

逻辑 AI 中的容变能力的研究^{*}

朱建中 戈也挺 骆斌 陈世福

(南京大学软件新技术国家重点实验室 南京210093)

Elaboration Tolerance in Logical AI

ZHU Jian-Zhong GE Ye-Ting LUO Bin CHEN Shi-Fu

(State Key Laboratory for Novel Software, Nanjing University, Nanjing 210093)

Abstract In this paper a survey of elaboration tolerance in logical AI is provided. John McCarthy views elaboration tolerance as the key property of any formalism that can represent information in the common sense informatic situation. The goal of studying elaboration tolerance is finding a formalism for describing problems logically that is as elaboration tolerant as natural language and the associated background knowledge. In the beginning, we introduce the missionaries and cannibals problem and its elaboration problems provided by John McCarthy as the test examples of studying elaboration tolerance. Then we introduce the study of elaboration tolerance from three aspects. First of all, the study of elaboration tolerance of the existing systems is introduced such as Causal Calculator and ABSFOL. Second the study of special elaboration is presented such as elaboration of actions. Last but not least a formal definition of elaboration tolerance and evaluation tools is provided.

Keywords Elaboration tolerance, Formal system, Common sense informatic situation

1 引言

John McCarthy^[1]提出要达到更高级的智能,关键之一在于系统能有效地处理常识信息情境(common sense informatic situation)。与常识信息情境相对的是有界信息情境(bounded informatic situation)。在有界信息情境中,与实现目标相关的信息在系统构建前已经确定,即系统初始时拥有所需的全部信息。例如在下棋程序中,规则和棋盘的初始格局与目标都是确定的。传统上,大多数 AI 系统只在有界信息情境中研究问题。而在常识信息情境中,已知的信息是不完全的。例如:现有一个气压计,要求测量大楼的高度。此问题中,我们不知道气压计是否坏了,也不知道是否有其它工具。在常识情境中求解问题与在有界情境中的本质差别在于使用缺省设置(default)。如在不知道气压计是否坏了的情况下默认它是好的,我们可以用它来测量气压。随着对问题的认识不断深入以及问题的不断变化,我们可能会得到进一步的信息,如气压计是坏的,则通过测量气压来解决问题就不对了。我们希望解决这类变化了的问题不需要完全从头开始,于是提出了容变能力(elaboration tolerance)的研究。因为目前的研究者主要是用逻辑的方法来研究容变能力,所以下文讨论的容变能力也局限于逻辑 AI。

McCarthy 及其他研究者认为容变能力主要包含两方面:1. 当问题发生变化时,表示变化了的问题不需完全从头开始,只需在表示原问题的逻辑系统中加几个句子(sentence)就足够了;2. 求解变化了的问题的过程不需要完全从头开始,可以通过对原问题的求解结果作某些修改得到。McCarthy 认为人类的自然语言是具有良好的容变能力。研究容变能力的目标是:开发出一个与自然语言具有同样容变能力的形式系统。

容变能力研究对人工智能有着广泛的意义。McCarthy^[2,3]认为容变能力是常识推理系统必须具备的能力,是常识推理系统的一般性质。Cimatti^[10]提出在情景 agent(situated agent)的研究中需要考虑容变能力。如果情景 agent 有容变能力,则它就能重用一个新的表示,而不是从头开始构造一个新的表示;Ipke Wachsmuth^[15]认为自然语言理解系统的设计需要考虑容变原则。在向知识库中增加新的事实来适应变化时,形式化的推理会因为要处理更多的公理而变慢。如何保证系统的推理过程不会随着不相关的知识加入而变慢,则是容变原则需要考虑的问题。

与容变能力研究密切相关的有非单调推理、行动推理、信念修正等。非单调推理是提高系统容变能力的重要技术^[2,5,7]。而作为常识推理重要基础的行动推理必须研究容变能力。与容变能力相比较,研究信念修正所关心的是在原有理论(或知识库)中增加或删除一条给定语句(或知识)所产生的结果,而研究容变能力所关心的是为达到想要的效果必需对原有理论(或知识库)作哪些改变。

2 野人过河问题及其19个变化问题

系统地研究人类容变能力是困难的,但可以给出一些研究系统容变能力的测试例子:McCarthy^[2]给出野人过河问题(Missionaries and Cannibals Problem)及其19个变化问题作为这样的测试例子。

野人过河问题:三个传教士和三个野人要过河,河边有一艘能容两个人的船。而在河的任一岸,如果野人数超过传教士的数目,传教士将被吃掉。问如何过河?

19个变化问题如下:

(1)船是划艇;

^{*} 本文得到国家自然科学基金(60003010)资助。朱建中 硕士,研究方向为人工智能与非经典逻辑。戈也挺 硕士,研究方向为人工智能与非经典逻辑。骆斌 博士,副教授,研究方向为人工智能,智能数据库。陈世福 教授,博士生导师,研究方向为人工智能。

- (2)每个人都戴有帽子,且各不相同,在没有进一步说明帽子的作用之前,此变化对原问题求解没有影响;
- (3)有4个野人4个传教士(此问题无解);
- (4)船可容纳3个人(则4个野人4个传教士可过河);
- (5)船每一岸有一个浆,两个人同时过河船上必需有两个浆;
- (6)只有一个野人一个传教士会划船;
- (7)传教士不会划船(此问题无解);
- (8)最大的野人不能和其它任何人一起过河;
- (9)如果最大的野人和最小的传教士单独在一起,传教士会被吃掉;
- (10)有一个传教士可涉水过河;
- (11)三个传教士与一个野人单独在一起时,野人会被转化为传教士;
- (12)野人单独在船上时有0.1的可能把船偷走,问如何才能保证过河成功的概率在0.9以上;
- (13)有座桥(任意数目的人可过);
- (14)船漏水,必须一边划船一边舀水;
- (15)船可能会坏,而且必需回到左岸修理;
- (16)河中有个孤岛;
- (17)有4个野人4个传教士,但最强壮的传教士划得足够快,以致野人还没饿到要吃传教士时他就能赶回来;
- (18)有4个野人4个传教士,但野人一开始不饿.传教士只有有限的食物,但他们能判断出哪个野人更饿,从而可通过给饿了的野人吃东西来避免麻烦;
- (19)另外还有一批野人和传教士,但两批人离得太远而不能交互。

McCarthy 对他所提的变化作了简要的分类.变化类型有:增加前提、动作、动作的结果及对象,1、2、5、13、16、17、18等变化可归入此类.有关这类变化将在第4节介绍;改变参数,如人数由三个变为四个,3、4、17、18等变化可归入此类;使得属性依赖情景,如野人可能被转化为传教士(11);简单并行,如(10);完全并行,如(14);情景比较,如(18)要求能在特定情景中比较哪个野人更饿等等.当然还可列出许多不同类型的变化.这种分类是比较原始的、随意的。

3 现有系统的容变能力的研究

围绕 McCarthy^[2]给出的野人过河问题及其19个变化问题,有不少研究者对现有形式系统的容变能力进行研究,宣称这些系统在不同程度上具有了 McCarthy 说的容变能力^[7,10,11,13].Lifschitz^[7]宣称他们的 ccalc 可解决其中10个问题.Adolfo Villafiorita^[13]介绍了基于 abstraction 理论的形式系统 ABSFOL 的容变能力.Gustafsson 和 Kvarnström^[11]提出用 TAL-C 逻辑结合面向对象机制可解决大部分问题。

3.1 Causal Calculator

Causal Calculator^[9]是一个以因果理论^[8]为基础的推理机,可用来查询与规划.Lifschitz 主要采用了如下三种技术处理使得系统具有较强的容变能力:

1. 引入 ab 谓词,即引入非单调推理机制.非单调推理对提高系统的容变能力是非常有用的.例如:现在有这样一条命题模式:always $\forall L : at(boat, L)$,表示同一时刻船只能在一个位置.假想现在有一动作把船打成碎片,并把一些碎片移到对岸.显然,在这种情况下“同一时刻船只能在一个位置”的命题就不再成立.需要某种机制使得命题 always $\forall L : at$

(boat, L)失效.非单调推理机制可解决上述问题。

2. 为了更方便地处理有关动作的变化,Lifschitz 把动作分解为动作及其属性.如第17个变化中提到:强壮的传教士划得快.这就要求在过河动作中增加对速度的描述,要区分划船快的和慢的.但是改变谓词或函数的参数个数意味着对原有描述的较大改变,是应该尽量避免的.解决办法是,把动作分解成动作及其属性(属性也可看作动作).这样增加限制条件就相当于增加动作的属性.只需加 cross-howfast(fast)作为动作 cross 的属性即可。

3. 因为除了野人和传教士外还有其他类别:会划船的、强壮的、瘦弱的等,所以 Causal Calculator 引入 group 的概念来加以区分各个类别。

Lifschitz 在解决第11、13个变化问题时分别限制:把野人转化为传教士的动作不能与过河动作同时进行;从桥上过河与坐船过河不能同时进行.Lifschitz 认为这两个问题中,并行的两个动作都会改变河两岸野人、传教士数目,这类并行是“difficult”的,在他的方案中不能处理.同理,第10个变化问题他也解决不了.另外,ccalc 在做规划时必需事先给出规划长度,也就是说 ccalc 只能告诉我们问题没有给定长度的解,而不能告诉我们问题无解,所以 ccalc 对无解问题(3,7)无能为力.对概率变化(12)和要求把事件分解成几部分的变化(15要求把过河动作分解为两部分),ccalc 也不能解决。

3.2 ABSFOL

ABSFOL 是以 abstraction 理论为基础交互式定理证明系统. Adolfo Villafiorita 研究了野人过河问题的两种表示方法,一种是用在初始岸传教士、野人以及船的数目表示状态(如初始状态表示为331),称为数值表示,另一种是给每个人起一个名字,称为命名表示(命名表示可看成是野人过河问题的一个变化,即每个人都有名字).他把数值表示法用 ABSFOL 的输入语言表示得 Σ_n ,并在 ABSFOL 上进行求得过河方案 Π_n 。

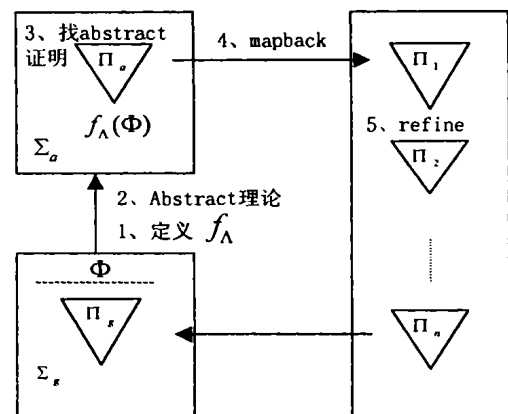


图1

如图1,首先把命名表示法用 ABSFOL 的输入语言表示得 Σ_n ;然后定义 abstraction 函数 f_Λ ,把命名表示 Σ_n 映射到数值表示 Σ_a ,命名表示的目标 Φ 映射为 $f_\Lambda(\Phi)$;接着在数值表示 Σ_a 上求解问题,即证明 $f_\Lambda(\Phi)$,证明过程 Π_a 就是过河方案,而数值表示的过河方案 Π_n 已求得;最后再把求解结果映射回(mapback)命名表示,由于函数 f_Λ 的逆映射可能有一对多的情况,所以 Π_a 通过 f_Λ 的逆映射得到的 Π_1 中可能有参数,参数表示可能的选择,把 Π_1 中的参数全部实例化就可得名命名表示的解 Π_n (这个过程称为 refinement)。

上述过程表明,命名表示的求解过程不需完全从头开始,而可以通过 abstraction 函数把数值表示的求解结果映射过来。Adolfo Villafiorita 认为可以以命名表示为基础用同样的方法去求解 McCarthy 提出的那些变化中的部分问题,如:2、5等。但是 abstraction 理论也不能解决所有变化问题,比如13(有座桥)。因为问题13的解的结构与命名表示的解的结构差别太大,以致无法建立如上面的映射机制。即 abstraction 理论不能解决那些要改变解的结构的变化。

3.3 扩展的 TAL-C

TAL-C 是时序逻辑(Temporal Action Logics)的一种。用 TAL-C 进行行动推理的过程如下:首先用 TAL-C 的表层语言(surface language)L(ND)表示问题,然后系统把问题描述翻译到 TAL-C 的底层语言(base language)L(FD)。L(FD)再被转化为逻辑等价的一阶逻辑理论。Gustafsson^[11,12]提出用面向对象机制扩充 TAL-C 的表层语言 L(ND),进而用面向对象方法来提高该系统的容变能力。他引入新的宏对 L(ND)加以扩充,而新的宏都可以翻译到 L(ND)。有了面向对象机制,我们可从如下几方面获得所需要的容变能力。

1. 新的类(可以是已有类的子类)可以很容易地被加入。
2. 类都可以通过增加新的属性和方法加以扩展,而不需要修改类定义的原有部分。
3. 超类中的方法实现可在子类中重定义(override)。

Gustafsson 用扩充了的 TAL-C 来解决野人过河问题及其各种变化问题。他只讨论 TAL-C 对问题的 domain specification 的容变能力。首先定义原问题的 domain specification,即问题涉及的类、类的属性、方法及类层次。要得到各种变化的 domain specification,只需在原问题的 domain specification 基础上通过上述方法增加新的内容即可。把面向对象机制引入推理系统可以很好地提高系统的容变能力,这方面的研究才刚刚开始。

4 对动作变化容变能力的研究

在常识推理中,对有关动作的变化具有重要的意义。动作变化(elaboration of actions)主要有增加或减少动作的前提和结果。目前对动作变化的研究主要有 Amir^[5]、Chitta Baral 与 Le-Chi Tuan^[14]。

4.1 Amir 对动作变化的研究

Amir 是在 Lifschitz 因果理论的基础上加以研究,但研究的结果也可推广到其它系统。他给每一个变化起一个名字,然后用一个名叫 elaborate 的函数把动作与变化名映射到变化了的动作。如 using-boat 是动作 go(group, bank) 的一个变化,那么变化了的动作 a' 用 elaborate(go(group, bank), using-boat) 表示。再如 Using-bridge 也是动作 go(group, bank) 的一个变化。有关一个动作的变化名的集合用 Elaborations(a) 表示。Elaborations(go(group, bank)) = {using-boat, using-bridge}。另外采用非单调推理技术对 Elaborations(a) 极小化,假设一个动作的所有变化就是在知识库中所提到的。

这样就得到由动作及其变化生成新的动作的变化公理: $a' = elaborate(a, e)$ 。这一变化公理在某种程度上给了我们所需的动态性: $property(a') = ((property(a) \wedge \varphi) \vee \Psi)$ 。这里的 φ, Ψ 是两个公式,描述了从动作 a 到 a' 的相对属性的变化。然而这也带来一个问题,就是系统如何在许多可能的变化动作中挑选合适的动作。在上面的例子中动作 go(group, bank) 就不能被选择,而两个变化了的动作都是可选的。因为两个变化

动作要求到河对岸必须坐船或从桥上走。Amir 引入新的谓词 chosen(a, s), 表示动作 a 在状态 s 下可以被选择。动作 a 在状态 s 下可执行谓词 Applicable(a, s) ($Applicable(a, s) \Leftrightarrow chosen(a, s) \wedge Poss(a, s)$)。而因果理论中的结果公理也变为:

$$Applicable(a, s) \wedge Precond^+(F, x, a, s) \Rightarrow F(x, result(a, s))$$

$$Applicable(a, s) \wedge Precond^-(F, x, a, s) \Rightarrow \neg F(x, result(a, s))$$

现在已经完成对因果理论的改造,用改造后的理论来描述动作变化是比较方便的。例如:描述“用马达船过河不需要桨”,只需加入如下公式就可以了 $is-motorboat \in Elaborations(elaborate(go(group, bank), using-boat))$ 。

4.2 有关概率的动作变化

一个动作执行可能产生多种结果,每种结果的出现有一定的概率。为了描述有关概率的动作变化,Chitta Baral 与 Le-Chi Tuan 在原有的因果理论基础上引入一个称为未知变量的集合 U。各未知变量之间相互独立。未知变量的值不受动作的影响,也不依赖于任何流(fluent),它们的值由模型以外的因素决定。但是作用在流上的动作所产生的结果却可能依赖于未知变量,同样在某一状态下的流的值也可能依赖于未知变量。

引入未知变量后,因果理论的域描述语言变为如下形式:

- (1) a causes Ψ if φ
- (2) φ causes Ψ
- (3) impossible a if φ

其中 a 表示动作, Ψ 表示流公式, φ 流公式或者未知变量。未知变量 u 值为 true 的概率表示为 probability of u is n (n 是 0、1 之间的有理数)。这样扩展后的因果理论就可以表示概率变化。

5 容变能力的比较模型

为了评价形式系统的容变能力, Amir^[4] 提出一个知识库的容变比较模型。

5.1 知识库变化模型

这里所讨论的知识库限于公理形式化系统(axiomatic formal systems)。

定义1 公理形式化系统 Σ 是个三元组 $\langle L, | \sim, \Gamma \rangle$, 其中 L 是语言, Γ 是公理集合, $| \sim$ 是推理关系。

本文中把语言看作是语句集合,用 $|L|$ 表示语言 L 的命题符号集。用 $C(\Sigma)$ 代表由形式系统 Σ 的公理推出的所有结论组成的理论: $C(\Sigma) = \{\Gamma | \sim \varphi\}$ 。

定义形式系统的相等关系: $\Sigma \equiv \Sigma'$ iff $C(\Sigma) = C(\Sigma')$ ($C(\Sigma) = C(\Sigma')$ 是集合相等关系)。

每个动作都将一个形式系统转变为另一个形式系统。这里限制只有增加和删除公理两个动作。如果 $\varphi \subset L$ 是一条公理,把 φ 加到 Σ 的结果定义为: $add(\varphi)(\Sigma) = \langle L, | \sim, \Gamma \cup \{\varphi\} \rangle$ 。

从知识库中删除公理,用 $delete(\varphi)(\Sigma)$ 表示: $delete(\varphi)(\Sigma) = \langle L, | \sim, \Gamma \setminus \{\varphi\} \rangle$ 。

为了表示在知识库上执行一系列动作后的结果,知识库上执行动作序列 $\alpha = \langle a_1, \dots, a_n \rangle$ ($\{a_i\}_{i \leq n}$ 都是形如 $add(\varphi)$ 或 $delete(\varphi)$ 的动作)表示为: $\alpha(\Sigma) = a_n(a_{n-1}(\dots(a_1(\Sigma))\dots))$ 。如: $\alpha = \langle delete(\Psi), add(\varphi) \rangle$ 则: $\alpha(\Sigma) = add(\varphi)(delete(\Psi)(\Sigma))$ 。Actions(L) 定义为所有能在 L 上执行的动作序列 α 组成的集

合。

5.2 比较容变能力

为了比较两个系统,需先把它们翻译到某个共同的基础之上。

定义2 翻译(Translation) 一个翻译 t 是个部分函数,它输入一个形式系统返回另一个形式系统。即 $t: S \rightarrow S$ 是个部分函数,其中 S 是公理化形式系统的集合。

形式系统之间基于 t 的恒等关系定义为: $\Sigma_1 \equiv_t \Sigma_2$ iff $C(t(\Sigma_1)) = C(t(\Sigma_2))$ 。

给定两个公理化形式系统, Σ, Σ_{target} 和翻译 t 。解决变化问题是找到形如 $\alpha(\Sigma) \equiv_t \Sigma_{target}$ 一个变化(也就是一串动作) α 。

定义3 句法距离(Syntactic Distance) Σ, Σ_{target} 是两个公理化形式系统。从 Σ 到 Σ_{target} 句法距离是: $dist_t(\Sigma, \Sigma_{target}) \stackrel{def}{=} \min\{len(\alpha) \mid \alpha \in Actions(L) \wedge \alpha(\Sigma) \equiv_t \Sigma_{target}\}$ 。这里 $len(\alpha)$ 是 α 中的动作个数。当没有任何动作序列可把 Σ 转变到 Σ_{target} 时,令 $dist_t(\Sigma, \Sigma_{target})$ 是 ∞ 。

定义4 变化比较(Elaboration Comparison) $\Sigma_1 = \langle L_1, |\sim_1, \Gamma_1 \rangle, \Sigma_2 = \langle L_2, |\sim_2, \Gamma_2 \rangle$ 是两个公理化形式系统, t 是一个形如 $\Sigma_1 \equiv_t \Sigma_2$ 的翻译。

给定 t, Σ_1 在动作序列 $\alpha \in Actions(L_1)$ 上比 Σ_2 更句法容变 (syntactically elaboration-tolerant) 定义为: $\Sigma_1 \leq_{t,a} \Sigma_2 \stackrel{def}{=} dist_t(\Sigma_1, \alpha(\Sigma_1)) \leq dist_t(\Sigma_2, \alpha(\Sigma_2))$ 。

给定 t, Σ_1 比 Σ_2 更句法容变(简称容变)可定义为:

$$\Sigma_1 \leq_t \Sigma_2 \stackrel{def}{=} \forall \alpha \in Actions(L_1) \Sigma_1 \leq_{t,a} \Sigma_2 \wedge \forall \alpha \in Actions(L_2) \exists \alpha' \in Actions(L_1) (\alpha'(\Sigma_1) \equiv_t \alpha(\Sigma_2))$$

5.3 一些结论与模型评价

运用上述比较模型 Amir 得到一些有用的结论: 在只允许增加知识的条件下,一个具有较大符号集的命题知识库比一个等价的但拥有较少符号的知识库更容变;非单调推理并不比等价的单调推理有更强的容变性。但对任意单调推理理论,能构造与之等价的但具有更强容变能力的非单调理论;如果增加、删除动作都允许的话,没有最容变的系统。

在 Amir 的比较模型中,只是比较两个系统的句法容变能力,而句法容变能力并不完全对应直觉上的容变能力。用句法距离作为比较容变能力的权函数也许不是最好的,可能有很多其它的选择。另外,在他的模型中假设已有一目标系统 Σ_{target} 的存在,如何找到 Σ_{target} 是至关重要的,但是在他的文章中却没有考虑。

总结与展望 目前对容变能力的研究还很不充分,没有一个系统能完美解决 McCarthy 提出的野人过河问题的19个变化问题。即使有系统能解决这全部的变化问题,也不能说该系统就有了足够的容变能力。已经有更多的例子更多的变化被提出。如:Ernie Davis^[16]提出“Egg Cracking”问题,并给出许多种变化。Murray Shanahan^[17]给出“Egg Cracking”问题及其变化问题的一个基于事件演算的形式化描述;McCarthy^[1]还提到他们现在用于讨论的野人过河问题的变化问题已有25

个。研究容变能力的最终目标是要系统能像自然语言那样容变各种变化,这一目标还远未达到。值得研究的问题还很多:利用面向对象的机制可以很好地提高系统的容变能力,这方面的研究还远未展开;动作变化研究有待进一步深入等。

容变能力已越来越引起研究者的注意,被视为表达常识信息情景的形式系统必须具备的关键属性。对容变能力的研究对我们构造新的 AI 系统具有重要的指导意义。相信将来会得到更多的研究。

参考文献

- 1 McCarthy J. From here to human-level AI. In Knowledge Representation 96, p. 640~646
- 2 McCarthy J. Elaboration tolerance. In Common Sense 98, London, Jan. 1998
- 3 McCarthy J. Mathematical logic in artificial intelligence. Daedalus, 1988, 117(1): 297~311
- 4 Amir E. Towards a formalization of elaboration tolerance: adding and deleting axioms. Symposium on Abstraction, Reformulation and Approximation (SARA98), 1998
- 5 Amir E. Machinery for elaborating action. Workshop on Non-monotonic Reasoning, Action, and Change, Nagoya, Japan 1997
- 6 Amir E. Elaboration tolerance of logical theories. In: Proc. of the Sixteenth National Conf. on Artificial Intelligence (AAAI'99), Doctoral Consortium, 1999
- 7 Lifschitz V. Missionaries and Cannibals in the Causal Calculator. In Principles of Knowledge Representation and Reasoning. In: Proc. of the Seventh Intl. Conf. (KR2000), April 2000. 85~96
- 8 McCain N, Turner H. Causal theories of action and change. on Artificial Intelligence (AAAI'97), AAAI, 1997. 460~465
- 9 McCain N, the Texas Action Group. The causal calculator. <http://www.cs.utexas.edu/users/tag/cc/>
- 10 Cimatti, Serafini. Multiagent reasoning with belief contexts II: elaboration tolerance. In: Proc. 1st Int. Conf. on Multi-Agent Systems (ICMAS-95), 1996. 57~64
- 11 Gustafsson J, Kvarnström. Elaboration tolerance through object-orientation. In Working Notes of Common Sense 2001
- 12 Gustafsson J. Object-oriented Reasoning about Action and Change. In: Proc. of the Seventh Scandinavian Conf. on Artificial Intelligence
- 13 Villafiorita A. Abstraction as a form of elaboration tolerance. In: 8th Intl. Conf. on Artificial Intelligence: Methodology, Systems, Applications - AIMS'98, Lecture Notes on Artificial Intelligence, Sozopol, Bulgaria, Springer-Verlag, 1998. 427~437
- 14 Baral C, Tuan L-C. Reasoning about actions in a probabilistic setting. In Common Sense 2001
- 15 Wachsmuth I, Gängler B. Knowledge Packets and Knowledge Packet Structures. Text Understanding in LILOG, LNCS, 1991, 546: 380~393
- 16 Davis E. Physical Reasoning: Cooking Problem. The Common Sense 98 Problem page. <http://www.dcs.qmw.ac.uk/conferences/CS98/CS98Problems.html>
- 17 Shanahan M. A Logical Formalisation of Ernie Davis's Egg Cracking Problem. <http://www.dcs.qmw.ac.uk/conferences/CS98/CS98Problems.html>