} / \sum_k Pr\{X_{3k}|X_{11}X_{21}\} =$$

$$0 / (0.3 + 0.4 + 0) = 0.0 \quad (16)$$

也就是说, X_3 在 $X_{11} X_{21}$ 条件下的 CPT 值保持不变. 现在来看看计算 $Pr\{X_{41} | X_{11} X_{21}\}$ 和 $Pr\{X_{42} | X_{11} X_{21}\}$ 的情况. 注意: 因为 X_4 是单赋值的, 无需应用式(1).

根据图 5, 我们有

$$\begin{aligned} Pr\{X_{41} | X_{11} X_{21}\} &= Pr\{X_{41} | X_{31}\} Pr\{X_{31} | X_{11} X_{21}\} + \\ &Pr\{X_{41} | X_{32}\} Pr\{X_{32} | X_{11} X_{21}\} + \\ &Pr\{X_{41} | X_{33}\} Pr\{X_{33} | X_{11} X_{21}\} \end{aligned} \quad (17)$$

根据图 6(d), 我们有

$$\begin{aligned} Pr\{X_{41} | X_{31}\} &= 0.7, \\ Pr\{X_{41} | X_{32}\} &= 0.0, \\ Pr\{X_{41} | X_{33}\} &= 0.0. \end{aligned}$$

于是, 式(17)成为

$$Pr\{X_{41} | X_{11} X_{21}\} = 0.7 Pr\{X_{31} | X_{11} X_{21}\} \quad (18)$$

对于计算 $Pr\{X_{31} | X_{11} X_{21}\}$, 我们有两种选择: 一种是用式(8)或式(14)中的 CPT 值. 在这种选择下, 我们有

$$Pr\{X_{41} | X_{11} X_{21}\} = 0.7 \times 0.4286 = 0.3 \quad (19)$$

$$\begin{aligned} Pr\{X_{42} | X_{11} X_{21}\} &= 1 - Pr\{X_{41} | X_{11} X_{21}\} \\ &= 1 - 0.3 = 0.7 \end{aligned} \quad (20)$$

式(20)是由于 X_4 是单赋值的. 这种选择将推理计算限定于使用 CPT 的值. 虽然本例的 CPT 值是按照简洁表达方式算出来的, 但我们却放弃了对应于该简洁表达方式的链式推理算法. 显然, 这种放弃是我们所不愿意的.

另一种选择是用式(5)或(11)的值, 即用归一化之前的概率值进行链式推理. 当采用式(5)的值时, 式(19)、(20)变成

$$Pr\{X_{41} | X_{11} X_{21}\} = 0.7 \times 0.6 = 0.42 \quad (21)$$

$$\begin{aligned} Pr\{X_{41} | X_{11} X_{21}\} &= 1 - Pr\{X_{41} | X_{11} X_{21}\} \\ &= 1 - 0.42 = 0.58 \end{aligned} \quad (22)$$

当采用式(11)的值时, 式(19)、(20)变成

$$Pr\{X_{41} | X_{11} X_{21}\} = 0.7 \times 0.3 = 0.21 \quad (23)$$

$$\begin{aligned} Pr\{X_{41} | X_{11} X_{21}\} &= 1 - Pr\{X_{41} | X_{11} X_{21}\} \\ &= 1 - 0.21 = 0.79 \end{aligned} \quad (24)$$

这里我们看到, 式(19)、(20)、式(21)、(22)和式(23)、(24)的值彼此不同. 换句话说, 虽然式(5)~(7)和式(11)~(13)所示的两套参数对应同一个 CPT, 但链式推理结果却不同. 事实上, 这种不同可能正是领域工程师给出不同参数时希望的效果, 但却与相同的 CPT 应得出相同的推理结果矛盾.

2.3 不精确性

应用式(1)必须分别计算多赋值子变量的所有状态的条件概率. 这里, 多赋值变量的每一个状态都被作为一个单赋值情况的 true 状态对待. 这就要求多赋值变量的所有状态都要按照单赋值变量的 true 状态分别说明其原因. 例如, X_{31} 、 X_{32} 和 X_{33} 的原因在图 6(a)~(c)中被分别说明.

然而, 在图 6(c)中说明的 X_{33} 的原因是不精确的, 因为 $X_{33} \neq \bar{X}_{11} \bar{X}_{21} = X_{12} X_{22}$. 例如, 即使 X_{11} 和 X_{21} 均发生, X_{33} 也仍可能发生, 因为 $X_{11} X_{21}$ 尚不足以导致 X_{31} 或 X_{32} . 精确的表达应当是 $X_{33} = \bar{X}_{31} \bar{X}_{32}$. 换句话说, X_{33} 是 $X_{31} + X_{32}$ 的补集. 其中, “+”代表互斥或(XOR). 于是,

$$\begin{aligned} Pr\{X_{33} | X_{11} X_{21}\} &= 1 - Pr\{X_{31} | X_{11} X_{21}\} - \\ &Pr\{X_{32} | X_{11} X_{21}\}. \end{aligned}$$

必须指出, 补集不是一种独立的原因说明, 不能用于式(1)的计算, 否则将会出现问题. 如上例, 要计算 $Pr\{X_{33} | X_{11} X_{21}\}$, 必须先知道 $Pr\{X_{31} | X_{11} X_{21}\}$ 和 $Pr\{X_{32} | X_{11} X_{21}\}$. 但要知道用式(1)算得的归一化的 $Pr\{X_{31} | X_{11} X_{21}\}$ 和 $Pr\{X_{32} | X_{11} X_{21}\}$, 就必须先知道 $Pr\{X_{33} | X_{11} X_{21}\}$. 这显然是矛盾的.

归一化前的 $Pr\{X_{31} | X_{11} X_{21}\}$ 和 $Pr\{X_{32} | X_{11} X_{21}\}$ 不能被用来计算 $Pr\{X_{33} | X_{11} X_{21}\}$. 这是因为按照式(5)和式(6), 我们有

$$\begin{aligned} Pr\{X_{33} | X_{11} X_{21}\} &= 1 - Pr\{X_{31} | X_{11} X_{21}\} - \\ &Pr\{X_{32} | X_{11} X_{21}\} \\ &= 1 - 0.6 - 0.8 \\ &= -0.4 \end{aligned} \quad (25)$$

这是一个不合理的值. 因此, 如果我们坚持应用式(1), 类似于图 6(c)的不精确表达就难以避免.

2.4 不可操作性

在上述计算中, 如果我们用式(11)和(12)的值, 式(25)就变成了式(26):

$$\begin{aligned} Pr\{X_{33} | X_{11} X_{21}\} &= 1 - Pr\{X_{31} | X_{11} X_{21}\} - \\ &Pr\{X_{32} | X_{11} X_{21}\} \\ &= 1 - 0.3 - 0.4 \\ &= 0.3 \end{aligned} \quad (26)$$

这是一个可接受的值. 尽管如此, 除了要冒式(25)的风险外, 如果有多于一个状态采用补集方式说明原因, 也会导致类似计算实际上无法操作. 例如, 如果上例中 X_{32} 和 X_{33} 的原因被分别说明为 $X_{11} + X_{33}$ 和 $X_{11} + X_{32}$ 的补集, 则式(26)的计算将无法操作. 这是因为

$$Pr\{X_{32} | X_{11} X_{21}\} = 1 - Pr\{X_{31} | X_{11} X_{21}\} -$$

$$\begin{aligned} & Pr\{X_{33} | X_{11} X_{21}\} \\ &= 1 - 0.3 - Pr\{X_{33} | X_{11} X_{21}\}, \\ Pr\{X_{33} | X_{11} X_{21}\} &= 1 - Pr\{X_{31} | X_{11} X_{21}\} - \\ & Pr\{X_{32} | X_{11} X_{21}\} \\ &= 1 - 0.3 - Pr\{X_{32} | X_{11} X_{21}\}. \end{aligned}$$

应当指出,领域工程师往往只关心所关注状态的原因(例如 X_{31} 是 X_{41} 的唯一有意义的原因),而对其它状态的原因并不关心. 这可能导致实际上无法说明不被关注的状态的原因. 这不仅是因为增加了不必要的工作量和难度,而且是因为领域工程师可能根本就不知道如何单独说明(不是作为其它状态的补集来说明)这些状态的原因.

2.5 小结

将单赋值情况下适用的简洁表达方式和推理算法简单地通过应用式(1)而推广于多赋值情况是不恰当的. 这一方面是因为理论上无依据,另一方面是因为实际操作中会出现不合理、不一致、不精确和不可操作的情况.

多赋值情况的本质是:简洁表达方式必须解决一对矛盾:(a)满足子变量各状态因互斥而相关(必须满足状态概率归一);(b)多赋值子变量的各状态的原因被独立说明.(a)中的相关性与(b)中的独立性是矛盾的. 然而(a)必须被满足,(b)也是简洁表达所要求的. 由于用式(1)来解决这个矛盾并不恰当,我们必须寻找新的解决方案. DUCG 正是本文提出的解决方案.

应当指出,本文只用了 CSI 为例来解释多赋值情况下应用式(1)存在的问题,但这些问题同样存在于应用式(1)的所有其它模型中,例如 noisy-OR、DCD 等等.

同样需要指出的是,单赋值和多赋值仅仅与子变量有关,而与父变量无关. 单赋值子变量的父变量的活跃状态可以有多个. 图 7 示例的是一个单赋值情况. 其父变量 X_1 有 3 个状态, X_1 和另一个父变量 X_2 有两个活跃状态,但子变量 X_3 却只有一个赋值状态(单赋值). 注意,图 1(a),图 2(a)和图 7(a)是完全相同的,但它们却隐含着不同的因果关系.

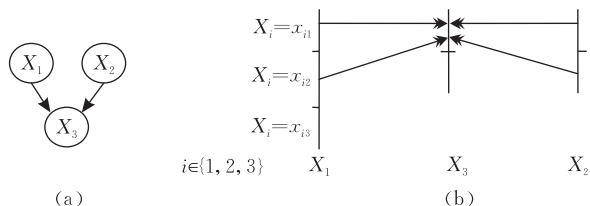


图 7 父变量活跃状态多于一个的单赋值情况示例

3 用于单赋值情况的 S-DUCG 模型

本节简要介绍 DCD 模型,因为它提供了 DUCG 模型的基本思想. 但是,在介绍 DCD 和提出 DUCG 之前,有必要定义一套新的符号体系.

在 DCD 和 DUCG 中,大写英文字母代表变量或事件,小写英文字母代表变量的概率分布或事件的发生概率;变量或事件的第 1 个下标用以区分变量,第 2 个下标用以区分变量的状态(即事件). 也就是说,第 1 和第 2 个下标共同确定某变量处于某状态这一事件.

例如, X_n 代表标号为 n 的变量, X_{nk} 代表变量 X_n 的第 k 个状态. X_{nk} 同时也是一个事件,即变量 X_n 处于状态 k . 相应地, $x_{nk} \equiv Pr\{X_{nk}\}$. 变量 X_n 与事件 X_{nk} 之间的差别在于 X_{nk} 有两个下标. 采用这套新的符号体系的原因在于方便变量状态或相应事件的表达. 这种方便在上一节的例子中已经显现.

为简单起见,除逻辑门变量外,本节的所有变量均为二状态变量. 读者须注意本节的有些表达式在多赋值情况下是不适用的.

3.1 DCD 的简要介绍

本节简要介绍 DCD 的基本概念. 理解 DCD 的基本概念非常重要,因为 DUCG 是从 DCD 发展而来的. 为便于读者理解,本节采用与 noisy-OR 和 CSI 对比的方式来介绍 DCD. 尽管 DCD 的提出比 CSI 早两年,但学术界对 DCD 并不熟悉,对 noisy-OR 和 CSI 却较熟悉. 所以,将三者进行简单对比有助于读者理解 DCD. 需要说明的是,本文不打算对三者进行系统比较,因为这种系统比较的工作量大,也超出了本文的范围.

学术界熟知的 noisy-OR 可用图 8(c)的示例说明,该图与文献[6]中的图 4.20 类似. 图中状态 1 代表 true 状态;状态 2 代表 false 状态. 例中, $Pr\{X_{31} | X_{11} X_{22}\} = 0.3$, $Pr\{X_{31} | X_{12} X_{21}\} = 0.6$ (这两式仅在单赋值情况下成立).

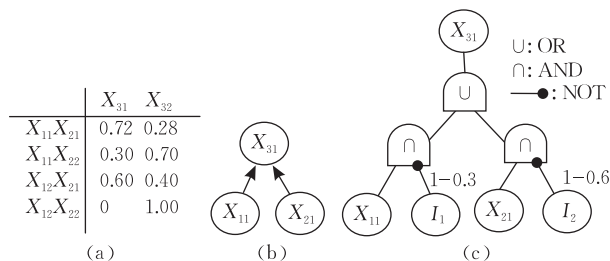


图 8 noisy-OR 示例

根据 noisy-OR,图 8(b)中的 CPT 可被算得如

图 8(a)所示. 其中,

$$Pr\{X_{31} | X_{11} X_{21}\} = 1 - (1 - Pr\{X_{31} | X_{12} X_{21}\})(1 - Pr\{X_{31} | X_{11} X_{22}\}) = 1 - (1 - 0.3)(1 - 0.7) = 0.72.$$

本例中, noisy-OR 只需要 2 个参数 (0.3 和 0.6) 就可以表达 CPT 中的 8 个参数.

然而, 对于图 9(a) 中所示的 CPT, noisy-OR 无能为力, 因为 X_{11} 和 X_{21} 彼此非逻辑或的关系. 图 9(b) 所示为 CSI 的解决方案, 其中左分支代表状态 1 (true 状态), 右分支代表状态 2 (false 状态).

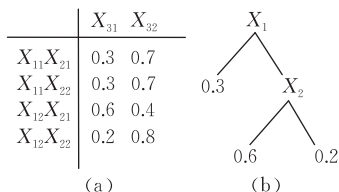


图 9 CSI 示例

本例中, CSI 只用了 3 个参数 (0.3, 0.6, 0.2) 就表达了 CPT 中的 8 个参数. 然而, 对于图 10(a) 所示的 CPT, 虽然 CSI 也能将其间接地表达如图 10(b) 所示, 但其简洁程度不够, 因为起作用的实际上只有一个参数: 0.3. 用 DCD, 本例可被更简洁地表达如图 11(a) 所示. 图 11(a) 直观地表达了 X_{11} 或 X_{21} 或两者共同都可以引起 X_{31} , 且其概率均为 0.3. 这正是图 10(a) 中的 CPT 实际上告诉我们的. 事实上, 图 8 和图 9 的情况也能用 DCD 分别表达如图 11(b) 和 (c) 所示. 按照文献 [13] 中的定义, 图 11(b) 中 X_{11} 和 X_{21} 的关系为逻辑或 (OR).

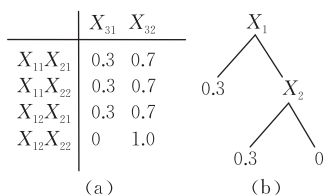


图 10 另一个 CSI 示例

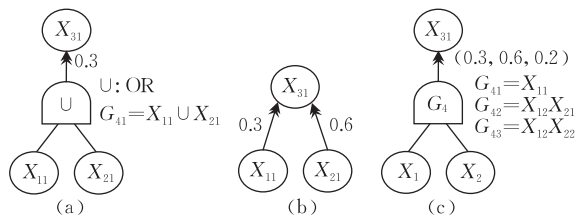


图 11 DCD 示例

表 3 逻辑门说明 (LGS_i)

j	G_{ij}
1	事件表达式 1
2	事件表达式 2
⋮	⋮
m	事件表达式 m

图 11(c) 中的逻辑门变量用 G_i 表达, 用以说明变量 X_1 和 X_2 之间复杂的逻辑关系. 本例中, G_4 是 X_3 的一个特殊父变量, 有 3 个互斥的状态: G_{41} 、 G_{42} 和 G_{43} . 这 3 个状态引起 X_{31} 的概率分别为 0.3、0.6 和 0.2. 作为对文献 [22] 中提出的 DCD 的发展, 本文提出任何 DCD 的逻辑门均可用表 3 来说明其逻辑关系. 这个表被称为逻辑门说明 LGS (Logic Gate Specification).

图 11(a) 中, LGS_4 说明 $G_{41} = X_{11} \cup X_{21}$. 这里, G_4 代表的是或门, 并且只有一个活跃状态. 但在图 11(c) 中, G_4 有 3 个活跃状态.

表达不同逻辑关系的逻辑门的图形符号不必有差别. 我们只需用统一的图形符号 \bigcirc_{G_i} 来代表任何逻辑门, 同时用表 3 定义的 LGS_i 来说明其中不同的逻辑关系. 这里, 下标 i 用以区分不同的逻辑门变量以及其它变量.

显然, 图 11(b) 与图 8(c) 是等价的. 图 11(b) 告诉我们的是: X_{11} 和 X_{21} 可独立或共同地引起 X_{31} , 因为按照 DCD 中的缺省定义, X_{11} 和 X_{21} 对 X_{31} 是逻辑或的关系. 在 DCD 中, 图 11(b) 的条件概率 $Pr\{X_{31} | X_{11} X_{22}\}$ 和 $Pr\{X_{31} | X_{12} X_{21}\}$ 被看成是两个独立的连接事件 $P_{31,11}$ 和 $P_{31,21}$ 的发生概率, 即

$$Pr\{X_{31} | X_{11} X_{22}\} = Pr\{P_{31,11}\} = p_{31,11},$$

$$Pr\{X_{31} | X_{12} X_{21}\} = Pr\{P_{31,21}\} = p_{31,21}.$$

实际上, 当人们给出 $Pr\{X_{31} | X_{11} X_{22}\}$ 时, 人们考虑的仅仅是 X_{11} 而不是 X_{22} , 甚至可能根本没有意识到 X_2 的存在. 对于 $Pr\{X_{31} | X_{12} X_{21}\}$ 来说, 情况也是同样的. 这是因为 X_1 和 X_2 通常处于不同的领域. 例如盗窃和地震尽管都能引起振动从而触发警报, 但却属于完全不同的领域, 人们很难把它们结合起来考虑, 通常是分别考虑. 这种情况可以用图 12 来进一步解释.

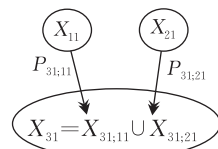


图 12 对图 11(b) 的解释

图中 X_{31} 既可以被 X_{11} 引起也可以被 X_{21} 引起. X_{31} 仅被 X_{11} 引起这一事件记为 $X_{31,11}$; X_{31} 仅被 X_{21} 引起这一事件记为 $X_{31,21}$. $X_{31,11}$ 或 $X_{31,21}$ 之一发生或同时发生均可引起 X_{31} 发生. 在 $X_{31,i}$ 和 X_{i1} ($i \in \{1, 2\}$) 之间存在一个不确定的作用机理. 给定 X_{i1} 发生, 当这个机理起作用时, $X_{31,i1}$ 发生; 当这个机理不

起作用时, $X_{31;i1}$ 不发生. 在 DCD 中, 这个不确定的作用机理由事件 $P_{31;i1}$ 来表达. $P_{31;i1}$ 可被理解为: 在 X_{i1} 发生的条件下, X_{i1} 引起 $X_{31;i1}$ 、从而引起 X_{31} 这样一个随机事件. $P_{31;i1}$ 的发生与否表达了 X_{i1} 引起 $X_{31;i1}$ 从而引起 X_{31} 的不确定性. 采用事件表达式的表述: $X_{31;i1} = P_{31;i1} X_{i1}$. 其中, $P_{31;i1}$ 和 X_{i1} 处于逻辑与的关系. 由于 $P_{31;i1}$ 是否发生是不确定的、且独立于任何其它因素, 可知 $P_{31;i1}$ 是一个独立的随机事件.

DCD 和 DUCG 中的一些特定类型变量的定义:

X 代表结果或效果事件或变量, 画为圆形. 它同时也可以作为原因事件或变量;

B 代表基本或根 (root) 事件或变量, 画为矩形. 它只能是一个独立的原因;

G 代表逻辑门事件或变量;

P 代表连接事件或变量. 其中, $\mathbf{P}_{n1;i}$ 代表事件矩阵 $[P_{n1;i1} \ P_{n1;i2} \ \dots \ P_{n1;im}]$.

图 13 是这些变量应用的示例. 如图 13(a) 所示, $P_{n1;i1}$ 、 $P_{n1;h1}$ 和 $P_{n1;g}$ 分别为从父事件或变量 X_{i1} 、 B_{h1} 和 G_g 到子事件 X_{n1} 之间的有向弧. 作为单赋值变量, X_{n1} 只有一个赋值状态: X_{n1} . 作为 X_{n1} 的父变量, X_{i1} 有一个活跃状态 X_{i1} , B_{h1} 有一个活跃状态 B_{h1} , G_g 有多个活跃状态. 这就是为什么连接事件 $P_{n1;i1}$ 、 $P_{n1;h1}$ 和连接事件矩阵 $\mathbf{P}_{n1;g}$ 被分别使用的原因. 为简单起见, $P_{n1;i1}$ 、 $P_{n1;h1}$ 、 $\mathbf{P}_{n1;g}$ 和变量状态通常被省略. 于是图 13(a) 通常被画成图 13(b).

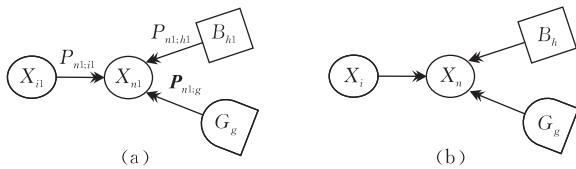


图 13 连接事件示例

借助于 P 事件, 我们可以将图 11(b) 的因果关系写为事件的积之和的表达式:

$$X_{31} = X_{31;i1} \cup X_{31;21} = P_{31;i1} X_{11} \cup P_{31;21} X_{21} \quad (27)$$

在 $X_{11} X_{21}$ 的条件下, 我们有

$$X_{31} | X_{11} X_{21} = P_{31;i1} \cup P_{31;21}.$$

应用概率论的容斥原理或迪·摩根定律, 我们有

$$\begin{aligned} Pr\{X_{31} | X_{11} X_{21}\} &= Pr\{P_{31;i1} \cup P_{31;21}\} \\ &= Pr\{P_{31;i1}\} + Pr\{P_{31;21}\} - Pr\{P_{31;i1}\} Pr\{P_{31;21}\} \\ &= p_{31;i1} + p_{31;21} - p_{31;i1} p_{31;21} \\ &= 0.3 + 0.6 - 0.3 \times 0.6 = 0.72, \end{aligned}$$

或

$$\begin{aligned} Pr\{X_{31} | X_{11} X_{21}\} &= Pr\{\overline{X_{31}} | X_{11} X_{21}\} \\ &= 1 - Pr\{\overline{X_{31}} | X_{11} X_{21}\} \end{aligned}$$

$$\begin{aligned} &= 1 - Pr\{\overline{P_{31;i1} X_{11} \cup P_{31;21} X_{21}} | X_{11} X_{21}\} \\ &= 1 - Pr\{\overline{P_{31;i1}} \cup \overline{P_{31;21}}\} \\ &= 1 - Pr\{\overline{P_{31;i1}} \overline{P_{31;21}}\} \\ &= 1 - Pr\{\overline{P_{31;i1}}\} Pr\{\overline{P_{31;21}}\} \\ &= 1 - (1 - 0.3)(1 - 0.6) \\ &= 0.72. \end{aligned}$$

后者正是 noisy-OR 告诉我们的. 在 noisy-OR 中, $1 - p_{nk;ij} \equiv Pr\{\overline{P_{nk;ij}}\}$ 被看成是图 8(c) 中的禁事件 (inhibitor) I_i 的发生概率.

我们可以将 X_{11} 和 X_{21} 又作为两个子事件, 继续按照式(27)展开 X_{31} 的方式将其展开为事件的积之和的表达式. 在 DAG 的情况下, 这个过程可以一直持续到 B 类型变量而结束. 这个过程被称为事件展开. 任何事件或以任何逻辑关系组成的事件的集合都可以按照这种方式展开为事件的积之和的表达式. 在展开过程中, 逻辑运算中的与、或、非、互斥或、吸收、互斥、取补等算法都可以应用. 例如, 设我们的问题是 “ $Pr\{H_{kj} | E\} = ?$ ”. 其中 H_{kj} 代表所关注的假设事件或事件表达式 (例如 $H_{11} = B_{12}$, $H_{21} = X_{11}$, $H_{22} = X_{12}$, $H_{31} = X_{21} B_{13}$, $H_{32} = X_{22} B_{12}$ 等等), E 代表一组处于逻辑与关系中的事件. 按照条件概率公式, 我们有

$$Pr\{H_{kj} | E\} = \frac{Pr\{H_{kj} E\}}{Pr\{E\}} \quad (28)$$

我们可以将 $H_{kj} E$ 和 E 分别展开为两个仅由 P 和 B 类型的事件组成的积之和的事件表达式, 通过逻辑运算使得表达式积内去重、积间互斥, 然后将表达式中的大写字母 (P 或 B 类型事件) 替换为小写字母 (p 或 b 类型参数), 即可算出 $Pr\{H_{kj} E\}$ 和 $Pr\{E\}$ 的值, 进而用式(28)算出 $Pr\{H_{kj} | E\}$ 的值. 这里, 进行概率计算的事件表达式必须是互斥的积之和的表达式. 为获得互斥积之和的事件表达式, 我们可以反复应用下述公式 (详见文献[22]):

$$C_1 \cup C_2 \cup \dots \cup C_n = C_1 + \overline{C_1} C_2 + \overline{C_1} \overline{C_2} C_3 + \dots + \overline{C_1} \overline{C_2} \dots \overline{C_{n-1}} C_n \quad (29)$$

$$\begin{aligned} \overline{C} &= \overline{V_{1j_1} V_{2j_2} \dots V_{mj_m}} \\ &= \overline{V_{1j_1}} + V_{1j_1} \overline{V_{2j_2}} + V_{1j_1} V_{2j_2} \overline{V_{3j_3}} + \dots + \\ &\quad V_{1j_1} \dots V_{m-1, j_{m-1}} \overline{V_{mj_m}} \end{aligned} \quad (30)$$

其中, $V \in \{X, P, G, B\}$, $C = V_{1j_1} V_{2j_2} \dots V_{mj_m}$, j_i 是变量 V_i 的状态下标, “+” 是互斥或算符. C 通常被称为割集 (cutset)、代表不同展开阶段的事件的积. 反复应用式(28)和(29), 同时应用事件的上述算法, 我们即可得到仅由 P 和 B 类型独立事件组成的互斥的积之和表达式. 简而言之, 上述算法由两步组成: (1) 通过事件展开获得仅由 P 和 B 类型事件组成的

互斥积之和的事件表达式; (2) 用小写字母替换表达式中相应的大写字母进行数值计算. 有时, 我们只需要第 1 步即可得到推理答案. 例如: 在一些诊断类问题中, 我们通过第 1 步发现 $H_{kj}E \neq 0, H_{g^y}E = 0$. 这里, “0”代表空集, H_{g^y} 代表除 H_{kj} 外的所有其它可能假设. 于是, 我们无须数值计算即知 H_{kj} 是给定证据 E 条件下的唯一可能的诊断结果, 概率参数无用. 在以下各节中, 我们将会看到这种算法带给我们的巨大的表达和推理方面的好处.

在结束介绍 DCD 之前, 必须指出, DCD 能够表达各种复杂的不确定因果关系, 例如但并不限于图 14~21 所示.

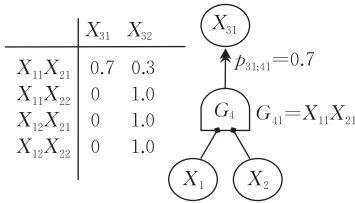


图 14 AND 门示例

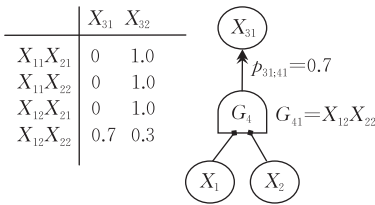


图 15 NOT 门示例

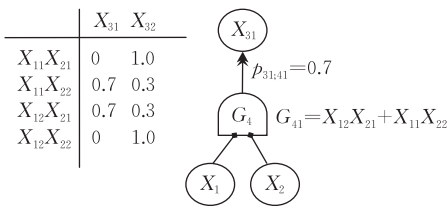


图 16 XOR 门示例

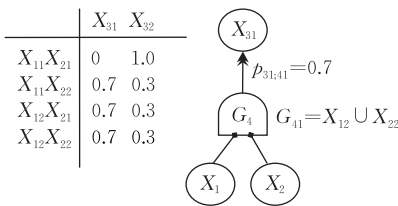


图 17 NOT-AND 门示例

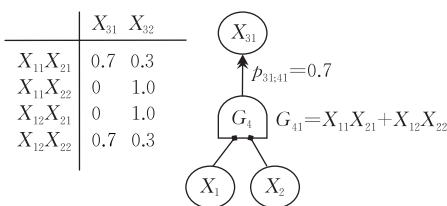


图 18 NOT-XOR 门示例

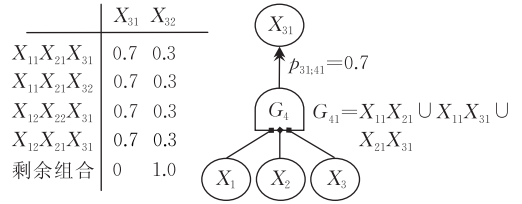


图 19 2/3 逻辑门示例

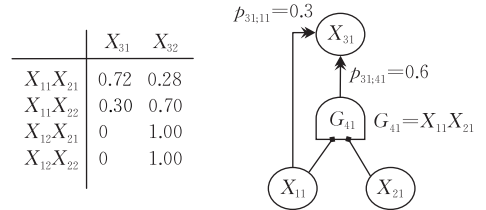


图 20 特殊逻辑组合示例

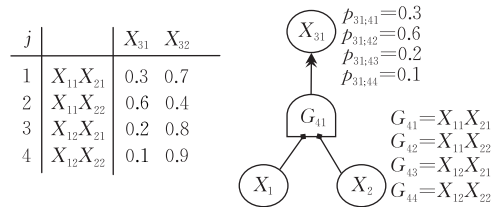


图 21 全组合逻辑门示例

在图 21 中, j 标识父变量的状态组合. 理论上, 任何 CPT 都可以用图 21 所示的 DCD 的全组合逻辑门来表达. 显然, 全组合逻辑门等价于 CPT. 这是最坏的情况, 因为应用全组合逻辑门实际上并不能获得表达上的简洁性. 然而, 全组合逻辑门却显示了 DCD 具有尽可能简洁地表达任何不确定因果关系的能力.

需要说明的是, 一个逻辑门可以是另一个逻辑门的输入变量, 多个逻辑门可以是同一子变量的父变量, 逻辑门可以是子变量的部分父变量 (如图 20 所示). 因此, 逻辑门的确是简洁表达变量间各种复杂的不确定因果关系的一种强有力的工具.

应当指出, 文献 [25] 提出的因果独立模型 (Causal Independence Model, CIM) 中的确定作用函数 (Deterministic Interaction Function) $f(I_1, \dots, I_n)$ 与逻辑门 G_g 有类似之处. 在 CIM 中, I_1, \dots, I_n 是子变量 X_g 和其父变量 V_1, \dots, V_n 之间的中介变量; $Pr\{f(I_1, \dots, I_n) | I_1, \dots, I_n\}$ 与 $Pr\{G_{gk} | SCPV_{g,j}\}$ 类似; 只能等于 1 或 0. 其中, $SCPV_{g,j}$ 代表逻辑门 G_g 的输入变量或 X_g 的父变量的第 j 个状态组合. $f(I_1, \dots, I_n)$ 与 G_g 的区别在于: (1) G_{gk} 不等于 X_{gk} , 但 $f(I_1, \dots, I_n)$ 等于 X_{gk} ; (2) 前者用 $Pr\{I_i | V_i\}$ 的乘积来计算不确定性, 其条件是 V_1, \dots, V_n 彼此独立. 后者的输入变量彼此可相关; (3) 对应于 $f(I_1, \dots,$

I_n)的 V_1, \dots, V_n 是子变量的所有父变量, 但逻辑门的输入变量可以是子变量的部分父变量, 且逻辑门可以作为其它逻辑门的输入变量。

有人认为逻辑门虽然统一了对任何逻辑关系的表达, 但其增加的图形表达比隐含了任何逻辑关系的 CPT 的表达更复杂, 从而否定引入逻辑门的意义. 但如果注意到逻辑门所具有的 CPT 所没有的强大的简洁表达功能, 这种否定就不成立了. 其实, 引入逻辑门的更重要的意义还在于其导致了 DCD 中连接事件的引入和事件展开算法的引入, 这使得 DCD 不仅能处理 DAG(有向无环图), 而且能处理 DCG(有向循环图, 详见文献[22]).

3.2 对 DCD 的进一步发展: S-DUCG

S-DUCG(Single-valued DUCG)在 DCD 的基础上增加了两个工具: 条件连接事件和缺省事件。

3.2.1 条件连接事件

在图 20 中我们注意到, X_{11} 和 X_{21} 同时发生引起 X_{31} 的概率是 0.72 而不是 0.6, 但 0.6 却可能是人们主观上想要表达的. 图 20 所展示的直观想法是: 当仅 X_{11} 发生时, X_{31} 的发生概率为 0.3; 当 $X_{11}X_{21}$ 发生时, X_{31} 的发生概率为 0.6. 然而, 由于 $P_{31;11}$ 和 $P_{31;41}$ 处于逻辑或的关系, $X_{11}X_{21}$ 引起 X_{31} 的概率从 0.6 增加到了 0.72, 而这并不是人们想要表达的. 为避免这种增加, 可将图 20 改为图 22. 其中, \rightarrow 被定义为条件连接事件。

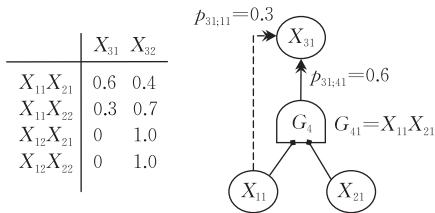


图 22 条件连接事件示例

$P_{31;11}$ 的条件被记为 $Z_{31;11} = \bar{X}_{21} = X_{22}$, 即, 当 X_{21} 不存在时, $P_{31;11}$ 存在, 否则 $P_{31;11}$ 不存在. 这里, $P_{31;11}$ 绑定了 $Z_{31;11}$, 并且可被表达为 $P_{31;11} Z_{31;11} = P_{31;11} X_{22}$. 一般来说, 条件连接事件 $P_{nk;ij}$ 和条件连接事件矢量 $\mathbf{P}_{nk;i}$ 的条件可被分别记为 $Z_{nk;ij}$ 和 $\mathbf{Z}_{nk;i}$, 带条件的 $P_{nk;ij}$ 和 $\mathbf{P}_{nk;i}$ 可被分别表达为 $P_{nk;ij} Z_{nk;ij}$ 和 $\mathbf{P}_{nk;i} \mathbf{Z}_{nk;i}$. 其中, 在单赋值情况下, k 标识子变量的赋值状态. 以图 22 为例, $n=3, k=1$, 事件表达式为

$$\begin{aligned} X_{31} &= P_{31;11} Z_{31;11} X_{11} \cup P_{31;41} G_{41} \\ &= P_{31;11} X_{22} X_{11} + P_{31;41} X_{11} X_{21}, \\ Pr\{X_{31}\} &= Pr\{P_{31;11} X_{22} X_{11} + P_{31;41} X_{11} X_{21}\} \\ &= Pr\{P_{31;11}\} Pr\{X_{11} X_{22}\} + \end{aligned}$$

$$\begin{aligned} &Pr\{P_{31;41}\} Pr\{X_{11} X_{21}\} \\ &= 0.3 Pr\{X_{11} X_{22}\} + 0.6 Pr\{X_{11} X_{21}\}, \\ Pr\{X_{31} | X_{11} X_{22}\} &= Pr\{(P_{31;11} X_{22} X_{11} + \\ &P_{31;41} X_{11} X_{21}) | X_{11} X_{22}\} \\ &= Pr\{P_{31;11}\} = 0.3, \\ Pr\{X_{31} | X_{11} X_{21}\} &= Pr\{(P_{31;11} X_{22} X_{11} + \\ &P_{31;41} X_{11} X_{21}) | X_{11} X_{21}\} \\ &= Pr\{P_{31;41}\} = 0.6. \end{aligned}$$

显然, 条件连接事件还可用于许多其它情况, DCD 的表达能力可由此得到极大扩展. 以文献[6]的盗窃警报系统为例, 如果盗窃(X_{11})和地震(X_{21})通过相同的作用机理(振动)引起警报(X_{31}), 我们可以用图 23 而非图 18 来表达其因果关系. 其中, $Z_{31;11} = \bar{X}_{21} = X_{22}$.

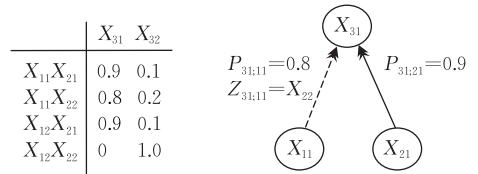


图 23 盗窃报警系统示例

图 23 中, 仅当无地震时, 盗窃产生的振动对于警报才有意义. 反之, 当同时出现地震时, 其振动远远超过盗窃的振动, 使得盗窃振动并不能增加报警响的几率. 在地震的强振动下, 仍然不确定的是报警装置是否正常. 因此, X_{11} 和 X_{21} 的同时发生并不能增加 X_{31} 发生的概率, 仍应为 0.9, 而不是 $0.9 + 0.8 - 0.9 \times 0.8 = 0.98$.

进而, 设老鼠(X_{41})也能引起振动, 其概率为 0.6. 即老鼠、盗窃和地震都能引起振动, 从而引起警报. 但老鼠振动并不能加强盗窃振动, 老鼠和盗窃振动并不能加强地震振动, 则报警系统的不确定因果关系将如图 24 所示. 其中, $Z_{31;41} = X_{12} X_{22}$, $Z_{31;11} = X_{22}$.

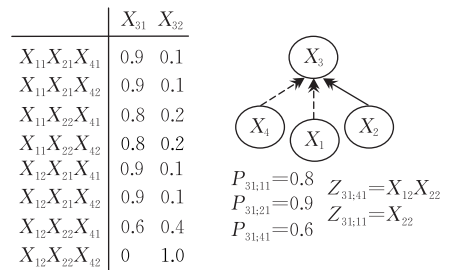


图 24 老鼠、盗窃和地震情况示例

应当指出, 绑定于 $P_{nk;ij}$ 的条件 $Z_{nk;ij}$ 的内容是很宽泛的. 事实上, $Z_{nk;ij}$ 可以是任何可判定其真伪的事件. 这个事件不仅包括子变量 X_n 的父变量处于

何种状态, 也包括因果图中任何地方的任何变量的状态, 甚至包括因果图中不能图形表达的事件. 例如,

$$\begin{aligned}
 Z_{nk,ij} &= \bar{P}_{hy,gm} X_{gm}, \\
 Z_{nk,ij} &= \{|\lambda - \beta| \leq \sigma\}, \\
 Z_{nk,ij} &= \{\lambda \geq \beta\}, \\
 Z_{nk,ij} &= \{\sqrt{\lambda^2 + \beta^2} \geq \sigma\}, \\
 Z_{nk,ij} &= \{E_1 \text{ 早于 } E_2 \text{ 出现}\}
 \end{aligned}$$

等等. 其中, X_{gm} 和 $P_{hy,gm}$ 可出现在因果图中的任何地方; λ, β 和 σ 可以是任何包含或未包含在因果图中的物理参数; E_1 和 E_2 可以是任何包含或未包含在因果图中的事件.

基于所观察到的证据或事件展开的结果, $Z_{nk,ij}$ 可以被判定为真 ($Z_{nk,ij} = 1$), 也可以被判定为假 ($Z_{nk,ij} = 0$). 如果 $Z_{nk,ij}$ 是否发生无法判定, 则包含 $Z_{nk,ij}$ 的事件表达式必须与 $(Z_{nk,ij} + \bar{Z}_{nk,ij})$ 相乘再展开, 并且 $Z_{nk,ij}$ 的验前概率分布也必须被给定. 否则该事件表达式的概率将无法计算. 当然, 这种情况并不常见, 因为定义 $Z_{nk,ij}$ 时就应当考虑到可判定其真伪.

由上可见, 本文提出的条件连接事件可以表达远比 CSI 或文献 [26] 提出的 CBN (Contingent Bayesian Network) 所能表达的更为复杂的逻辑关系. 仅当 $Z_{nk,ij}$ 只涉及 X_n 的父变量的状态时, 条件连接事件的表达才类似于 CSI 和 CBN, 但其推理算法仍不同. 实际上, 当条件连接事件仅仅涉及父变量的状态时, 其所表达的内容也可以被 DCD 的逻辑门替代. 但即使在这种简单情况下, 条件连接事件的表达也比逻辑门更加直观方便, 何况条件事件并不局限于父变量的状态.

用新增变量替代条件连接事件来表达相应的条件逻辑关系也是有可能的. 但是, 在许多情况下, 新增变量非常困难. 例如, X_{11} 早于 X_{21} 发生这一事件就很难用新增变量的方式来表达, 因为 X_{11} 和 X_{21} 在因果图中已经存在. 对于 $Z_{nk,ij} = \bar{P}_{hy,gm} X_{gm}$ 的情况, 由于 $P_{hy,gm}$ 是有向弧, 也很难用新增变量来表达. 实际上, $\bar{P}_{hy,gm} X_{gm}$ 意味着虽然 X_{gm} 发生, 但 X_{hy} 并未发生, 因为 $P_{hy,gm}$ 未发生. 这个绑定于普通连接事件的条件在实践中很常见, 却很难用新增变量的方式来表达. 可见, 条件连接事件为表达变量间复杂的逻辑关系提供了一个方便和直观的图形工具, 其引入是对现有不确定因果关系表达的实质性扩充.

最后应当指出, 条件连接事件的意义已经超出了对 CPT 的简洁表达的意义. 因为在一个 CPT 中, 一旦给定父变量的状态组合, 子变量的状态概率分

布就给定了. 但在条件连接事件包含父变量以外的事件、或包含父变量事件发生顺序的情况下, CPT 不仅取决于父变量的状态组合, 还取决于其它事件. 这已经超越了 BN 中 CPT 的定义. 所以, 本文引入的条件连接事件已经突破了 BN 的理论框架.

3.2.2 缺省事件

文献 [18-20] 指出, 在许多情况下, 子变量的原因并不能被全部表达在模型中. 换句话说, 即使子变量的所有父变量都处于 false 状态, 子变量仍可能存在缺省的非 (0, 1) 的状态概率分布. 这个概率分布是由某些未知的或未能明确表达的原因引起的. 这些原因可用 Leak^[18,20] 或 Dummy^[19] 变量表达. Leak 或 Dummy 变量的状态只有一个: true. 即这是一个发生概率恒等于 1 的必然事件. 但为方便起见, 我们仍称其为变量, 尽管它的状态永不改变.

为反映 Leak 变量对概率计算的影响, 文献 [18] 提出了如下公式

$$Pr\{X_n | \bigcap_{i=1}^m X_i\} = 1 - (1 - p_{n,0}) \prod_{i=1}^m \frac{1 - p_{n,i}}{1 - p_{n,0}}$$

其中, $X_i (i \in \{1, \dots, m\}, i \neq n)$ 是显式 (explicit) 表达的 X_n 的原因事件, $p_{n,0}$ 为 Leak 事件导致 X_n 事件发生的概率. 然而, 上述公式仅当不存在或只有一个显式原因时才好理解. 当显式原因多于一个时, 多出的原因的概率贡献就都被放大了 $1/(1 - p_{n,0})$. 这让人费解. 文献 [18] 并未给出解释.

在 S-DUCG 中, X_n 的这个未知或未明确表达的原因被定义为缺省变量 D_n , 并用图形 $\diamond D_n$ 表示. 与其它变量类似, 由一个连接变量 $P_{nk;nD}$ 将 D_n 与 X_n 相连. D_n 与其它变量的唯一区别在于 D_n 只有一个必然发生的状态, 即 $Pr\{D_n\} \equiv 1$. 在许多情况下, 仅当所有显式表达的原因事件均不存在时, D_n 才起作用, 并导致 X_n 的缺省概率分布. 如图 25(c) 所示, 设

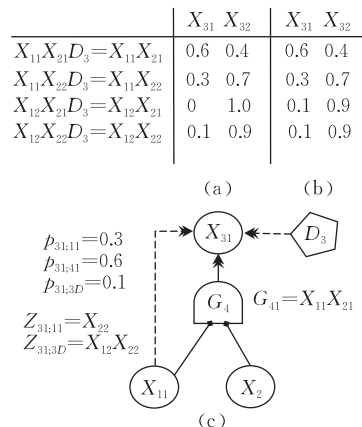


图 25 S-DUCG 中的缺省事件示例

$P_{31;11}$ 的条件是 $Z_{31;11} = \bar{X}_{21} = X_{22}$, $P_{31;3D}$ 的条件是 $Z_{31;3D} = \bar{X}_{11} \bar{X}_{21} = X_{12} X_{22}$, 则 CPT 如图 25(a) 所示. 与图 22 比较, $Pr\{X_{31} | X_{12} X_{22}\} = 0.1$ 而非 0. 这是因为有了来自 D_n 的贡献.

当然, D_n 也可以按其它方式使用. 如上例, 我们可以定义 $Z_{31;3D} = \bar{X}_{11} = X_{12}$, 则 CPT 将如图 25(b) 所示. 又如 D_n 可以被当作一个普通的变量对待, 用普通连接事件 $P_{nk;nD}$ 而非条件连接事件将其连接到 X_n , 则 D_n 将作为一个背景变量发挥作用.

4 多赋值情况下的 M-DUCG 模型

基于文献[18]提出的扩展 noisy-OR 方法的基本思想, 文献[19]和文献[20]分别提出了两个适用多赋值情况的具体方案, 但只能用于梯度变量. 本文提出的 M-DUCG 模型则可用于任何情况.

4.1 M-DUCG 的基本概念

作为本文提出的 DUCG 模型的组成部分, M-DUCG 是处理多赋值情况的一个新的理论模型. 本节和以后各节将循序予以说明. 首先, M-DUCG 基于如下假设.

假设 1. 设 $V_i (V_i \in \{X, B, G\})$ 是 X_n 的父变量, 则下式成立:

$$X_{nk} = \sum_i (r_{ni}/r_n) \sum_{j_i} A_{nk;ij_i} V_{ij_i} \quad (31)$$

其中, j_i 标识父变量 V_i 的状态; r_{ni} 定义为 X_n 和 V_i 之间的关联度; $r_n = \sum_i r_{ni}$; 与 $P_{nk;ij}$ 类似, $A_{nk;ij_i}$ 定义为在 V_{ij_i} 已发生的条件下 V_{ij_i} 独立导致 X_{nk} 发生这一随机事件. 为与 S-DUCG 中的连接事件相区别, $A_{nk;ij_i}$ 被称为父事件 V_{ij_i} 与子事件 X_{nk} 之间的作用事件. 值得注意的是, 式(31)是由绑定了系数 (r_{ni}/r_n) 的事件组成的积之和的表达式.

图 26 是对这个假设的解释. 其中, 为简单起见, $i \in \{1, 2, \dots, m\}$, $n \notin \{1, 2, \dots, m\}$.

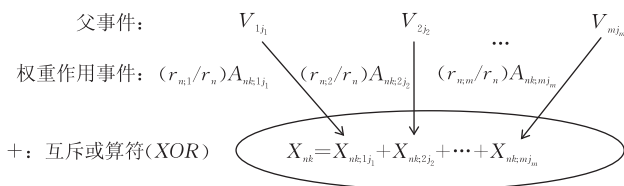


图 26 M-DUCG 模型示例

注意: 图中有向弧的形状是 \rightarrow , 而不是 \dashrightarrow , 也不是 \dashrightarrow . 前者表达的是权重后的独立因果链接. 换句话说, \rightarrow 标识了 CPT 中的父变量; \dashrightarrow 代表连接

事件变量; \dashrightarrow 代表权重后的作用事件变量 $\mathbf{F}_{n;i} \equiv (r_{ni}/r_n)\mathbf{A}_{n;i}$. 其中, $\mathbf{A}_{n;i}$ 是一个以 $A_{nk;ij_i}$ 为元素的事件矩阵, k 标识矩阵的行, j_i 标识矩阵的列; $\mathbf{F}_{n;i} \equiv (r_{ni}/r_n)\mathbf{A}_{n;i}$ 被称为权重作用事件变量, 是一个事件矩阵, 其元素为 $F_{nk;ij_i} \equiv (r_{ni}/r_n)A_{nk;ij_i}$. 为简单起见, 在不引起混淆的情况下, j_i 可简记为 j .

对 $A_{nk;ij}$ 的解释类似于图 12 中对 $P_{nk;ij}$ 的解释. 图 12 和图 26 有两个显著的区别: (1) 在图 12 中, $X_{31;11}$ 和 $X_{31;21}$ 的关系是逻辑或 (OR); 在图 26 中, $X_{nk;ij}$ 和 $X_{nk;ij'}$ ($j \neq j'$) 的关系是互斥或 (XOR); (2) 在图 26 中, 权重系数 (r_{ni}/r_n) 被绑定于 $A_{nk;ij}$; 在图 12 中, 没有任何系数被绑定于 $P_{nk;ij}$.

类似于 $P_{nk;ij}$, $A_{nk;ij}$ 代表 V_{ij} 已发生条件下 V_{ij} 确实引发 $X_{nk;ij}$ 的不确定作用机理. 如果这个机理起了作用, 则 $X_{nk;ij}$, 从而 X_{nk} 发生; 如果这个机理未起作用, 则 $X_{nk;ij} = 0$ (空集). 由于这个作用机理独立于任何其它因素, $A_{nk;ij}$ 是一个独立的随机事件. 当然, 对于 $k \neq k'$, $A_{nk;ij}$ 与 $A_{nk';ij}$ 互斥, 因为 X_{nk} 与 $X_{nk'}$ 互斥. 定义 $a_{nk;ij_i} \equiv Pr\{A_{nk;ij_i}\}$, 则 $a_{nk;ij_i}$ (简记为 $a_{nk;ij}$) 就是由领域工程师分别给定的原始参数. 通常情况下, 这些参数满足

$$\sum_k a_{nk;ij_i} = 1 \text{ 或 } \sum_k a_{nk;ij} = 1 \quad (32)$$

这对应于

$$\sum_k A_{nk;ij_i} = 1 \text{ 或 } \sum_k A_{nk;ij} = 1 \quad (33)$$

式(33)中, “1”代表全集.

对 r_{ni} 的解释如下: 在某些情况下, 领域工程师对 V_i 和 X_n 之间是否存在因果关系没有把握. 这类不确定性由 r_{ni} 表征. 即 (a) 当领域工程师确定两者之间存在因果关系时, $r_{ni} = 1$; (b) 当领域工程师确定两者之间不存在因果关系时, $r_{ni} = 0$; (c) 介于 (a) 和 (b) 之间的情况由 $0 < r_{ni} < 1$ 表达. 由于 (b) 不可能出现, 因为 $r_{ni} = 0$ 意味着 V_i 不是 X_n 的父变量, 所以 $r_{ni} \neq 0$. 于是我们有 $1 \geq r_{ni} > 0$. 又由于 r_{ni} 总是以 (r_{ni}/r_n) 的形式出现, 因而 r_{ni} 是否 ≤ 1 并不重要. 有时, 人们也可以用 $r_{ni} > 1$ 来强调 X_n 与 V_i 之间的关联度高于 X_n 与其它父变量之间的关联度. 于是, $1 \geq r_{ni} > 0$ 的限制条件被放松为 $r_{ni} > 0$. 实际上, 真正起作用的是 (r_{ni}/r_n) , 它是 V_i 对 X_n 的概率分布的贡献的权重. 由于权重系数 (r_{ni}/r_n) 的存在, 尽管 $A_{nk;ij}$ 能够导致 $X_{nk;ij}$ 从而导致 X_{nk} , 但其概率却因其它父变量的平衡而被缩减 (r_{ni}/r_n) . 注意: 具有不同的 k 或 j 但相同的 n 和 i 的 $A_{nk;ij}$ 在引起 $X_{nk;ij}$ 从而 X_{nk} 的时候绑定了相同的权重 (r_{ni}/r_n) .

按照式(31), 运用事件算法(例如 $V_{11} | V_{11} V_{22} = 1$ 和 $V_{11} | V_{12} V_{22} = 0$), 我们可以很容易地得到下述结果:

$$Pr\{X_{nk} | \bigcap_i V_{ij_i}\} = \sum_i (r_{ni}/r_n) a_{nk;ij_i} \quad (34)$$

$$\begin{aligned} x_{nk} &\equiv Pr\{X_{nk}\} = \sum_i (r_{ni}/r_n) \sum_{j_i} a_{nk;ij_i} Pr\{V_{ij_i}\} \\ &= \sum_i (r_{ni}/r_n) \sum_{j_i} a_{nk;ij_i} v_{ij_i} \end{aligned} \quad (35)$$

其中, $v_{ij_i} \equiv Pr\{V_{ij_i}\}$. 应用相同方法, 读者不难得出在部分父变量状态组合条件下的结果.

以上我们看到, 式(31)分别说明了子变量不同状态的原因, 这是多赋值情况的简洁表达所需要的. 下面我们证明式(31)自动满足子变量状态概率归一.

证明. 根据 $\sum_k A_{nk;ij_i} = 1$, $\sum_{j_i} V_{ij_i} = 1$ 和 $r_n \equiv \sum_i r_{ni}$, 由式(31), 我们有

$$\begin{aligned} \sum_k X_{nk} &= \sum_k \sum_i (r_{ni}/r_n) \sum_{j_i} A_{nk;ij_i} V_{ij_i} \\ &= \sum_i (r_{ni}/r_n) \sum_{j_i} V_{ij_i} \sum_k A_{nk;ij_i} = 1. \end{aligned}$$

证毕.

应当注意到: 在 M-DUCG 中, 父变量因权重系数 (r_{ni}/r_n) 的存在而相关, 同时参数 $a_{nk;ij_i}$ 和 r_{ni} 又是独立给定的, 可见, M-DUCG 巧妙地解决了子变量的状态因互斥而相关和各父变量与子变量的因果关系独立表达之间的矛盾. 这使得我们能够在多赋值情况下简洁表达 CPT. 例如, 设 1 个子变量和 5 个父变量各有 5 个状态, 则 CPT 的参数有 $5^6 = 15625$ 个. 采用 M-DUCG 的简洁表达, 其参数仅有 $5^3 + 5 = 130$ 个.

本质上, 假设 1 是建立在如下认识基础之上的: 每个父变量独立贡献一个被权重了的子变量的状态概率分布, 所有父变量贡献的权重后的状态概率分布之和构成子变量的最终状态概率分布, 子变量究竟处于哪个状态由这个概率分布随机决定.

这个认识非常简单, 实际上就是领域工程师对客观世界的直观认识. 因而, M-DUCG 很容易应用, 尤其适用于参数由领域工程师根据自己的经验直接给定的情况.

图 27 为 M-DUCG 的一个例子. 其中, 子变量 X_n 有两个父变量 $V_1 = B_1$ 和 $V_2 = B_2$; 3 个变量各有 3 个状态; X_n 的任何状态都可以由任何一个父变量的任何一个状态独立引起. 本例反映的是两个父变量属于两个独立的领域、且子变量的各状态处于完

全等价地位的情况.

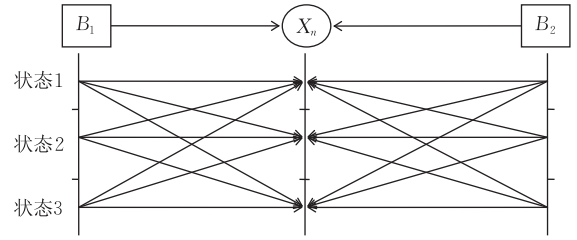


图 27 多赋值情况示例

基于 M-DUCG, 领域工程师可以很容易地给出参数 $a_{nk;ij_i}$, $b_{ij} \equiv Pr\{B_{ij}\}$ 和 r_{ni} . 例如, 图 27 的参数可被分别给定如下:

X_n 和 B_1 之间的参数:

$$a_{n;1} \equiv \begin{bmatrix} a_{n1;11} & a_{n1;12} & a_{n1;13} \\ a_{n2;11} & a_{n2;12} & a_{n2;13} \\ a_{n3;11} & a_{n3;12} & a_{n3;13} \end{bmatrix} = \begin{bmatrix} 0 & 0.1 & 0.3 \\ 0.4 & 0.2 & 0.3 \\ 0.6 & 0.7 & 0.4 \end{bmatrix}$$

$$b_1 \equiv [b_{11} \quad b_{12} \quad b_{13}]^T = [0.3 \quad 0.5 \quad 0.2]^T,$$

$$r_{n;1} = 1.$$

X_n 和 B_2 之间的参数:

$$a_{n;2} \equiv \begin{bmatrix} a_{n1;21} & a_{n1;22} & a_{n1;23} \\ a_{n2;21} & a_{n2;22} & a_{n2;23} \\ a_{n3;21} & a_{n3;22} & a_{n3;23} \end{bmatrix} = \begin{bmatrix} 0.8 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.3 \\ 0 & 0.2 & 0.5 \end{bmatrix}$$

$$b_2 \equiv [b_{21} \quad b_{22} \quad b_{23}]^T = [0.4 \quad 0.4 \quad 0.2]^T$$

$$r_{n;2} = 3,$$

这里假定了 X_n 与 B_2 的关联度是 X_n 与 B_1 的关联度的 3 倍. 于是, $r_n = r_{n;1} + r_{n;2} = 1 + 3 = 4$.

本例中我们看到, 与 B_1 和 B_2 有关的两组参数是分别独立给定的. 事实上, 究竟 $r_{n;1} = 1$ 和 $r_{n;2} = 3$ 还是 $r_{n;1} = 0.3$ 和 $r_{n;2} = 0.9$ 无关紧要, 因为两者都有 $(r_{n;1}/r_n) = 0.25$ 和 $(r_{n;2}/r_n) = 0.75$. 可见, r_{ni} 的大小只具有相对意义.

在 $B_{11} B_{21}$ 发生的条件下, 按照式(34),

$$\begin{aligned} Pr\{X_{n1} | B_{11} B_{21}\} &= (r_{n;1}/r_n) a_{n1;11} + (r_{n;2}/r_n) a_{n1;21} \\ &= 0.25 \times 0 + 0.75 \times 0.8 = 0.6, \end{aligned}$$

$$\begin{aligned} Pr\{X_{n2} | B_{11} B_{21}\} &= (r_{n;1}/r_n) a_{n2;11} + (r_{n;2}/r_n) a_{n2;21} \\ &= 0.25 \times 0.4 + 0.75 \times 0.2 = 0.25, \end{aligned}$$

$$\begin{aligned} Pr\{X_{n3} | B_{11} B_{21}\} &= (r_{n;1}/r_n) a_{n3;11} + (r_{n;2}/r_n) a_{n3;21} \\ &= 0.25 \times 0.6 + 0.75 \times 0 = 0.15, \end{aligned}$$

正如预期,

$$\sum_{k=1}^3 Pr\{X_{nk} | B_{11} B_{21}\} = 0.6 + 0.25 + 0.15 = 1.$$

以同样的方式, 我们可以算出 X_{nk} 在 B_1 和 B_2 的其它状态组合下的条件概率.

应用式(35), X_{nk} 的无条件概率也可以很容易地算出为 $Pr\{X_{n1}\} = 0.4475$, $Pr\{X_{n2}\} = 0.625$ 和 $Pr\{X_{n3}\} = 0.2875$. 正如预期,

$$\sum_{k=1}^3 Pr\{X_{nk}\} = 0.4475 + 0.625 + 0.2875 = 1.$$

应当指出,与现有方法不同,上述归一化结果是在没有应用任何归一化公式的情况下自动获得的.这意味着 M-DUCG 允许我们直接计算所关注的 $Pr\{X_{nk}\}$ 或 $Pr\{X_{nk} | E\}$, 而无须计算我们并不关注的 $Pr\{X_{nk'}\}$ 或 $Pr\{X_{nk'} | E\}$. 其中, E 代表任何证据或条件. 本文后面将表明, M-DUCG 的这一特性为人们建造知识库和进行概率推理提供了巨大的方便.

M-DUCG 除可用于多赋值情况外,还可用于单赋值情况. 在单赋值情况下,仅子变量的 true 状态的原因由式(31)说明. false 状态则是 true 状态的补集. 如图 8 的情况, 设 $a_{31;11} = 0.3, a_{31;21} = 0.6, r_{3;1} = 1, r_{3;2} = 1$, 我们有

$$\begin{aligned} Pr\{X_{31} | X_{11} X_{21}\} &= (r_{3;1}/r_3) a_{31;11} + (r_{3;2}/r_3) a_{31;21} \\ &= (1/2) \times 0.3 + (1/2) \times 0.6 \\ &= 0.45, \end{aligned}$$

$$Pr\{X_{32} | X_{11} X_{21}\} = 1 - 0.45 = 0.55,$$

$$\begin{aligned} Pr\{X_{31} | X_{11} X_{22}\} &= (r_{3;1}/r_3) a_{31;11} = (1/1) \times 0.3 \\ &= 0.3, \end{aligned}$$

$$Pr\{X_{32} | X_{11} X_{22}\} = 1 - 0.3 = 0.7,$$

$$\begin{aligned} Pr\{X_{31} | X_{12} X_{21}\} &= (r_{3;2}/r_3) a_{31;21} = (1/1) \times 0.6 \\ &= 0.6, \end{aligned}$$

$$Pr\{X_{32} | X_{12} X_{21}\} = 1 - 0.6 = 0.4,$$

$$Pr\{X_{31} | X_{12} X_{22}\} = 0;$$

$$Pr\{X_{32} | X_{12} X_{22}\} = 1 - 0 = 1,$$

所得 CPT 如图 28 所示.

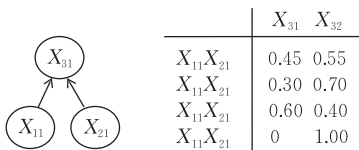


图 28 M-DUCG 所表达的单赋值情况示例

显然,图 28 中的 CPT 不同于图 8(a) 中的 CPT. 这意味着 M-DUCG 确实表达并响应了不同的知识. 注意,图 28 中 X_{11} 和 X_{21} 导致 X_{31} 的概率是 $(0.3 + 0.6)/2 = 0.45$, 而不是图 8(a) 中的 0.72. 这是因为在 M-DUCG 中, $r_{3;1} = r_{3;2} = 1, r_3 = r_{3;1} + r_{3;2} = 2$, 导致最终结果是 X_{11} 和 X_{21} 的贡献的平均值, 而不是 S-DUCG 中两者贡献的逻辑或所产生的值. 如果令 $r_{3;2} = 3$, 我们有 $r_3 = r_{3;1} + r_{3;2} = 4$, 则

$$\begin{aligned} Pr\{X_{31} | X_{11} X_{21}\} &= (1/4) \times 0.3 + (3/4) \times 0.6 \\ &= 0.525, \end{aligned}$$

这是因为 X_{21} 的贡献权重是 X_{11} 的 3 倍.

应当指出,式(34)和(35)看起来与文献[23]的

结果类似. 在文献[23]中,类似的公式为

$$Pr\{X_{nk} | \bigcap_i V_{ij_i}\} = \sum_i (r_{n;i}/r_n) Pr\{X_{nk} | V_{ij_i}\} \quad (36)$$

$$Pr\{X_{nk}\} = \sum_i (r_{n;i}/r_n) \sum_{j_i} Pr\{X_{nk} | V_{ij_i}\} Pr\{V_{ij_i}\} \quad (37)$$

其中,式(36)是一个假设. 但仔细分析可知,式(36)、式(37)与式(34)、(35)相比有本质的不同,因为

$$a_{nk;ij_i} \equiv Pr\{A_{nk;ij_i}\} \neq Pr\{X_{nk} | V_{ij_i}\}.$$

证明. 按照式(31),

$$\begin{aligned} Pr\{X_{nk} | V_{ij_i}\} &= (r_{n;i}/r_n) Pr\{A_{nk;ij_i}\} + \\ &\quad \sum_{i' \neq i} (r_{n;i'}/r_n) \sum_{j_{i'}} Pr\{A_{nk;i'j_{i'}}\} Pr\{V_{i'j_{i'}}\} \\ &\neq Pr\{A_{nk;ij_i}\} \equiv a_{nk;ij_i}. \quad \text{证毕.} \end{aligned}$$

最后,文献[23]指出简洁表达方式应当具备充分性(sufficiency)和可分离性(separability). 在 M-DUCG 中,这两者都实现了.

4.2 M-DUCG 中的逻辑门

实践中,子变量和其父变量之间的逻辑关系并非总如 4.2 节所述这般简单. 在复杂情况下,可利用逻辑门. 类似于 3.1 节所述,逻辑门 G_i 可以用表 3 所示的 LGS_i 说明. 如图 29 示例, 设 B_1, X_2, X_4, B_5 为二状态变量, LGS_3 如图 30 所示.

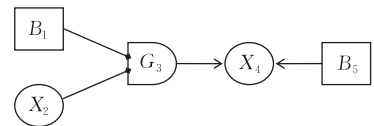


图 29 M-DUCG 中的逻辑门示例

j	G_{3j}
1	$B_{11} \cup X_{21} = B_{11} + B_{12} X_{21}$
2	$B_{12} X_{22}$

图 30 图 29 中的 LGS_3

图 29 中, X_4 有两个父变量: G_3 和 B_5 . 应用式(31)和图 30 所示之 LGS_3 , 我们有

$$\begin{aligned} X_{4k} &= (r_{4;3}/r_4) \sum_{j=1}^2 A_{4k;3j} G_{3j} + (r_{4;5}/r_4) \sum_{j=1}^2 A_{4k;5j} B_{5j} \\ &= (r_{4;3}/r_4) (A_{4k;31} (B_{11} \cup X_{21}) + \\ &\quad A_{4k;32} B_{12} X_{22}) + (r_{4;5}/r_4) \sum_{j=1}^2 A_{4k;5j} B_{5j} \\ &= (r_{4;3}/r_4) (A_{4k;31} B_{11} + A_{4k;31} B_{12} X_{21} + \\ &\quad A_{4k;32} B_{12} X_{22}) + (r_{4;5}/r_4) \sum_{j=1}^2 A_{4k;5j} B_{5j}. \end{aligned}$$

与 S-DUCG 中的情况类似,最复杂的情况可以用全组合逻辑门来表达. 如上例, 设 G_3 为全组合逻辑门, 我们有

$$\begin{aligned} X_{4k} &= (r_{4;3}/r_4) (A_{4k;31} B_{11} X_{21} + A_{4k;32} B_{12} X_{21} + \\ &\quad A_{4k;33} B_{11} X_{22} + A_{4k;34} B_{12} X_{22}) + \end{aligned}$$

$$(r_{4,5}/r_4) \sum_{j=1}^2 A_{4k,5j} B_{5j}.$$

全组合逻辑门等价于对应的 CPT. 然而, 在 DUCG 中, 一个 CPT 可以只包含部分父变量. 如上例, G_3 所表达的 CPT 仅包含 B_1 和 X_2 这两个父变量, 不包含 B_5 . 在 DUCG 中, 我们仍用有向弧来表达子变量与包含于一个 CPT 中的部分父变量之间的因果链接. 如上例, 当 G_3 是一个全组合逻辑门时, 图 29 可被简单且等价地表达为图 31.

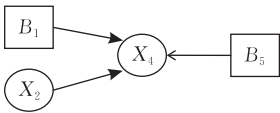


图 31 图 29 中的全组合逻辑门的等价表达

当然, 在图 31 中, B_5 也可以被包含在 CPT 中. 但在某些情况下, 我们需要这种分开表达. 例如, 设以 B_1 和 X_2 为父变量的 CPT 已从统计数据中获得. 如果我们想采取措施改变目标系统的状态 (例如, 采取某种政策来改变某经济系统的状态), 我们可能就需要图 31 所示的分开表达. 其中, B_5 代表拟采取的政策, 任务是预测各种政策选项的效果, 以便我们能够做出较好的决策. 在这种情况下, 我们没有包含了 B_1 、 X_2 和 B_5 的统计数据可用, 只有 X_2 和 B_5 的统计数据及独立的领域专家关于 B_5 影响 X_4 的经验知识可用. 注意, 我们在加入了 B_5 变量后, B_1 和 X_2 仍在继续影响 X_4 . 于是, 我们必须判断以 B_1 和 X_2 为父变量构成的 $CPT_{4,3}$ 影响 X_4 的权重 $r_{4,3}$ 和 B_5 影响 X_4 的权重 $r_{4,5}$. 这种情况正是图 31 所表达的. 以上对以 B_1 和 X_2 为父变量构成的 CPT 加上了下标“4;3”, 是因为这个 CPT 对应于图 29 所示的逻辑门 G_3 . 为简单起见, 类似的表达将不再逐一说明.

进而, 这种分开表达的同一个子变量的 CPT 可以不只一个. 图 32 是一个示例, 它等价于更容易被理解的图 33. 在图 33 中, G_3 和 G_{10} 是全组合逻辑门.

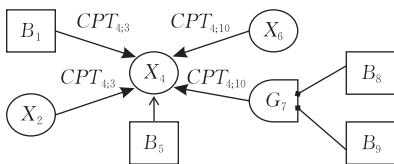


图 32 多于一个 CPT 的情况示例

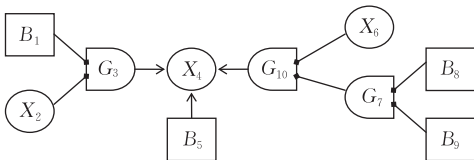


图 33 图 32 的等价表达

我们不需要一个 CPT 的原因在于不同的领域工程师可能有不同组别的统计数据. 这些不同组别的数据是由不同的领域工程师各自收集的, 无法合成为一组数据. 于是我们不得不面对由不同组别数据构成的不同的 CPT, 并按图 32 或图 33 所示方式将它们结合使用. 此外, 不同的 CPT 可以共享同一个父变量. 这种情况如图 34(a) 或 (b) 所示 (共享 X_2). 注意, 包含于 CPT 中的父变量也可以同时是独立的父变量 (如图中的 B_5).

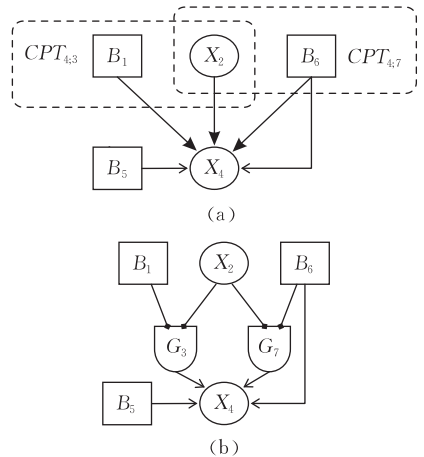


图 34 共享父变量且多于一个 CPT 的情况示例

读者可能已注意到, 图 32 中的 X_6 、 B_8 、 B_9 可被包含在一个 CPT 中, 但有时采用多重逻辑门可以使表达更简洁. 例如, 设 X_6 、 B_8 和 B_9 各有 3 个状态、 LGS_7 如图 35 所示. 则在 $CPT_{4,10}$ 中, 只需要 $3 \times 2 = 6$ 个参数. 如果我们将 X_6 、 B_8 、 B_9 作为一个 CPT 中的父变量对待, 该 CPT 的参数将达 $3^3 = 27$ 个.

j	G_{7j}
1	$B_{81} \cup B_{92} = B_{81} + B_{82}B_{92} + B_{83}B_{92}$
2	剩余状态组合

图 35 图 32 中的 LGS_7

最后, 领域工程师给出的 LGS 可不覆盖输入变量的所有状态组合, 因为领域工程师可能只关心这些输入变量状态组合中的一部分. 在这种情况下, 如图 35 所示的一个输入变量的剩余状态可被自动加入到逻辑门中, 该状态代表所有 LGS 中未显式表达的输入变量状态的剩余组合. 如图 30 示例, 设 B_1 和 X_2 各有 3 个状态, 则图 30 所示的 LGS_3 是不完备的, 必须加上剩余状态 $G_{33} = B_{13} \cup X_{23}$, 才能覆盖 B_1 和 X_2 的所有状态组合. 在加入了如图 35 所示的剩余状态后, 这时的 LGS 被称为完备的 (未必是全组合的). 对于完备的 LGS_i , 逻辑门变量 G_i 可被作为一个与 B_i 或 X_i 类似的普通变量对待, 即 G_i 的状态互斥且完备.

证明.

$$\sum_j Pr\{G_{ij}\} = \sum_j Pr\{\text{对应于 } G_{ij} \text{ 的逻辑表达式}\}$$

$$Pr\{\sum_j \text{对应于 } G_{ij} \text{ 的逻辑表达式}\}$$

$$Pr\{\text{输入变量的所有状态组合}\} = 1.$$

证毕.

应当注意, LGS 中的每个逻辑表达式中事件乘积的互斥可通过应用式(29)和(30)得到. 如图 35 示例,

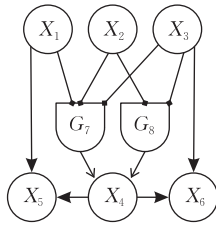


图 36 图 3 和图 4 示例的 M-DUCG 表达

j	G_{7j}	$a_{41;7j}$	$a_{42;7j}$	G_{8j}	$a_{41;8j}$	$a_{42;8j}$
1	X_{11}	0.2	0.8	X_{22}	0.3	0.7
2	$X_{12}X_{21}$	0.4	0.6	$X_{21}X_{31}$	0.6	0.4
3	$X_{12}X_{22}X_{31}$	0.6	0.4	$X_{21}X_{32}$	0.8	0.2
4	$X_{12}X_{22}X_{32}$	0.8	0.2			

LGS₇

LGS₈

j	$E_j = SCPV_{4j}$	用老的 p_i 计算		用新的 p_i 计算	
		$Pr\{X_{41} E_j\}$	$Pr\{X_{42} E_j\}$	$Pr\{X_{41} E_j\}$	$Pr\{X_{42} E_j\}$
1	$X_{11}X_{21}X_{31}$	2/5	3/5	9/20	11/20
2	$X_{11}X_{21}X_{32}$	1/2	1/2	1/2	1/2
	$X_{11}X_{22}X_{31}$	1/4	3/4	3/8	5/8
3	$X_{11}X_{22}X_{32}$	1/4	3/4	3/8	5/8
	$X_{12}X_{21}X_{31}$	1/2	1/2	1/2	1/2
4	$X_{12}X_{21}X_{32}$	3/5	2/5	11/20	9/20
	$X_{12}X_{22}X_{31}$	9/20	11/20	19/40	21/40
5	$X_{12}X_{22}X_{32}$	11/20	9/20	21/40	19/40

(a)

(b)

图 37 图 36 中的 CPT_4 (表达子变量 X_4 的 BN 类型的 CPT)

在本例中, 设 $r_{4;7} = r_{4;8} = 1$, 于是 $r_4 = r_{4;7} + r_{4;8} = 2$. 应用式(34), CPT_4 可算得如图 37(a) 所示. 其结果与表 1 显然不同. 如果将表 1 的 p_i 改为表 2 所示的新 p_i , 对应的改变是新的 $a_{4k;7j}$ 和 $a_{4k;8j}$, 则在 M-DUCG 中算得的新 CPT_4 如图 37(b) 所示. 显然, 结果与图 37(a) 不同.

注意: 第 2 节中用式(1)计算表 2 所示两套不同 p_i 参数所对应的 CPT 时, 结果均如表 1 所示, 无变化. 而本节我们看到, 在 M-DUCG 中, 这两套参数算出的 CPT 是不同的. 这反映了 M-DUCG 对参数变化的响应. 不仅如此, 如果 $r_{4;7} \neq r_{4;8}$, 计算结果也将与图 37(a) 和 (b) 所示不同. 这是因为在 M-DUCG 中, $(r_{n;i}/r_n)$ 是有实质意义的参数.

4.3 M-DUCG 中的条件作用事件

类似于 S-DUCG 中的条件连接事件, M-DUCG 引入了条件作用事件. 以图 38 为例, $A_{4k;1j}$ 为一个条件作用事件. 设其条件为 $Z_{4;1} = \bar{X}_{21}$, 即仅当 X_{21} 不存在时, $A_{4k;1j}$ 才可能发生. 一般地, $Z_{n;i}$ 为一个以 $Z_{nk;ij}$ 为元素的事件矩阵, 或为一个对所有 $A_{nk;ij}$ 都相同的

$$G_{71} = B_{81} \cup B_{92} = B_{81} + \bar{B}_{81} B_{92}$$

$$= B_{81} + B_{82} B_{92} + B_{83} B_{92},$$

表达式中所有事件的乘积均互斥的 LGS 称为互斥 LGS. 采用互斥 LGS, DUCG 中的事件表达式与计算该事件的概率的表达式相同. 因而在展开逻辑门的时候, 宜采用如图 30 和图 33 所示的互斥 LGS.

为进行比较, 下面用 M-DUCG 来处理图 3 和图 4 所示的多赋值情况. 在 M-DUCG 中, 图 3 和图 4 所示的多赋值情况被表达为图 36 和图 37.

条件事件. 于是, 当观测到或给定 X_{21} 后, $Z_{4;1} = \mathbf{0}$, 导致 $r_{4;1} = 0$, 即 X_4 和 X_1 之间无因果关系; 否则, $A_{4k;1j}$ 成为普通的作用事件.

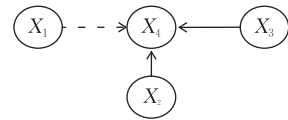


图 38 条件作用事件示例

本例中, 应用式(31), 人们可能直观地以为

$$X_{4k} = (r_{4;1}/r_4) Z_{4;1} \sum_{j_1} A_{4k;1j_1} X_{1j_1} +$$

$$(r_{4;2}/r_4) \sum_{j_2} A_{4k;2j_2} X_{2j_2} + (r_{4;3}/r_4) \sum_{j_3} A_{4k;3j_3} X_{3j_3}$$

$$= (r_{4;1}/r_4) \bar{X}_{2;1} \sum_{j_1} A_{4k;1j_1} X_{1j_1} +$$

$$(r_{4;2}/r_4) \sum_{j_2} A_{4k;2j_2} X_{2j_2} + (r_{4;3}/r_4) \sum_{j_3} A_{4k;3j_3} X_{3j_3},$$

但这一表达并不准确. 此外, 其中的 r_4 是一个取决于条件 $Z_{4;1}$ 是否满足的不确定参数. 通常情况下, 基于观测到的证据, 可判定条件 $Z_{n;i}$ 是否满足, 因为 $Z_{nk;ij}$ 的定义要求其可被观测. 如果没被观测到, 则 $Z_{n;i}$ 的验前概率必须被给定或通过某种方式被计算出来. 如本例, $Pr\{Z_{4;1}\} = Pr\{\bar{X}_{21}\} = 1 - Pr\{X_{21}\}$, 其中 $Pr\{X_{21}\}$ 要么被给定、要么通过对 X_{21} 的事件展开而被计算出来.

与 3.2.1 节相同, 当 $Z_{4;1}$ 未被观测到时, X_{4k} 的精确表达式应为

$$X_{4k} = (Z_{4;1} + \bar{Z}_{4;1}) X_{4k}$$

$$= (X_{21} + \bar{X}_{21}) X_{4k}$$

$$\begin{aligned}
&= (X_{21} + \bar{X}_{21}) \left((r_{4,1}/r_4) \bar{X}_{21} \sum_{j_1} A_{4k;1j_1} X_{1j_1} + \right. \\
&\quad \left. (r_{4,2}/r_4) \sum_{j_2} A_{4k;2j_2} X_{2j_2} + (r_{4,3}/r_4) \sum_{j_3} A_{4k;3j_3} X_{3j_3} \right) \\
&= (r_{4,2}/(r_{4,2} + r_{4,3})) A_{4k;21} X_{21} + \\
&\quad (r_{4,3}/(r_{4,2} + r_{4,3})) X_{21} \sum_{j_3} A_{4k;3j_3} X_{3j_3} + \\
&\quad (r_{4,1}/(r_{4,1} + r_{4,2} + r_{4,3})) \bar{X}_{21} \sum_{j_1} A_{4k;1j_1} X_{1j_1} + \\
&\quad (r_{4,2}/(r_{4,1} + r_{4,2} + r_{4,3})) \sum_{j_2 \neq 1} A_{4k;2j_2} X_{2j_2} + \\
&\quad (r_{4,3}/(r_{4,1} + r_{4,2} + r_{4,3})) \bar{X}_{21} \sum_{j_3} A_{4k;3j_3} X_{3j_3} \quad (38)
\end{aligned}$$

其中, $\bar{X}_{21} = \sum_{j_2 \neq 1} X_{2j_2}$, 因而上式满足 $\sum_k X_{4k} = 1$ (“1”为全集) 的归一化要求. 最后

$$\begin{aligned}
X_{4k} | X_{1j_1} X_{21} X_{3j_3} &= (r_{4,2}/(r_{4,2} + r_{4,3})) A_{4k;21} + \\
&\quad (r_{4,3}/(r_{4,2} + r_{4,3})) A_{4k;3j_3}, \quad j_2 = 1 \\
X_{4k} | X_{1j_1} X_{2j_2} X_{3j_3} &= (r_{4,1}/(r_{4,1} + r_{4,2} + r_{4,3})) A_{4k;1j_1} + \\
&\quad (r_{4,2}/(r_{4,1} + r_{4,2} + r_{4,3})) A_{4k;2j_2} + \\
&\quad (r_{4,3}/(r_{4,1} + r_{4,2} + r_{4,3})) A_{4k;3j_3}, \\
&\quad j_2 \neq 1 \quad (39)
\end{aligned}$$

将式(38)和(39)中的大写字母用小写字母替换, 我们即可很容易地算出 $Pr\{X_{4k}\}$ 和 $Pr\{X_{4k} | X_{1j_1} X_{2j_2} X_{3j_3}\}$. 按照如上所述方法, 读者可以自行求解条件作用事件的更复杂情况.

类似于 3.2.1 节中的条件事件, $\mathbf{A}_{n;i}$ 的条件 $\mathbf{Z}_{n;i}$ 也可以是多样的, 不限于 X_n 的父变量的状态. 这突破了简洁表达 BN 中的 CPT 这一常规的理论框架.

4.4 M-DUCG 中的缺省事件

文献[18]提到, 可将 leak 变量^[18,20] 作为一个新的父变量 L 对待. L 在文献[19]中被称为 dummy 变量. L 的特性在于它只有一个 true 状态, 即 $Pr\{L = \text{true}\} \equiv 1$. 类似于 L 变量, 在 M-DUCG 中定义了缺省变量 D_n , 其定义与 S-DUCG 中 D_n 的定义完全相同. 不同的是, S-DUCG 中的连接事件 $P_{n;nD}$ 在 M-DUCG 中变成了权重作用事件 $(r_{n;n}/r_n) A_{n;k;nD}$. 其中, $A_{n;k;nD}$ 为 X_{nk} 与 D_n 之间的作用事件, $r_{n;n}$ 为两者之间的关联度. 与 $A_{n;k;ij}$ 比较, $A_{n;k;nD}$ 的父变量只有一个状态, 因为 D_n 只有一个状态. 为方便起见, $A_{n;k;nD}$ 也可用 $A_{n;k;ij}$ 代表. 这时, $i=n, j$ 代表 D .

在多赋值情况下, 尽管子变量各状态是等价的, 仍可能存在一个特殊状态, 标识为 η . 通常, $\eta=0$. 这个特殊状态通常就是正常状态. 例如, 设变量 X_i 代表温度. 我们可以定义 X_{i0} = “正常”, X_{i1} = “偏高”, X_{i2} = “快速升高”, X_{i3} = “偏低”, X_{i4} = “快速降低”,

等等. 其中, X_{i0} 代表正常状态, X_{ij} ($j \neq 0$) 代表非正常状态.

实践中, 人们通常并不关心变量的正常状态, 而是关心其非正常状态. 也就是说, 正常状态通常不是赋值状态, 其原因和结果通常不被说明, 即: 领域工程师通常不给出 $a_{i0;mj}$ 和 $a_{nk;i0}$. 在这种情况下, 当所有非 D 类型父变量都处于正常状态时, 子变量的状态概率分布将仅由缺省变量 D_n 引起, 即 X_n 的状态概率分布为 $(r_{n;n}/r_n) Pr\{A_{n;k;nD}\} Pr\{D_n\} = Pr\{A_{n;k;nD}\} = a_{n;k;nD}$. 这是因为在这种情况下, 所有 $A_{n;k;ij}$ ($i \neq n, j \neq D$) 均不存在, 且 $r_n = r_{n;D}$. 通常, $A_{n;k;nD}$ 是一个条件作用事件, 其条件为 $Z_{n;k;nD} = \{X_n$ 的所有非 D_n 父事件对 X_n 均无影响}. 用矩阵的术语, $\mathbf{A}_{n;n}$ 以 $\mathbf{Z}_{n;n} = Z_{n;k;nD}$ 为条件, 其中 $A_{n;k;nD}$ 是事件矩阵 $\mathbf{A}_{n;n}$ 的元素, $Z_{n;k;nD}$ 是事件矩阵 $\mathbf{Z}_{n;n}$ 的元素, 所有元素均相同. 在图形中, $\mathbf{A}_{n;n}$ 被画成从 D_n 到 X_n 的虚线有向弧. 当然, D_n 也可作为常规父变量使用. 这时 D_n 的作用是给其它父变量增加了一个子变量的背景概率分布.

5 DUCG: M-DUCG 和 S-DUCG 的结合

对于完备地表达 CPD 来说, S-DUCG 或 M-DUCG 提供了对 CPD 的一种简洁表达. 至于究竟是选择 S-DUCG 还是 M-DUCG, 取决于所表达模块的具体情况. 模块 (module) 由一个子变量和其父变量 (包括逻辑门) 组成. 在某些文献中 (例如文献 [19,21]), “模块”被称为 “家庭” (family). 应当注意到, 家庭一词隐含着 DAG 结构, 即所有变量之间只存在单向的血缘关系, 不可能子孙同时又是祖先. 但 DUCG 模型允许存在有向循环图 DCG (将在后文讨论), 即子孙可能同时又是祖先. 因而在 DUCG 中, “家庭”一词被舍去而选用了 “模块” 一词. 鉴于本文仅限于处理 DAG 情况, “模块” 和 “家庭” 的含义大体相同. 唯一的不同在于 DUCG 的模块包括了起中间连接作用的逻辑门变量 (G 类型变量)、连接或作用变量 (P 或 A 类型变量) 及其条件变量 (Z 类型变量). 如图 34 示例, 子变量 X_4 及其父变量 B_1, X_2, B_5 和 B_6 构成一个模块, 既可以含逻辑门 G_3 和 G_7 (如图 34(b) 所示), 也可以不含 (如图 34(a) 所示). 如图 36 示例, 子变量 X_5 及其父变量 X_1 和 X_4 构成一个模块; 而子变量 X_4 及其父变量 X_1, X_2, X_3, G_7, G_8 则构成另一个模块. 图 11、图 14~21 也分别构成

模块。

对于一个特定的模块,当子变量为单赋值时,S-DUCG 和 M-DUCG 都可用;当子变量为多赋值时,仅 M-DUCG 可用.对不同的模块可自由选择应用 S-DUCG 或 M-DUCG,并将这些模块简单连接构成一个因果图,即为 DUCG. S-DUCG 和 M-DUCG 的共同思想是用独立事件 $P_{nk,ij}$ 或 $A_{nk,ij}$ 以及权重系数 $r_{n,i}$ 、逻辑门 G_i 、条件事件 $Z_{n,i}$ 和缺省事件 D_n 来简洁表达模块内子变量 (X 类型变量) 与其 $\{B, X\}$ 类型父变量 (中间可能存在逻辑门变量) 之间的不确定因果关系. 在 S-DUCG 中,采用连接事件 $P_{nk,ij}$; 在 M-DUCG 中,采用绑定了权重系数 ($r_{n,i}/r_n$) 的作用事件 $A_{nk,ij}$. $P_{nk,ij}$ 和 $A_{nk,ij}$ 都表达相同的作用机理: 父事件 V_{ij} ($V \in \{B, X, G, D\}$) 引发子事件 X_{nk} .

DUCG 模型有 3 个优点: (1) 能够直观简洁地表达模块内的各种复杂的不确定因果关系; (2) 由于引入独立的连接事件或作用事件,当获得证据后,因果图可通过删除无关的独立事件而被大大化简,推理也变得非常容易 (以后的章节将详细讨论); (3) 在建造 DUCG 的过程中,不同的模块可由不同的领域工程师分别独立建造,只要求相同变量的定义保持一致即可. 这样,分别独立建造的不同的模块可以由计算机自动进行简单连接而构成一个最终的 DUCG,从而将大型复杂的 DUCG 建造问题变得非常简单. 当然,如此建造的 DUCG 很可能出现 DCG 结构,这是由于各个模块的简单连接造成的. 限于篇幅,DCG 结构将在以后讨论,本文只讨论 DAG 结构. 需要指出的是,正是由于 DUCG 模型能够处理 DCG 结构,上述模块化的 DUCG 建造方法才是现实可行的.

一个 DAG 结构的 DUCG 的示例如图 39 所示. 该 DUCG 由 3 个模块构成: $\{X_4, B_1, B_2\}$ 及其内部有向弧; $\{X_5, B_1, B_2, B_3\}$ 及其内部有向弧; $\{X_7, G_6, X_4, X_5, D_7\}$ 及其内部有向弧. $\{ \}$ 中的第 1 个变量为该模块的子变量.

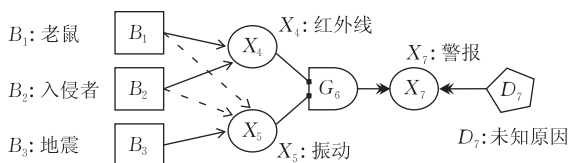


图 39 DAG 结构的 DUCG 示例

在图 39 中, $Z_{5,1} = Z_{5,2} = B_{32}$, 逻辑门 G_6 的说明 LGS_6 如图 40 所示.

j	G_{6j}
1	$(X_{42} + X_{43})X_{52}$
2	$X_{43}X_{53}$
3	剩余状态: $\bar{G}_{61}\bar{G}_{62} = X_{41} \cup X_{51} + X_{42}X_{53}$

图 40 图 39 中 G_6 的 LGS_6

作为子变量的 X_4 和 X_5 是多赋值变量,各有 3 个状态; B_1 和 B_2 是二状态变量; X_7 是单赋值变量 (仅 X_{71} 的原因被说明); $\{B, X, D\}$ 类型事件的定义如下:

$$B_{11} \equiv \{\text{老鼠出现}\}; B_{12} \equiv \{\text{无老鼠}\};$$

$$B_{21} \equiv \{\text{入侵者出现}\}; B_{22} \equiv \{\text{无入侵者}\};$$

$$B_{31} \equiv \{\text{地震发生}\}; B_{32} \equiv \{\text{无地震}\};$$

$$X_{41} \equiv \{\text{无红外线信号}\}; X_{42} \equiv \{\text{轻度红外线信号}\};$$

$$X_{43} \equiv \{\text{强红外线信号}\};$$

$$X_{51} \equiv \{\text{无振动信号}\}; X_{52} \equiv \{\text{轻度振动信号}\};$$

$$X_{53} \equiv \{\text{强振动信号}\};$$

$$X_{71} \equiv \{\text{警报响}\}; X_{72} \equiv \{\text{无警报}\};$$

$$D_7 \equiv \{\text{导致警报响的未知原因}\}.$$

这个入侵者探测系统有两个探测器: 即红外线探测器和振动探测器. 每个探测器的信号输出有 3 个状态: 某些状态可以触发警报, 另一些不能. 警报对两个探测器信号的响应遵从 LGS_6 所表达的逻辑关系. 其中 G_{63} 所表达的逻辑关系不能触发警报, 也就是说, $p_{71,63} = 0$. 然而, 即使 G_{63} 为真, 警报仍有可能被某些未知的原因触发, 这些原因可能是警报器的误响, 等等. 这些未知原因由 D_7 表达. X_{4j} ($j \in \{1, 2, 3\}$) 的原因有 2 个: B_{11} 和 B_{21} ; X_{5j} ($j \in \{1, 2, 3\}$) 的原因有 3 个: B_{11} , B_{21} 和 B_{31} . 仅当未发生地震时, B_{11} 和 B_{21} 起作用, 即 $A_{5j,11}$ 和 $A_{5j,21}$ 以 $Z_{5,1} = Z_{5,2} = B_{32}$ 为条件. 此外, B_{12} , B_{22} 和 B_{32} 与 X_4 和 X_5 之间无因果关系. 来自 B_1 , B_2 和 B_3 的关于 X_4 和 X_5 的关联度相同, 即 $r_{4,1} = r_{4,2} = r_{5,1} = r_{5,2} = r_{5,3} = 1$. 本例的其它相关参数给定如下:

$$\mathbf{a}_{4,11} = [0.7 \ 0.3 \ 0]^T; \mathbf{a}_{4,21} = [0 \ 0.3 \ 0.7]^T$$

$$\mathbf{a}_{5,11} = [0.6 \ 0.4 \ 0]^T; \mathbf{a}_{5,21} = [0 \ 0.6 \ 0.4]^T;$$

$$\mathbf{a}_{5,31} = [0 \ 0.1 \ 0.9]^T$$

$$p_{71,6} = [0.9 \ 0.7 \ 0]; p_{71,7D} = 0.005$$

$$\mathbf{b}_1 = [0.1 \ 0.9]^T; \mathbf{b}_2 = [0.1 \ 0.9]^T;$$

$$\mathbf{b}_3 = [0.01 \ 0.99]^T$$

其中,

$$\mathbf{a}_{n,i1} \equiv [a_{n1,i1} \ a_{n2,i1} \ a_{n3,i1}]^T,$$

$$p_{71,6} \equiv [p_{71,61} \ p_{71,61} \ p_{71,61}],$$

$$\mathbf{b}_i \equiv [b_{i1} \ b_{i2}]^T.$$

至此, 建造入侵者报警系统 DUCG 的任务完成. 对

应的 BN 如图 41 所示. 其中的 3 个 CPT (CPT_4 、 CPT_5 、 CPT_7) 涉及 54 个参数, 而图 39 中的 $\{a, p, b, r\}$ 类型参数只有 30 个.

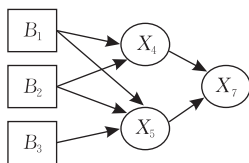


图 41 与图 39 的 DUCG 对应的 BN

需要注意的是, 本例并没有给出参数 $a_{ni;i2} \equiv [a_{n1;i2} \ a_{n2;i2} \ a_{n3;i2}]^T$. 这是因为 B_{12} 、 B_{22} 和 B_{32} 与 X_4 和 X_5 均无关. 可见, 本例的 DUCG 对于简洁表达一组变量 ($B_1, B_2, B_3, X_4, X_5, X_7$) 的联合概率分布来说是不完备的. 事实上, 根据图 39 和图 40 的 DUCG 及其参数无法算出图 41 的所有 CPT, 除非我们进一步给定 $a_{ni;i2} (n \in \{4, 5\}, i \in \{1, 2, 3\})$. 换句话说, 一个模块的 DUCG 简洁表达可能只表达了该模块 CPD 的部分内容, 而非完备. 含有表达不完备的模块的 DUCG 称为不完备 DUCG. 下一节我们将详细讨论不完备的 DUCG.

6 不完备 DUCG

所谓完备表达, 是指根据简洁表达所给定的参数即可算出对应的所有 CPT 或 CPD. 在上例中, DUCG 的某些参数并未给定, 导致并不能从给定的 DUCG 算出所有的 CPT. 不仅如此, 图 39 的参数还可给定为

$$\begin{aligned} a_{4;11} &= [0.7 \ 0.3 \ -]^T; \\ a_{4;21} &= [- \ 0.3 \ 0.7]^T \\ a_{5;11} &= [0.6 \ 0.4 \ -]^T; \\ a_{5;21} &= [- \ 0.6 \ 0.4]^T; \\ a_{5;31} &= [- \ 0.1 \ 0.9]^T \\ p_{71;6} &= [0.9 \ 0.7 \ -]; \quad p_{71;7D} = 0.005 \\ b_1 &= [0.1 \ -]^T; \\ b_2 &= [0.1 \ -]^T; \\ b_3 &= [0.01 \ -]^T \end{aligned}$$

其中, “-”代表缺失或无关. 我们甚至可以将

$$\begin{aligned} a_{5;11} &= [0.6 \ 0.4 \ -]^T; \\ a_{5;21} &= [- \ 0.6 \ 0.4]^T; \\ a_{5;31} &= [- \ 0.1 \ 0.9]^T \end{aligned}$$

改成

$$a_{5;11} = [0.6 \ 0.2 \ -]^T;$$

$$a_{5;21} = [- \ 0.6 \ 0.3]^T;$$

$$a_{5;31} = [- \ 0.1 \ 0.8]^T.$$

前者“-”的值实际上是“0”, 后者“-”的值却不应当是“0”, 但从效果上看等价于“0” (即没有因果关系). 注意, 这里的约束条件 $\sum_k a_{nk;ij} = 1$ 和 $\sum_j b_{ij} = 1$ 实际上已被放松为 $\sum_k a_{nk;ij} \leq 1$ 和 $\sum_j b_{ij} \leq 1$. 换言之, 在 DUCG 中, 人们可以只给出所关注的参数, 而不考虑不受关注的参数. DUCG 的这个特性导致 DUCG 尽管可以是、但也可以不是 CPD 的简洁表达. 从本质上看, DUCG 是人们关于现实世界的不确定知识的表达, 而不必是一组变量的联合概率分布的表达. 后者只是前者的一个特例. 可见, 不完备 DUCG 已经突破了 BN 的基本理论框架.

DUCG 可以不完备的原因在于 DUCG 的链式推理是自我依赖的. 换句话说, 在 DUCG 的推理中, $Pr\{X_{nk}\}$ 的计算与 $Pr\{X_{nk'}\}$ 无关, 与 $a_{nk';ij}$ 也无关. 这里, $k \neq k'$. 注意, 在所有应用式 (1) 的模型中, $Pr\{X_{nk}\}$ 的计算与 $Pr\{X_{nk'}\}$ 都是相关的. 在 DUCG 中, 当我们根据式 (35) 计算 $Pr\{X_{nk}\}$ 时, 我们无须知道 $a_{nk';ij} (k \neq k')$. 这意味着 DUCG 的某些参数可以缺失, 而不会影响我们所关注的推理计算 (精确计算).

例如, 设 B_1, B_2 和 X_3 为二状态变量、 $X_{32} = F_{32;11}B_{11} + F_{32;22}B_{22}$. 如果我们只关注 X_{32} , 领域工程师只需要给出 $a_{32;11}$ 和 $a_{32;22}$, 无须给出 $a_{32;12}$ 、 $a_{32;21}$ 、 $a_{31;11}$ 、 $a_{31;12}$ 、 $a_{31;21}$ 和 $a_{31;22}$. 用概率表达式, 我们有

$$\begin{aligned} Pr\{X_{32}\} &= Pr\{F_{32;11}\}Pr\{B_{11}\} + Pr\{F_{32;22}\}Pr\{B_{22}\} \\ &= f_{32;11}b_{11} + f_{32;22}b_{22} \\ &= (r_{3;1}/r_3)a_{32;11}b_{11} + (r_{3;2}/r_3)a_{32;22}b_{22}, \end{aligned}$$

其中, $f_{nk;ij} \equiv Pr\{F_{nk;ij}\} = Pr\{(r_{ni}/r_n)A_{nk;ij}\} \equiv (r_{ni}/r_n)Pr\{A_{nk;ij}\} = (r_{ni}/r_n)a_{nk;ij}$. 可见, $Pr\{X_{32}\}$ 的计算与 $Pr\{X_{3j}\} (j \neq 1)$ 及其输入作用事件无关. 正是由于 DUCG 推理的自我依赖特性, 才使得 DUCG 的表达可以是不完备的. 这种不完备性可以大大降低建造 DUCG 的工作量和难度, 再加上 DUCG 能够处理 DCG, 使得 DUCG 成为超越了 BN 的一个新的理论体系. 换句话说, BN 是 DUCG 的一个子集. 当 DUCG 完备且无 DCG 时, DUCG 是 BN 的一种简洁表达; 当 DUCG 不完备或包含 DCG 时, DUCG 就突破了 BN 的疆界.

7 在证据 E 条件下化简 DUCG

一旦收到证据 E , 我们可以通过固定在 E 中已

确定状态的变量来初步化简 DUCG. 例如, 设 $E = E_1 E_2 = B_{32} X_{71}$, 其中 $E_1 = B_{32}$, $E_2 = X_{71}$. 则图 39 的 DUCG 可被初步化简如图 42 所示.

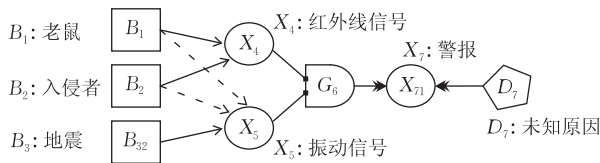


图 42 E 条件下初步化简的 DUCG (基于图 39)

图中, B_{32} 和 X_{71} 是被固定的状态. 这种初步化简的方法已在文献[27]中提出. 该文献阐明了利用证据编译(compiling)BN 的功效. 本文下面将提出进一步化简 DUCG 的方法. 其基本思想与文献[27-28]相同, 即: 将与 E 矛盾或与问题无关的变量删除.

如图 42 示例, 由于在 E 条件下 $Z_{5,1}$ 和 $Z_{5,2}$ 已满足, $A_{5,1}$ 和 $A_{5,2}$ 成为普通作用变量, 即从 B_1 和 B_2 到 X_5 的虚线有向弧变成了实线有向弧. 进而, 由于 $a_{5,32}$ 未给定(效果上与 $a_{5,32} = 0$ 相同), B_{32} 不是 X_5 的原因. 因而 $A_{5,3}$ 在化简 DUCG 的过程中被删除. 化简后的 DUCG 如图 43 所示.

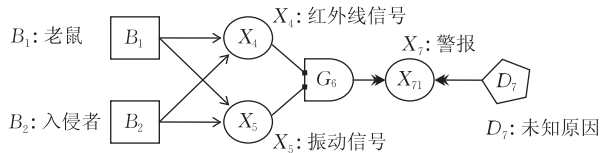


图 43 基于图 42 的在 E 条件下化简后的 DUCG

一般来说, 我们可以应用下述规则来进一步化简已经初步化简的 DUCG.

规则 1. 如果 E 显示 $Z_{n,i}$ 不满足, 将 $F_{n,i}$ 或 $P_{n,i}$ 从 DUCG 中删除; 如果 E 显示 $Z_{n,i}$ 已经满足, 则条件 $F_{n,i}$ 或 $P_{n,i}$ 成为普通 $F_{n,i}$ 或 $P_{n,i}$.

规则 2. 如果 E 显示 V_{ij} ($V \in \{B, X\}$) 为真, 但 V_{ij} 却不是 X_n 的父事件, 则将 $F_{n,i}$ 或 $P_{n,i}$ 从 DUCG 中删除.

例如, 设 X_{32} 不是 X_5 的父事件, 当 E 显示 V_{32} 为真时, 将 $F_{5,3}$ 从 DUCG 中删除.

规则 3. 如果 E 显示 X_{nk} 为真, 但 X_{nk} 不可能被 V_i ($V \in \{B, X, G\}$) 的任何状态引起, 则将 $F_{n,i}$ 或 $P_{n,i}$ 从 DUCG 中删除, 但 V_i 是待求解假设事件中的某变量或其后代变量、且其间的因果关系未被已知事件阻断的情况除外.

例如, 设 X_{53} 不可能被 X_2 的任何状态引起, X_2 也不是待求解假设事件中的变量或其后代, 或虽是其后代但其间的因果链被已知事件阻断. 则当 E 显示 X_{53} 为真时, $F_{5,2}$ 被删除.

规则 3 中的例外是指如下情况: 当 X_{nk} 不可能由 V_i 的任何状态引起、但预期 X_{nk} ($k \neq k'$) 可由以包含在假设事件中的 V_{hg} 为起点的因果链(通过 V_i)而引起时, X_{nk} 成为 V_{hg} 的负面证据, 从而降低 V_{hg} 发生的概率. 在这种情况下, X_{nk} 和 V_i 与假设事件中的 V_{hg} 是相关的. 此时 $F_{n,i}$ 或 $P_{n,i}$ 不能被删去. 如果 V_i 和 V_{hg} 之间的因果链被任何确知事件阻断, 则 X_n 与 V_i 之间的因果关系与假设事件不再相关. 这时 $F_{n,i}$ 或 $P_{n,i}$ 可被删去.

规则 4. 当 E 显示 X_{nk} 和 V_{ij} ($V \in \{B, X\}$) 为真, 同时 X_{nk} 不可能由 V_{ij} 引起时, 将 $F_{n,i}$ 或 $P_{n,i}$ 从 DUCG 中删除.

例如, 当 E 显示 X_{nk} 和 B_{ij} 为真, 同时 $a_{nk,ij} = 0$ 或未给定, 则 $F_{n,i}$ 被删除.

规则 5. 因任何原因导致 X_n 或 G_n 无输入变量, 则将 X_n 或 G_n 及其输出有向弧从 DUCG 中删除.

根据定义, X_n 或 G_n 必须有输入变量, 没有输入变量的 X_n 或 G_n 是无意义的, 从而也与待求解问题无关, 故被删除. X_n 或 G_n 被删除后, 以 X_n 或 G_n 为起点的输出有向弧当然也无意义, 从而也被删除. 需要说明的是, $F_{n,iD}$ 也是 X_n 的输入变量. 当其存在时, X_n 不能被删除.

规则 6. 因任何原因导致 G_i 没有输出, 则将 G_i 从 DUCG 中删除.

例如, 设 G_i 有两个子变量 X_1 和 X_2 . 如果 E 显示 X_{11} 和 X_{21} 为真, 同时 X_{11} 和 X_{21} 不可能由 G_i 的任何状态引起, 导致应用上述规则 3 删除了 $F_{1,i}$ 和 $F_{2,i}$, 进而导致 G_i 没有了输出, 则 G_i 被删除. 因为没有输出的逻辑门是无意义的, 也与待求解问题无关.

规则 7. 如果 (1) X_n 的状态未知; (2) X_n 无输出; (3) 未预知 X_n 包含在待求解的假设事件中, 则将 X_n 及其所有输入有向弧从 DUCG 中删除.

例如, 设 X_3 的状态未知、 X_3 无输出、且不是预定包含在待求解的假设事件中的变量, 则将 X_3 和 $F_{3,i}$ 从 DUCG 中删除. 这是因为在给定 E 后, X_3 和 $F_{3,i}$ 对确定所应当关注的假设事件以及计算这些假设事件在 E 条件下的概率没有任何影响.

规则 8. 如果 E 显示 X_{nk} 及其因果链上游事件 V_{ij} 为真, 且 X_{nk} 早于 V_{ij} 出现, 即 V_{ij} 不可能是 X_{nk} 的原因, 则将 X_{nk} 和 V_{ij} 之间因果链中的但与 X_{nk} 上游的其它因果链无关的 F 和 P 类型变量从 DUCG 中删除.

例如, 设 E 显示 X_{43} 的出现早于其爷爷辈事件 B_{22} , 在他们之间的变量为 X_1 . 如果无其它 X_{43} 的上

游变量通过 X_1 与 X_{43} 相连, 则将 $F_{4,1}$ 和 $F_{1,2}$ 删除. 然而, 如果 B_3 也是 X_1 的父变量 (即 B_3 也可以通过 $F_{43,1}$ 而导致 X_{43}), 则仅 $F_{1,2}$ 可被删除, $F_{4,1}$ 不能被删除, 因为 $F_{43,1} F_{1,3} B_3$ 仍是一条可能的因果链.

应当指出, 本规则比较特殊, 是关于事件发生时序的规则, 早在文献[22]中即已提出. 本文只是将其延伸到 M-DUCG. 这个规则之所以能够应用, 是基于 DUCG 将变量之间的不确定因果关系用 A 和 P 类型随机事件来表达.

规则 9. 如果存在一组变量 (称为独立组) 与 E 中涉及的变量无因果连通关系, 且不是预定关注的假设事件中的变量, 则将这组变量从 DUCG 中删除.

例如, 设 B_1, X_2 以及 $F_{2,1}$ 与 E 中涉及的任何变量无任何因果连通关系, 同时它们也非预定关注其状态的变量, 则删除 B_1, X_2 和 $F_{2,1}$. 这是因为在给定 E 的条件下, 它们与寻找应受关注的假设事件及计算其概率无关.

规则 10. 如果 (1) E 显示一组变量状态为正常, 即 $X_{n\eta}$ 为真, $n \in S_I, S_I$ 是这组变量的标识集; (2) 存在另一组状态未知的 $\{B, X, G, D, F, P\}$ 类型的变量, 这些变量只与 S_I 中的 $X_{n\eta}$ 变量相连. 换句话说, 这一组变量 (称为孤立组) 被 S_I 中的 $X_{n\eta}$ 事件阻断而独立于其它变量; (3) 孤立组中的变量非预定关注其状态的变量, 则将孤立组变量从 DUCG 中删除.

例如, 设 E 显示 X_{20} 和 X_{40} 为真, 即 $\eta=0, S_I = \{2, 4\}$, 变量 $\{V_3, V_5, F_{2,3}, F_{4,3}, F_{2,5}, F_{4,5}, F_{3,5}\}$ 的状态未知, 仅与 X_2 和 X_4 相连, 其中并无预定其状态受关注的变量, 则 $\{V_3, V_5, F_{2,3}, F_{4,3}, F_{2,5}, F_{4,5}, F_{3,5}\}$ 成为孤立组变量, 被从 DUCG 中删除. 这是因为孤立组变量与其它变量的相关性被 X_{20} 和 X_{40} 阻断了, 导致它们与寻找应受关注的假设事件及其概率计算的推理过程无关.

规则 1~10 可按任何顺序反复应用.

应当指出, 规则 1~10 所进行的化简不同于文献[28]中提出的 VE (Variable Elimination) 化简, 因为 VE 化简基于特定问题 " $Pr\{X_{nk} | E\} = ?$ ", 规则 1~10 则仅基于 E , 所化简的 DUCG 可用于化简后仍存在的任何所关注的问题. 当然, 基于化简后的 DUCG, 仍可针对某具体问题应用 VE 和上述规则进行进一步化简.

由于应用上述规则化简, 原 DUCG 中的变量被分成了两组, 分别用 S_{in} 和 S_{out} 的下标集合来标识.

包含在 S_{in} 中的变量就是仍留在化简后的 DUCG 中的变量, 包含在 S_{out} 中的变量则是应用规则 1~10 而被删除的变量. 被删除的变量实际上是在 E 条件下与待求解问题无关的变量. 也就是说, 与待求解问题有关的变量在化简 DUCG 后已从原来的 $S_{in} + S_{out}$ 减少为 S_{in} . 如图 39 和图 43 示例, $S_{in} = \{1, 2, 4, 5, 7\}, S_{out} = \{3\}$. 实际应用中, 往往 $S_{in} \ll S_{in} + S_{out}$. 因此, 应用规则 1~10 可以大大减少推理的工作量和难度.

本文以下将仅讨论包含在 S_{in} 中的变量. 换句话说, 下面的讨论将基于化简后的 DUCG, 即只有仍包含在化简后的 DUCG 中的假设事件属于被讨论的范围. 对于诊断 (逆向推理) 类问题, 这意味着可能的根 (root) 或原因事件被减少为仅包含在 S_{in} 中的状态未知的 B 类型变量的状态. 有时, S_{in} 中仅仅包含一个状态未知的 B 类型变量. 这时, 诊断工作实际上已经完成, 因为可能的原因事件已经被唯一并且精确地确定了. 注意, 这一结果的获得并不依赖于任何数值计算. 因而, 概率参数的精度在 DUCG 中显得不那么重要. 这是 DUCG 的又一个优点.

8 基于化简后的 DUCG 的概率推理

设 H_{kj} 是我们希望求其条件概率的假设事件, 条件为所观察到的证据 $E = \bigcap_h E_h = \bigcap_h V_{hy_h}$. H_{kj} 由 $\{X, B, P, A\}$ 类型事件组成 (P 和 A 类型事件可作为推理的假设事件是 DUCG 的又一贡献), k 标识在 H_{kj} 中的变量组合 (例如 $H_k = B_1 X_2$), j 标识这些变量的状态组合 (例如 $H_{kj} = B_{11} X_{23}$), $E_h = V_{hy_h}$ ($V \in \{X, B\}$), 则 H_{kj} 的条件概率的计算公式为

$$h_{kj}^s \equiv Pr\{H_{kj} | E\} = \frac{Pr\{H_{kj} E\}}{Pr\{E\}} = \frac{Pr\{H_{kj} \bigcap_h V_{hy_h}\}}{Pr\{\bigcap_h V_{hy_h}\}} \quad (40)$$

其中 h_{kj}^s 被称为 H_{kj} 的状态条件概率. 基于化简后的 DUCG, 我们可以采用两种方案计算 h_{kj}^s .

第 1 种方案用 CPT 进行计算: 当化简后的 DUCG 表达完备且无 DCG 时, 我们可计算出各模块的 CPT, 亦即将化简后的 DUCG 转化为一个标准的 BN, 然后用现有的计算 BN 的方法进行推理计算. 当化简后的 DUCG 表达不完备时, 无法计算出所有模块的 CPT (含 DCG 的情况将在下一篇文章中讨论), 这时必须采用本文提出的第 2 种方案.

第 2 种方案基于最早在文献[22]中提出的事件

运算算法,但文献[22]仅仅涉及 DCD 的情况,本文将将其推广到 DUCG 的任何情况.

第 2 种方案用式(40)进行计算.先将式中的 $\bigcap_h V_{h y_h}$ 和 $H_{kj} \bigcap_h V_{h y_h}$ 分别展开为由 $\{B, A, P, D\}$ 类型事件组成的互斥积之和的表达式,然后用对应的概率参数(对应的小写字母)替代表式中的事件(大写字母),从而算出 h_{kj} . 在事件展开过程中,下述规则可能被用到,其中“1”代表全集、“0”代表空集.

规则 11. 任给 $V \in \{B, X, G, D\}$, $j \neq j'$, 整数 $y \geq 2$, 则 $(V_{ij})^y = V_{ij}$, $V_{ij} V_{ij'} = 0$.

证明. 根据定义, V_{ij} 是一个事件. 根据逻辑运算法则, $(V_{ij})^y = V_{ij}$. 另据定义, 同一变量的不同状态互斥, 因而有 $V_{ij} V_{ij'} = 0$. 证毕.

规则 11 有多种应用方式. 例如, 设 $E_1 = X_{nk} = F_{nk;ij} X_{ij}$, $E_2 = X_{ij}$, 则有 $E_1 E_2 = F_{nk;ij} X_{ij} X_{ij} = F_{nk;ij} X_{ij} = E_1$. 又如, 设 $E_1 = X_{nk} = F_{nk;ij} X_{ij}$, $E_2 = X_{ij'}$, 其中 $j \neq j'$, 则有 $E_1 E_2 = F_{nk;ij} X_{ij} X_{ij'} = 0$.

规则 12. 给定整数 $y \geq 2$, $k \neq k'$, $j \neq j'$, 则 $(F_{nk;ij})^y = (r_{n;i}/r_n)^y A_{nk;ij}$, $F_{nk;ij} F_{nk';ij} = 0$, $F_{nk;ij} F_{nk';ij'} = 0$, $F_{nk';ij} F_{nk';ij'} = 0$.

证明. 根据定义, $(F_{nk;ij})^y = ((r_{n;i}/r_n) A_{nk;ij})^y$. 由于 $A_{nk;ij}$ 是一个事件, 我们有 $(A_{nk;ij})^y = A_{nk;ij}$, 从而有 $(F_{nk;ij})^y = ((r_{n;i}/r_n) A_{nk;ij})^y = (r_{n;i}/r_n)^y A_{nk;ij}$. 根据规则 11, 当 $k \neq k'$ 时, $X_{nk} X_{nk'} = 0$. 这意味着 $A_{nk;ij}$ 与 $A_{nk';ij}$ 不能并存, 即 $A_{nk;ij} A_{nk';ij} = 0$. 类似地, 当 $j \neq j'$ 时, $V_{ij} V_{ij'} = 0 (V \in \{X, B, G\})$. 这意味着 $A_{nk;ij}$ 与 $A_{nk';ij'}$ 不能并存, 即 $A_{nk;ij} A_{nk';ij'} = 0$. 因而我们有

$$\begin{aligned} F_{nk;ij} F_{nk';ij} &= (r_{n;i}/r_n)^2 A_{nk;ij} A_{nk';ij} = 0, \\ F_{nk;ij} F_{nk';ij'} &= (r_{n;i}/r_n)^2 A_{nk;ij} A_{nk';ij'} = 0, \\ F_{nk';ij} F_{nk';ij'} &= (r_{n;i}/r_n)^2 A_{nk';ij} A_{nk';ij'} = 0. \end{aligned} \quad \text{证毕.}$$

例如, 设 $E_1 = F_{nk;ij} X_{ij} + F_{nk;gy} B_{gy}$, $E_2 = F_{nk;gy} B_{gy}$, $E_3 = F_{nk';ij} X_{ij}$, 其中 $k \neq k'$. 我们有 $E_1 E_2 = F_{nk;ij} X_{ij} F_{nk;gy} B_{gy} + (F_{nk;gy} B_{gy})^2 = (r_{n;i} r_{n;g}/r_n^2) A_{nk;ij} X_{ij} A_{nk;gy} B_{gy} + (r_{n;g}/r_n)^2 A_{nk;gy} B_{gy} E_1 E_3 = F_{nk;ij} X_{ij} F_{nk';ij} X_{ij} + F_{nk;gy} B_{gy} F_{nk';ij} X_{ij} = 0$.

规则 13. 令 S_m 代表标号为 m 的变量标识集, $m \in \{1, 2, \dots, M\}$, $S_1 \subseteq S_2 \subseteq S_3 \dots \subseteq S_M$, 则

$$\sum_{m=1}^M \prod_{i \in S_m} F_{nk;ij_i} V_{ij_i} = \left(\sum_{m=1}^M \prod_{i \in S_m} (r_{n;i}/r_n) \right) \prod_{i \in S_1} A_{nk;ij_i} V_{ij_i}.$$

证明. 设 E_1 和 E_2 是两个事件. 根据集合理论, $E_1 \cup E_1 E_2 = E_1 = E_1 \cup E_2$, 即一旦 E_1 发生, 相当于表

达式中的 $E_1 E_2$ 也发生, 即有 $E_1 E_2 = E_1$. 类似地,

$$\begin{aligned} \sum_{m=1}^M \prod_{i \in S_m} F_{nk;ij_i} V_{ij_i} &= \sum_{m=1}^M \prod_{i \in S_m} (r_{n;i}/r_n) A_{nk;ij_i} V_{ij_i} \\ &= \sum_{m=1}^M \left(\prod_{i \in S_m} (r_{n;i}/r_n) \prod_{i \in S_m} A_{nk;ij_i} V_{ij_i} \right) \\ &= \sum_{m=1}^M \left(\prod_{i \in S_m} (r_{n;i}/r_n) \prod_{i \in S_1} A_{nk;ij_i} V_{ij_i} \right) \\ &= \left(\sum_{m=1}^M \prod_{i \in S_m} (r_{n;i}/r_n) \right) \prod_{i \in S_1} A_{nk;ij_i} V_{ij_i}, \end{aligned}$$

其中第 3 个“=”是因为在 $S_1 \subseteq S_2 \subseteq S_3 \dots \subseteq S_M$ 的条件下, 一旦 $\prod_{i \in S_1} A_{nk;ij_i} V_{ij_i}$ 成立, 相当于 $\prod_{i \in S_m} A_{nk;ij_i} V_{ij_i}$ 也成立, 即有 $\prod_{i \in S_m} A_{nk;ij_i} V_{ij_i} = \prod_{i \in S_1} A_{nk;ij_i} V_{ij_i}$. 证毕.

例如, 我们有

$$\begin{aligned} F_{31;11} V_{11} + F_{31;11} V_{11} F_{31;22} V_{22} &= (r_{3;1}/r_3) A_{31;11} V_{11} + \\ &= (r_{3;1}/r_3) A_{31;11} V_{11} (r_{3;2}/r_3) A_{31;22} V_{22} \\ &= (r_{3;1}/r_3 + (r_{3;1}/r_3)(r_{3;2}/r_3)) A_{31;11} V_{11}, \end{aligned}$$

其中 $S_1 = \{1\}$, $S_2 = \{1, 2\}$, $M = 2$.

应当指出, 规则 13 实际上定义了一种有别于常规集合理论的新算法. 这是因为在 M-DUCG 中, A 类型的事件总是绑定了权重系数 $(r_{n;i}/r_n)$. 规则 13 指出: 在保留集合理论的事件吸收算法的同时, 不应由于 A 类型事件的吸收而导致其权重系数消失. 如上例, $A_{31;11} V_{11} A_{31;22} V_{22}$ 应当被 $A_{31;11} V_{11}$ 吸收, 但绑定于 $A_{31;11} V_{11} A_{31;22} V_{22}$ 的权重系数 $(r_{3;1}/r_3)(r_{3;2}/r_3)$ 不应因此而消失. 换句话说, 事件运算和权重系数运算都应当进行.

规则 14. 令 $j = j_i$, 则

$$F_{nk;ij} V_{ij} \left(\sum_{i'} F_{nk;i'j'} V_{i'j'} \right) = F_{nk;ij} V_{ij}.$$

证明. i 是式中的 i' 的取值之一. 运用与规则 13 相同的概念, 我们有

$$\begin{aligned} F_{nk;ij} V_{ij} \left(\sum_{i'} F_{nk;i'j'} V_{i'j'} \right) &= (F_{nk;ij} V_{ij})^2 + F_{nk;ij} V_{ij} \sum_{i' \neq i} F_{nk;i'j'} V_{i'j'} \\ &= (r_{n;i}/r_n)^2 A_{nk;ij} V_{ij} + \\ &= (r_{n;i}/r_n) \sum_{i' \neq i} (r_{n;i'}/r_n) A_{nk;i'j'} V_{i'j'} A_{nk;ij} V_{ij} \\ &= (r_{n;i}/r_n)^2 A_{nk;ij} V_{ij} + (r_{n;i}/r_n) \sum_{i' \neq i} (r_{n;i'}/r_n) A_{nk;ij} V_{ij} \\ &= ((r_{n;i}/r_n)^2 + (r_{n;i}/r_n) \sum_{i' \neq i} (r_{n;i'}/r_n)) A_{nk;ij} V_{ij} \\ &= ((r_{n;i}/r_n) \sum_{i'} (r_{n;i'}/r_n)) A_{nk;ij} V_{ij} \\ &= (r_{n;i}/r_n) A_{nk;ij} V_{ij} \\ &= F_{nk;ij} V_{ij}. \end{aligned} \quad \text{证毕.}$$

规则 14 看起来似乎表明来自不同父变量的权重作用事件 $F_{nk;ij}$ 彼此互斥, 即给定 $i \neq i', F_{nk;ij} F_{nk;i'j'} = 0$, 同时 $(F_{nk;ij})^2 = F_{nk;ij}$. 但这种理解不对, 因为 (1) $A_{nk;ij}$ 和 $A_{nk;i'j'}$ 彼此是独立的; (2) 根据规则 12, $(F_{nk;ij})^2 = (r_{ni}/r_n) F_{nk;ij}$, 而不是 $(F_{nk;ij})^2 = F_{nk;ij}$.

限于篇幅, 效率更高的基于事件矩阵和概率矩阵的算法将在下一篇文章中介绍.

图 42 为一个应用示例. 通过化简得到图 43 所示的化简后的 DUCG 后, 剩下的受关注的假设事件为 $H_{11} \equiv B_{11}, H_{21} \equiv B_{21}, H_{71} \equiv P_{71,7D}$. 令 S_H 代表证据 E 条件下的所有可能假设事件的集合, 则 $S_H = \{H_{11}, H_{21}, H_{71}\}$. S_H 是 E 条件下可能的假设事件样本空间, 也是 DUCG 对诊断类问题的定性推理结果. 由于证据 $E_1 = B_{32}$ 的影响已被包含在图 43 中, B_{32} 已与化简后的 DUCG 无关, 从而有 $Pr\{B_{21} | B_{32} X_{71}\} = Pr\{B_{21} | X_{71}\}$. 根据式(40), 我们有

$$h_{21}^s \equiv Pr\{B_{21} | B_{32} X_{71}\} = Pr\{B_{21} | X_{71}\} = \frac{Pr\{B_{21} X_{71}\}}{Pr\{X_{71}\}},$$

分别展开 X_{71} 和 $B_{21} X_{71}$, 我们有

$$\begin{aligned} X_{71} &= (P_{71,61} G_{61} + P_{71,62} G_{62}) \cup P_{71,7D} D_7 \\ &= (P_{71,61} G_{61} + P_{71,62} G_{62}) \bar{P}_{71,7D} + P_{71,7D} \\ &= (P_{71,61} (X_{42} + X_{43}) X_{52} + P_{71,62} X_{43} X_{53}) \bar{P}_{71,7D} + P_{71,7D} \\ &= P_{71,61} \bar{P}_{71,7D} X_{42} X_{52} + P_{71,61} \bar{P}_{71,7D} X_{43} X_{52} + \\ &\quad P_{71,62} \bar{P}_{71,7D} X_{43} X_{53} + P_{71,7D} \\ &= P_{71,61} \bar{P}_{71,7D} (F_{42,11} B_{11} + F_{42,21} B_{21}) \cdot \\ &\quad (F_{52,11} B_{11} + F_{52,21} B_{21}) + \\ &\quad P_{71,61} \bar{P}_{71,7D} (F_{43,11} B_{11} + F_{43,21} B_{21}) \cdot \\ &\quad (F_{52,11} B_{11} + F_{52,21} B_{21}) + \\ &\quad P_{71,62} \bar{P}_{71,7D} (F_{43,11} B_{11} + F_{43,21} B_{21}) \cdot \\ &\quad (F_{53,11} B_{11} + F_{53,21} B_{21}) + P_{71,7D} \\ &= \bar{P}_{71,7D} (P_{71,61} (F_{42,11} F_{52,11} B_{11} + F_{42,21} F_{52,21} B_{21}) + \\ &\quad P_{71,61} (F_{43,11} F_{52,11} B_{11} + F_{43,21} F_{52,21} B_{21}) + \\ &\quad P_{71,62} (F_{43,11} F_{53,11} B_{11} + F_{43,21} F_{53,21} B_{21})) + P_{71,7D}; \\ B_{21} X_{71} &= \end{aligned}$$

$$\begin{aligned} &B_{21} (P_{71,61} \bar{P}_{71,7D} (F_{42,11} F_{52,11} B_{11} + F_{42,21} F_{52,21} B_{21}) + \\ &P_{71,61} \bar{P}_{71,7D} (F_{43,11} F_{52,11} B_{11} + F_{43,21} F_{52,21} B_{21}) + \\ &P_{71,62} \bar{P}_{71,7D} (F_{43,11} F_{53,11} B_{11} + F_{43,21} F_{53,21} B_{21}) + P_{71,7D}) \\ &= B_{21} (P_{71,61} \bar{P}_{71,7D} (F_{42,11} F_{52,11} B_{11} + F_{42,21} F_{52,21}) + \\ &P_{71,61} \bar{P}_{71,7D} (F_{43,11} F_{52,11} B_{11} + F_{43,21} F_{52,21}) + \\ &P_{71,62} \bar{P}_{71,7D} (F_{43,11} F_{53,11} B_{11} + F_{43,21} F_{53,21}) + P_{71,7D}). \end{aligned}$$

上面已经给定了 $\{b, p, a, r\}$ 类型参数. 注意到 $Pr\{F_{nk;ij}\} = f_{nk;ij} = (r_{ni}/r_n) a_{nk;ij}$, 将上式中的大写字母用相应的小写字母替换, 我们可以和很容易地算出 $Pr\{X_{71}\} = 0.02589, Pr\{B_{21} X_{71}\} = 0.01898$. 最

后, 根据上述条件概率公式可得 $h_{21}^s = 0.73$. 类似地, $h_{11}^s = Pr\{H_{11} | E\} = Pr\{B_{11} | X_{71}\} = 0.19, h_{71}^s = Pr\{H_{71} | E\} = Pr\{P_{71,7D} | X_{71}\} = 0.19$. 鉴于 $S_H = \{B_{11}, B_{21}, P_{71,7D}\}$ 是在 E 条件下仅存的 3 个可能的假设事件, $H_{11} = B_{11}, H_{21} = B_{21}$ 和 $H_{71} = P_{71,7D}$ 的排序概率可分别算得为 $h_{11}^r = 0.17, h_{21}^r = 0.66$ 和 $h_{71}^r = 0.17$. 其中, 排序条件概率的定义如下:

$$h_{kj}^r \equiv \frac{h_{kj}^s}{\sum_{H_{kj} \in S_H} h_{kj}^s} = \frac{Pr\{H_{kj} E\}}{\sum_{H_{kj} \in S_H} Pr\{H_{kj} E\}} \quad (41)$$

满足

$$\sum_{H_{kj} \in S_H} h_{kj}^r = 1 \quad (42)$$

显而易见, 如果 S_H 中只有一个假设事件 H_{kj} , 根据式(41), 我们无须计算 h_{kj}^s 即知 $h_{kj}^r = 1$. 对诊断类推理问题, 如果在 E 条件下化简 DUCG 后仅剩一个假设事件 H_{kj} , 则尽管 $H_{kj} E \neq E$, 诊断推理也无须数值计算即结束, 因为我们已确知 $h_{kj}^r = 1$. 同时, 由于(1)数值计算前已知定性推理结果 S_H ; (2)数值计算仅限于对这些包含在 S_H 中的假设事件, 数据精度对计算结果的影响被大大降低, 因而 DUCG 对参数精度的要求较低. 换句话说, DUCG 对参数精度具有较高的鲁棒性.

读者可能已经注意到: 图 43 是多连通的. 本例显示: DUCG 即使是在多联通情况下, 也无需像 BN 那样应用团算法 (clustering) 或割集条件算法 (cut-set conditioning), 因为 DUCG 无需用任何特殊算法, 仅仅通过事件展开就已自动将多连通分解了. 也就是说, DUCG 的算法与单连通和多连通无关. 这是 DUCG 的又一个优点.

9 结论和后续工作

适用于单赋值情况的不确定因果关系的简洁表达和推理算法并不能简单地用于多赋值情况, 因为其强制归一算法是不恰当的. 为解决多赋值问题, 本文提出了 DUCG 模型. 该模型既可用于单赋值情况 (S-DUCG), 也可用于多赋值情况 (M-DUCG). 其中 M-DUCG 模型也同时可用于单赋值情况. 文献 [23] 提出的简洁表达和推理应当满足的充分性和可分离性在 DUCG 模型中都得以实现. 此外, 基于所观察到的证据 (包括事件发生时序) 而化简后的 DUCG, 本文提出了一个新的概率推理的工具, 即事件展开算法. 该算法与化简后的 DUCG 是单连通还是多连通无关. 文中提出了一套化简的 DUCG 和进

行事件展开的运算规则. 通常, 化简后的 DUCG 可在进行数值计算之前提供一个定性的、有时甚至是最终的推理结果. 而且, 得益于事件展开算法, DUCG 对于表达 CPD 来说可以是不完备的, 即 DUCG 不必表达不受关注的因果关系, 尽管这些因果关系对于表达 CPD 来说是必须的. 用数学的语言来说, 尽管 DUCG 能够是(在没有 DCG 的情况下)、但并不必须是一组变量的联合概率分布的表达. DUCG 的不完备性、处理 DCG 的能力和与简洁表达普通 CPT 不同的条件连接事件和作用事件的引入, 使得 DUCG 是一个超越了 BN 的新的理论体系. 最后, 文中还提出了一个处理绑定了权重系数的事件的逻辑运算新算法.

限于篇幅, 本文仅处理静态、离散、证据确定和有向无环图 DAG 的情况. 然而, DUCG 的目标还在于处理有向循环图 DCG、证据在线动态变化、因果关系强度是时间的函数、过程系统所涉及的初因和非初因事件、权重后的以前因果作用在当前时刻的连续叠加、同一 DUCG 中确定和不确定因果关系的自由混合、不确定或模糊证据、连续和离散变量的自由混合等问题. 这些问题将在本文的后续论文中分别论述. 后续的另一篇论文将讨论 DUCG 在某核电站二回路系统动态故障诊断中的应用. 该应用涉及 100 多个真实变量并包含 DCG 结构.

参 考 文 献

- [1] Shortliffe E H, Buchanan B G. A model of inexact reason in medicine. *Mathematical Bioscience*, 1975, 23(3-4): 351-379
- [2] Shafer G. *A Mathematical Theory of Evidence*. Princeton, NJ: Princeton University Press, 1976
- [3] Duda R O. Development of the PROSPECTOR consultation system for mineral exploration. Final Report, SRI International, 1978
- [4] Zadeh L A. The role of fuzzy logic in the management of uncertainty in expert systems. *Fuzzy Sets and Systems*, 1983, 11(1-3): 199-227
- [5] Pearl J. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 1986, 29(3): 241-288
- [6] Pearl J. *Probabilistic Reasoning in Intelligent Systems*. San Mateo CA: Morgan Kaufmann, 1988
- [7] Srinivas S. A generalization of the noisy-OR model. *Uncertainty in Artificial Intelligence*. San Francisco: Morgan Kaufmann, 1993: 208-215
- [8] Gilio A, Scozzafava R. Conditional events in probability assessment and revision. *IEEE Transactions on Systems, Man and Cybernetics*, 1994, 24(12): 1741-1746
- [9] Jensen F V. *An Introduction to Bayesian Networks*. New York: Springer-Verlag, 1996
- [10] Heckerman D, Breese J S. Causal independence for probability assessment and inference using Bayesian networks. *IEEE Transactions on Systems Man Cybernetics*, 1996, 26(6): 826-831
- [11] Poole D et al. *Computational Intelligence*. New York: Oxford University Press, 1998
- [12] Cowell R G, Dawid A P, Lauritzen S L, Spiegelhalter D J. *Probabilistic Networks and Expert Systems*. New York: Springer, 1999
- [13] Jensen F V. *Bayesian Graphical Models*. *Encyclopedia of Environmetrics*, Wiley, Sussex, UK, 2000
- [14] Russell Stuart J, Norvig Peter. *Artificial Intelligence: A Modern Approach*. 2nd Edition. Prentice Hall, 2003
- [15] Pearl J, Russell S. *Bayesian networks*//Arbib Michael A ed. *The Handbook of Brain Theory and Neural Networks*. 2nd Edition, MIT Press, 2003
- [16] Poole D, Zhang N L. Exploiting contextual independence in probabilistic inference. *Journal of Artificial Intelligence Research*, 2003, 18: 263-313
- [17] Schubert L K.. A new characterization of probabilities in Bayesian networks//*Proceedings of the UAI-04*. Banff, Canada, 2004: 7-11
- [18] Henrion M. Practical issues in constructing a Bayes' belief network. *Uncertainty in Artificial Intelligence*, Amsterdam, North Holland, 1988: 132-139
- [19] Diez F J. Parameter adjustment in Bayes networks: The generalized noisy-OR gate. *Uncertainty in Artificial Intelligence*, Washington D. C., Morgan Kaufman, 1993, 99-105
- [20] Pradhan M, Provan G, Middleton B, Henrion M. Knowledge engineering for large belief networks//*Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence*. 1994: 1
- [21] Bouillier Craig, Friedman Nir, Goldszmidt Moises, Koller Daphne. Context-Specific Independence in Bayesian Network//*Proceedings of the UAI-96*, 1996
- [22] Zhang Qin. Probabilistic reasoning based on dynamic causality trees/diagrams. *Reliability Engineering and System Safety*, 1994, 46(3): 209-220
- [23] Pfeffer Avi. Sufficiency, separability and temporal probabilistic models//*Proceedings of the UAI-01*, 2001
- [24] D'Ambrosio Bruce. Local expression languages for probabilistic dependence. *Int. J. Approximate Reasoning*, 1995, 139(1): 61-68
- [25] Lucas Peter J F. Bayesian network modeling through qualitative patterns. *Artificial Intelligence*, 2005, 163(2): 233-263
- [26] Milch Brian, Marthi Bhaskara, Sontag David, Russell Stuart, Daniel L. Ong and andrey kolobov. Approximate inference for infinite contingent Bayesian networks//*Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*. Barbados, 2005

[27] Chavira Mark, Allen David, Darwiche Adnan. Exploiting evidence in probabilistic inference//Proceedings of the 21st UAI, 2005

[28] Zhang Navin Lianwen, Poole David. A simple approach to Bayesian network computation//Proceedings of the 10th Biennial Canadian Artificial Intelligence Conference, 1994



ZHANG Qin, born in 1956, Ph. D., professor, Ph. D. supervisor. His research interests include system reliability, safety of nuclear power plant, expert system for fault diagnoses and uncertain knowledge representation and reasoning, etc.

Background

The knowledge representation and reasoning dealing with uncertain causalities are crucial for intelligent systems. Many models have been developed such as Certainty Factor^[1], Evidence Reasoning^[2], PROSPECTOR^[3], Fuzzy Logic^[4], Bayesian Network (BN)^[5-20], etc. Among them, BN is becoming increasingly prominent^[12,16]. The compact knowledge representations and efficient inference algorithms are usually the core issues in the research of BN. It is well known that the typical representation of conditional probability distribution in BN is the conditional probability table (CPT). However, it is also noted that there are too many parameters to specify in a CPT. In order to provide compact representations, many efforts have been made, such as noisy-OR^[6], context-specific independence (CSI)^[21], dynamic cau-

sality diagram (DCD)^[22], etc. However, it is noted that they are presented for or illustrated with the binary variables, while these cases are actually single-valued instead of multi-valued. In this paper, it is pointed out that the single-valued cases are essentially different from the multi-valued cases, which means that the compact representations and the corresponding inference algorithms applicable in single-valued cases are not applicable in multi-valued cases. There is a need to develop a new model to overcome this problem. Moreover, the incompleteness of knowledge representation and the compact representation and inference algorithm for various complex logic relations among variables are also problems to be solved. DUCG (dynamical uncertainty causality graph) presented in this paper is such a solution to all these problems.