QUIOS的事件、进程和推理算法*

韦梓楚

(中国科学院数学研究所 北京 100080)

摘要 本文讨论开放型物理系统的定性推理方法. 文中阐释QUIOS机制下事件和进程的关系及两种视图的表达方式,用实例说明定性推理方法,并给出QUIOS的推理算法.

关键词 定性推理,开放系统,事件,进程,视图,推理算法.

中图法分类号 TP18

文献[1]提出了面向开放系统的定性推理的新描述机制QUIOS,以求消除现有定性推理方法的某些不足. 此机制在QPT的基础上建立个体、参量、事件和进程4种并行的视图体制,并考虑以事件结构为骨架来组织进程,启动推理操作. 该文侧重讨论了参量视图的有关问题,其他尚未具体涉及. 本文阐述事件和进程的关系及两种视图的表达方式,用一个实例描述这种机制下的推理方法和特点,说明它适用于一般的客观世界,同时也给出QUIOS的推理算法.

1 QUIOS的事件和进程体系

QPT方法只能描述封闭的世界,文献[2]中的一段话说得明白:"唯一机制假设"认为"物理系统中的所有变化都是直接或间接地由进程引起的". 因此,QPT虽有event(事件)的概念,但它们并不显式地出现于物理系统的描述中,只是作为进程演变的结果而隐式地存在. 在QUIOS中,事件走到前台,显式地出现于系统描述中. 它既可以由在此之前的某个进程激发,也可以突然、独立地产生. 同时,进程的启动和结束都是某个显式或隐式事件的结果. 因此,QUIOS中进程和事件的关系和作用相对于QPT来说倒了个. 对于事件和进程,我们也用视图描述.

1.1 事件视图

事件视图包含视图中涉及的个体和参量条件的描述,其参量条件采用前提条件(precondition)和后置条件(postcondition)的形式,分别表明该事件发生的必要充分条件,以及事件发生后出现什么新的条件. 下面的视图描述火箭"点火发射"的事件.

⟨Event View⟩ firelaunch(t,y,df) ⟨Event View⟩ combine(t,nf(y),f_1(y),f_2(y))

---

* 本文研究得到国家科委资助的中法合作PRA基金资助. 作者韦梓楚,1941年生,研究员,主要研究领域为程序设计方法学.

本文通讯联系人:韦梓楚,北京 100080,中国科学院数学研究所

本文1996-08-28收到修改稿

```
〈Individuals〉
  y:Rocket
〈Preconditions〉
  fire(y)=false
  value(lnr_intensity(drivforce(a)))=df
  sum(df,[3]_intensity(lnr_intensity(gr(y))))
    〉ZERO      /* 推力大于重力 */
〈Postconditions〉
  y H fire: bool
  fire(y)=True /* 火箭在 t 时刻点 */
  Call_event    /* 火,有向上净力 */
  combine(t,netforce(y),drivforce(y),gr(y))
〈End〉 firelaunch
```

```
〈Individuals〉 y:Object
〈Preconditions〉
  y H nf: Linear_force /* 所受合力 */
  applied_to(nf)=y
  y H f1,f2: Linear_force
  applied_to(f1)=y
  applied_to(f2)=y
〈Postconditions〉 /* f1 和 f2 组 */
  for i=1 to 3 do   /* 成合力 nf */
    get([i]_intensity(lnr_intensity(nf)),
      sum([i]_intensity(lnr_intensity(f1)),
          [i]_intensity(lnr_intensity(f2))))
〈End〉 combine
```

时间是定性进程理论中一个重要概念. 我们把它当作事件与进程中的普适实值参变量, 用 t 表示. 进程中还可用 Δt 表示进程延续的时间区间, 其长度为 $|\Delta t|$. 在事件视图 firelaunch 中出现的时间 t 是一个抽象的时间点. QUIOS 恒认为每个事件都是瞬时的, 因此可用一个只有相对意义的时间点来指定它是何时发生的, 并把它作为事件视图的第一个参数.

视图中用到了类似于普通程序设计语言中的求和(sum)、赋值(get)和循环(for)操作. 循环操作只是一种简化写法. 求和及赋值分别表现参量条件的综合或更新, 只在必要时作为定性推理的辅助手段, 以减少一些困难问题的描述复杂性(如判断多个力的合力是否为零), 引用少量这样的简单函数不至于与定性推理的普遍原则与和谐性发生根本矛盾.

1.2 进程视图

进程与事件不同, 它刻画一些参量条件得到满足时必然持续发生的过程以及这个过程对物理系统不断产生的影响. 因此, 进程视图包含对所涉及个体, 需满足的参量条件和所产生的影响的描述. 下面的视图描述因为力的作用引起的物体"加速"和"移动"进程, 其中不必关心物体质量 m 对加速度的影响.

```
〈Process View〉 acceleration(a,f)
  〈Individuals〉a:Object
  〈Quantity Conditions〉
    a H f:Linear_force /* 有作用力 */
    applied_to(f)=a
    lnr_intensity(netforce(a))=lnr_intensity(f(a))
  〈Influence〉        /* 速度的变化与力强相关 */
    for i=1 to 3 do /* 速度不变对应于力消失 */
    (D[speed(a)[i] ∝
        [i]_intensity(lnr_intensity(f))
    D[speed(a)[i]]=ZERO↔
        [i]_intensity(lnr_intensity(f))=ZERO)
  〈End〉 acceleration
```

```
〈Process View〉 move(y,spd)
  〈Individuals〉y:Rocket
  〈Quantity Conditions〉
   speed(y)=spd        /* 至少某个方 */
   (∃ i∈{1,2,3}) (speed(y)[i]≠ZERO) /* 向速度非零 */
  〈Influence〉
   for i=1 to 3 do
    (speed(y)[i]≠ZERO)∧(∀ xi>ZERO)→
      (∃ Δ[i]) (((|Δ[i]| ∝−speed(y)[i]) ∧
      (Δt<Δ[i]→place(y)[i]<xi)∧ /* 可按特指 */
      (Δt=Δ[i]→place(y)[i]=xi)∧ /* 位置划分 */
      (Δt>Δ[i]→place(y)[i]>xi))   /* 史段 */
      〈End〉 move
```

右边的进程表示, 物体运动速度非零时, 则任何前方位置都对应一个与速度有反相关关系的时间区间 $\Delta[i]$($i=1,2,3$, 指直角坐标系的 3 个方向).

进程视图是对进程类的统一描述, 使用不同的参数, 导出不同的进程. 实际用哪些参数, 完全按需要而定, 关键是使视图的参量条件成立. 为使用方便, 也可类似于参量视图, 定义进程视图的子视图(例见附录的 homoggent_motion).

上面的描述中用到了许多对 Object 及其子视图 Rocket 的一切实例来说是普适的参量. 它们的含义不难从名字中看出来. 这两个个体视图的描述见附录.

2　1个实例

作为例子,我们描述火箭发射过程和可能的事故问题.1995年某地卫星发射失败,运载火箭在空中爆炸.曾有专家认为是因为当时发射场上空突然出现西北切变风,造成火箭失控.我们姑不管此论的是非,而用QUIOS模拟一个假想的过程.简化后的情节可用几个显式事件描述:(1)火箭点火发射firelaunch,时间t_1;(2)导向redirect,时间t_2;(3)熄火outfire,时间t_3;(4)…….

显然,首事件不能是任何进程的结果,带有突发性.别的事件有可能是进程的结果,也可能由外部控制产生.即使是前者,由于情节中不计算具体的量(火箭的重量及推力的大小,导向时的实际高度和作用力的强弱等等),由进程推出的结果是不确定的.火箭可能正常运行入轨,也可能意外地出事故.把"爆炸"作为显式事件,可反过来推断在此之前的进程,讨论不确定因素的某些细节,这是以事件来规划进程这种做法的好处之一.

一个物理系统可由若干个体组成,把它看成组合的个体,可用一个视图表示其初态.我们的问题中只有一个个体rocket,附录中的个体视图System用参量条件表示火箭的长、火箭静止于座标中心、朝上(水平和垂直角座标均为零)、重力作用朝下.尚未描述的事件和进程的视图也在附录中给出.

现在可用QUIOS模拟上述情节.设$t_1<t_2<t_3$是时间轴上的几个参考点,我们依次有

(0)视图s:System提供火箭这个小系统的一种初态.

(1)事件firelaunch(t_1,rocket,df_0)发生,火箭点火,造成结果是作用在火箭上的净力是推力df_0和重力对消后的向上冲力.

(2)进程acceleration(rocket,向上冲力)的条件满足,被启动.火箭有了向上速度.

(3)进程move(rocket,speed(rocket))条件满足,被启动.火箭上升,能到达各种高度.

(4)事件redirect(t_2,rocket,h_0,wf,导向力强,$\pi/2$,$\pi/2$)发生.它对上述进程没有影响,但启动了一个并行的进程,即:

(5)进程rotate(rocket,wf,$\pi/2$)被启动.火箭以y轴为轴心向右转向,转动速度与切向力wf的强度有关.转角会不断加大,而且火箭的冲力和切向力也会不断改变.转角达到一定程度,引起新的事件:

(6)事件outfire(t_3,rocket,h_0,θ_0)发生,它使火箭熄火.

我们还可描述下去,并根据事件和进程含有的条件对火箭的行为继续作推理.也有可能出现这样的情节:当$t_1<t_5$时,(1) firelaunch(t_1,rocket,df_0),(2) explode(t_5,rocket),火箭发射后,在t_5时刻发生爆炸事件.这时,先前不一定成立的条件

fire(rocket)=True

direction(rocket)[2]>θ_0

现在是explode的前提条件,必须成立,成为已知条件.查找现有进程,从rotate中推知,要产生此结果,必须有某个α方向的水平力作用于rocket上,或是火箭内部作用引起,或是突然受了外力(如切变风).这个例子反映了QUIOS作反向推理的能力.

3 推理算法

定义. 进程称作活的,假如其参量条件部分的所有条件都被满足. 不是活的进程称作不活的.

现在归纳一下进程启动运行的原则.

(1)事件(显式事件以及由进程运行结果产生的隐式事件)的发生改变了参量的取值,使某些参量条件成立,另一些条件不成立. 结果,有些本来不活的进程变成活的,而有些运行着的进程因其参量条件不再成立而被打断,结束运行. 运行进程的参量条件未受发生的事件影响的,进程将继续运行,直到进程本身使某条件不成立而正常结束. 进程结束运行时因保留其影响而产生隐式事件.

(2) 如果有多个进程同时为活进程,则

〈1〉若这些进程涉及的个体的集合之交为空集,则它们被并行地启动.

〈2〉这些进程涉及的个体的集合之交非空,若属交集的个体的参量均不在这些进程的〈Influence〉部分被改变,则这些进程也被并行地启动;否则若有一组合进程(这些进程的共同父进程)能把它们组合起来,则选用组合后的进程;若不能组合,但其中存在一个进程,是此活进程集合中所有其他进程的子进程,则选用此子进程启动(如用 Gravitation 的进程优先于用 Linear_force 的进程);若不是上述情形,则以不确定方式选择其中之一启动.

基于这个原则,我们可概略给出 QUIOS 的推理算法:

(其中,某某条件集的"闭包"是指此条件集按参量视图集 QV 中的所有推理关系所能得到的最大集,"切片"是指一个时间点和系统在该时刻满足的所有条件所合成的元组)

(1)根据描述一个系统的所有个体、参量、进程和事件视图的定义,列出参量视图集 QV,进程视图集 PV,事件视图集 EV 和实在个体集 SI,以及按时间顺序排的实际事件队列 E_0(其中均删去虚假成份:未在 E_0 和系统初态中出现的个体,未实际用到的参量,用到不存在的个体和参量的进程和事件);SC 取系统初态满足的条件集的闭包. 活进程集 AP 置初态"空",推理方向取"向前",以 SC 为"新条件集".

(2)用"新条件集"对 PV 中的进程实例化,将参量条件均成立的进程加入活进程集 AP. AP 为空时转(4).

(3)按进程启动原则执行. 用启动进程的〈Influence〉部分 得到的条件集的闭包作"新条件集". 用此"新条件集"修改、补充 SC,并用 SC 建新史段和切片,转(9).

(4)当前不是向前推理,则转(7).

(5)集 E_0 为空,则算法终止;否则按序从队列 E0 中退出一个事件. 若其所有前提条件均成立则转(6),否则,作向后推理,用这些前提条件与 SC 不相交的部份,在未启动的进程的〈Influence〉部分作匹配检索,查不到则出错停机,否则取查到的一个进程的参量条件的闭包作"新条件集",转(8).

(6)建新史段.

(7)作向前推理,记录事件的时间点,用事件的后置条件的闭包作"新条件集".

(8)用"新条件集"修改、补充 SC,用 SC 建新切片.

(9)检查当前启动着的进程,若其中有其参量条件已经有不成立的,则终止此进程;检查活进程集,若其中有其参量条件已经有不成立的,则从AP中删除.转(2).

4 结　语

本文是文献[1]的继续,合在一起给出了关于开放型物理系统定性推理的描述手段QUIOS的完整说明.我们扼要阐述了QUIOS摆脱把事件仅当作进程运行结果的旧有观念,引进了随机和独立出现的事件概念,并反过来通过事件影响进程,这样就能自然地刻画非封闭的系统.一个关于用QUIOS描述的系统的推理算法已经给出.QUIOS能够描述和推断突发事件对系统行为的影响,其实质是系统状态间的非正常转换.反向推理在QUIOS下是自然的.还值得说明的是,在事件约束进程的机制下,对抽象时间点的偏序关系作各种调整,可作各种不同情况下的定性推理实验.

有了一种描述方法,就有实现的问题以及用它来自动地定性解决现实世界的各类实际大问题,这也就是De Kleer等人所阐述的定性推理研究的新潮流,在这方面QUIOS尚有许多工作要做.

致谢　与陆汝钤同志讨论总得益匪浅,笔者深为感谢.

参考文献

1　韦梓楚.关于开放型物理系统的QP理论.软件学报,1997,8(7):511～518.

2　Forbus K D. Qualitative process theory. Artificial Intelligence, 1984,24:84～168.

附录　文中用到的一些视图

```
〈Individual View〉 Object (a)
  〈Quantity Conditions〉
    a H place: vec(real:3)
    a H speed: vec(real:3)
    a H high: real
    a H volumn: real
    a H gr:Gravitation/*受重力*/
    applied_to(gr) = a
    a H netforce: Linear_force
      /*所受总合力*/
    applied_to(netforce) = a
    ..................
  〈Relations〉
    for i=1 to 3 do
    D[place(a)[i]]∝speed(a)[i]
  〈End〉 Object
```

```
〈Individual View〉 Rocket(a)
  〈Individuals〉
    a: Object
  〈Quantity Conditions〉
    a H long: real
    a H fire: boo /*是否点火*/
    a H drivforce:Linear_force /*火箭的推力*/
    applied_to(drivforce) = a
    a H direction: vec(real:2) /*由水平角和垂角定方向*/
  〈Relations〉
    [1]_intensity(lnr_intensity(drivforce(a)))和
    [2]_intensity(lnr_intensity(drivforce(a)))
    由value(lnr_intensity(drivforce(a)))与direction(a)估出
    [3]_intensity(lnr_intensity(drivforce(a)))由
    value(lnr_intensity(drivforce(a)))与direction(a)[2]估出
    〈End〉
```

```
〈Process View〉 homogen_motion(a,f)
  superclass = acceleration
  〈Quantity Conditions〉
  for i=1 to 3 do
    [i] intensity(lar_intensity(f))=ZERO
  〈End〉 homogen_motion
```

```
〈Individual View〉 System(s)
  〈Individuals〉 rocket: Rocket
  〈Quantity Conditions〉      /*系统初态*/
    place(rocket)=(ZERO,ZERO,ZERO) /*位于座标中心*/
    long(rocket) = l0
    direction(rocket)=(ZERO,ZERO) /*直立*/
```

```
〈Process View〉 rotate(y, rf, a)
  〈Individuals〉 y: Rocket
  〈Quantity Conditions〉
    y H rf: Circle_force /*受旋转力*/
    applied_to(rf) = y
    c_intensity(rf(y)) > ZERO
```

```
    speed(rocket)=(ZERO,ZERO,ZERO) /* 静候 */
    fire(rocket) = false          /* 未点火 */
    [1]_intensity(lnr_intensity(gr(rocket))) =
    [2]_intensity(lnr_intensity(gr(rocket))) = ZERO
    [3]_intensity(lnr_intensity(gr(rocket))) < ZERO
⟨End⟩                             /* 重力垂直朝下 */
        *          *          *
⟨Event View⟩ redirect(t,y,hh,rf,qd,α,β)
⟨Individuals⟩ y: Rocket
⟨Preconditions⟩

    place(y)[3] = hh /* 达指定高度 */
    qd > ZERO /* 切向受力强度非零 */
⟨Postconditions⟩
    y H rf: Circle_force
    applied_to(rf) = y
    c_intensity(rf(y))= qd
    axle(rf(y)) = (α,β)
    /* 在 t 时刻受旋转力,强度为 qd,转 */
    /* 轴的水平与垂直方向角为 α,β */
⟨End⟩ redirect
```

```
        axle(rf(y))=(α,π/2) /* 转轴方向,在水平面 */
⟨Influence⟩
        direction(y)[1] = sub(axle(rf(y))[1], π/2)
        D[ direction(y)[2] ] ∝ c_intensity(rf(y))
        ∀ θ⟩ ZERO → /* 可划分历史片断 */
          (∃ Δ) ((|Δ| ∝ c_intensity(rf(y)) ∧
          (Δt⟨Δ → direction(y)[2] ⟨θ) ∧
          (Δt=Δ → direction(y)[2] =θ) ∧
          (Δt⟩Δ → direction(y)[2] ⟩θ) )
          Call_event combine(netforce(y),drivforce
          f(y),gr(y))
        ⟨End⟩ rotate
```

```
        *          *          *
⟨Event View⟩ outfire(t, y, hh,θ)
  ⟨Individuals⟩ y: Rocket
  ⟨Preconditions⟩
   place(y)[3] ⟩ hh /* 超一定高度 */
   direction(y)[2]=θ /* 达一定偏角 */
⟨Postconditions⟩
   fire(y) = False /* 熄火 */
⟨End⟩ outfire
```

```
        *          *          *
⟨Event View⟩ explode(t, y,θ)
  ⟨Individuals⟩ y: Rocket
  ⟨Preconditions⟩
   fire(y) = True
  direction(y)[2] =θ
  ⟨Postconditions⟩y 爆炸的描述
  如 place(y)=(?,?,ZERO)
  ⟨End⟩ explode
```

THE MECHANISM QUIOS AND QUALITATIVE REASONING

WEI Zichu

(*Institute of Mathematics The Chinese Academy of Sciences Beijing* 100080)

Abstract This paper was focused to explore a qualitative reasoning method of open physical systems. This paper discussed the properties of events and processes in QUIOS, and the way to express their views. The mechanism of qualitative reasoning in QUIOS is shown with an example. A reasoning algorithm of QUIOS has also been outlined.

Key words Qualitative reasoning, open systems, events, processes, views, reasoning algorithm.

Class number TP18