摘 要

时段演算是区间时序逻辑的一个扩充，是由周巢尘， C.A.R. Hoare 和 A.P. Ravn 提出的。时段演算可以用来描述计算系统的实时需求，并可以用来验证计算系统的实时性质。在这方面人们已经做了大量工作，证明它是非常成功的。

期限驱动调度算法是 Liu 和 Layland 提出的一种动态调度策略。它假设若干任务周期性地向同一处理器请求执行时间，算法根据每个任务当前请求的期限动态地分配优先级，当前请求较急的任务将获得较高优先级，否则将获得较低优先级。在任意时刻，只有优先级最高且当前请求尚未完成的任务才能占有处理器。作为时段演算的一个应用，我们将用时段演算来形式描述这个算法，并证明 Liu 和 Layland 给出的关于该算法能行的充要条件。

本文的主要工作是研究如何用时段演算来刻画程序的实时行为。为了处理局部变量的声明，我们必须引进关于程序变量的量词。而在实时程序设计里，所有程序变量均看成时间域上的函数，因此，建立高阶时段演算是必要的。本文建立了高阶时段演算理论，包括它的语法，语义和证明系统。在假设所有程序变量均有穷可变的条件下，我们证明它在抽象时间域上是完备的。我们也将用高阶时段演算去验证一些程序的实时性质。例如，我们将说明在高阶时段演算里可以定义超稠密切割算子；我们将给出局部变量声明的形式描述；我们将证明程序顺序复合操作 “;” 具有单位元且满足结合律，以及可以将程序的实时语义分解成两部分：实时部分和非实时部分，从而可以将程序的实时语义看成是它的与时间无关语义的保守扩充。

关键词: 时段演算 高阶逻辑 实时系统 超稠密计算 完备性

Abstract

Duration Calculus (DC) is an extension of interval temproal logic which is proposed by Zhou Chaochen, C.A.R. Hoare and A.P. Ravn, and can specify the real-time requirements of computing systems and prove their real-time properties. DC has been proved to be a powerful logical frame by lots of case studies of it.

DDS (stands for Deadline Driven Scheduler) is about scheduling multiple tasks on a single processor dynamically which is proposed by Liu and Layland. It assumes that all tasks raise periodic requests for processor time, and priorities are dynamically assigned to tasks according to the deadlines of their current requests. A task will be assigned the highest priority if the deadline of its current request is the nearest, and will be assigned the lowest priority if the deadline of its current request is the furthest. At any instant, the task with the highest priority and yet unfulfilled request will occupy the processor. As an application, we will formally descripe DDS and prove the sufficient and necessary condition about DDS given by Liu and Layland in terms of DC.

The main results of this paper cover how to describe the real-time behaviours of programs in terms of duration calculus. In order to deal with hiding, e.g. local variable declaration and so on, quantifications over program variables are inevitable. So a higher-order duration calculus is established in this paper including its syntax, semantics and proof system. Its completeness on abstract domains will be proved under the assumption that all program variables have finite variability. We will explain how to apply the higher-order duration calculus to define semantics for real-time programs.

Keywords: duration calculus higher-order logic real-time system super-dense computation completeness

Contents

第 一 章　　引 言

1.1 实时系统

"实时 " (real time) 在不同的环境里具有不同的含义。 IEEE 的电子电器标准术语辞典 [24] 对它的定义如下：

> **real time: (software)** (A) Pertaining to the processing of data by a computer in connection with another process outside the computer according to the time requirements imposed by the outside process. This term is also used to describe operating systems in convensational mode and processes that can be influenced by human intervention while they are in progress. (B) Pertaining to the actual time during which a physical process transpires, for example, the performance of a computation during the actual time that the related physical process transpires, in order that results of the computation can be used in guiding the physical process.

我们这里用 "实时 " 来描述一类必须在指定时间内对外界产生的刺激或输入做出响应的计算系统。在实际生活中，实时系统 (Real-time Systems) 的例子很多，例如化工和核电的控制系统，银行数据库系统，导弹飞行控制系统，联网售票系统，机器人，铁路调度系统等等。对于这些系统，时间限制必须得到满足。如果时间响应提前或延迟，那么系统就会发生错误，而且这种错误有时会导致灾难性后果。例如核电站的控制系统，如果异常发生后，它的控制系统不能在响应时间内关闭核电站，那么将会招致灾难性后果。这种严格安全性系统 (Safety Critical Systems) 是很常见，同时又很重要的实时系统。

因此，实时系统的规范描述，精化和验证逐渐引起计算机界的广泛重视。在实时系统中，我们不仅要考虑系统的定性的 (qualitative) 性质，如系统行为的公平性，活性和同步等，而且还要考虑关于实时的定量的 (quantitative) 性质。我们首先需要考虑的是如何准确描述系统的需求，然后在有了需求规范后，我们怎样设计实现它，以及证明所设计的实现相对于需求是正确的。

, 对于严格安全性系统，我们最好使用形式化的方法来开发它。因为非形式化方法很难准确地描述系统的需求，从而也很难保证需求实现的正确性。这里我们用著名的达林顿例子 (Darlington Case) 来说明这一点。几年前，在加拿大安大略省达林顿建了一个核电站。当电站建成之后，政府部门因为电站无法保证其运行的安全性，而不允许此电站投入运行。其中的主交互控制系统是用来调节核反应堆的核反应的大小和监视电站的运行情况的。一旦系统出现异常，立刻就有两个相互独立的安全关闭系统关掉该控制系统。关闭系统负责收集

和显示系统参数，并判断核控制系统是否正常运行。两个安全关闭系统是运行在两台独立的计算机上，分别采用不同的传感器，不同的关闭机制。这两套安全关闭系统软件也是由两个不同的开发小组使用不同的程序设计语言开发的。然而，在由专家小组对其软件正确性和可靠性进行检查时，他们认为系统的可靠性是令人怀疑和担心的。他们发现原来关闭系统的控制模型是相对比较容易检查的，但计算机系统却变得相当复杂，软件设计也是难以理解，并且实现算法与原来模拟算法有着不同的结构。在该系统中，有一个用自然语言描述的需求：如果水位在 4 秒钟内持续保持 100 米，则水泵必须关闭。

我们可以在数学中有以下四种理解：

平均值模型：

$$\int_t^{t+4} WL(t)dt/4 > 100$$

中间值模型：

$$(max_{[t,t+4]}(WL(t)) + min_{[t,t+4]}(WL(t))/2 > 100$$

均方根模型：

$$\sqrt{\int_t^{t+4} WL(t)dt/4} > 100$$

最小值模型：

$$min_{[t,t+4]}(WL(t)) > 100$$

其中 $WL(t)$ 是水泵中水位随时间变化的函数。

两个独立的软件都是按最小值模型理解和实现的。不幸地是这种理解是非常危险的。因为有一种极端的情况是在 4 秒钟内只有很少一点时间，水位低于 100 米而绝大部分时间内水位都超过 100 米。然而如按最小值模型理解，则水泵不被关闭，并且还将继续往水泵中加水，这样系统将处于一种非常危险的状态。

总之，对于实时系统的开发来说，形式化方法是最有效和最可靠的方法。形式规范语言能够准确描述实时系统的需求，并且用形式化方法来验证实现满足严格安全性需求是可行的，而且也是严格的。这方面人们已经做了许多尝试，例如 [67, 76, 22, 68, 41, 63, 84, 66, 80, 81]。我们的理想是建立一个逻辑框架，它既可以作为规范语言，又可以定义程序设计语言。当它作为规范语言时，它能够准确，清晰地描述实时系统的需求；当用它来定义程序设计语言时，它能够准确说明实时程序的语义。并且它也提供了一个完备的逻辑推理系统，从而我们可以在统一的框架内既可以准确描述实时系统的需求，又可以给出它的实现，并可以严格地证明它的实现是满足它的需求的。

关于如何形式地开发实时系统，人们已经提出了许多方法。其中有些是基于代数的方法，如带时间的进程代数 ATP (Algebra of Timed Processes) [58, 59]，带时间的 CCS [85]，带时间的 CSP [69, 70] 等；有些是基于逻辑的方法。处理实时的逻辑称为时序逻辑，它

们可以分为两大类：一类是基于点的，例如 [65, 50, 77, 64, 19, 7]; 另一类是基于区间的，例如 [31, 2, 3, 53, 21, 13] 。还有一些是基于自动机的方法，例如 [4, 6, 5] 。

在这些方法中，时间有两种表示方式：显式 (explicit) 和隐式 (implicit) 。区间时序逻辑采用的是隐式时间。时段演算作为区间时序逻辑的一种扩充，用的也是隐式时间。

1.2 区间逻辑

1.2.1 区间时序逻辑

区间时序逻辑 (Interval Temporal Logic (下简称 ITL)) [31] 是用来描述实时系统需求并能够推导实时系统性质的一种模态逻辑。在 ITL 中，所有区间都是有穷长度的。不同于线性时序逻辑将公式在表示瞬间情况的状态上解释， ITL 是将公式在时间区间上进行解释。 ITL 包含唯一一个称为切割算子的模态操作 (记作 "$\frown$")，该操作被解释成将一个区间 "切割 " 成两部分：公式 $\phi\frown\psi$ 在解释 $\mathcal{J}$ 下在区间 $[b,e]$ 上为真当且仅当存在 $m\in[b,e]$ 使得 ϕ 在解释 $\mathcal{J}$ 下在 $[b,m]$ 上为真且 ψ 在解释 $\mathcal{J}$ 下在 $[m,e]$ 上为真。

1.2.2 时段演算

时段演算是区间时序逻辑 [31] 的一个扩充。在时段演算中，时序变量具有下面特殊结构：

$$\int S$$

其中 S 称为状态表达式，是由状态变量通过布尔运算符 $\neg,\vee,\ldots$ 连接而成的表达式。状态变量被解释为时间域上的布尔函数 $(\mathsf{R}\to\{0,1\})$，其中 R 是实数集合，我们用它来表示连续时间域。

$\int S$ 被解释为区间上的实函数。即给定一个状态表达式 S，亦为一个布尔函数 S，和区间 $[b,e]$ $(e\geq b)$，则 $\int S$ 在该区间上的值为：

$$\int_b^e \mathsf{S}(t)\,dt$$

其中 $\mathsf{S}(t)\in\{0,1\}$ 。使用此定义，可以验证任意一个区间的长度等于 $\int 1$ 在该区间上的值，其中 1 是一个被解释为处处等于常数 "1" 的状态。我们将用 ℓ 来表示区间长度。因此

$$\ell=\int 1$$

一个状态 S 在一个非点区间几乎处处为真可以表示为

$$\int S=\ell\ \wedge\ \ell>0 \quad (\text{简记为：}\ \llbracket S\rrbracket)$$

下面我们用 $\llbracket\ \rrbracket$ 表示 $(\ell=0)$ 。

时段演算引进了切割算子 [31], 记作 $\frown$ 。给定两个公式 ϕ 和 ψ 及一个解释 $\mathcal{I}$，公式

$$\phi\frown\psi$$

在解释 $\mathcal{I}$ 下在区间 $[b,e]$ 上为真当且仅当存在一个 m $(b \le m \le e)$ 使得 ϕ 在解释 $\mathcal{I}$ 下在 $[b,m]$ 上为真且 ψ 在解释 $\mathcal{I}$ 下在 $[m,e]$ 上为真。使用切割算子，我们可以定义一些其他模态算子。我们用 $\Diamond$ 表示存在某个子区间，$\Box$ 表示所有子区间，则

$$\begin{aligned} \Diamond\phi &\ \widehat{=}\ true^\frown\phi^\frown true \\ \Box\phi &\ \widehat{=}\ \neg\Diamond\neg\phi \end{aligned}$$

计算机内有时钟，计算系统的状态有一种称为有穷可变的稳定性。因此在时段演算里，我们假设所有状态具有有穷可变性，即在任意区间内，状态值的改变仅能有穷多次。换句话说，对任意状态 S (即为时间域上的一个布尔函数：$\mathsf{R} \to \{0,1\}$)，其有穷可变性可表示为：

$$\Box((\llbracket\ \rrbracket \vee \llbracket S\rrbracket^\frown true \vee \llbracket\neg S\rrbracket^\frown true) \wedge (\llbracket\ \rrbracket \vee true^\frown\llbracket S\rrbracket \vee true^\frown\llbracket\neg S\rrbracket))$$

也就是说，一个状态在任意非点区间总是在开始和结束的某段时间内要么几乎处处为真要么几乎处处为假。因此在时段演算里，我们不考虑一个状态在某个区间的一点上无穷摆动的情况，即发散情况 [38] 。

1.3 期限驱动调度算法

期限驱动调度算法 (Deadline Driven Scheduler (简称 DDS)) 是 Liu 和 Layland 提出的用来调度多个任务共享同一处理器的一种策略 [48] 。该算法假设所有任务周期地向处理器申请处理器时间。算法根据任务的当前请求的期限动态分配给它们优先级。如果一个任务当前的请求的期限较急，那么它可以分配到较高的优先级；否则，分配给它较低的优先级。在任意时刻，只有优先级高且当前周期的任务没有完成的进程才能占有处理器。因此，使用 DDS，通过根据当前任务的请求的期限动态地分配优先级的方法，可以充分利用处理器资源。

[48] 给出用 DDS 来调度多个任务能行的充要条件。该条件可以表述为下述定理：

定理 (Liu/Layland) *给定 m 个任务，期限驱动调度算法能行的充要条件是：*

$$(C_1/T_1) + \ldots + (C_m/T_m) \le 1 \quad (0 < C_i < T_i)$$

其中 C_i 是任务 p_i 申请的处理器时间，T_i 是它的请求周期 (假设 T_i $i = 1, \ldots, m$ 是正整数) 。

[48] 给出了该定理的一个非形式的证明。该定理的必要性是显然的，但充分性理解起来就非常困难。 [86] 用时段演算给出了 DDS 一个形式描述和一个形式证明，这是文献中关于 DDS 的第一个形式证明。该证明主要运用时段演算中的归纳规则，这样使得证明缺乏直观，从而难于理解。

这里我们使用 DC 给出 DDS 另外一个较直观的形式证明。该证明可参见 [56] 。我们通过三步来证明它的充分性：首先证明如果处理器在一个区间里没有空闲那么在这个区间上这个任务集合是可调度的；其次我们证明给定一个时间区间，如果在该时间区间的任意真前缀上这个任务集合都是可调度的，但是却在该时间区间上不可调度，那么处理器在该时间区间上没有空闲时间；最后，我们反设在某个时间区间上这个任务集合不可调度，那么存在该时间区间的一个前缀区间使得在该前缀区间的任意真前缀区间上这个任务集合都是可调度的，但是在该前缀区间上不不可调度。从而与第一，二步的结论矛盾。因此充分性得证。

利用这一实例，我们说明时段演算的表达和推理能力。

1.4 高阶时段演算及其完备性

周巢尘, C.A.R. Hoare 和 A.P. Ravn 等人为了刻画实时计算系统的需求并能够验证它的性质, 在一阶区间时序逻辑 [31] 的基础上进行扩充, 提出了时段演算 [13] 。自从时段演算理论建立以来, 它已经广泛应用于描述计算系统的实时需求, 其中包括嵌入式系统, 例如 [67, 76, 22, 68, 41, 63, 84, 66, 80, 81] 。另一方面, 时段演算也被用来定义程序设计语言的实时语义 [11, 46, 37, 73, 10, 15] 。人们正试图建立基于时段演算的程序设计方法。但是到目前为止, 所有这方面的工作都没有考虑局部变量和内部通道。原因在于还没有人研究如何量词化时序变量。因为在实时程序设计中, 程序变量被解释成时间域上的函数, 因此如果在时段演算中引进关于程序变量的量词, 我们必须建立高阶时段演算。一般而言, 高阶逻辑比一阶逻辑复杂的多。例如, 人们已经建立了完备的一阶谓词演算的证明系统, 但至今仍无人给出一个完备的二阶谓词演算的证明系统。事实上, 根据 Gödel 不完备性定理, 这也是不可能的。本文的主要工作就是研究高阶时段演算 (下简称 HDC) 。我们首先建立一个 HDC 的证明系统; 然后我们证明, 如果假设所有程序变量均有穷可变, 那么我们的证明系统在抽象域上是完备的; 最后, 我们将说明在 HDC 中如何描述局部变量声明, 并用 HDC 来证明一些程序的实时性质。同时, [45] 指出在 HDC 中可以定义超稠密切割算子。超稠密切割算子是在 [10] 中提出, 用于处理超稠密计算 [49] 。本文的主要结果已出现在 [9, 57] 。

为了描述实时程序的语义, 在 HDC 里, 我们引进程序变量 V, V_i ,$i = 1, 2, \ldots$, 并把它们看成时序变量。由程序变量, 全局变量和常量通过实算术运算符构成的表达式称为状态项。谓词作用到状态项构成的表达式是状态 $(\mathsf{R} \to \{0, 1\})$, 由状态通过逻辑连接词构成的表达式仍旧为状态。因此, 在 HDC 里, 状态具有内部结构。例如, $(V = x)$ 和 $(V > 1)$ 均可以作为状态。

在程序设计中, 值的传递涉及到程序变量在过去和将来时刻的值, 从上个语句接受初值, 并将计算的结果传递给下一个语句。因为切割算子是一个内敛算子, 它不能刻划一个状态在当前区间外面的性质, 因此我们在这里引进了左值函数 $\leftarrow$ 和右值函数 $\rightarrow$ 。左值函数 $\leftarrow$ 和右值函数 $\rightarrow$ 是以所有状态项的集合为定义域, 以状态项的的值为值域。例如 $(\overleftarrow{V} = 2)$ 表示在当前区间的一个左邻区间上, V 的值是 2 。因此它有下面的性质:

$$(\ell > 0)^\frown(\overleftarrow{V} = 2) \;\Rightarrow\; (true^\frown[\![V = 2]\!])^\frown(\overleftarrow{V} = 2)$$

对称地, $(\overrightarrow{V} = 2)$ 表示在当前区间的一个右邻区间上, V 的值是 2 。因此

$$(\overrightarrow{V} = 2)^\frown(\ell > 0) \;\Rightarrow\; (\overrightarrow{V} = 2)^\frown([\![V = 2]\!]^\frown true)$$

为了公理化左值函数 $\leftarrow$ 和右值函数 $\rightarrow$, 我们需要邻接规则。即:

如果 $(\ell = a)^\frown\Psi^\frown(\ell = b) \Rightarrow (\ell = a)^\frown\Upsilon^\frown(\ell = b)$, 那么 $\Psi \Rightarrow \Upsilon$。 $(a, b \geq 0)$

如果我们仅考虑程序变量 V, 赋值语句 $(V := V + 2)$ 的语义可描述为从前一个语句接受 V 的初值并加 2, 然后将终值传递给下一个语句。因此我们可以用 HDC 的公式表示为:

$$\exists x.\, (\overleftarrow{V} = x) \;\wedge\; [\![V = x + 2]\!] \;\wedge\; (\overrightarrow{V} = x + 2)$$

其中我们假设赋值语句 $(V := V + 2)$ 消耗一定时间, 且状态 $(V = x + 2)$ 在这段时间内是稳定的。

在区间时序逻辑 [21] 和时段演算及其变种 [13, 17, 18, 87] 中，采用了 [1, 25] 中的方法和术语，把符号分为两大类 –**刚性符号** 和 **柔性符号** 。刚性符号用来表示全局固定的实体，它们在所有区间上的解释都是相同的。相反地，柔性符号表示的实体在不同的区间上可能不同。这两类符号在 HDC 中的区别与它们在一阶时序逻辑中 [1, 25] 的区别是相同的。

区间时序逻辑和时段演算的完备性不仅依赖于时间域的选择，而且依赖于量词化那类变量。在实际应用中，我们需要选择实数作为时间。但是，如果这样，根据 Gödel 不完备性定理，我们不可能给出关于它们的完备的证明系统。因此，若我们选择实数作为时间，那么我们仅能得到这些系统的相对完备性。例如， [34] 已经证明若把所有关于实数和区间时序逻辑的永真公式当作时段演算的公理，那么时段演算是完备的。如果我们仅考虑量词化全局变量，那么区间时序逻辑在抽象时间域上是完备的 [21] 。 [29] 亦证明采用无穷规则，时段演算在抽象时间域上也是完备的。因为我们把程序变量解释成从时间域到时段域的函数，因此若我们一旦引进关于程序变量的量词，而且它的量词的作用域为从时间域到时段域的全体函数，那么我们不可能给出 HDC 的完备的证明系统。原因在于，一旦我们把程序变量的量词作用域解释为从时间域到时段域的全体函数，那么 HDC 就具有了二阶算术的表达能力，这样的系统不可能是完备的。因此，为了给出一个完备的 HDC 的证明系统，对程序变量加些限制是必要的。因而我们假设所有程序变量均具有有穷可变性，从而由程序变量作参数产生的状态也具有有穷可变性。这与在时段演算中假设所有状态变量都具有有穷可变性是一致的。如果这样，我们可以通过将 HDC 翻译到一阶区间时序逻辑中去，从而证明它在抽象域上的完备性。

1.5 实时程序的语义

在本文中，我们也将研究如何用 HDC 来描述实时程序的语义，包括如何用它来刻画超稠密计算和局部变量等。

1.5.1 超稠密计算

在计算系统中，许多程序是顺序执行的，且执行它的所有语句所消耗的时间用其环境中大的时间粒度很难测量，因为太小以至可以忽略不计。由一系列认为不消耗时间的操作构成的计算称为超稠密计算 [49] 。

在超稠密计算的假设下， $[\![V := V + 2]\!]$ 变成

$$\exists x.\ (\overleftarrow{V} = x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V} = x + 2)$$

因为状态 $(V = x + 2)$ 立即消失了。

根据切割算子的含义，描述程序语句间顺序复合操作 ";" 的语义自然应使用切割算子 "$\frown$" 。因此，两个连续赋值语句

$$V =: V + 2; V =: V + 1$$

的语义可表示成

$$[\![V =: V + 2]\!] \frown [\![V := V + 1]\!]$$

不幸地是，切割算子在点区间上将退化成逻辑联接词 “∧”, 因此上述公式变成

$$[\![V =: V+2]\!] \wedge [\![V := V+1]\!]$$

也就是,

$$(\exists x.\ (\overleftarrow{V} = x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V} = x+2))\ \wedge\ (\exists x.\ (\overleftarrow{V} = x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V} = x+1))$$

而这是一个矛盾。另一方面，根据合并赋值语句规则 [44], 上述两个赋值语句应该等价于单个赋值语句 $(V := V+3)$ 。也就是,

$$\exists x.\ (\overleftarrow{V} = x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V} = x+3)$$

因此，为了刻画程序语句顺序复合操作 $(;)$ 的语义， [10] 引进了超稠密切割算子 (记作 $\bullet$) 。使用超稠密切割算子 $\bullet$, 我们可以描述两个连续赋值语句

$$V =: V+2; V =: V+1$$

的语义如下:

$$[\![V =: V+2]\!] \bullet [\![V := V+1]\!]$$

即

$$(\exists x.\ (\overleftarrow{V} = x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V} = x+2)) \bullet (\exists x.\ (\overleftarrow{V} = x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V} = x+1)$$

对于任意给定 V 的一个初值 x, 不可见的中间状态 $(V = x+2)$ 可以将上述公式中 $\bullet$ 两边的子公式联系起来。上述公式可以简化成

$$\exists x.\ (\overleftarrow{V} = x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V} = x+3)$$

这正是赋值语句 $(V := V+3)$ 的语义。

本文中将说明，超稠密切割算子是可由二阶量词定义的。

1.5.2 局部变量

使程序变量 V 局部化在语义上相当于存在程序变量 V 。若我们用 $[\![\mathcal{P}]\!]$ 表示程序 $\mathcal{P}$ 的语义，那么程序 $(\mathsf{begin}\ V{:}\mathcal{P}\ \mathsf{end})$ 的语义可表述为

$$\exists V.\ [\![\mathcal{P}]\!]$$

其中 V 声明为程序 $\mathcal{P}$ 中的局部变量。例如，一个延迟 k 个时间单位的过程可以用程序段 $(\mathsf{begin}\ V{:}\mathcal{Q}(V,k)\ \mathsf{end})$ 来实现，其中 $\mathcal{Q}(V,k)$ 定义为

$$V := 0;\ \mathsf{while}\ V < k\ \mathsf{do}\ (\mathsf{tick};\ V := V+1)$$

在上述程序段中， k 是一个取值为正整数的参数， tick 是一个执行延迟一个时间单位的操作。直观上，我们期望

$$\begin{array}{rcl} [\![\mathsf{begin}\ V{:}\mathcal{Q}(V,k)\ \mathsf{end}]\!] & \widehat{=} & \exists V.\ [\![V := 0;\ \mathsf{while}\ V < k\ \mathsf{do}\ (\mathsf{tick}; V := V+1)]\!] \\ & \Leftrightarrow & (\ell = k) \end{array}$$

本文中提出的一些方法同样可以应用到处理通讯的实时行为，例如，我们可以用处理局部变量的方法来处理内部通道等。

1.6 论文结构

本文的剩下部分组织如下：第二章将简要介绍一下本文所依赖的理论基础 — 区间时序逻辑和时段演算，并给出一些在后面用到的定理；第三章将用期限驱动调度算法为例说明如何用时段演算来形式描述实时系统并证明实时系统的性质；第四章将建立高阶时段演算理论，包括它的语法，语义和证明系统；第五章将证明高阶时段演算在抽象时间域上的完备性；第六章将给出高阶时段演算的一些应用。我们首先给出在高阶时段演算里如何定义超稠密切割算子，然后给出局部变量声明的形式描述，最后我们证明一些实时程序的性质；第七章将首先对本文的工作做一个总结，然后介绍一下与本文有关的相关工作，最后讨论一下今后的进一步工作。

第 二 章 基 础 理 论

本章我们介绍一下本文所依赖的理论背景。我们首先介绍一下区间时序逻辑 (Interval Temporal Logic (简称 ITL)), 然后介绍一下时段演算 (Duration Calculus (简称 DC)) 。

2.1 区间时序逻辑

本节我们将介绍一下区间时序逻辑的语法, 语义和它的证明系统。我们这里介绍的区间时序逻辑主要依据 [21, 20] 的结果。

2.1.1 区间时序逻辑的语法

ITL 的字母表包括下面几种符号:

全局变量 我们用 $x, y, z, \ldots$, 表示全局变量。所有全局变量构成的集合记作 $GVar$ 。因为这些变量的语义解释不依赖于时间和时间区间, 所以称它们为全局变量。

时序变量 我们用 $v, v_1, v2, \ldots$ 表示时序变量。所有时序变量构成的集合记作 $TVar$ 。时序变量被解释成时间区间上的函数。

函数符号 我们用 $f^n, f_1^m, \ldots$ 表示全局函数符号, 其中, $n, m \geq 0$ 表示它们的元数。当 $n = 0$ 是, f 称为常量。我们用 $FSymb$ 表示所有函数符号的集合。 f^n 将被解释成时段域上的 n - 元函数, 且与时间和时间区间无关。

谓词符号 我们用 $R^n, R_1^m, \ldots$ 表示全局谓词符号, 其中, $n, m \geq 0$ 表示它们的元数。当 $n = 0$ 是, R 称为布尔常量。 **true** 和 **false** 是仅有的两个布尔常量。我们用 $RSymb$ 表示所有全局谓词符号的集合。 R^n 将被解释成时段上的 n - 元布尔函数, 且与时间和时间区间无关。

命题符号 我们用 $X, Y, \ldots$ 表示时序命题符号。所有时序变量的集合记作 $PLetter$ 。时序命题符号被解释成区间上的布尔函数。

长度变量 ℓ 表示区间的长度。

逻辑联结词 $\neg$ 和 $\vee$

量词符号 $\exists$

模态算子 切割算子 (chop) "$\frown$"

ITL 的项由下面抽象语法定义:

$$\theta \ ::= x \ \mid \ \ell \ \mid \ v \ \mid \ f_i^n(\theta_1, \ldots, \theta_n)$$

其中 f 是 n - 元函数。

ITL 的原子公式可以定义如下:

$$Atom \ ::= \ X \ \mid \ \mathbf{true} \ \mid \ R(\theta_1, \ldots, \theta_n)$$

其中 R 是 n - 元谓词。

它的公式可以由下述语法定义:

$$\phi \ ::= \ Atom \ \mid \neg\phi \ \mid \ \phi \vee \psi \ \mid \ (\phi \frown \psi) \ \mid \ \exists x.\phi$$

我们说 x 在 $\phi(x)$ 中相对于 θ 是 **自由的**, 如果 x 不在 $\exists y$ 和 $\forall y$ 的约束范围内出现, 其中 y 为任意在 θ 中出现的变量。

2.1.2 区间时序逻辑的语义

下面, 我们给出 ITL 在实数时间域上的语义, 即 $\mathbf{T}\text{ime} \ \widehat{=} \ \mathbf{R}$。它在抽象时间域上的语义我们在第四章中给出。

因为我们仅对实算术 (real arithmetics) 函数和关系感兴趣, 因此我们假设对每一个 n - 元函数符号 f_i^n 存在一个全函数 $\underline{f_i^n} \in \mathbf{R}^n \mapsto \mathbf{R}$ 与之对应; 对每一个 n - 元关系符 R_i^n 存在一个全关系函数 $\underline{R_i^n} \in \mathbf{R}^n \mapsto \{tt, ff\}$ 与之对应。特别地, 我们用 tt 和 ff 分别对应 **true** 和 **false**。

定义 2.1 $\mathbf{I}ntv \stackrel{\text{def}}{=} \{[b,e] \mid b, e \in \mathbf{T}ime, \ and \ \ b \leq e\}$ 其中, $[b,e] = \{t \mid t \in \mathbf{T}ime, and \ \ b \leq t \leq e\}$

全局变量, 时序变量和命题符号的语义由下面的解释给出。时序变量和命题符号都是依赖于时间区间的变量。

$$\mathcal{J} \in \begin{pmatrix} GVar \\ \bigcup \\ TVar \\ \bigcup \\ PLetter \end{pmatrix} \longmapsto \begin{pmatrix} \mathbf{R} \\ \bigcup \\ \mathbf{I}\text{ntv} \mapsto \mathbf{R} \\ \bigcup \\ \mathbf{I}\text{ntv} \mapsto \{tt, ff\} \end{pmatrix}$$

定义 2.2 $\mathcal{J}$ 和 $\mathcal{J}'$ 是两个解释。我们称它们是 x- 等价的当且仅当对任意不同于 x 的符号 z, $\mathcal{J}(z) = \mathcal{J}'(z)$。

项 θ 在解释 $\mathcal{J}$ 下的语义是类型为：

$$\mathcal{J}[\![\,\theta\,]\!] \in \mathbf{I}\mathrm{ntv} \mapsto \mathbf{R}$$

的函数，它可以根据项的结构递归定义如下：

$$\begin{array}{lcl}
\mathcal{J}[\![\,x\,]\!]([b,e]) & = & \mathcal{J}(x) \\
\mathcal{J}[\![\,\ell\,]\!]([b,e]) & = & e-b \\
\mathcal{J}[\![\,v\,]\!]([b,e]) & = & \mathcal{J}(v)([b,e]) \\
\mathcal{J}[\![\,f^n(\theta_1,\ldots,\theta_n)\,]\!]([b,e]) & = & \underline{f}^n(c_1,\ldots,c_n)
\end{array}$$

其中，$c_i = \mathcal{J}[\![\,\theta_i\,]\!]([b,e])$, 对所有 $1 \le i \le n$ 。

公式 ϕ 在解释 $\mathcal{J}$ 下的语义是类型为：

$$\mathcal{J}[\![\,\phi\,]\!] \in \mathbf{I}\mathrm{ntv} \mapsto \{tt, ff\}$$

的函数，它可以根据公式的结构递归定义如下：

1. $\mathcal{J}, [b,e] \models_{il} X$
   当且仅当 $\mathcal{J}[\![\,X\,]\!]([b,e]) = tt$
2. $\mathcal{J}, [b,e] \models_{il} R^n(\theta_1,\ldots,\theta_n)$
   当且仅当 $\underline{R}^n(\mathcal{J}[\![\,\theta_1\,]\!]([b,e]),\ldots,\mathcal{J}[\![\,\theta_n\,]\!]([b,e])) = tt$
3. $\mathcal{J}, [b,e] \models_{il} \neg\phi$
   当且仅当 $\mathcal{J}, [b,e] \not\models_{il} \phi$
4. $\mathcal{J}, [b,e] \models_{il} \phi \vee \psi$
   当且仅当 $\mathcal{J}, [b,e] \models_{il} \phi$ 或者 $\mathcal{J}, [b,e] \models_{il} \psi$
5. $\mathcal{J}, [b,e] \models_{il} \phi^\frown\psi$
   当且仅当存在 $m \in [b,e]$ 使得 $\mathcal{J}, [b,m] \models_{il} \phi$ 且 $\mathcal{J}, [m,e] \models_{il} \psi$
6. $\mathcal{J}, [b,e] \models_{il} \exists x.\phi$
   当且仅当存在一个解释 $\mathcal{J}'$ 且 $\mathcal{J}'$ x- 等价于 $\mathcal{J}$, 使得 $\mathcal{J}', [b,e] \models_{il} \phi$

其中，

$$\begin{array}{lcl}
\mathcal{J}, [b,e] \models_{il} \phi & \widehat{=} & \mathcal{J}[\![\,\phi\,]\!]([b,e]) = tt \\
\mathcal{J}, [b,e] \not\models_{il} \phi & \widehat{=} & \mathcal{J}[\![\,\phi\,]\!]([b,e]) = ff
\end{array}$$

一个公式 ϕ 是 永真的, 记作 $\models_{il} \phi$, 当且仅当对任意解释 $\mathcal{J}$ 和任意区间 $[b,e]$ 都有 $\mathcal{J}, [b,e] \models_{il} \phi$ 。一个公式 ψ 是可满足的当且仅当存在一个解释 $\mathcal{J}$ 和一个区间 $[b,e]$ 使得 $\mathcal{J}, [b,e] \models_{il} \psi$ 。

为了后面叙述的方便，下面我们定义一些记号。

$\Diamond\phi \;\widehat{=}\; \mathbf{true}^\frown(\phi^\frown\mathbf{true})$ 读作：“存在一个子区间：ϕ”
$\Box\phi \;\widehat{=}\; \neg\Diamond(\neg\phi)$ 读作：“对所有子区间：ϕ”
$\Diamond_p\phi \;\widehat{=}\; \phi^\frown\mathbf{true}$ 读作：“存在一个前缀：ϕ”
$\Box_p\phi \;\widehat{=}\; \neg\Diamond_p(\neg\phi)$ 读作：“对所有的前缀：ϕ”
$\Diamond_b\phi \;\widehat{=}\; \phi^\frown\ell > 0$ 读作：“存在一个真前缀：ϕ”
$\Box_b\phi \;\widehat{=}\; \neg\Diamond_b(\neg\phi)$ 读作：“对所有的真前缀：ϕ”

文中出现的一些逻辑记号和传统的一样，在此不赘述。

2.1.3 区间时序逻辑的证明系统

ITL 的公理包括：

(IL1) $\ell \geq 0$

(IL2) $((\phi^\frown\psi) \wedge \neg(\phi^\frown\varphi)) \;\Rightarrow\; (\phi^\frown(\psi \wedge \neg\varphi))$
$((\phi^\frown\psi) \wedge \neg(\varphi^\frown\psi)) \;\Rightarrow\; ((\phi \wedge \neg\varphi)^\frown\psi))$

(IL3) $((\phi^\frown\psi)^\frown\varphi) \;\Leftrightarrow\; (\phi^\frown(\psi^\frown\varphi))$

(IL4) $(\phi^\frown\psi) \Rightarrow \phi$ 若 ϕ 是刚性的
$(\phi^\frown\psi) \Rightarrow \psi$ 若 ψ 是刚性的

(IL5) $((\exists x.\phi)^\frown\psi) \Rightarrow \exists x.\ \phi^\frown\psi$ 若 x 在 ψ 中不是自由出现
$(\phi^\frown\exists x.\psi) \Rightarrow \exists x.\ \phi^\frown\psi$ 若 x 在 ϕ 中不是自由出现

(IL6) $((\ell = x)^\frown\phi) \Rightarrow \neg((\ell = x)^\frown\neg\phi)$
$(\phi^\frown(\ell = x)) \Rightarrow \neg(\neg\phi^\frown(\ell = x))$

(IL7) $(x \geq 0 \wedge y \geq 0) \Rightarrow ((\ell = x + y) \Leftrightarrow ((\ell = x)^\frown(\ell = y)))$

(IL8) $\phi \Rightarrow (\phi^\frown(\ell = 0))$
$\phi \Rightarrow ((\ell = 0)^\frown\phi)$

ITL 的推理规则包括：

(N)： 如果 ϕ 那么 $\neg((\neg\phi)^\frown\psi)$
如果 ϕ 那么 $\neg(\psi^\frown(\neg\phi))$

(M)： 如果 $\phi \Rightarrow \psi$ 那么 $(\phi^\frown\varphi) \Rightarrow (\psi^\frown\varphi)$
如果 $\phi \Rightarrow \psi$ 那么 $(\varphi^\frown\phi) \Rightarrow (\varphi^\frown\psi)$

ITL 的证明系统也包括命题逻辑，谓词逻辑和实算术的公理和推理规则。例如

$(\mathrm{G_x})$： 如果 ϕ 那么$\forall x.\phi$

然而，为了确保下面的公理在我们的模型中是永真的，我们须给出一个附加条件。

$(\mathrm{Q_x}) : (\forall x.\phi(x)) \;\Rightarrow\; \phi(\theta)$ 要么 x 在 $\phi(x)$ 中相对于 θ 是自由的，且 θ 是刚性的；要么 x 在 $\phi(x)$ 中相对于 θ 是自由的，且 $\phi(x)$ 中不含有切割算子。

上述的附加条件是必要的，原因在于在 ITL 和 DC 中，一个柔性符号在不同的区间上所指的对象可能不同。例如根据 IL7, 我们有

$$\vdash_{il} \forall x.(\ell = x)^\frown(\ell = x) \Rightarrow (\ell = 2x)$$

虽然 x 在上式中相对于 ℓ 是自由的，但是

$$\nvdash (\ell = \ell)^\frown(\ell = \ell) \Rightarrow (\ell = 2\ell)$$

因为 ℓ 在不同的区间上所指的对象不同。我们在这儿就不逐一列出实算术中用来处理等词，加运算等的公理和推理规则，可参见 [36, 21] 。

定理 2.1 区间时序逻辑的证明系统是可靠的，即 $\vdash_{il} \phi$ 蕴涵 $\models_{il} \phi$ 。

证明: 我们仅需证明每条公理是永真的，而每条规则保持永真性即可。 □

讨论: [21] 证明了区间时序逻辑的证明系统在抽象时间域上是完备的，即时间域满足全序可交换群的公理。当然, 实数集合是满足全序可交换群的公理的。不幸地是, 根据 Gödel 不完备性定理，我们是不可能建立一个证明系统，使得它仅以实数作为模型。

2.1.4 区间时序逻辑的一些定理

我们在下面证明一些将在后面用到的区间时序逻辑的定理。

(ILT1) $\phi^\frown(\ell = 0) \Leftrightarrow \phi$
$(\ell = 0)^\frown\phi \Leftrightarrow \phi$

(ILT2) $\exists x.\ell = x$

(ILT3) $\ell = 0 \Rightarrow \Box(\ell = 0)$

(ILT4) $(\phi \vee \psi)^\frown\varphi \Leftrightarrow ((\phi^\frown\varphi) \vee (\psi^\frown\varphi))$

证明: ILT1 可以由 IL6 和 IL8 证明; ILT2 可以由 G_x 证明; 根据 $\Box$ 的定义, ILT2 和 IL7, 我们可以证明 ILT3; ILT4 可证明如下：

(1)	$\phi \Rightarrow (\phi \vee \psi)$	(PL)
(2)	$\phi^\frown\varphi \Rightarrow (\phi \vee \psi)^\frown\varphi$	(M, (1))
(3)	$\psi \Rightarrow (\phi \vee \psi)$	(PL)
(4)	$\psi^\frown\varphi \Rightarrow (\phi \vee \psi)^\frown\varphi$	(M, (3))
(5)	$(\phi^\frown\varphi) \vee (\psi^\frown\varphi) \Rightarrow (\phi \vee \psi)^\frown\varphi$	(PL, (2), (4))
(6)	$(\phi \vee \psi)^\frown\varphi \wedge \neg((\phi^\frown\varphi) \vee (\psi^\frown\varphi))$	
	$\Rightarrow ((\phi \vee \psi) \wedge (\neg\phi \wedge \neg\psi))^\frown\varphi$	(IL2)
	$\Rightarrow \mathbf{false}^\frown\varphi$	(PL)
	$\Rightarrow \mathbf{false}$	(IL4)
(7)	$(\phi \vee \psi)^\frown\varphi \Rightarrow ((\phi^\frown\varphi) \vee (\psi^\frown\varphi))$	(PL,(6))
(8)	$(\phi \vee \psi)^\frown\varphi \Leftrightarrow ((\phi^\frown\varphi) \vee (\psi^\frown\varphi))$	(PL, (5), (7))

其中 PL 代表所有谓词逻辑和实算术的公理和推导规则。下同。

(ILT5) $\phi \Rightarrow \boxdot\phi$ 若 ϕ 是刚性的

(ILT6) $\boxdot\phi \Rightarrow \phi$

(ILT7) $\boxdot\phi \Rightarrow \boxdot\boxdot\phi$

(ILT8)
$$(\boxdot\phi \wedge (\psi^\frown\varphi)) \Rightarrow ((\phi \wedge \psi)^\frown\varphi)$$
$$(\Box\phi \wedge (\psi^\frown\varphi)) \Rightarrow (\psi^\frown(\phi \wedge \varphi))$$

其中, $\boxdot \in \{\Box, \Box_p\}$ 。

证明: 我们仅证明 $\boxdot = \Box$ 的情况, 其它情况可以类似证明。由 $\Box$ 的定义和 IL4, ILT5 立证; ILT6 和 ILT7 是显然的。下面我们给出 ILT8 的第一部分的证明, 第二部分可以类似地证明。

(1)
$$\begin{aligned}
&\neg(\mathbf{true}^\frown(\neg\phi)^\frown\mathbf{true}) \wedge (\psi^\frown\varphi) \wedge \neg((\phi \wedge \psi)^\frown\varphi) \\
&\Rightarrow \neg(\mathbf{true}^\frown(\neg\phi)^\frown\mathbf{true}) \wedge ((\neg\phi \vee \neg\psi) \wedge \psi)^\frown\varphi && \text{(IL2)} \\
&\Rightarrow \neg(\mathbf{true}^\frown(\neg\phi)^\frown\mathbf{true}) \wedge ((\neg\phi \wedge \psi)^\frown\varphi \vee (\mathbf{true}^\frown\varphi)) && \text{(PL)} \\
&\Rightarrow \neg(\mathbf{true}^\frown(\neg\phi)^\frown\mathbf{true}) \wedge \mathbf{true}^\frown(\neg\phi)^\frown\mathbf{true} && \text{(IL8, M, PL)} \\
&\Rightarrow \mathbf{false} && \text{(PL)}
\end{aligned}$$

(2) $(\Box\phi \wedge (\psi^\frown\varphi)) \Rightarrow ((\phi \wedge \psi)^\frown\varphi)$ ((1))

(ILT9)
$$(\ell = x^\frown\phi_1 \wedge \ell = x^\frown\phi_2) \Rightarrow (\ell = x^\frown(\phi_1 \wedge \phi_2))$$
$$(\phi_1\widehat{}\,\ell = x \wedge \phi_2\widehat{}\,\ell = x) \Rightarrow ((\phi_1 \wedge \phi_2)^\frown\ell = x)$$

(ILT10)
$$\left(\begin{array}{c}(\phi_1 \wedge \ell = x)^\frown\psi_1 \\ \wedge(\phi_2 \wedge \ell = x)^\frown\psi_2\end{array}\right) \Rightarrow ((\phi_1 \wedge \phi_2 \wedge \ell = x)^\frown(\psi_1 \wedge \psi_2))$$
$$\left(\begin{array}{c}\phi_1\widehat{}\,(\psi_1 \wedge \ell = x) \\ \wedge\phi_2\widehat{}\,(\psi_2 \wedge \ell = x)\end{array}\right) \Rightarrow ((\phi_1 \wedge \phi_2)^\frown(\psi_1 \wedge \psi_2 \wedge \ell = x))$$

证明: 我们仅证明 ILT9 和 ILT10 的第一部分, 它们的第二部分可以类似地证明。

ILT9 的证明如下:

$$\begin{aligned}
&\ell = x^\frown\phi_1 \wedge \ell = x^\frown\phi_2 \\
&\Rightarrow \neg(\ell = x^\frown(\neg\phi_1)) \wedge \ell = x^\frown\phi_2 && \text{(IL6)} \\
&\Rightarrow \ell = x^\frown(\phi_1 \wedge \phi_2) && \text{(IL2)}
\end{aligned}$$

ILT10 的证明如下:

$$
\begin{array}{ll}
(1) & \left(\begin{array}{ll} & (\phi_1 \wedge \ell = x)^\frown(\psi_1 \wedge \psi_2) \wedge (\phi_2 \wedge \ell = x)^\frown\psi_2 \\ \wedge & \neg((\phi_1 \wedge \phi_2)^\frown(\psi_1 \wedge \psi_2)) \end{array}\right) \\
 & \Rightarrow \left(\begin{array}{ll} & ((\phi_1 \wedge \neg\phi_2 \wedge \ell = x)^\frown(\psi_1 \wedge \psi_2) \vee \mathbf{false}^\frown(\psi_1 \wedge \psi_2)) \\ \wedge & (\phi_2 \wedge \ell = x)^\frown\psi_2 \end{array}\right) \quad \text{(IL2)} \\
(2) & (\phi_1 \wedge \neg\phi_2 \wedge \ell = x)^\frown(\psi_1 \wedge \psi_2) \wedge (\phi_2 \wedge \ell = x)^\frown\psi_2 \\
 & \Rightarrow (\phi_2 \wedge \ell = x)^\frown\psi_2 \wedge (\neg\phi_2 \wedge \ell = x)^\frown\psi_2 \quad \text{(M)} \\
 & \Rightarrow \exists y.\left(\begin{array}{ll} & \ell = y \wedge (\phi_2 \wedge \ell = x)^\frown(\ell = y - x \wedge \psi_2) \\ \wedge & (\neg\phi_2 \wedge \ell = x)^\frown(\ell = y - x \wedge \psi_2) \end{array}\right) \quad \text{(ILT2)} \\
 & \Rightarrow \exists y.\phi_2^\frown(\ell = y - x) \wedge \neg\phi_2^\frown(\ell = y - x) \quad \text{(M)} \\
 & \Rightarrow \exists y.(\phi_2 \wedge \neg\phi_2)^\frown(\ell = y - x)\text{(ILT9)} \\
 & \Rightarrow \mathbf{false} \quad \text{(IL4)} \\
(3) & \mathbf{false}^\frown(\psi_1 \wedge \psi_2) \\
 & \Rightarrow \mathbf{false} \quad \text{(IL4)} \\
(4) & \left(\begin{array}{ll} & (\phi_1 \wedge \ell = x)^\frown(\psi_1 \wedge \neg\psi_2)) \wedge (\phi_2 \wedge \ell = x)^\frown\psi_2 \\ \wedge & \neg((\phi_1 \wedge \phi_2)^\frown(\psi_1 \wedge \psi_2)) \end{array}\right) \\
 & \Rightarrow \ell = x^\frown\psi_2 \wedge \ell = x^\frown\neg\psi_2 \quad \text{(M)} \\
 & \Rightarrow \ell = x^\frown(\psi_2 \wedge \neg\psi_2) \quad \text{(ILT9)} \\
 & \Rightarrow \mathbf{false} \quad \text{(IL4)} \\
(5) & \left(\begin{array}{ll} & (\phi_1 \wedge \ell = x)^\frown\psi_1 \wedge (\phi_2 \wedge \ell = x)^\frown\psi_2 \\ \wedge & \neg((\phi_1 \wedge \phi_2)^\frown(\psi_1 \wedge \psi_2)) \end{array}\right) \\
 & \Rightarrow \left(\begin{array}{ll} & (\phi_1 \wedge \ell = x)^\frown(\psi_1 \wedge (\neg\psi_2 \vee \psi_2)) \\ \wedge & (\phi_2 \wedge \ell = x)^\frown\psi_2 \wedge \neg((\phi_1 \wedge \phi_2)^\frown(\psi_1 \wedge \psi_2)) \end{array}\right) \quad \text{(PL)} \\
 & \Rightarrow \mathbf{false} \quad \text{((1), (2), (3), (4))} \\
(6) & \left(\begin{array}{l} (\phi_1 \wedge \ell = x)^\frown\psi_1 \\ \wedge(\phi_2 \wedge \ell = x)^\frown\psi_2 \end{array}\right) \Rightarrow ((\phi_1 \wedge \phi_2 \wedge \ell = x) \quad \text{((3))}
\end{array}
$$

(ILT11) $\Box\phi \Rightarrow \Box\phi^\frown\Box\phi$

(ILT12) $\Box\phi^\frown\Box\phi \Rightarrow \Box(\phi^\frown\phi)$

证明:

ILT11 可以证明如下:

$$
\begin{array}{ll}
(1) & \Box\phi \wedge \neg(\Box\phi^\frown\Box\phi) \\
 & \Rightarrow (\Box\phi^\frown\ell = 0) \wedge \neg(\Box\phi^\frown\Box\phi) \wedge \Box\phi \quad \text{(IL8)} \\
 & \Rightarrow (\Box\phi^\frown(\ell = 0 \wedge \neg\Box\phi)) \wedge \Box\phi \quad \text{(IL2)} \\
 & \Rightarrow (\mathbf{true}^\frown\neg\phi^\frown\mathbf{true}) \wedge \Box\phi \quad \text{(M, Def)} \\
 & \Rightarrow \mathbf{false} \quad \text{(Def)} \\
(2) & \Box\phi \Rightarrow \Box\phi^\frown\Box\phi \quad \text{(PL, (1))}
\end{array}
$$

ILT12 可以证明如下:

$$
\begin{array}{lll}
(1) & \left(\begin{array}{ll} & (\Box\phi \wedge \ell = a)^\frown(\Box\phi \wedge \ell = b+c+d-a) \\ \wedge & (\ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) \\ \wedge & a \geq b+c \end{array}\right) & \\
& \Rightarrow \left(\begin{array}{ll} & (\ell = b^\frown(\Box\phi \wedge \ell = c)^\frown\ell = d) \\ \wedge & (\ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) \end{array}\right) & (\mathrm{M}, \mathrm{ILT11}) \\
& \Rightarrow (\ell = b^\frown(\Box\phi \wedge \neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) & (\mathrm{ILT10}) \\
& \Rightarrow (\ell = b^\frown(\Box\phi^\frown\Box\phi \wedge \neg(\phi^\frown\phi) \wedge \ell = \phi)^\frown\ell = d) & ((\mathrm{ILT11})) \\
& \Rightarrow (\ell = b^\frown((\phi^\frown\phi) \wedge \neg(\phi^\frown\phi))^\frown\ell = d) & (\mathrm{ILT6}) \\
& \Rightarrow \mathbf{false} & (\mathrm{PL},\ \mathrm{IL4}) \\
(2) & \left(\begin{array}{ll} & (\Box\phi \wedge \ell = a)^\frown(\Box\phi \wedge \ell = b+c+d-a) \\ \wedge & (\ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) \\ \wedge & b < a \leq b+c \end{array}\right) & \\
& \Rightarrow \left(\begin{array}{ll} & ((\Box\phi \wedge \ell = b)^\frown(\Box\phi \wedge \ell = a-b) \\ \frown & (\Box\phi \wedge \ell = b+c-a)^\frown(\Box\phi \wedge \ell = d)) \\ \wedge & (\ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) \end{array}\right) & ((\mathrm{ILT11})) \\
& \Rightarrow (\ell = b^\frown(\Box\phi^\frown\Box\phi \wedge \neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) & (\mathrm{IL3},\ \mathrm{ILT10}) \\
& \Rightarrow (\ell = b^\frown((\phi^\frown\phi) \wedge \neg(\phi^\frown\phi))^\frown\ell = d) & (\mathrm{ILT6}) \\
& \Rightarrow \mathbf{false} & (\mathrm{PL},\ \mathrm{IL4}) \\
(3) & \left(\begin{array}{ll} & (\Box\phi \wedge \ell = a)^\frown(\Box\phi \wedge \ell = b+c+d-a) \\ \wedge & (\ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) \\ \wedge & a \leq b \end{array}\right) & \\
& \Rightarrow \left(\begin{array}{ll} & (\Box\phi \wedge \ell = a)^\frown(\Box\phi \wedge \ell = b-a) \\ \frown & (\Box\phi \wedge \ell = c)^\frown(\Box\phi \wedge \ell = d) \\ \wedge & (\ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) \end{array}\right) & ((\mathrm{ILT11})) \\
& \Rightarrow (\ell = b^\frown(\Box\phi \wedge \neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d) & (\mathrm{IL3},\ \mathrm{ILT6}) \\
& \Rightarrow (\ell = b^\frown((\phi^\frown\phi) \wedge \neg(\phi^\frown\phi))^\frown\ell = d) & (\mathrm{ILT11},\ \mathrm{ILT6}) \\
& \Rightarrow \mathbf{false} & (\mathrm{PL},\ \mathrm{IL4}) \\
(4) & (\Box\phi^\frown\Box\phi) \wedge \neg\Box(\phi^\frown\phi) & \\
& \Rightarrow (\Box\phi^\frown\Box\phi) \wedge (\mathbf{true}^\frown\neg(\phi^\frown\phi)^\frown\mathbf{true}) & (\mathrm{Def.}) \\
& \Rightarrow \exists a,b,c,d.\left(\begin{array}{ll} & (\Box\phi \wedge \ell = a)^\frown\Box\phi \\ \wedge & \ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d \end{array}\right) & (\mathrm{ILT2}) \\
& \Rightarrow \exists a,b,c,d.\left(\begin{array}{ll} & (\Box\phi \wedge \ell = a) \\ \frown & (\Box\phi \wedge \ell = b+c+d-a) \\ \wedge & \ell = b^\frown(\neg(\phi^\frown\phi) \wedge \ell = c)^\frown\ell = d \\ \wedge & (a \geq b+c \vee b+c > a > b \vee a \leq b) \end{array}\right) & (\mathrm{PL}) \\
& \Rightarrow \mathbf{false} & ((1),(2),(3),\mathrm{PL}) \\
(5) & \Box\phi^\frown\Box\phi \Rightarrow \Box(\phi^\frown\phi) &
\end{array}
$$

2.2 时段演算

在这部分我们简要介绍一下 DC, 包括它的语法, 语义和证明系统。关于 DC 更多的结果可参见 [36] 。

2.2.1 时段演算的语法

DC 是区间时序逻辑的一个扩充，其中的每个时序变量都有下面的特殊结构：

$$\int S$$

其中的 S 称为状态表达式，它由状态变量 $P_i, i \geq 0$ 和状态常量 0 及 1 通过下面语法构成：

$$S \quad ::= \quad 0 \quad | \quad 1 \quad | \quad P \quad | \quad S_1 \vee S_2 \quad | \quad \neg S$$

我们用 $SVar$ 表示所有状态变量的集合。

其它语法成分的定义和 ITL 的相同。

注意： 逻辑联结词 $\neg$ 和 $\vee$ 即在公式中出现，又在状态表达式中出现。但是，我们将在后面见到，它们的语义是不同的。但因为状态表达式总是出现在 $\int$ 里出现，因此这不会引起麻烦。

2.2.2 时段演算的语义

因为在 DC 中，时序变量具有特殊的结构，即 $\int S$, 因此它的语义应该由状态变量的语义导出。为了这个目的，我们给出关于全局变量，状态变量和时序命题符号的解释如下：

$$\mathcal{I} \in \left(\begin{array}{l} GVar \\ \bigcup \\ SVar \\ \bigcup \\ PLetter \end{array} \right) \longmapsto \left(\begin{array}{l} \mathbf{R} \\ \bigcup \mathbf{T}\text{ime} \mapsto \{0,1\} \\ \bigcup \\ \mathbf{I}\text{ntv} \mapsto \{tt, ff\} \end{array} \right)$$

其中，$\mathcal{I}(x) \in \mathbf{R}, \mathcal{I}(P) \in \mathbf{T}\text{ime} \mapsto \{0,1\}, \mathcal{I}(X) \in \mathbf{I}\text{ntv} \mapsto \{tt, ff\}$。并且 $\mathcal{I}(P)$ 在任意有穷区间 $[b, e]$ 上至多有有穷多个不连续点，从而 $\mathcal{I}(P)$ 在任意有穷区间上均可积。

给定一个解释 $\mathcal{I}$, 状态表达式 S 在其下的语义是类型为

$$\mathcal{I}(S) \in \mathbf{T}\text{ime} \ \mapsto \ \{0,1\}$$

的函数，它可以根据状态表达式的结构递归定义如下：

$$\begin{array}{lcl} \mathcal{I}[\![\ 0\ ]\!](t) & = & 0 \\ \mathcal{I}[\![\ 1\ ]\!](t) & = & 1 \\ \mathcal{I}[\![\ P\ ]\!](t) & = & \mathcal{I}(P)(t) \\ \mathcal{I}[\![\ (\neg S)\ ]\!](t) & = & 1 - \mathcal{I}[\![\ S\ ]\!](t) \\ \mathcal{I}[\![\ (S_1 \vee S_2)\ ]\!](t) & = & \left\{ \begin{array}{l} 0 \text{ 如果 } \mathcal{I}[\![\ S_1\ ]\!](t) = 0 \text{ 且 } \mathcal{I}[\![\ S_2\ ]\!](t) = 0 \\ 1 \text{ 其它} \end{array} \right. \end{array}$$

下面，我们将用 $S_{\mathcal{I}}$ 表示 $\mathcal{I}[\![\ S\ ]\!]$。由上述语义定义，易见 $S_{\mathcal{I}}$ 在任意有穷区间上仅有有穷多个不连续点，因而它在任意有穷区间上都是可积的。

一个具有 $\int S$ 结构的时序变量的语义是由函数 $\mathcal{I}[\![\int S]\!] \in \mathbf{I}\text{ntv} \mapsto \mathbf{R}$ 给出，它定义如下：

$$\mathcal{I}[\![\int S]\!]([b,e]) \quad = \quad \int_b^e S_{\mathcal{I}}(t)dt$$

其它语法成分的语义和 ITL 中的是一致的。

2.2.3 时段演算的证明系统

DC 除了包含所有 ITL 的公理和规则外，它还包括下面的公理和规则。关于时段的公理包括：

(DC1) $\int 0 = 0$

(DC2) $\int 1 = \ell$

(DC3) $\int S \geq 0$

(DC4) $\int S_1 + \int S_2 = \int(S_1 \vee S_2) + \int(S_1 \wedge S_2)$

(DC5) $((\int S = x_1)^\frown(\int S = x_2)) \Rightarrow (\int S = x_1 + x_2)$

(DC6) $\int S_1 = \int S_2$, 如果 $S_1 \Leftrightarrow S_2$ 成立

公理 DC1-6 指明了如何计算时段的值和推导与时段有关的性质。公理 DC1-3 和 DC6 的永真性是明显的。由于积分的可加性，所以公理 DC4 和 DC5 也是永真的。

DC 还包括下面两个归纳规则：设 $H(X)$ 是包含命题符号 X 的公式，而 S 为 DC 的任意状态表达式。

IR1

若$H(\llbracket\ \rrbracket)$ 且$H(X) \Rightarrow H(X \vee X^\frown\llbracket S\rrbracket \vee X^\frown\llbracket\neg S\rrbracket)$
那么$H(\mathbf{true})$

IR2

若$H(\llbracket\ \rrbracket)$ 且$H(X) \Rightarrow H(X \vee \llbracket S\rrbracket^\frown X \vee \llbracket\neg S\rrbracket^\frown X)$
那么$H(\mathbf{true})$

由上面两条归纳规则和相关公理，我们可以得到下面结论：

定理 2.2 设 $S_1, \ldots, S_n$ 为 DC 的 n 个状态表达式，且 $\bigvee_{i=1}^n S_i = 1$ 。则

I 若 $H(\llbracket\ \rrbracket)$ 且 $H(X) \vdash_{dc} H(X \vee \bigvee_{i=1}^n X^\frown\llbracket S_i\rrbracket)$ 那么 $H(\mathbf{true})$

II 若 $H(\llbracket\ \rrbracket)$ 且 $H(X) \vdash_{dc} H(X \vee \bigvee_{i=1}^n \llbracket S_i\rrbracket^\frown X)$ 那么 $H(\mathbf{true})$

证明： 见 [36] 。

2.2.4 时段演算的一些定理

本节我们将给出一些关于 DC 的定理。

DCT1 如果 $S_1 \Rightarrow S_2$ 那么 $\int S_1 \leq \int S_2$

证明:

(1)	$\neg S_1 \vee S_2$	(前提)
(2)	$\int \neg S_1 \vee S_2 = \int 1 = \ell$	(DC6, DC2)
(3)	$\int \neg S_1 \vee S_2 = \int \neg S_1 + \int S_2 - \int \neg S_1 \wedge S_2$	(DC4)
(4)	$\int \neg S_1 = \ell - \int S_1$	(DC4)
(5)	$\int \neg S_1 \vee S_2 = \int S_2 - \int S_1 + \ell - \int \neg S_1 \wedge S_2$	(PL)
(6)	$\int S_1 \leq \int S_2$	((2), (4), (5), DC3)

DCT2 $a > 0 \wedge b > 0 \Rightarrow ((\ell = a + b \wedge [\![S]\!]) \Leftrightarrow ((\ell = a \wedge [\![S]\!]) ^\frown (\ell = b \wedge [\![S]\!])))$

证明:

(1)	$a > 0 \wedge b > 0 \wedge (\ell = a \wedge [\![S]\!]) ^\frown (\ell = b \wedge [\![S]\!])$	
	$\Rightarrow a > 0 \wedge b > 0 \wedge (\ell = a \wedge \int S = a) ^\frown (\ell = b \wedge \int S = b)$	(Def.)
	$\Rightarrow \ell = a + b \wedge \int S = a + b \wedge a + b > 0$	(IL7, DC5, PL)
	$\Rightarrow \ell = a + b \wedge [\![S]\!]$	(Def.)
(2)	$a > 0 \wedge b > 0 \wedge \ell = a + b \wedge [\![S]\!]$	
	$\Rightarrow \ell = a ^\frown \ell = b \wedge \int S = a + b \wedge a > 0 \wedge b > 0$	(IL7, Def.)
	$\Rightarrow \begin{pmatrix} (\ell = a \wedge (\int S = a \vee \int S < a)) \\ ^\frown \quad (\ell = b \wedge \int S \leq b) \wedge \int S = a + b \end{pmatrix}$	(PL, DCT1)
(3)	$(\ell = a \wedge \int S < a) ^\frown (\ell = b \wedge \int S \leq b) \wedge \int S = a + b$	
	$\Rightarrow \ell = a + b \wedge \int S < a + b \wedge \int S = a + b$	(DC5)
	$\Rightarrow \mathbf{false}$	(PL)
(4)	$a > 0 \wedge b > 0 \wedge \ell = a + b \wedge [\![S]\!]$	
	$\Rightarrow \begin{pmatrix} (\ell = a \wedge \int S = a) ^\frown (\ell = b) \\ \wedge \quad a > 0 \wedge b > 0 \wedge \int S = a + b \end{pmatrix}$	((2), (3))
(5)	$a > 0 \wedge b > 0 \wedge \ell = a + b \wedge [\![S]\!]$	
	$\Rightarrow (\ell = a \wedge \int S = a) ^\frown (\ell = b \wedge \int S = b)$	(类似于(2), (3))
(6)	$a > 0 \wedge b > 0 \Rightarrow ((\ell = a + b \wedge [\![S]\!])$	
	$\Leftrightarrow (\ell = a \wedge [\![S]\!]) ^\frown (\ell = b \wedge [\![S]\!]))$	((1), (5))

DCT3 $\int S \geq x \Rightarrow \int S = x ^\frown \mathbf{true}$

证明: 我们关于下面命题进行归纳。设

$H(X) \hat{=} X \Rightarrow (\int S \geq x \Rightarrow \int S = x ^\frown \mathbf{true})$

根据定理 2.2, 我们仅需证明:

A: $H(\llbracket\ \rrbracket)$;

B: $H(X) \vdash H(X^\frown\llbracket\neg S\rrbracket)$;

C: $H(X) \vdash H(X^\frown\llbracket S\rrbracket)$ 。

- A: 显然。
- B:

$$
\begin{array}{ll}
& X^\frown\llbracket\neg S\rrbracket \wedge \int S \geq x \\
\Rightarrow & \exists a, b.(\ell = a \wedge X)^\frown(\ell = b \wedge \llbracket\neg S\rrbracket) \wedge \int S \geq x & \text{(ILT2)} \\
\Rightarrow & \exists a, b.(\ell = a \wedge \int S \geq x \wedge X)^\frown(\ell = b \wedge \int\neg S = \ell) & \text{(DC5)} \\
\Rightarrow & \exists a, b.(\ell = a \wedge \int S = x^\frown\mathbf{true})^\frown(\ell = b \wedge \llbracket S\rrbracket) & \text{(归纳假设)} \\
\Rightarrow & (\int S = x^\frown\mathbf{true})^\frown\mathbf{true} & \text{(M)} \\
\Rightarrow & \int S = x^\frown\mathbf{true} & \text{(IL3)}
\end{array}
$$

- C:

$$
\begin{array}{lll}
(1) & X^\frown\llbracket S\rrbracket \wedge \int S \geq x & \\
& \Rightarrow \exists a, b.(\ell = a \wedge X)^\frown(\ell = b \wedge \llbracket S\rrbracket) \wedge \int S \geq x & \text{(ILT2)} \\
& \Rightarrow \exists a, b, c. \begin{pmatrix} & (\ell = a \wedge X \wedge \int S = c) \\ ^\frown & (\ell = b \wedge \llbracket S\rrbracket) \\ \wedge & \int S = b + c \geq x \end{pmatrix} & \text{(PL, DC5)} \\
(2) & (\ell = a \wedge X \wedge \int S = c \geq x)^\frown(\ell = b \wedge \llbracket S\rrbracket) & \\
& \Rightarrow (\int S = x^\frown\mathbf{true})^\frown(\ell = b \wedge \int S = b) & \text{(归纳假设)} \\
& \Rightarrow (\int S = x^\frown\mathbf{true})^\frown\mathbf{true} & \text{(M)} \\
& \Rightarrow \int S = x^\frown\mathbf{true} & \text{(IL3, PL)} \\
(3) & \begin{pmatrix} & (\ell = a \wedge X \wedge \int S = c < x) \\ ^\frown & (\ell = b \wedge \llbracket S\rrbracket) \\ \wedge & \int S = b + c \geq x \end{pmatrix} & \\
& \Rightarrow \begin{pmatrix} & \int S = c \\ ^\frown & (\ell = (x - c) + (b + c - x) \wedge \llbracket S\rrbracket) \\ \wedge & \int S = b + c \geq x \wedge x > c \end{pmatrix} & \text{(PL)} \\
& \Rightarrow \begin{pmatrix} & \int S = 0^\frown((\ell = x - c \wedge \llbracket S\rrbracket) \\ ^\frown & (\ell = b + c - x \wedge \llbracket S\rrbracket)) \\ \wedge & b + c \geq x \wedge x > c \end{pmatrix} & \text{(DCT2)} \\
& \Rightarrow (\int S = c^\frown\int S = x - c)^\frown(\ell = b + c - x \wedge \llbracket S\rrbracket) & \text{(M, Def., IL3)} \\
& \Rightarrow \int S = x^\frown\mathbf{true} & \text{(DC5, M)} \\
(4) & X^\frown\llbracket S\rrbracket \wedge \int S \geq x \Rightarrow \int S = x^\frown\mathbf{true} & ((1), (2), (3))
\end{array}
$$

DCT4 $x > 0 \wedge y > 0 \Rightarrow (\int S = x + y \Leftrightarrow \int S = x^\frown\int S = y)$

证明:

$$
\begin{array}{lll}
(1) & x>0\wedge y>0\wedge \int S=x^\frown\int S=y & \\
& \Rightarrow \int S=x+y & (\mathrm{DC5})\\
(2) & x>0\wedge y>0\int S=x+y\wedge\neg(\int S=x^\frown\int S=y) & \\
& \Rightarrow \int S=x+y\wedge \int S=x^\frown\mathbf{true}\wedge\neg(\int S=x^\frown\int S=y) & (\mathrm{DCT3})\\
& \Rightarrow \int S=x+y\wedge(\int S=x^\frown\int S\neq y) & (\mathrm{IL2})\\
& \Rightarrow \int S=x+y\wedge\int S\neq x+y & (\mathrm{DC5})\\
& \Rightarrow \mathbf{false} & \\
(3) & x>0\wedge y>0\wedge\int S=x+y\Rightarrow\int S=x^\frown\int S=y & ((2),\ \mathrm{PL})\\
(4) & x>0\wedge y>0\Rightarrow(\int S=x+y\Leftrightarrow\int S=x^\frown\int S=y) & ((1),(3))
\end{array}
$$

下面我们证明几个将在后面用到的关于 $[\![S]\!]$ 的简单性质。

$$
\begin{array}{ll}
(\mathrm{DCT5}) & [\![1]\!]\vee[\![\ ]\!]\\
(\mathrm{DCT6}) & [\![S]\!]\vee[\![\ ]\!]\ \Rightarrow\ \Box([\![S]\!]\vee[\![\ ]\!])\\
(\mathrm{DCT7}) & [\![\neg S]\!]\Rightarrow\int S=0\\
(\mathrm{DCT8}) & (\int S\geq x^\frown\int S\geq y)\Rightarrow\int S\geq x+y\\
(\mathrm{DCT9}) & \int S\neq\ell\Rightarrow(\mathbf{true}^\frown[\![\neg S]\!]^\frown([\![S]\!]\vee[\![\ ]\!]))\\
(\mathrm{DCT10}) & (\sum_{i=1}^m\int S_i\leq\ell^\frown\sum_{i=1}^m\int S_i\leq\ell)\Rightarrow\sum_{i=1}^m\int S_i\leq\ell
\end{array}
$$

证明: DCT5 是显然的。 DCT6 的证明如下:

$$
\begin{array}{lll}
(1) & \ell=0\wedge([\![S]\!]\vee[\![\ ]\!]) & \\
& \Rightarrow\Box(\ell=0) & (\mathrm{ILT3})\\
& \Rightarrow\Box([\![S]\!]\vee[\![\ ]\!]) & (\mathrm{PL})\\
(2) & \ell>0\wedge([\![S]\!]\vee[\![\ ]\!])\wedge\Diamond\neg([\![S]\!]\vee[\![\ ]\!]) & \\
& \Rightarrow[\![S]\!]\ \wedge\ \mathbf{true}^\frown(\int S<\ell)^\frown\mathbf{true} & (\mathrm{IL1})\\
& \Rightarrow[\![S]\!]\ \wedge\ (\int S\leq\ell)^\frown(\int S<\ell)^\frown(\int S\leq\ell) & (\mathrm{DCT1})\\
& \Rightarrow[\![S]\!]\ \wedge\ (\int S<\ell) & (\mathrm{DC5})\\
& \Rightarrow\mathbf{false} & \\
(3) & \ell>0\wedge([\![S]\!]\vee[\![\ ]\!])\Rightarrow\Box([\![S]\!]\vee[\![\ ]\!]) & ((2))\\
(4) & [\![S]\!]\vee[\![\ ]\!]\Rightarrow\Box([\![S]\!]\vee[\![\ ]\!]) & ((1),(3))
\end{array}
$$

DCT7 可以证明如下:

$$
\begin{array}{lll}
(1) & \int S+\int\neg S=\int S\wedge\neg S+\int S\vee\neg S & (\mathrm{DC4})\\
(2) & \int S+\int\neg S=\ell & ((1),\mathrm{DC1},\mathrm{DC2},\mathrm{DC6})\\
(3) & [\![\neg S]\!]\Rightarrow\int S=0 & ((2),\ \mathrm{Def.})
\end{array}
$$

DCT8 的证明是显然的，我们略去。

DCT9 可以归纳证明如下:

令

$$H(X)\ \widehat{=}\ X\Rightarrow(\int S\neq\ell\Rightarrow(\mathbf{true}^\frown[\![\neg S]\!]^\frown([\![S]\!]\vee[\![\ ]\!]))$$

(1)		$X^\frown[\![\neg S]\!] \wedge \int S \neq \ell$	
	$\Rightarrow$	$\mathbf{true}^\frown[\![\neg S]\!]$	(M, PL)
	$\Rightarrow$	$\mathbf{true}^\frown[\![\neg S]\!]^\frown \ell = 0$	(IL8)
	$\Rightarrow$	$\mathbf{true}^\frown[\![\neg S]\!]^\frown([\![S]\!] \vee [\![\ ]\!])$	(M)
(2)		$X^\frown[\![S]\!] \wedge \int S \neq \ell$	
	$\Rightarrow$	$(X \wedge \int S \neq \ell)^\frown[\![S]\!]$	(DC5)
	$\Rightarrow$	$(\mathbf{true}^\frown[\![\neg S]\!]^\frown([\![S]\!] \vee [\![\ ]\!]))^\frown[\![S]\!]$	(归纳假设)
	$\Rightarrow$	$\mathbf{true}^\frown[\![\neg S]\!]^\frown[\![S]\!]$	(IL3, DCT2)
	$\Rightarrow$	$\mathbf{true}^\frown[\![\neg S]\!]^\frown([\![S]\!] \vee [\![\ ]\!])$	(M)
(3)		$\mathbf{true} \Rightarrow (\int S \neq \ell \Rightarrow (\mathbf{true}^\frown[\![\neg S]\!]^\frown([\![S]\!] \vee [\![]\!]))$	(IR1)
(4)		$\int S \neq \ell \Rightarrow (\mathbf{true}^\frown[\![\neg S]\!]^\frown([\![S]\!] \vee [\![\ ]\!]))$	((3), PL)

DCT10 的证明类似于 DCT8 。

第 三 章
实例: 期限驱动调度算法的形式证明

本章, 我们将用时段演算来形式描述期限驱动调度算法并证明其正确性。我们想用这个比较复杂的例子来说明如何使用时段演算来刻画实时系统并验证实时系统的实时性质。

期限驱动调度算法是 Liu 和 Layland 提出的用来调度多个任务共享同一处理器的一种策略 [48] 。该算法假设所有任务周期地向处理器申请处理器时间。算法根据任务的当前请求的期限动态分配给它们优先级。如果一个任务它当前的请求的期限较急, 那么它可以分配到较高的优先级; 否则, 分配给它较低的优先级。在任意时刻, 只有优先级高且当前周期的任务没有完成的进程才能占有处理器。因此, 使用 DDS, 通过根据当前任务的请求的期限动态地分配优先级的方法, 可以充分利用处理器资源。

[48] 给出一个用 DDS 来调度多个任务能行的充要条件。该条件可以表述为下述定理:

定理 (Liu/Layland) *给定 m 个任务, 期限驱动调度算法能行的充要条件是:*

$$(C_1/T_1) + \ldots + (C_m/T_m) \leq 1 \quad (0 < C_i < T_i)$$

其中 C_i 是任务 p_i 申请的处理器时间, T_i 是它的请求周期 (假设 T_i $i = 1, \ldots, m$ 是正整数) 。

[48] 给出了该定理的一个非形式的证明。该定理的必要性部分是显然的, 但充分性部分理解起来就非常困难。 [86] 用时段演算给出了 DDS 一个形式描述和一个形式证明, 这是文献中关于 DDS 的第一个形式证明。该证明主要运用时段演算中的归纳规则, 这样使得证明缺乏直观, 从而难于理解。

这里我们使用 DC 给出 DDS 另外一个较直观的形式证明。该证明可参见 [56] 。我们通过三步来证明它的充分性: 首先证明如果处理器在一个区间里没有空闲那么在这个区间上这个任务集合是可调度的; 其次我们证明给定一个时间区间, 如果在该时间区间的任意真前缀上这个任务集合都是可调度的, 但是却在该时间区间上不可调度, 那么处理器在该时间区间上没有空闲时间; 最后, 我们反设在某个时间区间上这个任务集合不可调度, 那么存在该时间区间的一个前缀区间使得在该前缀区间的任意真前缀区间上这个任务集合是可调度的, 但是在该前缀区间上不可调度。从而与第一, 二步的结论矛盾。因而充分性得证。

3.1 形式描述

我们对期限调度算法的描述包括下述四个部分:

- **若干进程共享同一处理器,**
- **每个进程周期性地申请处理器时间,**
- **每个进程的需求,**
- **调度算法。**

3.1.1 共享处理器

假设 $p_1, \ldots, p_m$ 是给定的 m 个进程，它们运行在同一个处理器上。令

$$\alpha = \{1, \ldots, m\}$$

我们用下面两个状态变量来刻画进程 p_i 的行为:

$$\begin{array}{lll} \mathrm{Run}_i: & \mathbf{T}\mathrm{ime} \rightarrow & \{0,1\} \\ \mathrm{Std}_i: & \mathbf{T}\mathrm{ime} \rightarrow & \{0,1\} \end{array} \qquad \text{其中 } i \in \alpha$$

它们的含义是：如果 p_i 在 t 时刻在处理器上执行，那么 $\mathrm{Run}_i(t) = 1$; 否则 $\mathrm{Run}_i(t) = 0$ 。 $\mathrm{Std}_i(t) = 1$ 意味着 p_i 在 t 时刻对处理器有请求；而 $\mathrm{Std}_i(t) = 0$ 指 p_i 在 t 时刻没有向处理器请求，换句话说，就是 p_i 在 t 时刻已经完成了当前周期的任务。

一个进程只有当它保持请求时才能运行:

$$A_1 \mathrel{\widehat{=}} [\![\mathrm{Run}_i]\!] \Rightarrow [\![\mathrm{Std}_i]\!]$$

在任意时刻至多只有一个进程占有处理器:

$$A_2 \mathrel{\widehat{=}} [\![\mathrm{Run}_i]\!] \Rightarrow \bigwedge_{j \neq i} [\![\neg \mathrm{Run}_j]\!]$$

任意进程在任意子区间内都必须满足这些性质:

$$ShProc \mathrel{\widehat{=}} \Box \bigwedge_{i \in \alpha} (A_1 \wedge A_2)$$

由上述共享处理器的描述我们可以得出所有进程的执行时间的总和不会超过该区间的长度:

引理 3.1

$$ShProc \Rightarrow (\sum_{i \in \alpha} \int Run_i) \leq \ell$$

证明: 我们以下述命题作为归纳假设进行归纳

$$X \Rightarrow (ShProc \Rightarrow (\sum_{i \in \alpha} \int \mathrm{Run}_i) \leq \ell)$$

因为

$$\bigvee_{i\in\alpha} \mathrm{Run}_i \vee (\bigwedge_{i\in\alpha} \neg\mathrm{Run}_i) = 1$$

所以根据定理 2.2, 我们仅需证明:

A: $H(\llbracket\ \rrbracket)$,

B: $H(X) \vdash H(X^\frown\llbracket\mathrm{Run}_i\rrbracket)$, 对任意 $i \in \alpha$,

C: $H(X) \vdash H(X^\frown\llbracket\bigwedge_{i\in\alpha}\neg\mathrm{Run}_i\rrbracket)$ 。

A: 因为 $\llbracket\ \rrbracket \Rightarrow \int\mathrm{Run}_i = 0$, 所以显然。

B:

(1)	$\llbracket\mathrm{Run}_i\rrbracket$	
	$\Rightarrow \llbracket\mathrm{Run}_i\rrbracket \wedge \bigwedge_{j\neq i}\llbracket\neg\mathrm{Run}_j\rrbracket$	$(ShProc)$
	$\Rightarrow \int\mathrm{Run}_i = \ell \wedge \bigwedge_{j\neq i}\int Run_j = 0$	(DCT2)
	$\Rightarrow \sum_{i\in\alpha}\mathrm{Run}_i \leq \ell$	(PL)
(2)	$(X^\frown\llbracket\mathrm{Run}_i\rrbracket) \wedge ShProc$	
	$\Rightarrow\ (X^\frown\llbracket\mathrm{Run}_i\rrbracket) \wedge \Box ShProc$	(ILT7)
	$\Rightarrow\ (X \wedge ShProc)^\frown\llbracket\mathrm{Run}_i\rrbracket$	(ILT8)
	$\Rightarrow\ ((\sum_{i\in\alpha}\int\mathrm{Run}_i) \leq \ell)^\frown\llbracket\mathrm{Run}_i\rrbracket$	$(H(X))$
	$\Rightarrow\ ((\sum_{i\in\alpha}\int\mathrm{Run}_i) \leq \ell)^\frown((\sum_{i\in\alpha}\int\mathrm{Run}_i) \leq \ell)$	$((1), \mathrm{M})$
	$\Rightarrow\ (\sum_{i\in\alpha}\int\mathrm{Run}_i) \leq \ell$	(DCT10)
(3)	$H(X^\frown\llbracket\mathrm{Run}_i\rrbracket)$	$((2))$

C: 由 DCT2, DCT7, ILT7, ILT1 和归纳假设易证。 □

3.1.2 周期性请求

每个进程 p_i 都周期地请求处理器时间, 这些请求开始在 $k\cdot T_i$ 时刻, 其中 $k = 0, 1, 2, 3, \ldots$ 。每个周期开始时, 进程开始对处理器进行请求。而一旦进程 p_i 在该周期内累积执行时间达到 C_i, 即 $\int\mathrm{Run}_i = C_i$, 那么它就不再请求, 也就是说当前周期内的任务已经执行完毕。

考虑下面公式:

$$mult_i \mathrel{\widehat{=}} T_i \mid \ell$$

其中 $a \mid b$ 读作 "a 整除 b", 或者说 "b 是 a" 的整数倍。如果存在一个自然数 k 使得 $k\cdot a = b$, 那么它为真。这样, $mult_i$ 在所有长度为 T_i 的整数倍的区间上为真。

进程 p_i 在每个周期开始时必须开始请求处理器可以形式地表示成:

$$StartR_i \mathrel{\widehat{=}} \Box_p(mult_i^\frown(\ell \leq T_i \wedge ((\llbracket\mathrm{Std}_i\rrbracket^\frown\mathbf{true}) \vee \llbracket\ \rrbracket)))$$

一个进程在一个周期内只有当它的请求被执行完了之后它才能放弃对处理器的请求。也就是说，如果 Std_i 变成了 0, 那么 p_i 在当前周期内的任务必然已经执行完毕。

$$HoldR_i \mathrel{\widehat{=}} \Box_p((mult_i^{\frown}\begin{pmatrix}\ell \le T_i\\ \wedge(\Diamond[\![\neg\mathrm{Std}_i]\!])\end{pmatrix}) \Rightarrow (mult_i^{\frown}\begin{pmatrix}\ell \le T_i\\ \wedge\int\mathrm{Run}_i = C_i\end{pmatrix}))$$

当进程在当前周期内的任务完成之后，进程必须放弃对处理器的请求。等价地说，就是进程完成了当前周期的任务后，在当前周期内 Std_i 不能再保持为 1 。

$$DisappearR_i \mathrel{\widehat{=}} \Box_p(\neg(mult_i^{\frown}\begin{pmatrix}\ell \le T_i\\ \wedge((\int\mathrm{Run}_i = C_i)^{\frown}(\Diamond[\![\mathrm{Std}_i]\!]))\end{pmatrix}))$$

对于任意进程，上述三个公式必须满足。因此 m 个进程周期性请求同一处理器可以形式地表述为:

$$PeriodicR \mathrel{\widehat{=}} \bigwedge_{i\in\alpha}(StartR_i \wedge HoldR_i \wedge DisappearR_i)$$

下面引理说明每个进程的执行时间不会超过它的最大需求。

引理 3.2

$$ShProc \wedge PeriodicR \Rightarrow \bigwedge_{i\in\alpha}\int Run_i \le \lceil \ell/T_i \rceil \cdot C_i$$

证明: 对每个进程 p_i, 对 $\lceil \ell/T_i \rceil$ 作数学归纳，结论立得。 □

3.1.3 需求

期限驱动调度算法的需求是每个进程必须在它的每个周期内完成任务。

进程 p_i 的需求可以形式地表述为

$$Req_i \mathrel{\widehat{=}} \int\mathrm{Run}_i \ge \lfloor \ell/T_i \rfloor \cdot C_i$$

这个需求对每个进程在每个前缀区间上必须满足，即:

$$Req \mathrel{\widehat{=}} \Box_p \bigwedge_{i\in\alpha} Req_i$$

3.1.4 算法

算法的作用是分配处理器时间使得每个进程在它的所有期限处均满足它的需求。进程 p_i 的期限定义为给定周期的终止，或者说给定周期的下一个周期的开始。

在下面的讨论中我们始终假设所有进程在 0 时刻开始它的第一个周期。

如果在 t 时刻进程 p_i 的期限比 p_j 更近，则说在 t 时刻 p_i 比 p_j 更紧急。其中 $t \geq 0$ 。即

$$urgent(i,j)(t) \mathrel{\widehat{=}} ((\lfloor t/T_i \rfloor + 1)T_i - t) < ((\lfloor t/T_j \rfloor + 1)T_j - t)$$

$urgent(i,j)$ 的特征函数可以看成是一个状态变量：

$$urgent(i,j): \quad \mathbf{T}\mathrm{ime} \ \rightarrow \ \{0,1\} \ \text{其中} \ i,j \in \alpha$$

注意：由上述定义可知一个没有请求的进程可能比正在请求的进程更紧急。

算法必须保证在任意时刻在最紧急且没有完成任务的几个进程中有一个占有处理器。这可以通过两步来形式化它。

公式：

$$Sch_1 \mathrel{\widehat{=}} \bigwedge_{i,j\in\alpha} \Box\neg(\llbracket urgent(i,j) \rrbracket \wedge \llbracket \mathrm{Run}_j \wedge \mathrm{Std}_i \rrbracket)$$

表示处于请求状态且期限较近的进程具有较高的优先级。

然而，当没有进程在区间上运行，尽管有些进程处于请求状态时，Sch_1 仍旧为真。下面的公式是说只要有进程处于请求状态，那么必然有某个进程在执行。

$$Sch_2 \mathrel{\widehat{=}} \bigwedge_{i\in\alpha} \Box(\llbracket Std_i \rrbracket \Rightarrow \llbracket \bigvee_{j\in\alpha} \mathrm{Run}_j \rrbracket)$$

公式 Sch_1 和 Sch_2 一起说明几个最紧急且没有完成任务的进程中有一个将被执行。

$$Sch \mathrel{\widehat{=}} Sch_1 \wedge Sch_2$$

3.2 Liu/Layland 定理的形式证明

Liu/Layland 定理包含两个部分：一部分是证明条件的必要性，即要证 $\sum_{i\in\alpha} \frac{C_i}{T_i} \leq 1$; 另一部分是证明该条件的充分性。为了证明必要性，我们只需找到一个特殊长度的区间使得必要性在该区间上可证即可。

3.2.1 必要性

在长度为 $a = T_1 \cdot T_2 \cdot \cdots \cdot T_m$ 的区间上，我们容易证明该条件的必要性。注意因为我们假设每个 T_i 是正整数，所以 T_i 整除 a 。

证明：

$$\begin{array}{lll}
 & \begin{pmatrix} \ell = T_1 \cdot T_2 \cdot \cdots \cdot T_m \\ \wedge\, ShProc \wedge PeriodicR \wedge Sch \wedge Req \end{pmatrix} & \\
\Rightarrow & \bigwedge_{i\in\alpha} \int \mathrm{Run}_i \geq \lfloor \ell/T_i \rfloor \cdot C_i & (Req, \text{ILT6}) \\
\Rightarrow & \bigwedge_{i\in\alpha} \int \mathrm{Run}_i \geq \ell/T_i \cdot C_i & (\lfloor \ell/T_i \rfloor = \ell/T_i) \\
\Rightarrow & (\sum_{i\in\alpha} \int \mathrm{Run}_i) \geq (\sum_{i\in\alpha} \ell/T_i \cdot C_i) & \\
\Rightarrow & \ell \cdot (\sum_{i\in\alpha} C_i/T_i) \leq \ell & (\text{引理 } 3.1) \\
\Rightarrow & \sum_{i\in\alpha} C_i/T_i \leq 1 &
\end{array}$$

□

3.2.2 充分性

为了简化证明，我们令

$$Cond \quad \widehat{=} \quad ShProc \wedge PeriodicR \wedge Sch \wedge \sum_{i\in\alpha} C_i/T_i \leq 1$$

我们将通过三步来证明充分性：首先证明如果处理器在某个时间区间里没有空闲那么需求在这段时间里是满足的；其次我们证明给定一个时间区间，如果需求在该时间区间的任意真前缀上都是满足的，但是却在该时间区间上不满足，那么处理器在该时间区间上没有空闲时间；最后，我们反设需求在某个时间区间上不满足，那么存在该时间区间的一个前缀区间使得需求在该前缀区间的任意真前缀区间上都是满足的，但是在该前缀区间上不满足。根据第一，二步的结论，这是不可能的。因此充分性得证。

为了证明充分性，我们首先证明一些引理。

下面的引理是说如果进程 p_i 不再请求，那么它在当前周期内的任务已经执行完毕 ($HoldR_i$)。而且，如果 p_i 的需求在前面的所有周期内都满足，那么它在整个区间上的所有请求都已执行完毕。

引理 3.3 对任意 $i \in \alpha$:

$$\begin{pmatrix} ShProc \wedge PeriodicR \\ \wedge\ (\Box_p Req_i \frown [\![\neg \mathrm{Std}_i]\!]) \end{pmatrix} \ \Rightarrow\ \int \mathrm{Run}_i = \lceil \ell/T_i \rceil \cdot C_i$$

证明:

$$\begin{array}{lll} & ShProc \wedge PeriodicR \wedge (\Box_p Req_i \frown [\![\neg \mathrm{Std}_i]\!]) & \\ \Rightarrow & \begin{pmatrix} mult_i \\ \wedge \int \mathrm{Run}_i = \lfloor \ell/T_i \rfloor \cdot C_i \end{pmatrix} \frown \begin{pmatrix} \ell \leq T_i \\ \wedge\ [\![\mathrm{Std}_i]\!] \frown \mathbf{true} \frown [\![\neg \mathrm{Std}_i]\!] \end{pmatrix} & (\mathrm{ILT8}, StartR_i) \\ \Rightarrow & \begin{pmatrix} mult_i \\ \wedge \int \mathrm{Run}_i = (\ell/T_i) \cdot C_i \end{pmatrix} \frown \begin{pmatrix} \ell \leq T_i \\ \wedge \int \mathrm{Run}_i = C_i \end{pmatrix} & (HoldR_i) \\ \Rightarrow & \int \mathrm{Run}_i = \lceil \ell/T_i \rceil \cdot C_i & (\mathrm{DC5}) \end{array}$$

□

在下面的引理中，我们将证明如果需求不满足，那么一定存在一个最小前缀子区间使得需求首次在该区间上不满足。

引理 3.4

$$(Cond \wedge \exists i_0 \in \alpha \Diamond_p \neg Req_{i_0}) \Rightarrow\ \exists j_0 \in \alpha \exists m_{j_0} \in \mathcal{N} \begin{pmatrix} \ell = m_{j_0} T_{j_0} \\ \wedge \int Run_{j_0} < \lfloor \ell/T_{j_0} \rfloor \cdot C_{j_0} \\ \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{pmatrix} \frown \mathbf{true}$$

证明: 易证。 □

我们将在后面用到下面两个事实：

$$(P_1)\quad \lfloor a_\ell/a_T\rfloor \le \lfloor (a_\ell - a_x)/a_T\rfloor + \lceil a_x/a_T\rceil \quad \text{若 } a_\ell \ge a_x \ge 0 \text{ 且 } a_T > 0$$

和

$$(P_2)\quad \text{如果}\quad A(x)\Rightarrow B(x)\quad \text{那么}\quad \exists x A(x)\Rightarrow \exists x B(x)$$

下面，我们将证明一个非常重要的引理。给定一个区间，如果需求在它的所有真前缀子区间上都可满足，即 $\bigwedge_{i\in\alpha}\Box_b\mathbf{Req}_i$; 并且存在一个前缀子区间使得所有进程在此子区间上的所有请求都执行完毕，即 $\bigwedge_{i\in\alpha}\int \mathrm{Run}_i = \lceil \ell/T_i\rceil\cdot C_i$; 而且处理器在该区间的剩下部分没有空闲，即 $[\![\bigvee_{i\in\alpha}\mathrm{Run}_i]\!]$, 那么需求在整个区间上可满足。

引理 3.5

$$(Cond \wedge ((\bigwedge_{i\in\alpha}\int Run_i = \lceil \ell/T_i\rceil\cdot C_i)^\frown[\![\bigvee_{i\in\alpha} Run_i]\!]) \wedge \bigwedge_{i\in\alpha}\Box_b\mathbf{Req}_i) \Rightarrow Req$$

证明:

$$(1)\quad \begin{pmatrix} Cond \wedge ((\bigwedge_{i\in\alpha}\int \mathrm{Run}_i = \lceil \ell/T_i\rceil\cdot C_i)^\frown[\![\bigvee_{i\in\alpha}\mathrm{Run}_i]\!]) \\ \wedge\bigwedge_{i\in\alpha}\Box_b\mathbf{Req}_i \wedge \exists i_0\in\alpha\Diamond_p\neg Req_{i_0}\end{pmatrix}$$
$$\Rightarrow \exists i_0\in\alpha\exists d\begin{pmatrix} Cond\wedge\int Run_{i_0} < m_{i_0}C_{i_0}\wedge\ell = m_{i_0}T_{i_0}\\ \wedge((\bigwedge_{i\in\alpha}\int\mathrm{Run}_i = \lceil\ell/T_i\rceil\cdot C_i)\\ ^\frown([\![\bigvee_{i\in\alpha}\mathrm{Run}_i]\!]\wedge\ell = d\ge T_{i_0}))\\ \wedge\bigwedge_{i\in\alpha}\Box_b\mathbf{Req}_i\end{pmatrix}\qquad (\mathrm{ILT2})$$

为了简单起见，令

$$Abbr\ \widehat{=}\ \begin{pmatrix}\ell = m_{i_0}T_{i_0}\wedge\ \ Cond\wedge\bigwedge_{i\in\alpha}\Box_b\mathbf{Req}_i\\ \wedge((\bigwedge_{i\in\alpha}\int\mathrm{Run}_i = \lceil\ell/T_i\rceil\cdot C_i)^\frown([\![\bigvee_{i\in\alpha}\mathrm{Run}_i]\!]\wedge\ell = d\ge T_{i_0}))\\ \wedge\int Run_{i_0} < m_{i_0}C_{i_0}\end{pmatrix}$$

我们将进程集合 α 分成四个互不相交子集，即 β_1,β_2,β_3 和 β_4 。该分法如下：

$$\begin{aligned}
\beta_1 &\ \widehat{=}\ \{i \mid i\in\alpha\wedge\Box_p Req_i\wedge(mult_i^\frown(\ell < T_i\wedge\ell\le T_{i_0})\}\\
\beta_2 &\ \widehat{=}\ \{i \mid i\in\alpha\wedge\Box_p Req_i\wedge(mult_i^\frown(T_{i_0}<\ell<T_i\wedge[\![\neg\mathrm{Run}_i]\!]))\}\\
\beta_3 &\ \widehat{=}\ \{i \mid i\in\alpha\wedge\Box_p Req_i\wedge(mult_i^\frown(T_{i_0}<\ell<T_i\wedge(\Diamond[\![\mathrm{Run}_i]\!])))\}\\
\beta_4 &\ \widehat{=}\ \{i \mid i\in\alpha\wedge\int Run_i < \lfloor m_{i_0}T_{i_0}/T_i\rfloor\cdot C_i\}
\end{aligned}$$

显然地，$\alpha = \bigcup_{i=1}^4\beta_i$ 且 $\beta_i\cap\beta_j = \emptyset$, 对所有 $i\neq j$ 且 $i,j = 1,2,3,4$ 。

下面我们将证明在条件 $Abbr$ 下对每个 $i \in \beta_1 \cup \beta_2$ 中的进程 p_i, 它的执行时间不超过 $\lfloor \ell/T_i \rfloor \cdot C_i$ 。

$$
\begin{array}{lll}
(2) & Abbr \wedge i \in \beta_1 & \\
& \Rightarrow \exists x_i (mult_i^\frown (\ell = x_i \wedge x_i < T_i \wedge x_i \le T_{i_0})) & \text{(ILT2, Def.)} \\
& \Rightarrow \exists x_i \left(\begin{array}{c} mult_i \\ \frown \left(\begin{array}{c} (\ell = x_i \wedge x_i < T_i \wedge 0 < x_i \le T_{i_0} \\ \wedge [\![urgent(i_0, i)]\!]) \vee (\ell = x_i \wedge x_i = 0) \end{array} \right) \end{array} \right) & (P_2, \text{Sch}) \\
& \Rightarrow (\int \mathrm{Run}_i \le \lfloor \ell/T_i \rfloor \cdot C_i)^\frown (\int \mathrm{Run}_i = 0) & \text{(引理 3.2, Sch)} \\
& \Rightarrow \int \mathrm{Run}_i \le \lfloor \ell/T_i \rfloor \cdot C_i & \text{(DCT8)} \\
(3) & Abbr & \\
& \Rightarrow \bigwedge_{i \in \beta_1} \int \mathrm{Run}_i \le \lfloor \ell/T_i \rfloor \cdot C_i & ((2), \text{PL}) \\
(4) & Abbr \wedge i \in \beta_2 & \\
& \Rightarrow \exists x_i (mult_i^\frown (\ell = x_i \wedge x_i < T_i \wedge x_i > T_{i_0} \wedge [\![\neg \mathrm{Run}_i]\!])) & \text{(ILT2, Def.)} \\
& \Rightarrow ((\int \mathrm{Run}_i \le \lfloor \ell/T_i \rfloor \cdot C_i)^\frown (\int \mathrm{Run}_i = 0)) & \text{(引理 3.2, DCT7)} \\
& \Rightarrow \int \mathrm{Run}_i \le \lfloor \ell/T_i \rfloor \cdot C_i & \text{(DCT8)} \\
(5) & Abbr & \\
& \Rightarrow \bigwedge_{i \in \beta_2} \int \mathrm{Run}_i \le \lfloor \ell/T_i \rfloor \cdot C_i & ((4), \text{PL})
\end{array}
$$

下面我们将证明如果 $\beta_3 = \emptyset$ 那么 $\sum_{i \in \alpha} \frac{C_i}{T_i} > 1$ 。

$$
\begin{array}{lll}
(6) & Abbr \wedge \beta_3 = \emptyset & \\
& \Rightarrow \mathbf{true}^\frown \left(\begin{array}{c} \ell = d \wedge (\sum_{i \in \beta_1} \lfloor \ell/T_i \rfloor \cdot C_i + \\ \sum_{i \in \beta_2} \lfloor \ell/T_i \rfloor \cdot C_i + \sum_{i \in \beta_4} \lfloor \ell/T_i \rfloor \cdot C_i > \ell) \end{array} \right) & ((3),\ (5),\ P_1) \\
& \Rightarrow \sum_{i \in \alpha} \frac{C_i}{T_i} > 1 & (\lfloor \ell/T \rfloor \le \ell/T)
\end{array}
$$

下一步，我们将证明如果 $\beta_3 \neq \emptyset$ 那么在 $Abbr$ 条件下，我们可以根据在 β_3 中进程的执行情况把区间分成三段使得：第一段可以任意；在第二段里，处理器一直被 β_3 中的进程占据；在第三段里，没有 β_3 中的进程占用处理器，显然它的长度界于 T_{i_0} 和 d 之间。

$$
\begin{array}{lll}
(7) & Abbr \wedge \beta_3 \neq \emptyset & \\
& \Rightarrow \exists j_0 \in \beta_3 \exists x_{j_0} (mult_{j_0}^\frown \left(\begin{array}{c} \ell = x_{j_0} \wedge T_{i_0} < x_{j_0} < T_{j_0} \\ \wedge (\Diamond [\![\mathrm{Run}_{j_0}]\!]) \end{array} \right)) & \text{(ILT2, Def.)} \\
& \Rightarrow \exists j_0 \in \beta_3 \exists x_{j_0} (mult_{j_0}^\frown \left(\begin{array}{l} \ell = x_{j_0} \wedge T_{i_0} < x_{j_0} < T_{j_0} \\ \wedge (\Diamond [\![\mathrm{Run}_{j_0}]\!]) \\ \wedge (\mathbf{true}^\frown (\ell = T_{i_0} \\ \wedge \bigwedge_{i \in \beta_3} [\![urgent(i_0, i)]\!])) \end{array} \right) & (P_2, \text{Sch}) \\
& \Rightarrow \exists j_0 \in \beta_3 \exists x_{j_0} (mult_{j_0}^\frown \left(\begin{array}{c} \ell = x_{j_0} \wedge T_{i_0} < x_{j_0} < T_{j_0} \wedge (\Diamond [\![\mathrm{Run}_{j_0}]\!]) \\ \wedge (\mathbf{true}^\frown (\ell = T_{i_0} \wedge [\![\bigwedge_{i \in \beta_3} \neg \mathrm{Run}_i]\!])) \end{array} \right) & (\text{Sch}, P_2)
\end{array}
$$

$$\Rightarrow \exists j_0 \in \beta_3 \exists x_{j_0} (mult^{\frown}_{j_0} \left(\begin{array}{c} (\ell = x_{j_0} - T_{i_0} \wedge (\Diamond [\![\mathrm{Run}_{j_0}]\!])) \\ \frown (\ell = T_{i_0} \wedge [\![\bigwedge_{i \in \beta_3} \neg \mathrm{Run}_i]\!])) \end{array} \right) \quad (\mathrm{M}, P_2)$$

$$\Rightarrow \exists j_0 \in \beta_3 \exists x_{j_0} \left(\begin{array}{l} mult^{\frown}_{j_0} (\ell = x_{j_0} - T_{i_0} \wedge \\ (\mathbf{true} \frown [\![\bigvee_{i \in \beta_3} \mathrm{Run}_i]\!] \\ \frown ([\![\bigwedge_{i \in \beta_3} \neg \mathrm{Run}_i]\!] \vee [\![ ]\!]))) \\ \frown (\ell = T_{i_0} \wedge [\![\bigwedge_{i \in \beta_3} \neg \mathrm{Run}_i]\!]) \end{array} \right) \quad (\mathrm{DCT9},\ P_2)$$

$$\Rightarrow ((\ell = a) \frown (\ell = b \wedge [\![\bigvee_{i \in \beta_3} \mathrm{Run}_i]\!]) \frown \left(\begin{array}{c} T_{i_0} \leq \ell = c \leq d \\ \wedge [\![\bigwedge_{i \in \beta_3} \neg \mathrm{Run}_i]\!] \end{array} \right)) \quad (\mathrm{PL})$$

下面，我们将证明对所有 $\beta_1 \cup \beta_2 \cup \beta_4$ 中的进程 p_i，它在最后一段时间内的执行时间不超过 $\lfloor \ell / T_i \rfloor \cdot C_i$

(8) $Abbr \wedge \beta_3 \neq \emptyset \wedge i \in \beta_1$

$$\Rightarrow \left(\begin{array}{ll} & (\ell = a) \\ \frown & (\ell = b \wedge [\![\bigvee_{j \in \beta_3} \mathrm{Run}_j]\!] \\ \wedge & \bigwedge_{j \in \beta_3} [\![urgent(i,j)]\!]) \\ \frown & (\ell = c \wedge [\![\bigwedge_{j \in \beta_3} \neg \mathrm{Run}_j]\!]) \end{array} \right) \quad ((7), \mathrm{ILT8}, \mathrm{Sch}, \mathrm{Def.})$$

$\Rightarrow (\ell = a \frown (\ell = b \wedge [\![\neg \mathrm{Std}_i]\!]) \frown (\ell = c))$ (Sch)

$\Rightarrow ((\ell = a + b \wedge \int \mathrm{Run}_i = \lceil \ell / T_i \rceil \cdot C_i) \frown (\ell = c))$ (引理 3.3)

(9) $Abbr \wedge \beta_3 \neq \emptyset$

$\Rightarrow \bigwedge_{i \in \beta_1} (\ell = a + b) \frown (\ell = c \wedge \int \mathrm{Run}_i \leq \lfloor \ell / T_i \rfloor \cdot C_i)$ $((3),\ (8),\ P_1)$

(10) $Abbr \wedge \beta_3 \neq \emptyset \wedge j \in \beta_2 \wedge mult^{\frown}_j (\ell < T_j \wedge T_j \geq c)$

$\Rightarrow (\ell = a + b) \frown (\ell = c \wedge \int \mathrm{Run}_j = 0)$ (DCT2, Def.)

$\Rightarrow (\ell = a + b) \frown (\ell = c \wedge \int \mathrm{Run}_j \leq \lfloor \ell / T_j \rfloor \cdot C_j)$ (PL)

(11) $Abbr \wedge \beta_3 \neq \emptyset \wedge j \in \beta_2 \wedge mult^{\frown}_j (\ell < T_j \wedge T_j < c)$

$$\Rightarrow (\ell = a) \frown \left(\begin{array}{c} \ell = b \wedge [\![\bigvee_{i \in \beta_3} \mathrm{Run}_i]\!] \\ \wedge \bigwedge_{i \in \beta_3} [\![urgent(j,i)]\!] \end{array} \right)^{\frown} (\ell = c)) \quad (\mathrm{Sch})$$

$\Rightarrow (\ell = a + b \wedge \int \mathrm{Run}_j = \lceil \ell / T_j \rceil \cdot C_j) \frown (\ell = c)$ (引理 3.3)

$\Rightarrow (\ell = a + b) \frown (\ell = c \wedge \int \mathrm{Run}_j \leq \lfloor \ell / T_j \rfloor \cdot C_j)$ $((5),\ P_1)$

(12) $Abbr \wedge \beta_3 \neq \emptyset$

$\Rightarrow \bigwedge_{j \in \beta_2} ((\ell = a + b) \frown (\ell = c \wedge \int \mathrm{Run}_j \leq \lfloor \ell / T_i \rfloor \cdot C_j)$ $((10), (11))$

(13) $Abbr \wedge \beta_3 \neq \emptyset \wedge i \in \beta_4$

$$\Rightarrow \left(\begin{array}{c} (\ell = a) \frown \\ (\ell = b \wedge [\![\bigvee_{j \in \beta_3} \mathrm{Run}_j]\!] \wedge \bigwedge_{j \in \beta_3} [\![urgent(i,j)]\!]) \\ \frown (\ell = c) \end{array} \right) \quad (\mathrm{Sch})$$

$\Rightarrow ((\ell = a + b \wedge \int \mathrm{Run}_i = \lceil \ell / T_i \rceil \cdot C_i) \frown (\ell = c))$ (引理 3.3)

(14) $Abbr \wedge \beta_3 \neq \emptyset$

$\Rightarrow \bigwedge_{i \in \beta_4} (\ell = a + b) \frown (\ell = c \wedge \int \mathrm{Run}_i < \lfloor \ell / T_i \rfloor \cdot C_i)$ $((13), P_1, \mathrm{Def.})$

下面是该引理的余下部分的证明:

$$
\begin{array}{lll}
(15) & Abbr \wedge \beta_3 \neq \emptyset & \\
& \Rightarrow \left(\begin{array}{l} (\ell = a + b) \\ \frown(\sum_{i\in\beta_1} \lfloor c/T_i \rfloor \cdot C_i + \\ \sum_{i\in\beta_2} \lfloor c/T_i \rfloor \cdot C_i + \ \sum_{i\in\beta_4} \lfloor c/T_i \rfloor \cdot C_i > c) \end{array} \right) & \left(\begin{array}{c} (9), (12), \\ (14),\ \mathrm{ILT10} \end{array} \right) \\
& \Rightarrow (\ell = a + b) \frown \sum_{i\in\alpha} \frac{C_i}{T_i} > 1 & (\lfloor \ell/T \rfloor \leq \ell/T) \\
& \Rightarrow \sum_{i\in\alpha} \frac{C_i}{T_i} > 1 & (\mathrm{IL4}) \\
(16) & Abbr & \\
& \Rightarrow \sum_{i\in\alpha} \frac{C_i}{T_i} > 1 & ((6), (15)) \\
& \Rightarrow \mathbf{false} & \\
(17) & \left(\begin{array}{l} Cond \wedge ((\bigwedge_{i\in\alpha} \int \mathrm{Run}_i = \lceil \ell/T_i \rceil \cdot C_i) \frown [\![\bigvee_{i\in\alpha} \mathrm{Run}_i]\!]) \\ \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \wedge \exists i_0 \in \alpha \Diamond_p \neg Req_{i_0} \end{array} \right) & \\
& \Rightarrow \mathbf{false} & ((1), (16), P_2) \\
(18) & \left(\begin{array}{l} Cond \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \\ \wedge((\bigwedge_{i\in\alpha} \int \mathrm{Run}_i = \lceil \ell/T_i \rceil \cdot C_i) \frown [\![\bigvee_{i\in\alpha} \mathrm{Run}_i]\!]) \end{array} \right) & \\
& \Rightarrow \bigwedge_{i\in\alpha} \Box_p \mathbf{Req}_i & \Box
\end{array}
$$

下面的定理给出了充分性证明的第一步，即如果处理器在一个区间上没有空闲时间那么需求在该区间上可满足。

定理 3.1

$$Cond \wedge \sum_{i\in\alpha} \int Run_i = \ell \ \Rightarrow \ Req$$

Proof:

$$
\begin{array}{lll}
(1) & \exists i_0 \in \alpha \Diamond_p \neg Req_{i_0} \wedge Cond \wedge \sum_{i\in\alpha} \int Run_i = \ell & \\
& \Rightarrow \exists i_0 \exists m_{i_0} \in \mathcal{N} \left(\begin{array}{c} \ell = m_{i_0} T_{i_0} \wedge \ \ Cond \\ \wedge \ \int Run_{i_0} < m_{i_0} C_{i_0} \wedge [\![\bigvee_{i\in\alpha} \mathrm{Run}_i]\!] \\ \wedge \ \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{array} \right)^{\frown} \mathbf{true} & (\text{引理 } 3.4) \\
& \Rightarrow \exists i_0 \exists m_{i_0} \in \mathcal{N} \left(\begin{array}{c} \ell = m_{i_0} T_{i_0} \wedge \ \int Run_{i_0} < m_{i_0} C_{i_0} \\ \wedge \left(\begin{array}{l} (\ell = 0 \\ \wedge(\bigwedge_{i\in\alpha} \int \mathrm{Run}_i = \lceil \ell/T_i \rceil \cdot C_i)) \\ \frown [\![\bigvee_{i\in\alpha} \mathrm{Run}_i]\!] \end{array} \right) \\ \wedge Cond \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{array} \right)^{\frown} \mathbf{true} & (\mathrm{IL8},\ P_2) \\
& \Rightarrow \exists i_0 (\bigwedge_{i\in\alpha} \Box_p \mathbf{Req}_i \wedge \int Run_{i_0} < m_{i_0} C_{i_0}) & (\text{引理 } 3.5) \\
& \Rightarrow \mathbf{false} & \\
(2) & Cond \wedge \sum_{i\in\alpha} \int Run_i = \ell \ \Rightarrow \ Req & ((1))
\end{array}
$$

现在我们给出关于充分性证明的第二步，即给定一个区间，如果需求在它的所有真前缀子区间上是可满足的，但在它自己上却不可满足，那么处理器在它上面没有空闲时间。

定理 3.2

$$(\int Run_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \wedge Cond) \;\Rightarrow\; \sum_{i\in\alpha} \int Run_i = \ell$$

证明:

$$\begin{array}{lll}
(1) & \left(\begin{array}{c} Cond \wedge (\mathbf{true}^\frown[\![\bigwedge_{i\in\alpha} \neg \mathrm{Run}_i]\!]) \\ \wedge \int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{array}\right) & \\
& \Rightarrow \int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \wedge \bigwedge_{i\in\alpha} \int \mathrm{Run}_i = \lceil \ell/T_i \rceil \cdot C_i & (\text{引理 } 3.3) \\
& \Rightarrow \mathbf{false} & \\
(2) & \left(\begin{array}{c} Cond \wedge (\mathbf{true}^\frown[\![\bigwedge_{i\in\alpha} \neg \mathrm{Run}_i]\!]^\frown[\![\bigvee_{i\in\alpha} \mathrm{Run}_i]\!]) \\ \wedge \int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{array}\right) & \\
& \Rightarrow \left(\begin{array}{c} Cond \wedge \int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \\ \wedge((\bigwedge_{i\in\alpha} \int \mathrm{Run}_i = \lceil \ell/T_i \rceil \cdot C_i)^\frown[\![\bigvee_{i\in\alpha} \mathrm{Run}_i]\!]) \end{array}\right) & (\text{引理 } 3.3, \mathrm{ILT8}) \\
& \Rightarrow (\int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \wedge Req) & (\text{引理 } 3.5) \\
& \Rightarrow \mathbf{true}^\frown\mathbf{false} & \\
& \Rightarrow \mathbf{false} & \\
(3) & \left(\begin{array}{c} Cond \wedge \sum_{i\in\alpha} \int \mathrm{Run}_i \neq \ell \\ \wedge \int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{array}\right) & \\
& \Rightarrow \left(\begin{array}{c} Cond \\ \wedge(\mathbf{true}^\frown[\![\bigwedge_{i\in\alpha} \neg \mathrm{Run}_i]\!]^\frown([\![\bigvee_{i\in\alpha} \mathrm{Run}_i]\!] \vee [\![]\!])) \\ \wedge \int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{array}\right) & (\mathrm{DCT9}) \\
& \Rightarrow \mathbf{false} & ((1),(2)) \\
(4) & \left(\begin{array}{c} Cond \wedge \int \mathrm{Run}_{i_0} < \lfloor \ell/T_{i_0} \rfloor \cdot C_{i_0} \\ \wedge \bigwedge_{i\in\alpha} \Box_b \mathbf{Req}_i \end{array}\right) \Rightarrow \sum_{i\in\alpha} \int \mathrm{Run}_i = \ell & (3)
\end{array}$$

下面我们给出充分性的证明。我们的方法如下：假设需求在某个区间上不可满足，那么存在一个它的前缀子区间使得需求在该前缀子区间的所有真前缀子区间上都可满足，但是在该前缀子区间上不可满足。但这与我们上面两步的结论矛盾，从而充分性得证。

定理 3.3

$$Cond \quad \Rightarrow \quad Req$$

Proof:

(1) $Cond \wedge \exists i_0 \in \alpha \Diamond_p \neg Req_{i_0}$

$$\Rightarrow \exists j_0 \in \alpha \exists m_{j_0} \in \mathcal{N} \left(\begin{array}{l} Cond \wedge \ell = m_{j_0} T_{j_0} \\ \wedge \int Run_{j_0} < \lfloor \ell / T_{j_0} \rfloor \cdot C_{j_0} \\ \wedge \bigwedge_{i \in \alpha} \Box_b \mathbf{Req}_i \end{array} \right)^\frown \mathbf{true} \quad (\text{引理 } 3.4)$$

$$\Rightarrow \exists j_0 \in \alpha \exists m_{j_0} \in \mathcal{N} \left(\begin{array}{l} Cond \wedge \ell = m_{j_0} T_{j_0} \\ \wedge \int Run_{j_0} < \lfloor \ell / T_{j_0} \rfloor \cdot C_{j_0} \\ \wedge \sum_{i \in \alpha} \int Run_i = \ell \end{array} \right)^\frown \mathbf{true} \quad (\text{定理 } 3.2, P_2)$$

$$\Rightarrow \exists j_0 \in \alpha \exists m_{j_0} \in \mathcal{N} \left(\begin{array}{l} \int Run_{j_0} < \lfloor \ell / T_{j_0} \rfloor \cdot C_{j_0} \\ \wedge \bigwedge_{i \in \alpha} \Box_p \mathbf{Req}_i \end{array} \right)^\frown \mathbf{true} \quad (\text{定理 } 3.1, P_2)$$

$\Rightarrow$ **false**

(2) $Cond \Rightarrow Req$ ((1))

第 四 章 高阶时段演算 (HDC)

在这部分，我们将建立高阶时段演算理论，包括它的语法，语义和证明系统。因为在实际应用中，人们需要用实数作为时间，因此，我们在本章讨论语义时选择实数作为时间。同时，我们也取实数集合作为时段域。当然，我们也可以给出 HDC 的更一般的语义，例如，在下一章我们将给出它在抽象时间域上的语义。为了处理实时程序语义，我们在 HDC 中引进程序变量，并把它解释成时间域上的函数。在 HDC 中，由程序变量，全局变量和常量通过实算术运算符构成的表达式称为状态项。谓词作用到状态项生成的表达式是状态，由状态通过布尔运算符构成的表达式亦是状态。因此，在 HDC 中，状态是有内部结构的。例如，$V \geq 2$ 和 $x = y$ 等。

4.1 高阶时段演算的语法和语义

变量:

- **全局变量**, $x, y, z, x_1, y_1, z_1, \ldots$ 。全局变量被解释成实数 (R) 。全体全局变量的集合记作 $GVar$ 。
- **程序变量**, $V, V_1, V_2, \ldots$ 。它们被解释成时间域上的有穷可变的实函数 ($\mathsf{R} \rightarrow_{fv} \mathsf{R}$) 。这里，我们用实数来表示时间。这里所说的有穷可变性是指任意一个程序变量在任意一个有穷时间区间内，其值不能改变无穷多次。因而程序变量在任意一个有穷区间上都是一个仅有有穷多个不连续点的阶梯函数。全体程序变量的集合记作 $PVar$ 。
- **函数符号**, $f_i^n,\ i = 1, \ldots,\ n \geq 0$ 。当 $n = 0$ 时， f_i^n 表示一个常量; 当 $n > 0$ 时, f_i^n 被解释成实数 $\mathbf{R}$ 上的 n- 元实算术函数，且为全函数。所有这样的函数的集合记作 $FSymb$ 。它们都是刚性符号，即它们的解释是与时间和区间无关。
- **谓词符号**, $R_i^n,\ i = 1, \ldots,\ n \geq 0$ 。当 $n = 0$ 时， R_i^n 表示一个布尔常量，即为 **true** 或者 **false**; 当 $n > 0$ 时， R_i^n 被解释成实数 $\mathbf{R}$ 上的 n- 元谓词。所有这样的谓词符号的集合记作 $RSymb$ 。它们都是刚性符号，即它们的解释是与时间和区间无关。
- **长度变量**, ℓ 。它是一个从区间到实数的一个函数 ($\mathbf{Intv} \rightarrow \mathsf{R}$) 。它表示区间的长度。
- **命题符合**, $X, X_1, X_2, \ldots$ 。它们可以代表任意公式, 被解释成区间上的布尔函数 ($\mathbf{Intv} \rightarrow \{\mathbf{tt}, \mathbf{ff}\}$) 。

状态项:

状态项 ϑ 是由全局变量和程序变量及实算术函数符号构成的特殊项，其定义如下：

$$\vartheta ::= x \mid V \mid f(\vartheta_1, \ldots, \vartheta_n)$$

它被解释成时间域上的实函数。

状态：

谓词作用到状态项得到的表达式是状态，由状态通过逻辑联结词组合成的表达式仍旧是状态。它们被解释成时间域上的布尔函数 $(\mathsf{R} \to \{0,1\})$。例如，$(V = x)$ 和 $(V > 3)$ 均是状态。由于我们假设程序变量是有穷可变的，很明显状态也是有穷可变的。这与在一阶时段演算 [13] 中假设状态具有有穷可变性是一致的。下面是状态的语法定义。

$$S ::= 0 \mid 1 \mid R(\vartheta_1, \ldots, \vartheta_n) \mid \neg S \mid S_1 \vee S_2$$

项：

我们首先引进两个特殊项 — 状态项的初值和终值，$\overleftarrow{\vartheta}, \overrightarrow{\vartheta}, \overleftarrow{\vartheta_1}, \overrightarrow{\vartheta_1}, \ldots$。其中 $\leftarrow$ 和 $\rightarrow$ 是两个把状态项映射到 $(\mathbf{Intv} \to \mathsf{R})$ 的函数。给定一个解释 $\mathcal{I}$ 和一个区间 $[b,e]$，$\mathcal{I}(\overleftarrow{V})[b,e] = d$ $(\mathcal{I}(\overrightarrow{V})[b,e] = d)$ 当且仅当存在 $\delta > 0$ 使得 $\mathcal{I}(V)$ 在区间 $[b-\delta, b]$ 上几乎处处等于 d(相应地，$\mathcal{I}(V)$ 在区间 $[e, e+\delta]$ 上几乎处处等于 d)。同理，$(\overleftarrow{x} = d)$ 为真指在给定区间的一个左邻区间内，$(x = d)$ 为真，也就是 x 的赋值为 d。类似地，$(V_1 \overrightarrow{+} V_2 = d)$ 为真指在给定区间的一个右邻区间内，$(V_1 + V_2 = d)$ 几乎处处为真。

这样，项可以定义为：

$$\theta ::= x \mid \ell \mid \overleftarrow{\vartheta} \mid \overrightarrow{\vartheta} \mid \textstyle\int S \mid f(\theta_1, \ldots, \theta_n)$$

其中，$\int S$ 表示状态表达式 S 的时段。也就是，给定一个区间 $[b,e]$ 和一个状态表达式 S 的解释，S，$\int S$ 在区间 $[b,e]$ 上的值定义为：

$$\textstyle\int_b^e \mathsf{S}(t)\,dt$$

其中 $\mathsf{S}(t) \in \{0,1\}$。项被解释成区间上的实函数。

公式：

公式是由命题符号，谓词符号和项通过下述语法构造而成的：

$$\phi ::= X \mid R(\theta_1, \ldots, \theta_n) \mid \neg\phi \mid \phi_1 \vee \phi_2 \mid \phi_1 \frown \phi_2 \mid \exists x.\phi \mid \exists V.\phi$$

给定了命题符号，程序变量和全局变量的解释后，公式被解释成区间上的布尔函数 $(\mathbf{Intv} \to \{\mathbf{tt}, \mathbf{ff}\})$。在给定一个解释 $\mathcal{I}$ 和一个区间 $[b,e]$ 下，$\phi \frown \psi$ 为真的充要条件是存在一个 m $(b \le m \le e)$ 使得 ϕ 在解释 $\mathcal{I}$ 下，在区间 $[b,m]$ 上为真，且 ψ 在解释 $\mathcal{I}$ 下，在区间 $[m,e]$ 上为真。$\exists x$ 和 $\exists V$ 的语义和传统的一样，我们可以通过分别改变在解释中 x 和 V 的值来定义，可以参见第二章中的类似定义。

一个公式称为 **刚性的**，如果它不含有 ℓ，程序变量和命题符号。也就是说，它的值仅依赖于对刚性符号的解释，而不会随着时间区间的变化而改变。类似地，我们可以定义 **刚性的** 状态项，项和状态。当状态表达式 S 是刚性的，那么根据我们的语法定义，它也是一个公式。例如，根据语法定义，$(x + y > 1)$ 既可以看作状态，又可以看作公式。为了避免混乱，当 S 是刚性时，我们用 ϕ_S 表示它对应的刚性公式 S。

4.2 证明系统

我们的证明系统包含四组公理和规则。

4.2.1 关于区间时序逻辑的公理和规则

我们的证明系统仍旧包含一阶区间时序逻辑 [21, 20] 的公理和推理规则。一阶区间时序逻辑的公理包括：

(IL1) $\ell \geq 0$

(IL2) $((\phi^\frown\psi) \wedge \neg(\phi^\frown\varphi)) \Rightarrow (\phi^\frown(\psi \wedge \neg\varphi))$
$((\phi^\frown\psi) \wedge \neg(\varphi^\frown\psi)) \Rightarrow ((\phi \wedge \neg\varphi)^\frown\psi))$

(IL3) $((\phi^\frown\psi)^\frown\varphi) \Leftrightarrow (\phi^\frown(\psi^\frown\varphi))$

(IL4) $(\phi^\frown\psi) \Rightarrow \phi$ 若 ϕ 是刚性的
$(\phi^\frown\psi) \Rightarrow \psi$ 若 ψ 是刚性的

(IL5) $((\exists x.\phi)^\frown\psi) \Rightarrow \exists x.\ \phi^\frown\psi$ 若 x 在 ψ 中不是自由出现
$(\phi^\frown\exists x.\psi) \Rightarrow \exists x.\ \phi^\frown\psi$ 若 x 在 ϕ 中不是自由出现

(IL6) $((\ell = x)^\frown\phi) \Rightarrow \neg((\ell = x)^\frown\neg\phi)$
$(\phi^\frown(\ell = x)) \Rightarrow \neg(\neg\phi^\frown(\ell = x))$

(IL7) $(x \geq 0 \wedge y \geq 0) \Rightarrow ((\ell = x + y) \Leftrightarrow ((\ell = x)^\frown(\ell = y)))$

(IL8) $\phi \Rightarrow (\phi^\frown(\ell = 0))$
$\phi \Rightarrow ((\ell = 0)^\frown\phi)$

区间时序逻辑的推理规则包括：

(N)： 如果 ϕ 那么 $\neg((\neg\phi)^\frown\psi)$
如果 ϕ 那么 $\neg(\psi^\frown(\neg\phi))$

(M)： 如果 $\phi \Rightarrow \psi$ 那么 $(\phi^\frown\varphi) \Rightarrow (\psi^\frown\varphi)$
如果 $\phi \Rightarrow \psi$ 那么 $(\varphi^\frown\phi) \Rightarrow (\varphi^\frown\psi)$

区间时序逻辑的证明系统也包括命题逻辑和谓词逻辑及实算术的公理和推理规则。例如

$(\mathrm{G_x})$： 如果 ϕ 那么$\forall x.\phi$

然而，为了确保下面的公理在我们的模型中是永真的，我们须给出一个附加条件。

$$(\mathrm{Q_x}) : (\forall x.\phi(x)) \Rightarrow \phi(\theta) \begin{cases} \text{要么 } x \text{ 在 } \phi(x) \text{ 中相对于 } \theta \text{ 是自由的，且 } \theta \text{ 是刚性的；} \\ \text{要么 } x \text{ 在 } \phi(x) \text{ 中相对于 } \theta \text{ 是自由的，} \\ \text{且 } \phi(x) \text{ 中不含有切割算子。} \end{cases}$$

我们在这儿就不逐一列出实算术中用来处理等词，加运算等的公理和推理规则，可参见 [36] 。

4.2.2 关于时段的公理和规则

下面是一组关于时段的公理。

(DC1) $\int 0 = 0$

(DC2) $\int 1 = \ell$

(DC3) $\int S \geq 0$

(DC4) $\int S_1 + \int S_2 = \int(S_1 \vee S_2) + \int(S_1 \wedge S_2)$

(DC5) $((\int S = x_1)^\frown(\int S = x_2)) \Rightarrow (\int S = x_1 + x_2)$

(DC6) $\int S_1 = \int S_2$, 如果 $S_1 \Leftrightarrow S_2$ 成立

(DC7) $[\![S]\!] \Leftrightarrow (\phi_S \wedge \ell > 0)$, 如果 S 是刚性的

公理 DC1-6 指明了如何计算时段的值和推导与时段有关的性质，可参见 [13] 。公理 DC7 指出了刚性状态和它所对应的刚性公式之间的关系。公理 DC1-3 和 DC6-7 的永真性是明显的。由于积分的可加性，所以公理 DC4 和 DC5 也是永真的。

下面我们证明几个将在后面用到的关于 $[\![S]\!]$ 的简单性质。

DCT11 设 S 是一个状态表达式且 $V_1, \ldots, V_n$ 是所有在它里面出现的程序变量。那么

$$[\![V_1 = x_1]\!] \wedge \ldots \wedge [\![V_n = x_n]\!] \Rightarrow (\int S(V_1, \ldots, V_n) = \int S[x_1/V_1, \ldots, x_n/V_n])$$

证明:

(1) $V_1 = x_1 \wedge \ldots \wedge V_n = x_n$
$\Rightarrow (S(V_1, \ldots, V_n) \Leftrightarrow R[x_1/V_1, \ldots, x_n/V_n])$ (Ident)

(2) $[\![V_1 = x_1]\!] \wedge \ldots \wedge [\![V_n = x_n]\!]$
$\Rightarrow [\![S(V_1, \ldots, V_n) \Leftrightarrow S[x_1/V_1, \ldots, x_n/V_n]]\!]$ ((1), DCT1)

(3) $[\![S(V_1, \ldots, V_n) \Leftrightarrow S[x_1/V_1, \ldots, x_n/V_n]]\!]$
$\Rightarrow \begin{pmatrix} [\![S(V_1, \ldots, V_n) \vee \neg S[x_1/V_1, \ldots, x_n/V_n]]\!] \\ \wedge [\![\neg S(V_1, \ldots, V_n) \vee S[x_1/V_1, \ldots, x_n/V_n]]\!] \end{pmatrix}$ (DCT1)

(4) $\int S[x_1/V_1, \ldots, x_n/V_n] = 0 \vee \int S[x_1/V_1, \ldots, x_n/V_n] = \ell$ (DC7, DCT1)

(5) $[\![V_1 = x_1]\!] \wedge \ldots \wedge [\![V_n = x_n]\!]$
$\Rightarrow \int S(V_1, \ldots, V_n) = \int S[x_1/V_1, \ldots, x_n/V_n]$ ((2), (3), (4))

□

4.2.3 关于程序变量的公理和推理规则

为了把程序变量定义成有穷可变函数，我们引进两条归纳规则。设 $H(X)$ 是一个包含命题符号 X 的公式。

(DCR1)

若 $H([\![\,]\!])$ **且** $H(X) \Rightarrow H(X \vee (X^\frown \exists x.[\![V = x]\!]))$
那么 $H(true)$

(DCR2)

若 $H(\lceil\!\rceil)$ **且** $H(X) \Rightarrow H(X \vee ((\exists x.\lceil V = x\rceil)^\frown X))$
那么 $H(true)$

$\lceil S\rceil$ 指 S 在任意一个非点区间上几乎处处为真。因此，$\lceil V = x\rceil$ 指程序变量 V 在非点区间上几乎处处等于 x 。因而，上述两个规则正好说明了任意有穷区间均存在一个有穷分割，使得程序变量 V 在分割中的每个子区间上是个常量。这正好表达了程序变量的有穷可变性。

注意: 到目前为止，我们讨论的语义均是把时间看成实数，时段域也取实数集合。就象在引言中谈到的那样，这样我们不能得到一个绝对完备的高阶时段演算的证明系统，我们仅能证明它的相对完备性，就象 [34] 一样。上述两条规则足够保证我们的系统在 [34] 意义上的相对完备性，见 [57] 。[21] 已经证明在抽象时间域上我们不能用有穷规则来公理化有穷可变性，因此如果要讨论 HDC 在抽象时间域上的完备性，上述两条规则是不能公理化程序变量的有穷可变性的。为了在抽象时间域上公理化有穷可变性，[29] 引进了 ω - 规则。因此，在下部分我们研究 HDC 在抽象时间域上的完备性时，我们引进了 ω - 规则来公理化程序变量的有穷可变性。同时，我们也可以证明 ω - 规则要比上述两条归纳规则要强，因此下面所有基于 DCR1 和 DCR2 的结论，当我们用 ω - 规则代替 DCR1 和 DCR2 后，仍旧成立。

设

$$H(X) \;\widehat{=}\; \Box(X \;\Rightarrow\; \lceil\!\rceil \vee \exists x.\lceil V = x\rceil^\frown true)$$

使用归纳规则，易证

$$\Box(\lceil\!\rceil \;\vee\; \exists x.\; \lceil V = x\rceil^\frown true)$$

类似地，也易证

$$\Box(\lceil\!\rceil \;\vee\; \exists x.\; true^\frown\lceil V = x\rceil)$$

这两个公式说明程序变量 V 在任意非点区间的开始一段和结尾一段保持常值。这也是程序有穷可变性的另一种表示形式。

注意: 使用 DCR1 和 DCR2 我们可以证明第二章中关于状态有穷可变性的两条归纳规则 IR1 和 IR2 。并且我们也可以给出 DCR1 和 DCR2 的类似定理 2.2 的另一种表示形式。

为了处理状态项 ϑ 的初值和终值，即 $\overleftarrow{\vartheta}$ 和 $\overrightarrow{\vartheta}$, 我们建立下面几条公理。

(PV1) $\quad (\ell > 0)^\frown((\overleftarrow{\vartheta} = x_1) \wedge (\ell = x_2)) \;\Leftrightarrow\; true^\frown\lceil\vartheta = x_1\rceil^\frown(\ell = x_2)$

(PV2) $\quad ((\overrightarrow{\vartheta} = x_1) \wedge (\ell = x_2))^\frown(\ell > 0) \;\Leftrightarrow\; (\ell = x_2)^\frown\lceil\vartheta = x_1\rceil^\frown true$

PV1 和 PV2 指出了状态项的初值和终值的含义就是从前一个时间段接收一个值，并把计算结果传递给下一个时间段。因为函数 $\leftarrow$ ($\rightarrow$) 涉及到状态项在当前区间的左邻区间 (右邻区间) 的值，为了能够推导与它们有关的性质，引进邻接规则 [18] 是必要的。

邻接规则

如果 $(\ell = a)^\frown\Psi^\frown(\ell = b) \Rightarrow (\ell = a)^\frown\Upsilon^\frown(\ell = b)$, 那么 $\Psi \Rightarrow \Upsilon$。 $(a, b \geq 0)$

使用邻接规则, 我们可以证明下面几个有用的定理。

定理 4.1

(I) $(\overleftarrow{V} = x)^\frown(\ell = y) \Leftrightarrow (\overleftarrow{V} = x) \wedge (\ell \geq y)$

(II) $(\ell = y)^\frown(\overrightarrow{V} = x) \Leftrightarrow (\overrightarrow{V} = x) \wedge (\ell \geq y)$

(III) $(\overleftarrow{\vartheta} = \vartheta) \wedge (\overrightarrow{\vartheta} = \vartheta)$ 如果 ϑ 是刚性的

(IV) $(f(\vartheta_1, \overleftarrow{\ldots}, \vartheta_n) = f(\overleftarrow{\vartheta_1}, \ldots, \overleftarrow{\vartheta_n}))$ 其中 $f \in Fsymb$

(V) $(f(\vartheta_1, \overrightarrow{\ldots}, \vartheta_n) = f(\overrightarrow{\vartheta_1}, \ldots, \overrightarrow{\vartheta_n}))$ 其中 $f \in Fsymb$

证明: 下面我们给出 (I) 的证明, 其它可以类似地证明。

(1)		$(\ell = a)^\frown(\overleftarrow{V} = x)^\frown(\ell = y)$	
	$\Leftrightarrow$	$\exists b.(\ell = a)^\frown(\overleftarrow{V} = x \wedge \ell = b)^\frown(\ell = y)$	(ILT2)
	$\Leftrightarrow$	$\exists b.((\ell = y)^\frown(\overleftarrow{V} = x \wedge \ell = b))^\frown(\ell = y)$	(IL3)
	$\Leftrightarrow$	$\exists b.((\ell = a \wedge \mathbf{true}^\frown[\![V = x]\!])^\frown(\ell = b))^\frown(\ell = y)$	(PV1)
	$\Leftrightarrow$	$\exists b.(\ell = a \wedge \mathbf{true}^\frown[\![V = x]\!])^\frown(\ell = b + y)$	(IL3, IL7)
	$\Leftrightarrow$	$\exists b.(\ell = a)^\frown(\overleftarrow{V} = x \wedge \ell = b + y)$	(PV1)
	$\Leftrightarrow$	$(\ell = a)^\frown(\overleftarrow{V} = x \wedge \ell \geq y)$	(PL)
(2)		$\overleftarrow{V} = x^\frown(\ell = y) \Leftrightarrow (\overleftarrow{V} = x \wedge \ell \geq y)$	((1), 邻接规则)

在这个定理中, (I) 和 (II) 说明一个区间和它的前缀 (后缀) 子区间享有相同的左邻区间 (右邻区间) 。 (III), (IV) 和 (V) 告诉如何计算复合状态项的初值和终值。

公式一般是用来描述程序变量和它们的初值及终值的性质的。假设 ϕ 中含有 V, $\overleftarrow{V}$ 和 $\overrightarrow{V}$, 而 ψ 仅含有 V, 而不包含 $\overleftarrow{V}$ 和 $\overrightarrow{V}$ 。那么我们可以证明

定理 4.2

(I) $([\![V_1 = V_2]\!] \vee [\![\ ]\!]) \wedge (\overleftarrow{V_1} = \overleftarrow{V_2}) \wedge (\overrightarrow{V_1} = \overrightarrow{V_2}) \Rightarrow \phi(V_1) \Leftrightarrow \phi(V_2)$

(II) $[\![V_1 = V_2]\!] \vee [\![\ ]\!] \Rightarrow \psi(V_1) \Leftrightarrow \psi(V_2)$

证明: 我们仅给出证明的一个概要, 而省略掉证明的具体细节。

我们先证

$$([\![V_1 = V_2]\!] \vee [\![\ ]\!]) \wedge (\overleftarrow{V_1} = \overleftarrow{V_2}) \wedge (\overrightarrow{V_1} = \overrightarrow{V_2}) \Rightarrow \Box(([\![V_1 = V_2]\!] \vee [\![\ ]\!]) \wedge (\overleftarrow{V_1} = \overleftarrow{V_2}) \wedge (\overrightarrow{V_1} = \overrightarrow{V_2})) \qquad (\star)$$

由 DCT6, 我们可以证明 $\Box(\lceil V_1 = V_2 \rceil \vee \lceil\!\lceil\ \rceil\!\rceil)$ 。用 PV1 和定理 4.1(I), 我们可以证明 $\Box(\overleftarrow{V_1}{=}\overleftarrow{V_2})$ 。类似地, 用 PV2 和定理 4.1 (II), 我们可以证明 $\Box(\overrightarrow{V_1}{=}\overrightarrow{V_2})$ 。

用定理 4.1, 归纳于项 $\theta(V)$ 的结构, 我们可以证明

$$(\lceil V_1 = V_2 \rceil \vee \lceil\!\lceil\ \rceil\!\rceil) \wedge (\overleftarrow{V_1}{=}\overleftarrow{V_2}) \wedge (\overrightarrow{V_1}{=}\overrightarrow{V_2}) \ \Rightarrow\ \theta(V_1) = \theta(V_2)$$

然后根据 $(\star)$, 再归纳于公式 ϕ 的结构, 我们即可以证明 (I) 。

(II) 可以类似地证明。

4.2.4 关于高阶量词的公理和规则

最后我们给出关于程序变量的量词的公理和规则。它们刻画了 V , $\overleftarrow{V}$ 和 $\overrightarrow{V}$ 在高阶量词环境中的语义。

下面的公理和规则与二阶逻辑中关于高阶量词的公理和规则的形式是一样的。

$\mathrm{G_V}$: **如果** ϕ **那么** $\forall V.\phi$
$\mathrm{Q_V}$: $(\forall V.\phi(V)) \ \Rightarrow\ \phi(\vartheta)$

下面两条公理是说如果 $\overleftarrow{V}$ $(\overrightarrow{V})$ 不在公式 ϕ 中出现, 那么它可以取任意值。原因在于 $\overleftarrow{V}$ $(\overrightarrow{V})$ 的值是由 V 在当前区间的外面的值决定的。

(HDC1) $\quad \exists V.\phi \ \Rightarrow\ \exists V.\ \phi \wedge (\overleftarrow{V}{=}\, x)$ 如果 $\overleftarrow{V} \notin \phi$

(HDC2) $\quad \exists V.\phi \ \Rightarrow\ \exists V.\ \phi \wedge (\overrightarrow{V}{=}\, x)$ 如果 $\overrightarrow{V} \notin \phi$

$\exists V$ 相对于切割算子的可分布性是把程序变量看成是时间域上的函数的本质属性。 $\exists V$ 相对于切割算子是可分布的充要条件是在切割算子左侧出现的 $\overrightarrow{V}$ 的值和在切割算子右侧出现的 V 的值要一致。对称地, 在切割算子右侧出现的 $\overleftarrow{V}$ 的值和在切割算子左侧出现的 V 的值要一致。

(HDC3)

$$\left(\begin{array}{ll} & (\exists V.\ \phi \wedge (true^\frown\lceil V = x_1 \rceil \vee \lceil\!\lceil\ \rceil\!\rceil) \wedge (\overrightarrow{V}{=}\, x_2)) \\ ^\frown & (\exists V.\ \psi \wedge (\lceil V = x_2 \rceil^\frown \mathbf{true} \vee \lceil\!\lceil\ \rceil\!\rceil) \wedge (\overleftarrow{V}{=}\, x_1)) \end{array}\right) \ \Rightarrow\ \exists V.\ \phi^\frown\psi$$

当 $\overrightarrow{V} \notin \phi$ 且 $\overleftarrow{V} \notin \psi$, 由上述公理, 我们可以得到下述结论

$$(\exists V.\phi)^\frown\exists V.\psi \ \Rightarrow\ \exists V.\ \phi^\frown\psi$$

因为 V 被解释成时间域上的函数, ϕ 和 ψ 是关于程序变量 V 的谓词的积分的公式, 且它们都不考虑 V 在单独一个点上的值, 因而上述公式的正确性是显而易见的。因此我们可以把在两个相连的子区间上的两个函数合并为一个在整个区间上的函数, 而不考虑新函数在这两个相连子区间公共点上的值。

关于 $\exists V$ 相对于切割算子的可分布性, 我们有下面一个定理

定理 4.3

(I) $(\exists V.\phi)^\frown\psi \Leftrightarrow \exists V.(\phi^\frown\psi)$ 若 $V \notin \psi$
(II) $\phi^\frown\exists V.\psi \Leftrightarrow \exists V.(\phi^\frown\psi)$ 若 $V \notin \phi$

证明: 我们仅证 **(I), (II)** 可以类似地证明。

(1) $\exists V.(\phi^\frown\psi)$
$\Rightarrow \exists V.((\exists V.\phi)^\frown(\exists V.\psi))$ (**M**)
$\Rightarrow (\exists V.\phi)^\frown\psi$ (**PL**)

(2) $(\exists V.\phi)^\frown\psi$
$\Rightarrow (\exists V.(\phi \wedge (\ell > 0 \vee \ell = 0) \wedge \overleftarrow{V}{=}\overleftarrow{V})^\frown\psi$ (**IL1, PL**)
$\Rightarrow (\exists V, x1, x2.(\phi \wedge (\mathbf{true}^\frown[\![V = x_1]\!] \vee [\![\ ]\!]) \wedge \overleftarrow{V} = x_2))^\frown\psi$ (**M, PL,** $\mathrm{G_x}$)
$\Rightarrow \exists x_1, x_2.((\exists V.(\phi \wedge (\mathbf{true}^\frown[\![V = x_1]\!] \vee [\![\ ]\!]) \wedge \overleftarrow{V} = x_2))^\frown\psi)$ (**IL5**)
$\Rightarrow \exists x_1, x_2.\left(\begin{array}{c} \exists V.(\phi \wedge (\mathbf{true}^\frown[\![V = x_1]\!] \vee [\![\ ]\!]) \wedge \overleftarrow{V} = x_2) \\ {}^\frown \quad \exists V.(\psi \wedge ([\![V = x_2]\!]^\frown\mathbf{true} \vee [\![\ ]\!]) \wedge \overrightarrow{V} = x_1) \end{array}\right)$ ($\mathrm{G_V}$, **HDC1**)
$\Rightarrow \exists x_1, x_2.(\exists V.(\phi^\frown\psi))$ (**HDC3**)
$\Rightarrow \exists V.(\phi^\frown\psi)$ (**PL**)

(3) $(\exists V.\phi)^\frown\psi \Leftrightarrow \exists V.(\phi^\frown\psi)$ ((1), (2))

第 五 章
高阶时段演算在抽象时间域上的完备性

5.1 主要思想

区间时序逻辑和时段演算的完备性不仅依赖于时间域的选择, 而且依赖于量词化那类变量。在实际应用中, 我们需要选择实数作为时间。但是, 如果这样, 根据 **Gödel** 不完备性定理, 我们不可能给出关于它们的完备的证明系统。因此, 若我们选择实数作为时间, 那么我们仅能得到这些系统的相对完备性。例如, [**34**] 已经证明若把所有关于实数和区间时序逻辑的永真公式当作时段演算的公理, 那么时段演算是完备的。如果我们仅考虑量词化全局变量, 那么时段演算在抽象时间域上是完备的 [**29**] 。因为我们把程序变量解释成从时间域到时段域的函数, 因此若我们一旦引进关于程序变量的量词, 而且该量词的作用域为从时间域到时段域的全体函数, 那么我们不可能给出 **HDC** 的完备的证明系统。原因在于, 一旦我们把程序变量的量词作用域解释为从时间域到时段域的全体函数, 那么 **HDC** 就具有了二阶算术的表达能力。因此, 为了给出一个完备的 **HDC** 的证明系统, 对程序变量加些限制是必要的。因而我们假设所有程序变量均具有有穷可变性, 从而由程序变量作参数产生的状态也具有有穷可变性。这与在时段演算中假设所有状态变量都具有有穷可变性是一致的。如果这样, 我们可以通过将 **HDC** 翻译到一阶区间时序逻辑中去的方法来证明它在抽象时间域上的完备性。下面简要解释一下我们的方法:

一个容易想到的将二阶逻辑归结到一阶逻辑中去的方法是对所有 n 元谓词 $H^n(x_1,...,x_n)$ 引进一个新的 $n+1$ 元谓词 $E^{n+1}(z,x_1,...,x_n)$ 去枚举所有 n 元谓词 $H^n(x_1,...,x_n)$ 。这样,

$$\exists H^n.\phi$$

可以变为

$$\exists z.\phi[E^{n+1}(z,x_1,...,x_n)/H^n(x_1,...,x_n)]$$

因而二阶逻辑可以归结到一阶逻辑中去。关于这种方法的详细讨论, 可参见 [**23**] 。然而, 为了定义这个用来枚举所有 n 元谓词的 $(n+1)$ 元谓词 $E^{n+1}(z,x_1,\ldots,x_n)$, 我们必须有下面的假设。为了讨论的方便, 我们设 $n=1$ 。首先,

$$\exists z.E(z,x_1)$$

和

$$\exists z.\neg E(z, x_1)$$

假设对于单元论域, E 能够枚举所有 H 。而且, 它们和下面公式

$$\exists z.\ (x_1 \neq x_2) \Rightarrow (E(z, x_1) \Leftrightarrow E(z_1, x_1) \wedge E(z, x_2) \Leftrightarrow E(z_2, x_2))$$

一起规定了在有穷论域中 E 能够枚举所有 H 。不幸地是我们不能在无穷论域上用这种方法来定义一个 E 去枚举所有的 H, 因为如果这样, E 的第一个参数的值域的势不能小于 $2^{\aleph_0}$, 那么相应的一阶逻辑所对应的语言必然是不可数的。因此, 一般而言, 用这种方法不能把二阶逻辑翻译到一阶逻辑中去。

然而, 因为我们假设了程序变量具有有穷可变性, 因此, 任给一个区间, 任意一个程序变量在该区间上的值可以由它在该区间的一个有穷分割上的值组合而成, 而且它在该分割中的每一个子区间上的值是一个常量。因此, 我们可以构造一个一元的柔性函数 $g(y)$, 使它满足下述假设

$$[\![\ ]\!] \ \vee \ \exists y.[\![g(y) = c]\!] \quad \text{对任意常量 } c$$

和

$$[\![\ ]\!] \ \vee \ \exists y.\ [\![g(y) \Leftrightarrow g(y_1)]\!] \frown [\![g(y) \Leftrightarrow g(y_2)]\!]$$

这样, $g(y)$ 可以用来枚举所有程序变量。

用这种方法, $\exists V.\phi$ 可以归约成 $\exists y_V.\phi'$, 其中 ϕ' 是 ϕ 在一阶区间时序逻辑中所对应的公式。这样, 我们可以建立一个完备的高阶时段演算。这个想法是受 [27] 启发。

为了证明 **HDC** 的完备性, 我们首先建立一个多种类的区间时序逻辑; 用 [21, 29] 中的方法, 我们可以证明它在抽象时间域上是完备的; 然后, 我们将 **HDC** 翻译到这个多种类的区间时序逻辑中去。因为我们可以证明 **HDC** 的一个公式集合 Γ 在 **HDC** 中是协调的充要条件是 $\Gamma' \cup Axiom_{hdc}$ 在这个多种类的区间时序逻辑中是协调的。其中, Γ' 和 $Axiom_{hdc}$ 分别代表 Γ 和 **HDC** 的所有公理的实例集合翻译到这个多种类的区间时序逻辑中的对应部分; 由这个区间时序逻辑的完备性, 我们可以得到它的一个模型 $< \mathcal{F}, \mathcal{J} >$ 使得它满足 $\Gamma' \cup Axiom_{hdc}$ 。根据 $< \mathcal{F}, \mathcal{J} >$, 我们可以构造一个 **HDC** 的模型 $< \mathcal{F}', \mathcal{J} >$ 使得它满足 Γ 。这样, 我们就证明了 **HDC** 在抽象时间域上是完备的。

5.2 一阶多种类区间时序逻辑 (IL_2)

在这里, 我们首先建立一个完备的带两个种类的区间时序逻辑 (以下简称 IL_2)。

5.2.1 IL_2 的语法

IL_2 的语法和 **ITL** 的语法基本一样。区别在于, 在 IL_2 里, 全局变量, 常量和函数分属于两个不同的种类。我们用 x_i, $i \geq 0$ 表示第一类全局变量, 所有第一类全局变量的集合记

作 Var^1; 我们用 $y_i, i \geq 0$ 表示第二类全局变量, Var^2 表示第二类全局变量的全体; 我们用 $c_i, i \geq 0$ 表示第一类常量; 我们用 $d_i, i \geq 0$ 表示第二类全局变量; 我们用 $f_i^n, i, n \geq 0$ 表示第一类 n- 元函数, $Fsymb^1$ 表示第一类函数全体; 我们用 $g_i^n, i, n \geq 0$ 表示第二类 n- 元函数, $Fsymb^2$ 表示第二类函数的全体。我们用 ϑ 表示第二类项, 它要么是第二类全局变量, 要么是第二类常量。

5.2.2 IL_2 的抽象语义

下面, 我们给出 IL_2 在抽象时间域上的语义。这里, 和 **ITL** 在实数时间域上的语义稍微不同的是, 我们将函数解释成区间上的函数。即函数符号可能是“柔性符号 ”。当然, 后前可以看成前者的一种特殊情况, 即所有函数均为区间上的常函数。这主要是为了后面证明的需要。

定义 5.1 一个时间域 ***(time domain)*** 是一个线性序 $< T, \leq >$ 。

定义 5.2 给定一个时间域 $< T, \leq >$, 由它构成的区间 ***(interval)*** 的集合为 $\mathbf{Intv}(T) = \{[t_1, t_2] \mid t_1, t_2 \in T \text{ 且} t_1 \leq t_2\}$, 其中, $[t_1, t_2] = \{t \mid t \in T \text{ 且} t_1 \leq t \leq t_2\}$ 。

定义 5.3 一个时段域 ***(duration domain)*** 是一个类型为 $< D, +, 0 >$ 的系统, 且它必须满足下述公理:

(D1) $a + (b + c) = (a + b) + c$
(D2) $a + 0 = a = 0 + a$
(D3) $a + b = a + c \Rightarrow b = c, \quad a + c = b + c \Rightarrow a = b$
(D4) $a + b = 0 \Rightarrow a = 0 = b$
(D5) $\exists c.a + c = b \vee b + c = a, \exists c.c + a = b \vee c + b = a$

也就是说, $< D, +, 0 >$ 是一个全序可交换群。

定义 5.4 给定一个时间域 $< T, \leq >$ 和一个时段域 $< D, +, 0 >$, 我们称一个从 T 到 D 的函数 m 是一个测度 ***(measure)***, 若它满足下述条件:

(M1) $m([t_1, t_2]) = m([t_1, t_2']) \Rightarrow t_2 = t_2'$
(M2) $m([t_1, t]) + m([t, t_2]) = m([t_1, t_2])$
(M3) $m([t_1, t_2]) = a + b \Rightarrow \exists t.m([t_1, t]) = a \wedge (t_1 \leq t \leq t_2)$

定义 5.5 一个 $\mathbf{IL}_2$ 的语义框架是一个四元组 $<< T, \leq >, < D, +, 0 >, D_1, m >$, 其中, $< T, \leq >$ 是时间域, $< D, +, 0 >$ 是时段域, D_1 称为内附论域 ***(inhabited domain)***, m 是一个测度。

定义 5.6 一个 $\mathbf{IL}_2$ 的模型是一个五元组 $<< T, \leq >, < D, +, 0 >, D_1, m, \mathcal{J} >$, 其中, $<< T, \leq >, < D, +, 0 >, D_1, m >$ 一个语义框架, $\mathcal{J}$ 是一个解释, 它满足下述条件:

(I) 对任意 $X \in PLetter$, $\mathcal{J}(X):\ \mathbf{Intv}(T) \to \{0,1\}$;

(II) 对任意 $v \in TVar$, $\mathcal{J}(v):\ \mathbf{Intv}(T) \to D$;

(III) 对所有 $R_i^n \in RSymb$ $\mathcal{J}(R_i^n):\ D^n \to \{0,1\}$;

(IV) 对所有 $f_i^n \in FSymb^1$, $\mathcal{J}(f_i^n): D^n \times \mathbf{Intv}(T) \to D$; 对所有 $g_i^n \in FSymb^2$, $\mathcal{J}(g_i^n): D_1^n \times \mathbf{Intv}(T) \to D$;

(V) 对任意 $x \in Var^1$, $\mathcal{J}(x) \in D$, 且对任意 $y \in Var^2$, $\mathcal{J}(y) \in D_1$ 。

(VI) $\mathcal{J}(0) = 0, \mathcal{J}(+) = +, \mathcal{J}(=)$ 就是 $=$, 而 $\mathcal{J}(\ell) = m$ 。

定义 5.7 设 $\mathcal{J}$ 和 $\mathcal{J}'$ 是满足上述条件的两个解释。如果对除 z 外的所有符号, 在这两个解释下拥有相同的值, 则我们说 $\mathcal{J}$ z- 等价于 $\mathcal{J}'$,

给定一个 $\boldsymbol{IL}_2$ 的模型 $\mathcal{M} =<< T, \leq>, < D, +, 0 >, D_1, m, \mathcal{J} >$ 和区间 $[t_1, t_2]$, $\mathcal{J}_{t_1}^{t_2}(\theta)$ 表示项 θ 在模型 $\mathcal{M}$ 下在区间 $[t_1, t_2]$ 上的值。其定义如下:

$$
\begin{array}{ll}
\mathcal{J}_{t_1}^{t_2}(x) = \mathcal{J}(x) & \text{其中} x \in Var^1 \\
\mathcal{J}_{t_1}^{t_2}(y) = \mathcal{J}(y) & \text{其中} y \in Var^2 \\
\mathcal{J}_{t_1}^{t_2}(v) = \mathcal{J}(v)([t_1, t_2]) & \text{其中} v \in TVar \\
\mathcal{J}_{t_1}^{t_2}(f_i^n(\theta_1, \ldots, \theta_n) = \mathcal{J}(f_i^n)([t_1, t_2], \mathcal{J}_{t_1}^{t_2}(\theta) \ldots \mathcal{J}_{t_1}^{t_2}(\theta_n)) & \text{其中} f_i^n \in FSymb^1 \\
\mathcal{J}_{t_1}^{t_2}(g_i^n(\vartheta_1, \ldots, \vartheta_n) = \mathcal{J}(g_i^n)([t_1, t_2], \mathcal{J}_{t_1}^{t_2}(\vartheta) \ldots \mathcal{J}_{t_1}^{t_2}(\vartheta_n)) & \text{其中} g_i^n \in FSymb^2
\end{array}
$$

公式 ϕ 在模型 $\mathcal{M} =< \mathcal{F}, \mathcal{J} >$ 下的解释可由下面几条规则给出, 其中 $\mathcal{F} =<< T, \leq>, < D, +, 0 >, D_1, m >$ 。

1. $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} X$
当且仅当 $\mathcal{J}(X)([t_1, t_2]) = tt$;

2. $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} R^n(\theta_1, \ldots, \theta_n)$
当且仅当 $\mathcal{J}(R^n)(\mathcal{J}_{t_1}^{t_2}(\theta_1), \ldots, \mathcal{J}_{t_1}^{t_2}(\theta_n)) = tt$;

3. $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} \neg\phi$
当且仅当 $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \not\models_{\boldsymbol{IL}_2} \phi$;

4. $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi \vee \psi$
当且仅当 $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi$ 或者 $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} \psi$;

5. $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi^\frown\psi$
当且仅当存在 $t \in [t_1, t_2]$, 使得 $< \mathcal{F}, \mathcal{J} >, [t_1, t] \models_{\boldsymbol{IL}_2} \phi$ 且 $< \mathcal{F}, \mathcal{J} >, [t, t_2] \models_{\boldsymbol{IL}_2} \psi$;

6. $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\boldsymbol{IL}_2} \exists z.\phi$
当且仅当存在一个 z- 等价于 $\mathcal{J}$ 的解释 $\mathcal{J}'$, 使得 $< \mathcal{F}, \mathcal{J}' >, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi$

给定一个模型 $\mathcal{M} =<< T, \leq>, < D, +0 >, D_1, m, \mathcal{J} >$ 和一个公式 ϕ，如果存在一个区间 $[t_1, t_2] \in \mathbf{Intv}(T)$ 使得 $\mathcal{M}, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi$，则说 $\mathcal{M}$ **满足** ϕ 。我们可以立即将这个概念推广到一类模型上去： ϕ 在一类模型 $\mathcal{C}$ 上是 **可满足的**，如果它在 $\mathcal{C}$ 中的某个模型上是可满足的。给定一个公式集合 Γ，我们说 $\mathcal{M}$ 是 Γ 的一个模型或者说 $\mathcal{M}$ 满足 Γ，如果存在一个区间 $[t_1, t_2] \in \mathbf{Intv}(T)$，使得对 Γ 中的任意公式 ϕ，$\mathcal{M}, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi$ 。我们说公式 ϕ 在模型 $\mathcal{M}$ 上是 **永真的**，当且仅当对于任意区间 $[t_1, t_2] \in \mathbf{Intv}(T)$，$\mathcal{M}, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi$ 。我们说公式 ψ 在一类模型 $\mathcal{C}$ 上是 **永真的**，如果它在 $\mathcal{C}$ 的每个成员上均是永真的。如果 ϕ 在所有模型构成的类上是永真的，则称 ϕ 是 **永真的** 。也就是说，对于任意模型 $\mathcal{M} =<< T, \leq>, < D, +, 0 >, D_1, m, \mathcal{J} >$ 和任意区间 $[t_1, t_2] \in \mathbf{Intv}(T)$，$\mathcal{M}, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi$ 。记作 $\models_{\boldsymbol{IL}_2} \phi$ 。如果存在一个模型 $\mathcal{M} =<< T, \leq>, < D, +, 0 >, D_1, m, \mathcal{J} >$ 和一个区间 $[t_1, t_2] \in \mathbf{Intv}(T)$，使得 $\mathcal{M}, [t_1, t_2] \models_{\boldsymbol{IL}_2} \phi$，则称 ϕ **可满足** 。

定义 5.8 设 Φ 是 $\mathbf{IL}_2$ 的一个公式， $\mathcal{M} =< \mathcal{F}, \mathcal{J} >$ 是 $\mathbf{IL}_2$ 的模型，其中 $\mathcal{F} =<< T, \leq>, < D, +, 0 >, D_1, m >$ 是它对应的语义框架。如果对任意区间 $[t_1, t_2] \in \mathbf{I}\boldsymbol{ntv}(T)$，$\mathcal{M}, [t_1, t_2] \models_{\mathbf{IL}_2} \Phi \Rightarrow \Box\Phi$，且存在一个序列 $t'_1, \ldots, t'_n$，使得 $t_1 = t'_1 \leq \ldots \leq t'_n = t_2$ 且对所有 $i = 1, \ldots, n-1$ $\mathcal{M}, [t'_i, t'_{i+1} \models_{\mathbf{IL}_2} \Box\Phi$，则称 Φ 在模型 $\mathcal{M}$ 上有 **有穷可变性** 。如果 Φ 在一类模型 $\mathcal{C}$ 中的任意成员上都有有穷可变性，则称 Φ 在 $\mathcal{C}$ 上有 **有穷可变性** 。

定义 5.9 设 Φ 是 $\mathbf{IL}_2$ 的一个公式。我们定义公式序列 $\{\Phi^k\}_{k<\omega}$ 如下：

$$\Phi^0 \ \widehat{=}\ \ell = 0, \quad \Phi^{k+1} \ \widehat{=}\ (\Phi^k \frown \Box\Phi)$$

在本章的余下部分，我们固定一个 $\boldsymbol{IL}_2$ 的公式集合 Ω，并且我们仅考虑对任意 $\Phi \in \Omega$，Φ 在它上面有有穷可变性的 $\boldsymbol{IL}_2$ 的模型，我们记所有这样模型构成的类为 $\mathcal{C}_{\mathcal{M}}(\Omega)$ 。当然，我们下面的所有讨论也适用于任意 $\boldsymbol{IL}_2$ 的公式集合 Ω 。例如，当 $\Omega = \emptyset$，我们的证明和 [21] 中的证明是相同的。因此，我们可以认为 [21] 中的证明是我们的一种特殊情况，即对应于 $\Omega = \emptyset$ 。我们说公式 Φ 具有有穷可变性是指对任意区间，我们可以把它分割成有穷多个毗连子区间，使得 $\Box\Phi$ 在每个子区间上成立。

5.2.3 $\boldsymbol{IL}_2$ 的证明系统

[21] 证明在抽象时间域上我们不能够用有穷规则来公理化有穷可变性。为了在 **ITL** 中公理化有穷可变性， [29] 引进了 ω - 规则。因此，在 $\boldsymbol{IL}_2$ 的证明系统除了包含 **ITL** 的证明系统外，还增加下面两条公理和规则：

Ω **公理：**

$$\Phi \Rightarrow \Box\Phi \qquad \text{对任意} \Phi \in \Omega$$

ω **- 规则：**

$$\mathrm{IR}^{\Phi} \quad \frac{H(\Phi^0/X) \quad \forall k < \omega . H(\Phi^k/X) \Rightarrow H(\Phi^{k+1}/X)}{H(\mathbf{true}/X)} \qquad \text{其中} \Phi \in \Omega$$

为了证明 IL_2 的证明系统相对于 $\mathcal{C}_{\mathcal{M}}(\Omega)$ 是 **可靠的**，我们首先证明几个关于有穷可变性的性质。[29] 已经给出了它们的证明，但为了论文的完整性，我们这里仍旧给出了它们的具体证明。

性质 5.1 令 Φ 在模型 $\mathcal{M} =< \mathcal{F}, \mathcal{J} >$ 上具有有穷可变性。其中，$\mathcal{F} =<< T, \leq >, < D, +, 0 >, D_1, m >$ 是它对应的语义框架。那么，对任意 $[t_1, t_2] \in \mathbf{Intv}(T)$，存在 $k_0 < \omega$ 使得对所有 $k \geq k_0$ 有 $\mathcal{M}, [t_1, t_2] \models_{\mathbf{IL}_2} \Box\Phi^k$ 。

证明: 设 $[t_1, t_2] \in \mathbf{Intv}(T)$ 。因为 Φ 在 $\mathcal{M}$ 上具有有穷可变性，所以存在一个序列 $t_1', \ldots, t_n'$ 使得 $t_1 = t_1' \leq \ldots \leq t_n' = t_2$ 且对所有 $i = 1, \ldots, n-1$ 有 $\mathcal{M}, [t_i', t_{i+1}'] \models_{IL_2} \Box\Phi$ 。根据 **ILT11** 知对所有 $k \geq n$ 有 $\mathcal{M}, [t_1, t_2] \models_{IL_2} \Phi^k$ 。从而命题得证。 □

推论 5.1 设 Φ 在模型 $\mathcal{M} =< \mathcal{F}, \mathcal{J} >$ 上有有穷可变性。那么对任意 $[t_1, t_2] \in \mathbf{Intv}\ (T)$，存在 $k_0 < \omega$ 使得 $\mathcal{M}, [t_1, t_2] \models_{\mathbf{IL}_2} \Box(\Phi^k \Leftrightarrow \mathbf{true})$ 对所有 $k \geq k_0$ 。

引理 5.1 如 $\Phi \in \Omega$，那么 $\vdash_{\mathbf{IL}_2} \Phi^k \Rightarrow \Box\Phi^k$ 。

证明: 由 Ω 公理和 **ILT12** 立得。 □

引理 5.2 设 $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\mathbf{IL}_2} \Box(\Phi \Leftrightarrow \psi)$，$H$ 是一个包含命题符号 X 的 $\mathbf{IL}_2$ 公式。那么 $< \mathcal{F}, \mathcal{J} >, [t_1, t_2] \models_{\mathbf{IL}_2} \Box(H(\Phi/X) \Leftrightarrow H(\psi/X))$ 。

证明: 归纳于 H 的结构可得。 □

定理 5.1 $\mathbf{IL}_2$ 的证明系统相对于 $\mathcal{C}_m(\Omega)$ 是可靠的。

证明: 我们仅需要证明对于 $\Phi \in \Omega$，规则 IR^{Φ} 保持永真性即可。其它情况非常简单。

设 $\mathcal{M} =< \mathcal{F}, \mathcal{J} > \in \mathcal{C}_{\mathcal{M}}(\Omega)$，$\mathcal{M}, [t_1, t_2] \models_{IL_2} H(\Phi^0/X)$，且对任意 $k < \omega$ 有 $\mathcal{M}, [t_1, t_2] \models_{IL_2} H(\Phi^k/X) \Rightarrow H(\Phi^{k+1}/X)$ 。因此，对任意 $k < \omega$，$\mathcal{M}, [t_1, t_2] \models_{IL_2} H(\Phi^k/X)$ 。根据推论 **5.1**，存在 $k_0 < \omega$ 使得 $\mathcal{M}, [t_1, t_2] \models_{IL_2} \Box(\Phi^{k_0} \Leftrightarrow \mathbf{true})$ 。由引理 **5.2**，可得 $\mathcal{M}, [t_1, t_2] \models_{IL_2} \Box(H(\Phi^{k_0}/X) \Leftrightarrow H(\mathbf{true}/X))$ 。因此，$\mathcal{M}, [t_1, t_2] \models_{IL_2} H(\mathbf{true}/X)$ 。 □

5.3 IL_2 在抽象论域上的完备性

在这部分，我们将证明 IL_2 的完备性。我们的证明非常类似于 [21, 29] 中的证明。相同或仅仅细枝末节不同于 [21, 29] 中的技术结论，我们将分别标明。我们的证明分成两部分：第一部分主要处理极大协调理论及它的相关性质；第二部分构造 IL_2 的经典模型。

5.3.1 预备知识

定义 5.10 如果 Γ 是 $\mathbf{IL}_2$ 的一个公式集合，且它包含所有 $\mathbf{IL}_2$ 的定理，并且它在 $\mathbf{IL}_2$ 的证明系统里封闭，则称 Γ 是 $\mathbf{IL}_2$ 的一个理论。

定义 5.11 一个理论 Γ 称为 **协调的** ***(consistent)***，如果 $\mathbf{false} \notin \Gamma$ 。否则，称为 **不协调的** ***(inconsistent)*** 。设 Γ 是 $\mathbf{IL}_2$ 的一个公式集合。如果 $\Gamma \nvdash_{\mathbf{IL}_2} \mathbf{false}$，则称 Γ 是协调的。也就是说， Γ 是某个协调理论的子集。

定义 5.12 设 Γ 是一个理论。如果 Γ 是协调的且不是任何协调理论的真子集，则说 Γ 是极大协调理论 。

定义 5.13 设 Γ 是一个理论。如果 $\exists z\phi \in \Gamma$ 蕴涵存在一个常量 f 且 f 和 z 具有相同的种类，使得 $\phi(f/z) \in \Gamma$ 。则称 Γ 是 ***Henkin*** 理论。

定义 5.14 设 Γ 是 $\mathbf{IL}_2$ 的一个公式集合。我们用 $Cn(\Gamma)$ 表示 $\{\ \phi \mid \Gamma \vdash_{\mathbf{IL}_2} \phi\}$，用 $Cn(\Gamma + \phi)$ 表示 $Cn(\Gamma \cup \{\phi\})$ 。

引理 5.3

$$\Gamma + \phi \ \widehat{=}\ \{\psi \mid \phi \Rightarrow \psi \in Cn(\Gamma)\} = Cn(\Gamma + \phi)$$

证明: 因为 $Cn(\Gamma \cup \{\phi\})$ 关于 **MP** 规则封闭，所以 $\Gamma + \phi \subset Cn(\Gamma \cup \{\phi\})$ 。下面我们关于 ψ 的证明进行归纳证明 $Cn(\Gamma \cup \{\phi\}) \subset \Gamma + \phi$. **1)** 若 $\psi \in Cn(\Gamma) \cup \{\phi\}$，则因为 $\psi \Rightarrow (\phi \Rightarrow \psi) \in Cn(\Gamma)$，因而 $\phi \Rightarrow \psi \in Cn(\Gamma)$; **2)** 若 ψ 由 **MP** 规则由 $\chi \Rightarrow \psi, \chi \in Cn(\Gamma \cup \{\phi\})$ 得到。由归纳假设和逻辑可得 $\phi \Rightarrow (\chi \Rightarrow \psi), \phi \Rightarrow \chi, (\phi \Rightarrow (\chi \Rightarrow \psi)) \Rightarrow ((\phi \Rightarrow \chi) \Rightarrow (\phi \Rightarrow \psi)) \in Cn(\Gamma)$，而且 $Cn(\Gamma)$ 关于 **MP** 封闭，所以 $\phi \Rightarrow \psi \in \Gamma$ 。从而 $\psi \in \Gamma + \phi$ 。 **3)** 若 $\psi = H(\mathbf{true}/X) \in Cn(\Gamma \cup \{\phi\})$ 由 IR^{Φ} 从 $H(\Phi^k/X) \in Cn(\Gamma \cup \{\phi\}), k < \omega$ 得到，其中 $\Phi \in \Omega$ 。由归纳假设知对任意 $k < \omega$ 有 $\phi \Rightarrow H(\Phi^k/X) \in Cn(\Gamma)$ 。因为 $Cn(\Gamma)$ 在 IR^{Φ} 下封闭，所以 $\phi \Rightarrow H(\mathbf{true}/X) \in Cn(\Gamma)$ 。因而 $\psi = H(\mathbf{true}/X) \in \Gamma + \phi$ 。 □

从现在开始，我们固定一个 $\boldsymbol{IL}_2$ 的语言 $\mathcal{L}$ 和两个可数刚性常量符号集合 C_1, C_2，使得所有 $C_1 \cup C_2$ 的符号不在 $\mathcal{L}$ 中出现，且 $C_1 \cap C_2 = \emptyset$ 。由 $\mathcal{L}$，C_1 和 C_2 中的符号构成 $\boldsymbol{IL}_2$ 的一个语言，我们用 $\mathcal{L}(C_1 + C_2)$ 表示。显然， $\mathcal{L}(C_1 + C_2)$ 也是可数的。

定理 5.2 设 $\Gamma_0 \subset \mathcal{L}$ 是协调的。那么存在一个极大协调 ***Henkin*** 理论 $\Gamma \in \mathcal{L}(C_1 + C_2)$ 使得 $\Gamma_0 \subset \Gamma$ 。

证明: 如果 Γ_0 是一个理论，令 $\Gamma_0' = \Gamma_0$; 否则，令 $\Gamma_0' = \{\phi \mid \Gamma_0 \vdash_{\boldsymbol{IL}_2} \phi\}$ 。明显地， $\Gamma_0' \subset \mathcal{L}$ 且是一个理论。当然， $\Gamma_0 \subset \Gamma_0'$ 。令 $\Omega = \{\Phi_k \mid k < \omega\}$ 和 $\mathcal{L}(C_1 + C_2) = \{\psi_k \mid k < \omega\}$ 。设 $\pi_1, \pi_2 : \mathcal{N} \to \mathcal{N}$ 分别是配对函数 $\pi : \mathcal{N} \times \mathcal{N} \to \mathcal{N}$ 的左，右函数。因此，对任意 $k_1, k_2 < \omega$，存在一个 k 使得 $k = \pi(k_1, k_2)$ 且 $\pi_1(k) = k_1$, $\pi_2(k) = k_2$ 。现在，我们定义理论序列 $\Gamma_0 \subseteq \ldots \subseteq \Gamma_k \subseteq \ldots$ 如下： Γ_0 就是 Γ_0' 。假设 Γ_k 已经定义。令 $k_1 = \pi_1(k)$ 和 $k_2 = \pi_2(k)$ 。下面，我们将考虑如何定义 Γ_{k+1}:

1. $\Gamma_k + \psi_{k_1}$ 不是协调的。那么 $\Gamma_{k+1} = \Gamma_k$ 。

2. $\Gamma_k + \psi_{k_1}$ 是协调的。

2.1. $\psi_{k_1} = \exists z.\psi$, 其中 z 是第 j 种类的全局变量, ψ 是另外一个公式, $j = 1, 2$ 。我们选择一个常量 $f \in C_j$ 使得 f 不在 $\Gamma_k \cup \{\psi_{\pi_1(n)} \mid n < k\}$ 中的公式中出现。我们令 $\Gamma_{k+1} = \Gamma_k + \psi_{k_1} + \psi(f/z)$ 。

2.2. $\psi_{k_1} = \neg H(\mathbf{true}/X)$, 其中 H 是包含命题符号 X 的公式。那么至多有 $2^m - 1$ 个不同的公式 H 满足上述等式, 其中 m 是 **true** 在 ψ_{k_1} 中出现的次数。令满足上述等式的不同公式为 $H_1, \ldots, H_p$ 。那么存在 $n_1, \ldots, n_p < \omega$ 使得 $\Gamma_k + \psi_{k_1} + \neg H_1(\Phi_{k_2}^{n_1}/X) + \ldots + \neg H_p(\Phi_{k_2}^{n_p}/X)$ 是协调的。否则, 必存在 $q \leq p$ 使得对所有 $i < \omega$ $H_q(\Phi_{k_2}^{i}/X) \in \Gamma_k + \psi_{k_1}$ 。因而, 有 $IR^{\Phi_{k_2}}$, 我们有 $H_q(\mathbf{true}/X) \in \Gamma_k + \psi_{k_1}$ 。而这与 $\Gamma_k + \psi_{k_1}$ 的协调性矛盾。因此, 对任意 $q = 1, \ldots, p$, 我们可以找到一个 n_q 使得它满足上述性质。令 $\Gamma_{k+1} = \Gamma_k + \psi_{k_1} + \neg H_1(\Phi_{k_2}^{n_1}/X) + \ldots + \neg H_p(\Phi_{k_2}^{n_p})$ 。

2.3. 其它, 令 $\Gamma_{k+1} = \Gamma_k + \psi_{k_1}$ 。

根据定义 **5.14**, 对于任意 $k \leq \omega$, Γ_k 是协调的。下面, 我们将证明 $\Gamma = \bigcup_{k<\omega}\Gamma_k$ 是一个极大协调理论。因为对所有 $k < \omega$, Γ_k 关于规则 MP 是封闭的, 所以 Γ 关于 MP 也封闭。假设存在 $\Phi \in \Omega$, Γ 在 IR^{Φ} 下不封闭。令 $\Phi = \Phi_{k_2}$, 其中 $k_2 < \omega$, 且对任意 $k < \omega$, $H(\Phi_{k_2}^{k}/X) \in \Gamma$, 但是 $H(\mathbf{true}/X) \notin \Gamma$ 。令 $H(\mathbf{true}/X) = \psi_{k_1} \notin \Gamma$, 其中 $k_1 < \omega$ 。令 $k = \pi(k_1, k_2)$ 。那么 $H(\mathbf{true}/X) \notin \Gamma_{k+1} \subset \Gamma$ 。因而 $\Gamma_k + H(\mathbf{true}/X)$ 不是协调的。令 $\neg H(\mathbf{true}/X) = \psi_{k_1'}$, $k' = \pi(k_1', k_2)$ 。那么 $\Gamma_{k'} + \neg H(\mathbf{true}/X)$ 是协调的。因为, 否则, $\Gamma_{max(k,k')} + (\psi_{k_1} \vee \psi_{k_1'})$ 将是不协调的, 且 $\psi_{k_1} \vee \psi_{k_1'}$ 是一个永真公式。根据情况 **2.2.**, 那么存在一个 $n < \omega$ 使得 $\neg H(\Phi^n/X) \in \Gamma_{k'+1}$ 。这与假设 $\mathbf{false} \notin \Gamma$ 矛盾, 因为对所有 $k < \omega$, $\mathbf{false} \notin \Gamma_k$ 。因此, Γ 是协调的。

假设 $\Gamma' \supset \Gamma$ 也是协调的, 且 $\psi \in \Gamma'/\Gamma$ 。那么, 存在 $k < \omega$ 使得 $\psi = \psi_{k_1}$ 。很明显地, $\psi \notin \Gamma_{k+1}$ 。因此, $\psi + \Gamma_k$ 不是协调的。由于 $\Gamma_k \subset \Gamma \subset \Gamma'$, 所以 Γ' 不是协调的。这与假设矛盾。因而 Γ 是极大协调的。

Γ_{k+1} 的构造可以确保 Γ 也是 **Henkin** 理论。 □

引理 5.4 设 Γ 是 $\mathcal{L}(C_1+C_2)$ 的一个极大协调理论, 且它在 IR^{Φ} 下封闭。那么存在 $k < \omega$ 使得 $\Phi^k \in \Gamma$ 。

证明: ([29]) 令 $H \mathrel{\widehat{=}} \neg X$ 。因为 $\neg H(\mathbf{true}/X) \Leftrightarrow \neg\neg\mathbf{true}$, 则我们有 $\vdash_{\boldsymbol{IL}_2} \neg H(\mathbf{true}/X)$ 。因为 Γ 是协调的且在 IR^{Φ} 下封闭, 所以存在 $k < \omega$ 使得 $H(\Phi^k/X) \notin \Gamma$, 否则, $H(\mathbf{true}/X) \in \Gamma$ 。这与 Γ 的协调性矛盾。那么, 对这个 k, 我们有 $\neg H(\Phi^k/X) \in \Gamma$ 。因此, $\Phi^k \in \Gamma$ 。 □

定义 5.15 设 $\Gamma \subset \mathcal{L}(C_1+C_2)$ 。我们用 $\Gamma^{\widehat{}}$ 表示 $\{\psi_1 \wedge \ldots \wedge \psi_n \mid n < \omega, \psi_1, \ldots, \psi_n \in \Gamma\}$ 。

定义 5.16 设 $\Gamma_1, \Gamma_2 \subset \mathcal{L}(C_1 + C_2)$ 。我们用 $\Gamma_1 \widehat{\ } \Gamma_2$ 表示 $\{(\phi \widehat{\ } \psi) \mid \phi \in \Gamma_1, \psi \in \Gamma_2\}$ 。

引理 5.5 设 $\Gamma \subset \mathcal{L}(C_1 + C_2)$ 。如果 Γ 在 MP 下封闭且 $\Phi^k \in \Gamma$, 那么 Γ 在 IR^Φ 下封闭当且仅当 Γ 在下面有穷规则下是封闭的

$$(IR_k^\Phi) \qquad \frac{\forall i \leq k H(\Phi^i/X)}{H(\mathbf{true}/X)}$$

证明: ([29]) 根据 **ILT11**, 如果 Γ 在 IR_k^Φ 下封闭, 那么 Γ 在 IR^Φ 下封闭。为了证明反方向, 令

$$H' \;\widehat{=}\; \left(\bigwedge_{i\leq k} H(\Phi^i/X)\right) \;\Rightarrow\; H$$

那么, 当 $i \leq k$ 时, $H'(\Phi^i/X)$ 是一个永真公式。此外, 当 $k' \geq k$ 时, 由 **ILT11** 知 $\Box(\Phi^{k'} \Leftrightarrow \Phi^k) \in \Gamma$ 。根据引理 **5.2**, 可得 $\Box(H'(\Phi^k/X) \Leftrightarrow H'(\Phi^{k'})) \in \Gamma$ 。因此对所有 $i < \omega$, $H'(\Phi^i/X) \in \Gamma$ 。因而, 由 IR^Φ 可得 $H'(\mathbf{true}/X) \in \Gamma$ 。这蕴涵了 Γ 在 IR_k^Φ 下是封闭的。 □

推论 5.2 设 $\Gamma \subset \mathcal{L}(C_1 + C_2)$ 且 $\Phi^k \in \Gamma$ 。那么如果 $\Gamma \vdash_{\mathbf{IL}_2} \phi$, 那么存在一个 Γ 的有穷子集 $\Gamma_0 \subseteq \Gamma$ 使得 $\Gamma_0 \vdash_{\mathbf{IL}_2} \phi$ 。

证明: 归纳于由 Γ 推导 ϕ 的证明过程。由引理 **5.5**, 我们可以将每次运用 IR^Φ 规则的地方用 IR_k^Φ 规则去替换。这样, 我们可以得到 ϕ 的一个证明, 在这个证明中, 仅有有穷多个前提在 Γ 中。 □

引理 5.6 设 Γ 是一个极大协调理论, $\Gamma_1, \Gamma_2 \neq \emptyset, \Gamma_1\hat{}\Gamma_2\hat{} \subset \Gamma$ 。那么 Γ_1 和 Γ_2 都是协调的且 $Cn(\Gamma_1)^\frown Cn(\Gamma_2) \subset \Gamma$ 。

证明: ([29]) 由引理 **5.4**, 存在 $k < \omega$ 使得 $\Phi^k \in \Gamma$ 。反设 $\Gamma_1 \cup \{\Phi^k\}$ 不是协调的。由推论 **5.2** 和引理 **5.3**, 存在 $\phi_1, \ldots, \phi_n \in \Gamma_1$ 使得 $\vdash_{\boldsymbol{IL}_2} \neg(\phi_1 \wedge \ldots \wedge \phi_n \wedge \Phi^k)$ 。根据规则 **N**, 可得对任意的公式 $\psi \in \Gamma_2$, $\vdash_{\boldsymbol{IL}_2} \neg((\phi_1 \wedge \ldots \wedge \phi_n \wedge \Phi^k)^\frown\psi)$ 且 $\neg(\neg\Phi^{k\frown}\psi) \in \Gamma$ 。由 **ILT4** 可得 $\vdash_{\boldsymbol{IL}_2} (\neg((\phi_1 \wedge \ldots \wedge \phi_n \wedge \Phi^k)^\frown\psi) \wedge \neg(\neg\Phi^{k\frown}\psi)) \Rightarrow \neg((\phi_1 \wedge \ldots \phi_n)^\frown\psi)$, 所以 $\neg((\phi_1 \wedge \ldots \phi_n)^\frown\psi) \in \Gamma$ 。这与 $\Gamma_1\hat{}\Gamma_2\hat{} \subset \Gamma$ 矛盾。因此 $\Gamma_1 \cup \{\Phi^k\}$ 是协调的, 相应地, Γ_1 也是协调的。同理可证 $\Gamma_2 \cup \{\Phi^k\}$ 和 Γ_2 是协调的。

现在可令 $\phi \in Cn(\Gamma_1)$, $\psi \in Cn(\Gamma_2)$ 。那么 $\phi \in Cn(\Gamma_1 \cup \{\Phi^k\})$ 且 $\psi \in Cn(\Gamma_2 \cup \{\Phi^k\})$ 。由推论 **5.2** 和引理 **5.3** 可得 $\vdash_{\boldsymbol{IL}_2} (\phi_1 \wedge \ldots \wedge \phi_n \wedge \Phi^k) \Rightarrow \phi$ 且 $\vdash_{\boldsymbol{IL}_2} (\psi_1 \wedge \ldots \wedge \psi_m \wedge \Phi^k) \Rightarrow \psi$, 其中 $\phi_1, \ldots, \phi_n \in \Gamma_1, \psi_1, \ldots, \psi_m \in \Gamma_2$ 。由单调规则 **M**, 我们有 $\vdash_{\boldsymbol{IL}_2} ((\phi_1 \wedge \ldots \wedge \phi_n \wedge \Phi^k)^\frown(\psi_1 \wedge \ldots \wedge \psi_m \wedge \Phi^k)) \Rightarrow (\phi^\frown\psi)$ 。因为 $\Gamma_1\hat{}\Gamma_1\hat{} \subset \Gamma$, 所以 $((\phi_1 \wedge \ldots \wedge \phi_n)^\frown(\psi_1 \wedge \ldots \wedge \psi_m)) \in \Gamma$ 。因为 Γ 是一个理论, 且 $\neg(\neg\Phi^{k\frown}\mathbf{true}), \neg(\mathbf{true}^\frown\neg\Phi^k) \in \Gamma$, 所以由公理 **IL2**, $((\phi_1 \wedge \ldots \wedge \phi_n \wedge \Phi^k)^\frown(\psi_1 \wedge \ldots \wedge \psi_m \wedge \Phi^k)) \in \Gamma$ 。从而 $(\phi^\frown\psi) \in \Gamma$ 。因此 $Cn(\Gamma_1)^\frown Cn(\Gamma_2) \subset \Gamma$ 。 □

类似于 [21], 我们定义两个作用在 $\boldsymbol{IL}_2$ 公式的集合配对上的函数 δ_1 和 δ_2 如下:

$$\delta_1(\Gamma, \Delta) = \{\psi \mid \neg(\phi^\frown\neg\psi) \in \Gamma, \phi \in \Delta\} \quad \delta_2(\Gamma, \Delta) = \{\phi \mid \neg(\neg\phi^\frown\psi) \in \Gamma, \psi \in \Delta\}$$

引理 5.7 设 $\Gamma \subset \mathcal{L}(C_1+C_2)$ 是极大协调理论，$\Gamma_1,\Gamma_2 \subset \mathcal{L}(C_1+C_2)$ 使得 $\Gamma_1\hat{}\frown\Gamma_2\hat{} \subset \Gamma$。令 $\Gamma_1' = \Gamma_1 \cup \delta_2(\Gamma,\Gamma_2)$，$\Gamma_2' = \Gamma_2 \cup \delta_1(\Gamma,\Gamma_1)$。那么 $(\Gamma_1')\hat{}\frown\Gamma_2\hat{} \subset \Gamma$ 且 $\Gamma_1\hat{}\frown(\Gamma_2')\hat{} \subset \Gamma$

证明: ([21]) 设 $\phi_1,\ldots\ \phi_n \in \Gamma_1', \psi_1,\ldots,\psi_l \in \Gamma_2$。如果 $\phi_1,\ldots,\phi_n \in \Gamma_1$，那么由假设 $((\phi_1 \wedge \ldots \wedge \phi_n)\frown(\psi_1 \wedge \ldots \wedge \psi_l)) \in \Gamma$。设 $\phi_1,\ldots\phi_m \in \delta_2(\Gamma,\Gamma_2), m \leq n$。那么存在 $\chi_1,\ldots,\chi_m \in \Gamma_2$ 使得对任意 $i = 1,\ldots,m$ 有 $\neg(\neg\phi_i\frown\chi_i) \in \Gamma$。设 $\psi \widehat{=} \psi_1 \wedge \ldots \wedge \psi_l \wedge \chi_1 \wedge \ldots \wedge \chi_m$。那么 $\psi \Rightarrow \chi_i$ 是永真公式。因此由单调规则 **M** 可得 $\vdash_{\boldsymbol{IL}_2}\ (\neg\phi_i\frown\psi) \Rightarrow (\neg\phi_i\frown\chi_i)$。因而对所有 $i = 1,\ldots,m$。$\vdash_{\boldsymbol{IL}_2}\ \neg(\neg\phi_i\frown\chi_i) \Rightarrow \neg(\neg\phi_i\frown\psi)$。因为 Γ 是极大协调的，因而在 MP 下封闭，所以对任意 $i = 1,\ldots,m$，我们有 $\neg(\neg\phi_i\frown\psi) \in \Gamma$。

如果 $m < n$，令 $\phi \widehat{=} \phi_{m+1} \wedge \ldots \wedge \phi_n$。否则，令 ϕ 为 Γ_1 中的任意公式。因为 $\phi \in \Gamma_1\hat{}$ 且 $\psi \in \Gamma_2\hat{}$，所以 $(\phi\frown\psi) \in \Gamma$。利用 $\vdash_{\boldsymbol{IL}_2}\ (\phi\frown\psi) \wedge \neg(\neg\phi_1\frown\psi) \wedge \ldots \wedge \neg(\neg\phi_m\frown\psi) \Rightarrow ((\phi \wedge \phi_1 \wedge \ldots \wedge \phi_m)\frown\psi)$，我们可以得到 $((\phi_1 \wedge \ldots \wedge \phi_n)\frown\psi) \in \Gamma$。因此 $((\phi_1 \wedge \ldots \wedge \phi_n)\frown(\psi_1 \wedge \ldots \wedge \psi_l)) \in \Gamma$。所以 $(\Gamma_1')\hat{}\frown\Gamma_2\hat{} \subset \Gamma$。

同理可证 $\Gamma_1\hat{}\frown(\Gamma_2')\hat{} \subset \Gamma$。 □

推论 5.3 设 $\Gamma \subset \mathcal{L}(C_1+C_2)$ 是极大协调理论，且 $\Gamma_1,\Gamma_2 \subset \mathcal{L}(C_1+C_2)$ 使得 $\Gamma_1\hat{}\frown\Gamma_2\hat{} \in \Gamma$。令 $\Gamma_1' = \Gamma_1 \cup \delta_2(\Gamma,\Gamma_2)$，$\Gamma_2' = \Gamma_2 \cup \delta_2(\Gamma,\Gamma_1)$。那么 $Cn(\Gamma_1')\frown Cn(\Gamma_2) \subset \Gamma$ 且 $Cn(\Gamma_1)\frown Cn(\Gamma_2') \subset \Gamma$。

证明: 由引理 **5.6** 和引理 **5.7** 立证。 □

定理 5.3 设 $\Gamma \subset \mathcal{L}(C_1+C_2)$ 是一个极大协调理论。已知 $\Gamma_1,\Gamma_2 \subset \mathcal{L}(C_1+C_2)$ 且 $\Gamma_1\hat{}\frown\Gamma_2\hat{} \subset \Gamma$。那么存在两个极大协调理论 $\Gamma_1^\star \supseteq \Gamma_1$ 和 $\Gamma_2^\star \supseteq \Gamma_2$ 使得 $\Gamma_1^\star\frown\Gamma_2^\star \subset \Gamma$。

证明: ([21]) 考虑下面两个递归构造的集合序列 $\Gamma_1^n,\Gamma_2^n, n < \omega$:

$$\begin{array}{ll}\Gamma_1^0 = Cn(\Gamma_1) & \Gamma_2^0 = Cn(\Gamma_2)\\ \Gamma_1^{k+1} = Cn(\Gamma_1^k \cup \delta_2(\Gamma,\Gamma_2^k)) & \Gamma_2^{k+1} = Cn(\Gamma_2^k \cup \delta_1(\Gamma,\Gamma_1^k))\end{array}$$

根据引理 **5.6** 和引理 **5.7**，对上标 n 进行归纳，我们可以得到对所有 $n < \omega$，$\Gamma_1^n\frown\Gamma_2^n \subset \Gamma$。令 $\Gamma_1^\omega \widehat{=} \bigcup_{n<\omega}\Gamma_1^n, \Gamma_2^\omega \widehat{=} \bigcup_{n<\omega}\Gamma_2^n$。设 $\phi_1,\ldots,\phi_m \in \Gamma_1^\omega$，$\psi_1,\ldots,\psi_l \in \Gamma_2^\omega$。那么存在一个 n 使得 $\phi_1,\ldots,\phi_m \in \Gamma_1^n$ 且 $\psi_1,\ldots,\psi_l \in \Gamma_2^n$。因而 $((\phi_1\wedge\ldots\wedge\phi_m)\frown(\psi_1\wedge\ldots\wedge\psi_l)) \in \Gamma$。因此 $(\Gamma_1^\omega)\hat{}\frown(\Gamma_2^\omega)\hat{} \subset \Gamma$。从而 Γ_1^ω 和 Γ_2^ω 是协调的。

由定理 **5.2**，存在一个极大协调理论 $\Gamma_1^\star \supseteq \Gamma_1^\omega$。令 $\psi \in \Gamma_2^\omega$，ϕ 是任意公式使得 $\neg(\neg\phi\frown\psi) \in \Gamma$。那么存在一个 $n < \omega$ 使得 $\psi \in \Gamma_2^n$ 且 $\phi \in \delta_2(\Gamma,\Gamma_2^n)$。因而 $\phi \in \Gamma_1^\omega \subseteq \Gamma_1^\star$。这表明 $\Gamma_1^\star\frown\Gamma_2^\omega \subset \Gamma$。因为 $\Gamma_1^\star$ 是极大协调的，所以 $(\Gamma_1^\star)\hat{} = \Gamma_1^\star$。此外，$\Gamma_2^\omega = (\Gamma_2^\omega)\hat{}$。这样，$(\Gamma_1^\star)\hat{}\frown(\Gamma_2^\omega)\hat{} \subset \Gamma$。令 $\Gamma_2' = \Gamma_2^\omega \cup \delta_1(\Gamma,\Gamma_1^\star)$。由引理 **5.7** 可得，$\Gamma_1^\star\frown\Gamma_2' \subset \Gamma$。根据引理 **5.6**，$\Gamma_2'$ 是协调的。根据定理 **5.2**，存在一个极大协调理论 $\Gamma_2^\star$ 使得 $\Gamma_2^\star \supseteq \Gamma_2'$。如果 $\phi \in \Gamma_1^\star$ 且 ψ 一个 $\boldsymbol{IL}_2$ 公式使得 $\neg(\phi,\neg\psi) \in \Gamma$，那么 $\psi \in \delta_1(\Gamma,\Gamma_1^\star) \subset \Gamma_2' \subseteq \Gamma_2^\star$。这证明了 $\Gamma_1^\star\frown\Gamma_2^\star \subset \Gamma$。 □

引理 5.8 已知 $\Gamma_1,\Gamma_2,\Gamma \subset \mathcal{L}(C_1+C_2)$ 是极大协调理论。$\ell = c_1 \in \Gamma_1$ 且 $\ell = c_2 \in \Gamma_2$，其中 $c_1,c_2 \in C_1$。$\Gamma_1\frown\Gamma_2 \subset \Gamma$。那么如果 ϕ 是一个刚性公式，则 $\phi \in \Gamma$ 当且仅当 $\phi \in \Gamma_1$ 当且仅当 $\phi \in \Gamma_2$。

证明: ([21]) 令 $\phi \in \Gamma$ 。那么, 因为 $\neg\phi$ 也是刚性的, 所以由规则 **R** 可得 $\vdash_{\boldsymbol{IL}_2} (\neg\phi^\frown\ell = c_2^1) \Rightarrow \neg\phi$ 。假设 $\phi \notin \Gamma_1$ 。那么 $\neg\phi \in \Gamma_1$, 因此 $(\neg\phi^\frown\ell = c_2^1) \in \Gamma$ 且 $\neg\phi \in \Gamma$ 。从而矛盾, 因而 $\phi \in \Gamma_1$ 。类似地可以证明 $\phi \in \Gamma_2$ 。反方向显然。 □

引理 5.9 设 $\Gamma \subset \mathcal{L}(C_1 + C_2)$ 是极大协调理论, $(\ell = c^\frown\mathbf{true}) \in \Gamma$ 其中 $c \in C_1$ 。那么 $\Delta_1 = \{\phi \mid (\ell = c^\frown\phi) \in \Gamma\}$ 也是一个极大协调理论。对称地, 如果 $(\mathbf{true}^\frown\ell = c) \in \Gamma$, 那么 $\Delta_2 = \{\phi \mid (\phi^\frown\ell = c) \in \Gamma\}$ 也是一个极大协调理论。

证明: 我们易证下面事实: 由 **ILT9** 知如果 $\phi_1, \ldots, \phi_n \in \Delta_1$, 那么 $\phi_1 \wedge \ldots \wedge \phi_n \in \Delta_1$ 。从而 $\{\ell = c\}^\frown\hat{\Delta_1} \subset \Gamma$ 。根据引理 **5.6** 可得 Δ_1 是协调的。下面, 我们仅需证明对任意公式 ϕ, 要么 $\phi \in \Delta_1$ 要么 $\neg\phi \in \Delta_1$ 。因为 $(\ell = c^\frown\mathbf{true}) \in \Gamma$, 所以 $(\ell = c^\frown\phi) \vee (\ell = c^\frown\neg\phi) \in \Gamma$ 。因为 Γ 是极大协调的, 所以要么 $(\ell = c^\frown\phi) \in \Gamma$ 要么 $(\ell = c^\frown\neg\phi) \in \Gamma$ 。从而要么 $\phi \in \Delta_1$ 要么 $\neg\phi \in \Delta_1$ 。对于 Δ_2 的证明可以类似地完成。 □

推论 5.4 设 Γ, Δ' 和 Δ'' 是 $\mathcal{L}(C_1 + C_2)$ 里的极大协调理论, 且存在 $c \in C_1$ 使得 $\{\ell = c\}^\frown\Delta'$, $\{\ell = c\}^\frown\Delta'' \subset \Gamma$ 。那么 $\Delta' = \Delta''$ 。类似地, 如果 $\Delta'^\frown\{\ell = c\}$, $\Delta''^\frown\{\ell = c\} \subset \Gamma$, 那么 $\Delta' = \Delta''$ 。

证明: 由引理 **5.9** 立得。 □

5.3.2 经典模型的构造

让我们把下面的讨论中固定在一个包含在 $\mathcal{L}(C_1 + C_2)$ 里的极大协调 **Henkin** 理论 Γ_0 上。

定义 5.17 我们让集合 Σ 为: $\Sigma \widehat{=} \{\phi \in \Gamma_0 \mid \phi$ 是刚性的$\}$ 。

定义 5.18 现在, 我们在 C_j 上定义一个关系 $\equiv^j$ 为 $f_1^j \equiv^j f_2^j$ 当且仅当 $f_1^j = f_2^j \in \Sigma$, 其中 $j = 1, 2$ 。

易证上述分别定义于 C_1 和 C_2 上的关系是等价关系。因为 C_1 和 C_2 是不相交的, 因此 $\equiv^j$ 也互不相交, 其中 $j = 1, 2$ 。

定义 5.19 我们定义集合 W, 关系 $R \subset W \times W \times W$, 和集合 T, D, D_1 如下:

- $W \widehat{=} \{\Gamma \mid \Gamma$ 是一个极大协调 ***Henkin*** 理论且 $\Sigma \subseteq \Gamma\}$
- $R(\Delta_1, \Delta_2, \Gamma)$ 当且仅当 $\Delta_1^\frown\Delta_2 \subseteq \Gamma$
- $T \widehat{=} \{< \Delta_1, \Delta_2 > \mid \Delta_1, \Delta_2 \in W, \Delta_1^\frown\Delta_2 \subseteq \Gamma_0\}$
- $D \widehat{=} \{[c]_{\equiv^1} \mid c \in C_1\}$
- $D_1 \widehat{=} \{[d]_{\equiv^2} \mid d \in C_2\}$

引理 5.10 设 $\Delta, \Delta_1, \Delta_2 \in W$, $\Gamma_1, \Gamma_2 \subset \mathcal{L}(C_1+C_2)$ 是两个极大协调理论。那么 $\delta_1(\Delta, \Delta_1) \subset \Gamma_2$ 蕴涵 $\Gamma_2 \in W$ 且 $R(\Delta_1, \Gamma_2, \Delta)$; 对称地, $\delta_2(\Delta, \Delta_2) \subset \Gamma_1$ 蕴涵 $\Gamma_1 \in W$ 且 $R(\Gamma_1, \Delta_2, \Delta)$ 。

证明: ([21]) 设 $\delta_1(\Delta, \Delta_1) \subset \Gamma_2$, $\phi \in \Delta_1, \psi \in \Gamma_2$ 。如果 $\neg(\phi^\frown\psi) \in \Delta$, 那么由 δ_1 的定义可得 $\neg\psi \in \Gamma_2$, 从而矛盾。因此 $\Delta_1{}^\frown\Gamma_2 \subset \Delta$ 。另外, 由引理 **5.8** 可得 $\Gamma_2 \in W$ 。因而 $R(\Delta_1, \Gamma_2, \Delta)$ 。另一部分可以类似地证明。 □

性质 5.2 设 $\Delta, \Delta_1, \Delta_2, \Delta_3 \in W$ 。那么如果存在一个 $\Delta' \in W$ 使得 $R(\Delta_1, \Delta_2, \Delta')$ 且 $R(\Delta', \Delta_3, \Delta)$, 那么存在一个 $\Delta'' \in W$ 使得 $R(\Delta_1, \Delta'', \Delta)$ 且 $R(\Delta_2, \Delta_3, \Delta'')$ 。反之亦然。

证明: 假设存在一个 $\Delta' \in W$ 使得 $R(\Delta_1, \Delta_2, \Delta')$ 且 $R(\Delta', \Delta_3, \Delta)$ 。也就是, $\Delta_1{}^\frown\Delta_2 \subset \Delta'$ 且 $\Delta'{}^\frown\Delta_3 \subset \Delta$ 。因 Δ_1, Δ_2 和 Δ_3 都是极大协调 **Henkin** 理论, 由 **ILT2** 知存在 $c_1, c_2, c_3 \in C_1$ 使得 $\ell = c_1 \in \Delta_1$, $\ell = c_2 \in \Delta_2$ 且 $\ell = c_3 \in \Delta_3$ 。令 $A \mathrel{\widehat{=}} \{(\ell = c_2{}^\frown\ell = c_3)\} \cup \delta_1(\Delta, \Delta_1)$ 。若 A 是协调的, 那么由定理 **5.2** 知存在一个极大协调 **Henkin** 理论 $\Delta'' \supset A$ 。由引理 **5.10** 知 $\Delta'' \in W$ 且 $R(\Delta_1, \Delta'', \Delta)$ 。设 $\chi_2 \in \Delta_2$, $\chi_3 \in \Delta_3$ 。我们有 $(\ell = c_1{}^\frown\chi_2) \in \Delta'$ 且 $((\ell = c_1{}^\frown\chi_2){}^\frown\chi_3) \in \Delta$ 。因而, 根据公理 **IL3** 可得 $(\ell = c_1{}^\frown(\chi_2{}^\frown\chi_3)) \in \Delta$ 。再由公理 **IL7** 得 $\neg(\ell = c_1{}^\frown\neg(\chi_2{}^\frown\chi_3)) \in \Delta$ 。所以 $\neg\neg(\chi_2{}^\frown\chi_3) \in \delta_1(\Delta, \Delta_1)$, 从而 $(\chi_2{}^\frown\chi_3) \in \Delta''$ 。这证明了 $\Delta_2{}^\frown\Delta_3 \subset \Delta''$, 即 $R(\Delta_2, \Delta_3, \Delta'')$ 。

现在, 我们仅需要去证明 A 是协调的。令 $\phi_1, \ldots, \phi_n \in \delta_1(\Delta, \Delta_1)$ 。那么对所有 $i = 1, \ldots, n$, 存在 $\psi_i \in \Delta_1$ 使得 $\neg(\psi_i{}^\frown\phi_i) \in \Delta$ 。令 $\psi \mathrel{\widehat{=}} \psi_1 \wedge \ldots \wedge \psi_n$ 。明显地, $\psi \in \Delta_1$ 。所以由规则 N, 对所有 $i = 1, \ldots, n$, $\neg(\psi^\frown\neg\phi_i) \in \Delta$ 。因为 $\Delta_1{}^\frown\Delta_2 \subset \Delta'$ 且 $\Delta'{}^\frown\Delta_3 \subset \Delta$, 我们有 $(\psi^\frown\ell = c_2) \in \Delta'$ 且 $((\psi^\frown\ell = c_2)^\frown\ell = c_3) \in \Delta$ 。那么, 由公理 **IL3** 可得 $(\psi^\frown(\ell = c_2{}^\frown\ell = c_3)) \in \Delta$ 。使用公理 **IL2** $n-1$ 次, 我们可以得到 $\vdash_{\boldsymbol{IL}_2}$ $(\psi^\frown(\ell = c_2{}^\frown\ell = c_3)) \wedge \neg(\psi^\frown\neg\phi_1) \wedge \ldots \wedge \neg(\psi^\frown\neg\phi_n) \Rightarrow (\psi^\frown((\ell = c_2{}^\frown\ell = c_3) \wedge \phi_1 \wedge \ldots \wedge \phi_n))$ 。因而 $\psi^\frown((\ell = c_2{}^\frown\ell = c_3) \wedge \phi_1 \wedge \ldots \wedge \phi_n) \in \Delta$ 。从而 $\nvdash_{\boldsymbol{IL}_2}$ $\neg((\ell = c_2{}^\frown\ell = c_3) \wedge \phi_1 \wedge \ldots \wedge \phi_n)$, 否则根据规则 N, 我们有 $\vdash_{\boldsymbol{IL}_2}$ $\neg(\psi^\frown((\ell = c_2{}^\frown\ell = c_3) \wedge \phi_1 \wedge \ldots \wedge \phi_n))$ 。这与 Δ 是协调的相矛盾。因此 A 的任意有穷子集都是协调的, 由推论 **5.2** 可得 A 是协调的。

剩下部分可以类似地证明。 □

性质 5.3 设 $\Delta \in W$ 。那么存在 $\Delta_1, \Delta_2 \in W$ 使得 $\ell = 0 \in \Delta_1, \Delta_2$, $R(\Delta_1, \Delta, \Delta)$ 且 $R(\Delta, \Delta_2, \Delta)$ 。

证明: ([21]) 类似于性质 **5.2** 的证明。我们仅需证明 Δ_1 的存在性。令 $A \mathrel{\widehat{=}} \{\ell = 0\} \cup \delta_2(\Delta, \Delta)$ 。我们首先证明 A 是协调的。根据推论 **5.2**, 我们只需证明 A 的每个有穷子集是协调的即可。设 $\phi_1, \ldots, \phi_n \in \delta_2(\Delta, \Delta)$ 。那么存在 $\psi_1, \ldots, \psi_n \in \Delta$ 使得 $\neg(\neg\phi_i{}^\frown\psi_i) \in \Delta$, 其中 $i = 1, \ldots, n$ 。令 $\psi \mathrel{\widehat{=}} \psi_1 \wedge \ldots \wedge \psi_n$ 。那么由规则 N 和 M 可得 $\psi \in \Delta$ 且 $\neg(\neg\phi_i{}^\frown\psi) \in \Delta$ 。由公理 **IL8** 可得 $(\ell = 0{}^\frown\psi) \in \Delta$ 。象性质 **5.2** 中的证明一样可得 $((\ell = 0 \wedge \phi_1 \wedge \ldots \wedge \phi_n){}^\frown\psi) \in \Delta$ 。因而 $\nvdash_{\boldsymbol{IL}_2}$ $\neg(\ell = 0 \wedge \phi_1 \wedge \ldots \wedge \phi_n)$ 。这表明 A 的任意有穷子集是协调的。

因此, 根据定理 **5.2**, 存在一个极大协调理论 $\Delta_1 \supseteq A$ 。我们有 $\ell = 0 \in \Delta_1$ 。由引理 **5.10** 可得 $\Delta_1 \in W$ 且 $\Delta_1{}^\frown\Delta \subset \Delta$ 。

类似地可证明 $\ell = 0 \in \Delta_2$ 且 $R(\Delta, \Delta_2, \Delta)$ 。 □

定义 5.20 我们定义一个 T 上的序关系 $\leq$ 如下：如果 $\exists c_1, c_2 \in C_1 \quad \ell = c_1 \in \Delta_1, \ell = c_1 + c_2 \in \Delta_1'$，那么 $< \Delta_1, \Delta_2 > \quad \leq \quad < \Delta_1', \Delta_2' >$ 。

性质 5.4 $\leq$ 是 T 上的全序。

证明： ([21]) 设 $< \Delta_1, \Delta_2 >, < \Delta_1', \Delta_2' > \in T$ 。根据 $\leq$ 的定义，我们有 $< \Delta_1, \Delta_2 > \quad \leq \quad < \Delta_1', \Delta_2' >$ 当且仅当存在 $c_1, c_2 \in C_1$ 使得 $\ell = c_1 \in \Delta_1, \ell = c_2 \in \Delta_1'$ 且 $\exists x. c_1 + x = c_2$ 。使用 **D1-D5**，容易证明 $\leq$ 是一个全序。 □

性质 5.5 设 $< \Delta_1, \Delta_2 >, < \Delta_1', \Delta_2' > \in T$ 且 $< \Delta_1, \Delta_2 > \quad \leq \quad < \Delta_1', \Delta_2' >$ 。那么存在唯一的 $\Delta \in W$ 使得 $R(\Delta_1, \Delta, \Delta_1')$ 且 $R(\Delta, \Delta_2', \Delta_2)$ 。

证明： ([21]) 因 $\Delta_1, \Delta_2, \Delta_1'$ 和 Δ_2' 是极大协调 **Henkin** 理论，由 **ILT2** 知存在 $c_1, c_2, c_1', c_2' \in C_1$ 使得 $\ell = c_1 \in \Delta_1, \ell = c_2 \in \Delta_2, \ell = c_1' \in \Delta_1', \ell = c_2' \in \Delta_2'$ 。因为 $< \Delta_1, \Delta_2 > \quad \leq \quad < \Delta_1', \Delta_2' >$，所以存在一个 $c \in C_1$ 使得 $c_1 + c = c_1' \in \Sigma$ 。因而 $c_1 + c = c_1' \in \Delta_1'$ 。

令 $A \hat{=} \{\ell = c\} \cup \delta_1(\Delta_1', \Delta_1) \cup \delta_2(\Delta_2, \Delta_2')$ 。我们将证明 A 是协调的。设 $\phi_1, \ldots, \phi_n \in \delta_1(\Delta_1', \Delta_1), \psi_1, \ldots, \psi_m \in \delta_2(\Delta_2, \Delta_2')$ 。那么存在公式 $\phi_1', \ldots, \phi_n' \in \Delta_1, \psi_1', \ldots, \psi_m' \in \Delta_2'$ 使得 $\neg(\phi_i' \frown \neg\phi_i) \in \Delta_1', \ i = 1, \ldots, n$，$\neg(\neg\psi_i \frown \psi_i') \in \Delta_2, \ i = 1, \ldots, m$ 。令 $\phi' \hat{=} \phi_1' \wedge \ldots \wedge \phi_n'$，$\psi' \hat{=} \psi_1' \wedge \ldots \wedge \psi_m'$ 。因为 Δ_1 和 Δ_2' 都是极大协调理论，所以我们有 $\phi' \wedge \ell = c_1 \in \Delta_1$ 和 $\psi' \wedge \ell = c_2' \in \Delta_2'$ 。象在性质 **5.2** 和性质 **5.3** 的证明一样可得对所有 $i = 1, \ldots, n$ 有 $\neg((\phi' \wedge \ell = c_1) \frown \neg\phi_i) \in \Delta_1'$ 和对所有 $i = 1, \ldots, m$ 有 $\neg(\neg\psi_i \frown (\psi' \wedge \ell = c_2')) \in \Delta_2$ 。根据 T 的定义可得 $\Delta_1 \frown \Delta_2, \Delta_1' \frown \Delta_2' \subset \Gamma_0$，因此我们有 $((\phi' \wedge \ell = c_1) \frown (\neg(\neg\psi_1 \frown (\psi' \wedge \ell = c_2')) \wedge \ldots \wedge \neg(\neg\psi_m \frown (\psi' \wedge \ell = c_2')))) \in \Gamma_0$ 且 $((\neg((\phi' \wedge \ell = c_1) \frown \neg\phi_1) \wedge \ldots \wedge \neg((\phi' \wedge \ell = c_1) \frown \neg\psi_n)) \frown (\psi' \wedge \ell = c_2')) \in \Gamma_0$ 。此外，$\vdash_{IL_2} \ \ell = c_1' \wedge c_1 + c = c_1' \Rightarrow \ell = c_1 + c$，由公理 **IL7** 可得 $\vdash_{IL_2} \ \ell = c_1 + c \Rightarrow (\ell = c_1 \frown \ell = c)$ 。因此 $(\ell = c_1 \frown \ell = c) \in \Delta_1'$ 且 $((\ell = c_1 \frown \ell = c) \frown (\psi' \wedge \ell = c_2')) \in \Gamma_0$ 。根据公理 **IL3**，$(\ell = c_1 \frown (\ell = c \frown (\psi' \wedge \ell = c_2'))) \in \Gamma_0$ 。

根据 **ILT9** 可得 $((\phi' \wedge \ell = c_1) \frown (\neg(\neg\psi_1 \frown (\psi' \wedge \ell = c_2')) \wedge \ldots \wedge \neg(\neg\psi_m \frown (\psi' \wedge \ell = c_2')) \wedge (\ell = c \frown (\psi' \wedge \ell = c_2')))) \in \Gamma_0$ 。由公理 **IL2**，规则 M 和 Γ_0 是极大协调理论的事实，我们可以得到 $((\phi' \wedge \ell = c_1) \frown (\ell = c \wedge \psi_1 \wedge \ldots \wedge \psi_m \frown \psi' \wedge \ell = c_2')) \in \Gamma_0$ 。由公理 **IL3** 可得 $((\phi' \wedge \ell = c_1) \frown ((\ell = c \wedge \psi_1 \wedge \ldots \wedge \psi_m) \frown (\psi' \wedge \ell = c_2'))) \in \Gamma_0$ 。

同样根据 **ILT9** 可得 $((((\phi' \wedge \ell = c_1) \frown (\ell = c \wedge \psi_1 \wedge \ldots \wedge \psi_m)) \wedge \neg((\phi' \wedge \ell = c_1) \frown \neg\phi_1) \wedge \ldots \wedge \neg((\phi' \wedge \ell = c_1) \frown \neg\psi_n)) \frown (\psi' \wedge \ell = c_2')) \in \Gamma_0$ 。使用公理 **IL2** $n-1$ 次，我们可得 $(((\phi' \wedge \ell = c_1) \frown (\ell = c \wedge \psi_1 \wedge \ldots \wedge \psi_m \wedge \phi_1 \wedge \ldots \wedge \phi_n)) \frown (\psi' \wedge \ell = c_2^{1\prime})) \in \Gamma_0$ 。这证明 $\ell = c \wedge \psi_1 \wedge \ldots \wedge \psi_m \wedge \phi_1 \wedge \ldots \wedge \phi_n$ 是协调的。因此，A 的每个有穷子集都是协调的。根据推论 **5.2** 知 A 是协调的。根据定理 **5.2**，存在一个极大协调理论 $\Delta \supset A$ 。根据引理 **5.10**，因而 $\delta_1(\Delta_1', \Delta_1), \delta_2(\Delta_2, \Delta_2') \subset \Delta, \ \Delta \in W$，$R(\Delta_1, \Delta, \Delta_1')$ 且 $R(\Delta, \Delta_2', \Delta_2)$ 。 □

定义 5.21 令 $\mathbf{Intv}(T) \hat{=} \{[t_1, t_2] \mid t_1, t_2 \in T, t_1 \leq t_2\}$ 。我们定义函数 $\mu : \mathbf{Intv}(T) \to W$ 如下：对 $t_1 =< \Delta_1, \Delta_2 >$ 和 $t_2 =< \Delta_1', \Delta_2' >$，$\mu([t_1, t_2]) = \Delta$ 当且仅当 $R(\Delta_1, \Delta, \Delta_1')$ 且 $R(\Delta, \Delta_2', \Delta_2)$ 。性质 ***5.5*** 确保该定义是合适的。

性质 5.6 设 $t_1, t_2, t_3 \in T, t_1 \leq t_2 \leq t_3$ 。那么 $R(\mu([t_1,t_2]), \mu([t_2,t_3]), \mu([t_1,t_3]))$ 。

证明: ([21]) 设 $t_1 =< \Delta_1, \Delta_2 >, t_2 =< \Delta_1', \Delta_2' >$, $t_3 =< \Delta_1'', \Delta_2'' >$ 。由 μ 的定义, 我们有 $R(\Delta_1, \mu([t_1,t_2]), \Delta_1'), R(\Delta_1', \mu([t_2,t_3]), \Delta_1'')$ 和 $R(\Delta_1, \mu([t_1,t_3]), \Delta_1'')$ 。这意味着 $\Delta_1{}^\frown\mu([t_1,t_2]) \subset \Delta_1', \Delta_1'{}^\frown\mu([t_2,t_3]) \subset \Delta_1''$ 和 $\Delta_1{}^\frown\mu([t_1,t_3]) \subset \Delta_1''$ 。设 $\phi \in \mu([t_1,t_2]), \psi \in \mu([t_2,t_3])$ 。令 $c \in C_1$ 使得 $\ell = c \in \Delta_1$ 。那么 $(\ell = c{}^\frown\phi) \in \Delta_1'$ 且 $((\ell = c{}^\frown\phi){}^\frown\psi) \in \Delta_1''$ 。由公理 **IL3**, 可得 $(\ell = c{}^\frown(\phi{}^\frown\psi)) \in \Delta_1''$ 。运用公理 **IL6**, $\neg(\ell = c{}^\frown\neg(\phi{}^\frown\psi)) \in \Delta_1''$ 。因为 $\ell = c \in \Delta_1$ 和 $\Delta_1{}^\frown\mu([t_1,t_3]) \subset \Delta_1''$, 因此 $(\phi{}^\frown\psi) \in \mu([t_1,t_3])$ 。这证明 $\mu([t_1,t_2]){}^\frown\mu([t_2,t_3]) \subset \mu([t_1,t_3])$ 。 □

性质 5.7 设 $t_1, t_2 \in T, t_1 \leq t_2$, $\Gamma_1, \Gamma_2 \in W$ 且 $R(\Gamma_1, \Gamma_2, \mu([t_1,t_2]))$ 。那么存在一个 $t \in T$ 使得 $t_1 \leq t \leq t_2$ 且 $\Gamma_1 = \mu([t_1,t]), \Gamma_2 = \mu([t,t_2])$ 。

证明: ([21]) 设 $t_1 =< \Delta_1, \Delta_2 >$, $t_2 =< \Delta_1'', \Delta_2'' >$ 。由 T 的定义可得 $\Delta_1{}^\frown\Delta_2, \Delta_1''{}^\frown\Delta_2'' \subset \Gamma_0$ 。由 μ 的定义可得 $\Delta_1{}^\frown\mu([t_1,t_2]) \subset \Delta_1'', \mu([t_1,t_2]){}^\frown\Delta_2'' \subset \Delta_2$ 。因 $\Gamma_1, \Gamma_2 \in W$ 且 $R(\Gamma_1, \Gamma_2, \mu([t_1,t_2]))$, 即 $\Gamma_1{}^\frown\Gamma_2 \subset \mu([t_1,t_2])$, 根据性质 **5.1** 知存在一个 $\Delta_1' \in W$ 使得 $\Delta_1{}^\frown\Gamma_1 \subset \Delta_1'$ 且 $\Delta_1'{}^\frown\Gamma_2 \subset \Delta_1''$; 存在一个 Δ_2' 使得 $\Gamma_1{}^\frown\Delta_2' \subset \Delta_2$ 且 $\Gamma_2{}^\frown\Delta_2'' \subset \Delta_2'$ 。我们仅需要证明 $t =< \Delta_1', \Delta_2' > \in T$ 和 $t_1 \leq t \leq t_2$ 。根据性质 **5.5**, 这证明了 $\mu([t_1,t]) = \Gamma_1$ 且 $\mu([t,t_2]) = \Gamma_2$ 。

因为 $\Delta_1, \Delta_2'', \Gamma_1$ 和 Γ_2 是极大协调协调 **Henkin** 理论, 根据 **ILT2** 知存在 $c_1, c_2'', b_1, b_2 \in C_1$ 使得 $\ell = c_1 \in \Delta_1, \ell = c_2'' \in \Delta_2'', \ell = b_1 \in \Gamma_1, \ell = b_2 \in \Gamma_2$ 。设 $\phi \in \Delta_1', \psi \in \Delta_2'$ 。我们有 $(\phi{}^\frown\ell = b_2) \in \Delta_1'', ((\phi{}^\frown\ell = b_2){}^\frown\ell = c_2'') \in \Gamma_0, (\ell = b_1{}^\frown\psi) \in \Delta_2$ 和 $(\ell = c_1{}^\frown(\ell = b_1{}^\frown\psi)) \in \Gamma_0$ 。此外, $(\ell = b_1{}^\frown\ell = b_2) \in \mu([t_1,t_2]), (\ell = c_1{}^\frown(\ell = b_1{}^\frown\ell = b_2)) \in \Delta_1''$ 且 $((\ell = c_1{}^\frown(\ell = b_1{}^\frown\ell = b_2)){}^\frown\ell = c_2'') \in \Gamma_0$ 。现在, 根据公理 **IL3** 和 **IL7** 可得 $(\phi{}^\frown\ell = b_2 + c_2''), (\ell = c_1 + b_1{}^\frown\psi), (\ell = c_1 + b_1{}^\frown\ell = b_2 + c_2'') \in \Gamma_0$ 。由 **ILT10** 可得 $((\phi \wedge \ell = c_1 + b_1){}^\frown(\psi \wedge \ell = b_2 + c_2'')) \in \Gamma_0$ 。因而 $(\phi{}^\frown\psi) \in \Gamma_0$ 。从而 $\Delta_1'{}^\frown\Delta_2' \subset \Gamma_0$, 即 $t =< \Delta_1', \Delta_2' > \in T$ 。

因为 $\Delta_1{}^\frown\Gamma_1 \subset \Delta_1', \ell = c_1 + b_1 \in \Delta_1'$, 因而 $t_1 \leq t$ 。又因为 $\Delta_1'{}^\frown\Gamma_2 \subset \Delta_1'', \ell = c_1 + (b_1 + b_2) \in \Delta_1''$, 所以 $t \leq t_2$ 。 □

定义 5.22 我们定义测度函数 $m : \boldsymbol{Intv}(T) \to D$ 如下: $m([t_1,t_2]) = [c]_{\equiv^1}$ 当且仅当 $\ell = c \in \mu([t_1,t_2])$ 。

定义 5.23 我们定义函数 $+ : D \times D \to D$ 如下: $[c_1]_{\equiv^1} + [c_2]_{\equiv^1} = [c_3]_{\equiv^1}$ 当且仅当 $c_1 + c_2 = c_3 \in \Sigma$ 。

定理 5.4 设 $\mathcal{F} =<< T, \leq >, < D, [0]_{\equiv^1}, + >, D_1, m >$ 。那么 $\mathcal{F}$ 是 $\mathbf{IL}_2$ 的语义框架。

证明: ([29]) 根据性质 **5.4**, $< T, \leq >$ 是一个时间域。因为 **D1-D5** 的所有实例都在 Σ 里, 所以 D 必然满足时段域的几条公理 **D1-D5** 。因此我们仅需证明上述定义的测度 m 满足 **M1 - M3** 即可。

对于 **M1**, 设 $t_1 =< \Delta_1, \Delta_2 >, t_2 =< \Delta_1', \Delta_2' >$, $t_3 =< \Delta_1'', \Delta_2'' >$ 。令 $m([t_1,t_2]) = m([t_1,t_3]) = [c]_{\equiv^1}$, 其中 $c \in C_1$ 。那么 $\ell = c \in \mu([t_1,t_2]), \mu([t_1,t_3])$ 。令 $b \in C_1$ 使得

$\ell = b \in \Delta_1$。根据 μ 的定义，$\Delta_1 \frown \mu([t_1, t_2]) \subset \Delta_1'$ 所以 $(\ell = b \frown \ell = c) \in \Delta_1'$。因而由公理 **IL7** 可得 $\ell = b + c \in \Delta_1'$。类似可证 $\ell = b + c \in \Delta_1''$。令 $a \in C_1$ 使得 $a = b + c \in \Sigma$。因为 Γ_0 是一个 **Henkin** 理论，所以这样的 a 是存在的。因而 $\ell = a \in \Delta_1', \Delta_1''$。又因为 $\Delta_1' \frown \Delta_2', \Delta_1'' \frown \Delta_2'' \subset \Gamma_0$，所以可得 $\{\ell = a\} \frown \Delta_2', \{\ell = a\} \frown \Delta_2'' \subset \Gamma_0$。因而根据推论 **5.4**，$\Delta_2' = \Delta_2''$。令 $e \in C_1$ 使得 $\ell = e \in \Delta_2', \Delta_2''$。那么 $\Delta_1' \frown \{\ell = e\}, \Delta_1'' \frown \{\ell = e\} \subset \Gamma_0$，因而再次运用推论 **5.4** 可得 $\Delta_1' = \Delta_1''$。因此 $t_2 = t_3$。

对于 **M2**，假设存在 $c_1, c_2 \in C_1$ 使得 $\ell = c_1 \in \mu([t_1, t_2])$ 且 $\ell = c_2 \in \mu([t_2, t_3])$。根据性质 **5.6** 可得 $\mu([t_1, t_2]) \frown \mu([t_2, t_3]) \subset \mu([t_1, t_3])$，所以 $(\ell = c_1 \frown \ell = c_2) \in \mu([t_1, t_3])$。因而根据公理 **IL7** 可得 $\ell = c_1 + c_2 \in \mu([t_1, t_3])$。令 $c \in C_1$ 使得 $c = c_1 + c_2 \in \Sigma$。那么明显地 $\ell = c \in \mu([t_1, t_3])$，因此 $m([t_1, t_3]) = [c]_{\equiv^1} = [c_1]_{\equiv^1} + [c_2]_{\equiv^1}$。

对于 **M3**，设存在 $c_1, c_2 \in C_1$ 使得 $m([t_1, t_2]) = [c_1]_{\equiv^1} + [c_2]_{\equiv^1}$。那么 $\ell = c_1 + c_2 \in \mu([t_1, t_2])$，因而由公理 **IL7** 可得 $(\ell = c_1 \frown \ell = c_2) \in \mu([t_1, t_2])$。根据定理 **5.3**，存在两个极大协调理论 Γ_1 和 Γ_2 使得 $\ell = c_1 \in \Gamma_1, \ell = c_2 \in \Gamma_2$ 且 $\Gamma_1 \frown \Gamma_2 \subset \mu([t_1, t_2])$。由引理 **5.8** 可得 $\Sigma \subset \Gamma_1, \Gamma_2$，因而 $\Gamma_1, \Gamma_2 \in W$。由性质 **5.7** 知存在一个 $t \in T$ 使得 $t_1 \leq t \leq t_2, \Gamma_1 = \mu([t_1, t])$ 且 $\Gamma_2 = \mu([t, t_2])$。显然 $m([t_1, t]) = [c_1]_{\equiv^1}$ 且 $m([t, t_2]) = [c_2]_{\equiv^1}$。 □

定义 5.24 我们定义从 $\mathcal{L}(C_1 + C_2)$ 到 $\mathcal{F}$ 的解释 $\mathcal{J}$ 如下：

$\mathcal{J}(c) = [c]_{\equiv^1}$	对所有$c \in C_1$;
$\mathcal{J}(d) = [d]_{\equiv^2}$	对所有$d \in C_2$;
$\mathcal{J}(x) = [c]_{\equiv^1}$	当且仅当$x = c \in \Sigma$ 对所有$x \in Var^1$;
$\mathcal{J}(y) = [d]_{\equiv^2}$	当且仅当$y = d \in \Sigma$ 对所有$y \in Var^2$;
$\mathcal{J}(v)([t_1, t_2]) = [c]_{\equiv^1}$	当且仅当$v = c \in \mu([t_1, t_2])$, 对所有$v \in TVar$;
$\mathcal{J}(f)([c_1]_{\equiv^1}, \ldots, [c_n]_{\equiv^1}) = [c_{n+1}]_{\equiv^1}$	当且仅当$f(c_1, \ldots, c_n) = c_{n+1} \in \mu([t_1, t_2])$, 对所有$f \in FSymb^1$;
$\mathcal{J}(g)([t_1, t_2])([d_1]_{\equiv^2}, \ldots, [d_n]_{\equiv^2}) = [c]_{\equiv^1}$	当且仅当$g(d_1, \ldots, d_n) = c \in \mu([t_1, t_2])$, 对所有$g \in FSymb^2$;
$\mathcal{J}(R)([c_1]_{\equiv^1}, \ldots, [c_n]_{\equiv^1}) = 1$	当且仅当$R(c_1, \ldots, c_n) \in \Sigma$ 对所有$R \in RSymb$;
$\mathcal{J}(X)([t_1, t_2]) = 1$	当且仅当$X \in \mu([t_1, t_2])$, 对所有$X \in PLetter$。

上述定义是合适的。例如，对于 $Var^1 \cup Var^2$ 中的全局变量 z，因为 $\vdash_{\boldsymbol{IL}_2} \exists z'(z' = z)$ 且 $\Sigma = \{\phi \in \bigcap \mathcal{W} \mid \phi \text{ 是刚性的}\}$，所以 $\exists z'(z' = z) \in \Sigma$。这样，存在一个刚性常量 $f \in C_j$，$j = 1, 2$ 使得 $z = f \in \Sigma$。而且，这是不依赖于 f 的选取。因为，如果 f_1 是 C^j 中的另一个常量使得 $z = f_1 \in \Sigma$，那么我们可得 $\vdash_{\boldsymbol{IL}_2} (z = f) \wedge (z = f_1) \Rightarrow (f = f_1)$。所以 $[f]_{\equiv^j} = [f_1]_{\equiv^j}$。其它情况可以类似地讨论。而且，刚性符号在不同的区间上的解释是相同的。

定理 5.5 $\mathcal{M} =< \mathcal{F}, \mathcal{J} >$ 是 $\mathbf{IL}_2$ 的一个模型。

证明： 因为 W 是由极大协调 **Henkin** 理论构成，所以 $\mathcal{J}$ 是处处有定义的。易证 $\mathcal{J}(\ell) = m, \mathcal{J}(0) = [0^1]_{\equiv^1}, \mathcal{J}(+) = +^1$，$\mathcal{J}(=)$ 就是 $=$。 □

定理 5.6 设 t^j 是 $\mathcal{L}(C_1+C_2)$ 中第 j 种类的项，$[t_1,t_2]\in \mathbf{Intv}(T)$。那么 $\mathcal{J}_{t_1}^{t_2}(t^j)=[f]_{\equiv^j}$ 当且仅当 $t^j=f\in\mu([t_1,t_2])$，其中 $j=1,2$。

证明: 对 t^j 的结构进行归纳即可。 □

定理 5.7 设 $\phi\in\mathcal{L}(C_1+C_2)$。那么 $<\mathcal{F},\mathcal{J}>,[t_1,t_2]\models_{\mathbf{IL}_2}\ \phi$ 当且仅当 $\phi\in\mu([t_1,t_2])$。

证明: 对 ϕ 的构成进行归纳。当 ϕ 是原子公式时，由 $\mathcal{J}$ 的定义直接可得；当 $\phi=\neg\psi$ 或 $\phi=\varphi\wedge\psi$ 时，利用 $\mu([t_1,t_2])$ 是极大协调理论即可证；当 $\phi=\exists z.\psi$ 时，可以由 $\mu([t_1,t_2])$ 是 **Henkin** 理论得到；当 $\phi=(\varphi^\frown\psi)$ 时，根据 **ILT2** 可得 $\exists x,x'.((\varphi\wedge\ell=x)^\frown(\psi\wedge\ell=x'))\in\mu([t_1,t_2])$。因此存在 $c_1,c_2\in C_1$ 使得 $((\varphi\wedge\ell=c_1)^\frown(\psi\wedge\ell=c_2))\in\mu([t_1,t_2])$。由定理 **5.3** 知存在 $\Gamma_1,\Gamma_2\in\mathcal{W}$ 使得 $\Gamma_1{}^\frown\Gamma_2\in\mu([t_1,t_2])$，$\varphi\wedge\ell=c_1\in\Gamma_1,\psi\wedge\ell=c_2\in\Gamma_2$。根据性质 **5.7**，存在一个 $t\in T$ 使得 $\Gamma_1=\mu([t_1,t])$，$\Gamma_2=\mu([t,t_2])$。因而由归纳假设可得 $\mathcal{M},[t_1,t]\models_{\boldsymbol{IL}_2}\ \varphi$ 且 $\mathcal{M},[t,t_2]\models_{\boldsymbol{IL}_2}\ \psi$，因此 $\mathcal{M},[t_1,t_2]\models_{\boldsymbol{IL}_2}\ \phi$。为了证明反方向，令 $\mathcal{M},[t_1,t_2]\models_{\boldsymbol{IL}_2}\ \phi$。那么存在一个 $t\in T$ 使得 $\mathcal{M},[t_1,t]\models_{\boldsymbol{IL}_2}\ \varphi$ 且 $\mathcal{M},[t,t_2]\models_{\boldsymbol{IL}_2}\ \psi$。根据归纳假设，$\varphi\in\mu([t_1,t]),\psi\in\mu([t,t_2])$，因而 $(\varphi^\frown\psi)\in\mu([t_1,t])^\frown\mu([t,t_2])$。由性质 **5.6** 可得 $\mu([t_1,t])^\frown\mu([t,t_2])\subset\mu([t_1,t_2])$，因而 $\phi\in\mu([t_1,t_2])$ □

定理 5.8 对每个 $\Phi\in\Omega$，它在 $\mathcal{M}=<\mathcal{F},\mathcal{J}>$ 上具有有穷可变性。

证明: 设 $\Phi\in\Omega,[t_1,t_2]\in\mathbf{Intv}(T)$。那么 $\mu([t_1,t_2])$ 是一个极大协调理论且在 IR^Φ 下封闭。因而，根据引理 **5.4** 可得存在 $k<\omega$ 使得 $\mathcal{M},[t_1,t_2]\models_{\boldsymbol{IL}_2}\ \Phi^k$。因而存在 $t_1'<\ldots<t_n'$ 使得 $t_1=t_1',t_2=t_n'$，且对 $i=1,\ldots,n-1$ 有 $\mathcal{M},[t_i',t_{i+1}']\models_{\boldsymbol{IL}_2}\ \Box\Phi$。这表明对 $i=1,\ldots,n$，$\mathcal{M},[t_1,t_i']\models_{\boldsymbol{IL}_2}\ \Phi^{i-1}$，因而 Φ 在 $\mathcal{M}$ 上具有有穷可变性。 □

定理 5.9 (完备性) 如果 Γ 是协调的，那么存在一个模型 $<\mathcal{F},\mathcal{J}>$ 和一个区间 $[t_1,t_2]\in\mathbf{Intv}(T)$ 使得 $<\mathcal{F},\mathcal{J}>,[t_1,t_2]\models_{\mathbf{IL}_2}\ \Gamma$ 且 $\Phi\in\Omega$ 在该模型上具有有穷可变性。

证明: 根据定理 **5.2**，存在一个极大协调 **Henkin** 理论 $\Gamma_0\supseteq\Gamma$。令 $t_1,t_2\in T$ 使得 $\mu([t_1,t_2])=\Gamma_0$，$<\mathcal{F},\mathcal{J}>$ 是基于 Γ_0 用上述方法构造的模型，其中 $\mathcal{F}=<<T,\leq>,<D,+,0>,D_1,m>$ 是它对应的语义框架。因此，根据定理 **5.7**，我们有 $<\mathcal{F},\mathcal{J}>,[t_1,t_2]\models_{\boldsymbol{IL}_2}\ \Gamma$。再根据定理 **5.8** 可得对所有 $\Phi\in\Omega$ 在 $\mathcal{M}=<\mathcal{F},\mathcal{J}>$ 上具有有穷可变性。 □

推论 5.5 如果 $\models_{\mathbf{IL}_2}\ \phi$，那么 $\vdash_{\mathbf{IL}_2}\ \phi$。

证明: 假设 $\nvdash_{\boldsymbol{IL}_2}\ \phi$，因此 $\{\neg\phi\}$ 在 $\boldsymbol{IL}_2$ 的证明系统里是协调的。根据定理 **5.9**，存在一个模型 $<\mathcal{F},\mathcal{J}>$ 和一个区间 $[t_1,t_2]$ 使得 $<\mathcal{F},\mathcal{J}>,[t_1,t_2]\models_{\boldsymbol{IL}_2}\ \neg\phi$。从而与 $\models_{\boldsymbol{IL}_2}\ \phi$ 矛盾。因此 $\vdash_{\boldsymbol{IL}_2}\ \phi$。 □

5.4 HDC 在抽象时间域上的语义

在这里，我们首先给出 **HDC** 在抽象时间域上的语义。

HDC 的语义框架本质上和 IL_2 的语义框架是相同的，差别仅在于在 **HDC** 的语义框架里不再需要内附论域。即 **HDC** 的语义框架是一个三元组 $<< T, \leq >, < D, +, 0 >, m >$，其中，$< T, \leq >$ 是时间域，$< D, +, 0 >$ 是时段域，m 是测度。

定义 5.25 一个 ***HDC*** 的模型是一个四元组 $<< T, \leq >, < D, +, 0 >, m, \mathcal{I} >$，其中，$<< T, \leq >, < D, +, 0 >, m >$ 是语义框架，$\mathcal{I}$ 是 **解释**，它满足下述条件：

(I) 对任意 $X \in PLetter$ 有 $\mathcal{I}(X) : \mathbf{Intv}(T) \to \{0, 1\}$；

(II) 对任意 $R_i^n \in RSymb$ 有 $\mathcal{I}(R_i^n) : D^n \to \{0, 1\}$；

(III) 对任意 $f_i^n \in FSymb$ 有 $\mathcal{I}(f_i^n) : D^n \to D$；

(IV) 对任意 $V \in PVar$ 和任意区间 $[t_1, t_2] \in \mathbf{Intv}(T)$，存在一个序列 $t_1', \ldots, t_n'$ 使得 $t_1 = t_1' \leq \ldots \leq t_n' = t_2$，且对任意 $t, t' \in [t_i', t_{i+1}')$ 有 $\mathcal{I}(V)(t) = \mathcal{I}(V)(t')$；

(V) $\mathcal{I}(0) = 0, \mathcal{I}(+) = +, \mathcal{I}(=)$ 就是 $=$，而 $\mathcal{I}(\ell) = m$ 。

上述的 **(IV)** 是说，在任意一个解释下，程序变量都有有穷可变性。

定义 5.26 设 $\mathcal{I}$ 和 $\mathcal{I}'$ 是满足上述条件的两个解释。如果对除 z 外的所有符号，在这两个解释下有相同的值，则我们说 $\mathcal{I}$ z- 等价于 $\mathcal{I}'$，

给定一个模型 $\mathcal{M} =<< T, \leq >, < D, +, 0 >, m, \mathcal{I} >$，全局变量，程序变量和命题符号的语义由 $\mathcal{I}$ 给出。对所有 $x \in GVar$，$\mathcal{I}(x) \in D$；对所有 $V \in PVar$，$\mathcal{I}(V) \in T \to D$；对所有 $X \in PLetter$，$\mathcal{I}(X) \in \mathbf{Intv}(T) \to \{tt, ff\}$ 。

给定一个模型 $\mathcal{M} =< \mathcal{F}, \mathcal{I} >$，状态项 ϑ 的语义是一个类型为

$$\mathcal{I}(\vartheta) \in T \ \to \ D$$

的函数，该函数可以根据状态项的结构递归定义如下：

$$\begin{array}{lcl} \mathcal{I}(x)(t) & = & \mathcal{I}(x) \\ \mathcal{I}(V)(t) & = & \mathcal{I}(V)(t) \\ \mathcal{I}(f^n(\vartheta_1, \ldots, \vartheta_n))(t) & = & \mathcal{I}(f^n)(c_1, \ldots, c_n) \end{array}$$

其中，$c_i = \mathcal{I}(\vartheta_i)(t)$，$1 \leq i \leq n$ 。

给定一个模型 $\mathcal{M} =< \mathcal{F}, \mathcal{I} >$，状态表达式 S 被解释成一个类型为

$$\mathcal{I}(S) \in T \ \to \ \{0, 1\}$$

的函数，该函数可以根据状态表达式的结构递归定义如下：

$$\begin{array}{lcl}
\mathcal{I}(0)(t) & = & 0 \\
\mathcal{I}(1)(t) & = & 1 \\
\mathcal{I}(R^n(\vartheta_1,\ldots,\vartheta_n))(t) & = & \mathcal{I}(R^n)(\mathcal{I}(\vartheta_1)(t),\ldots,\mathcal{I}(\vartheta_n)(t)) \\
\mathcal{I}(\neg S)(t) & = & 1-\mathcal{I}(S)(t) \\
\mathcal{I}(S_1\vee S_2)(t) & = & \begin{cases} 0 \text{ 如果 } \mathcal{I}(S_1)(t)=0 \text{ 且 } \mathcal{I}(S_2)(t)=0 \\ 1 \text{ 其它} \end{cases}
\end{array}$$

引理 5.11 设 S 是一个状态表达式，$\mathcal{I}$ 是基于语义框架 $\mathcal{F}=<<T,\leq>,<D,+,0>,m>$ 上的一个 ***HDC*** 的解释。那么，对任意区间 $[t_1,t_2]\in \mathbf{Intv}(T)$，存在一个序列 $t_1',\ldots,t_n'$ 使得 $t_1=t_1'\leq\ldots\leq t_n'=t_2$，且对任意 $t,t'\in[t_i',t_{i+1}')$ 有 $\mathcal{I}(S)(t)=\mathcal{I}(S)(t')$，其中 $i=1,\ldots,n-1$ 。

证明: 归纳于 S 的结构，立得。

运用引理 **5.11**，我们可以给出 $\int S$ 在模型 $\mathcal{M}=<\mathcal{F},\mathcal{I}>$ 下的解释。设 $[t_1,t_2]\in \mathbf{Intv}(T)$，$t_1',\ldots,t_n'$ 是它的一个分割，且满足引理 **5.11** 中描述的性质。那么 $\mathcal{I}(\int S)([t_1,t_2])=\sum_{i=1}^{n-1}\mathcal{I}(S)(t_i')\bullet m([t_i',t_{i+1}'])$ 。其中，$p\bullet c$ 定义为：

$$p\bullet c \quad = \quad \begin{cases} 0 & \text{若} p=0 \\ c & \text{若} p=1 \end{cases}$$

我们很容易证明上述定义是不依赖于对 $t_1',\ldots,t_n'$ 的选择。

给定一个模型 $\mathcal{M}=<\mathcal{F},\mathcal{I}>$ 和一个区间 $[t_1,t_2]\in\mathbf{Intv}(T)$，状态项的初值和终值 $\overleftarrow{\vartheta},\overrightarrow{\vartheta},\overleftarrow{\vartheta_1},\overrightarrow{\vartheta_1},\ldots$ 被解释成类型为 $\mathbf{Intv}(T)\ \to\ D$ 的函数，该函数可定义为：

$\mathcal{I}(\overleftarrow{\vartheta},[t_1,t_2])=d$，
当且仅当 存在一个 $\delta>0$，使得 $<\mathcal{F},\mathcal{I}>,[t_1-\delta,t_1]\models_{\boldsymbol{HDC}}[\![\vartheta=d]\!]$ 。

$\mathcal{I}(\overrightarrow{\vartheta},[t_1,t_2])=d$，
当且仅当 存在一个 $\delta>0$，使得 $<\mathcal{F},\mathcal{I}>,[t_2,t_2+\delta]\models_{\boldsymbol{HDC}}[\![\vartheta=d]\!]$ 。

其中，

$$\begin{array}{lcl}
\mathcal{M},[t_1,t_2]\models_{\boldsymbol{HDC}}\phi & \widehat{=} & \mathcal{I}(\phi)([t_1,t_2])=tt \\
\mathcal{M},[t_1,t_2]\not\models_{\boldsymbol{HDC}}\phi & \widehat{=} & \mathcal{I}(\phi)([t_1,t_2])=ff
\end{array}$$

给定一个模型 $\mathcal{M}=<\mathcal{F},\mathcal{I}>$，公式 ϕ 被解释成类型为

$$\mathcal{I}(\phi)\in\mathbf{Intv}(T)\to\{tt,ff\}$$

的函数，它可以根据公式的结构递归定义为：

1. $\mathcal{M},[t_1,t_2]\models_{\boldsymbol{HDC}}X$
   当且仅当 $\mathcal{I}(X)([t_1,t_2])=tt$

2. $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} R^n(\theta_1,\ldots,\theta_n)$
当且仅当 $\mathcal{I}(R^n)(\mathcal{I}(\theta_1)([t_1,t_2]),\ldots,\mathcal{I}(\theta_n)([t_1,t_2])) = tt$

3. $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \neg\phi$
当且仅当 $\mathcal{M},[t_1,t_2] \not\models_{\boldsymbol{HDC}} \phi$

4. $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi \vee \psi$
当且仅当 $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi$ 或者 $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \psi$

5. $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi^\frown\psi$
当且仅当存在 $t \in [t_1,t_2]$, 使得 $\mathcal{M},[t_1,t] \models_{\boldsymbol{HDC}} \phi$ 且 $\mathcal{M},[t,t_2] \models_{\boldsymbol{HDC}} \psi$

6. $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \exists z.\phi$
当且仅当存在一个 z- 等价于 $\mathcal{I}$ 的解释 $\mathcal{I}'$ 使得 $< \mathcal{F},\mathcal{I}' >,[t_1,t_2] \models_{\boldsymbol{HDC}} \phi$ 。其中, $z \in GVar \cup PVar$ 。

给定一个模型 $\mathcal{M} =<< T,\leq>,< D,+0>,m,\mathcal{I} >$ 和一个公式 ϕ, 如果存在一个区间 $[t_1,t_2] \in \mathbf{Intv}(T)$ 使得 $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi$, 则说 $\mathcal{M}$ **满足** ϕ 。我们可以立即将这个概念推广到一类模型上去, 即 ϕ 在一类模型 $\mathcal{C}$ 上是 **可满足的**, 仅当它在 $\mathcal{C}$ 中的某个模型上是可满足的。给定一个公式集合 Γ, 我们说 $\mathcal{M}$ 是 Γ 的一个模型或者说 $\mathcal{M}$ 满足 Γ, 仅当存在一个区间 $[t_1,t_2] \in \mathbf{Intv}(T)$, 使得对 Γ 中的任意公式 ϕ, $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi$ 。我们说公式 ϕ 在模型 $\mathcal{M}$ 上是 **永真的**, 当且仅当对于任意区间 $[t_1,t_2] \in \mathbf{Intv}(T)$, $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi$ 。我们说公式 ψ 在一类模型 $\mathcal{C}$ 上是 **永真的**, 仅当它在 $\mathcal{C}$ 的每个成员上均是永真的。如果 ϕ 在所有模型构成的类上是永真的, 则称 ϕ 是 **永真的** 。也就是说, 对于任意模型 $\mathcal{M} =<< T,\leq>,< D,+,0>,m,\mathcal{I} >$ 和任意区间 $[t_1,t_2] \in \mathbf{Intv}(T)$, $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi$ 。记作 $\models_{\boldsymbol{HDC}} \phi$ 。如果存在一个模型 $\mathcal{M} =<< T,\leq>,< D,+,0>,m,\mathcal{I} >$ 和一个区间 $[t_1,t_2] \in \mathbf{Intv}(T)$, 使得 $\mathcal{M},[t_1,t_2] \models_{\boldsymbol{HDC}} \phi$, 则称 ϕ **可满足** 。

5.5 ω - 规则

[21] 已经证明在抽象时间域上用有穷规则来公理化有穷可变性是不可能的。因此, 当我们讨论 **HDC** 在抽象时间域上的完备性时, **DCR1** 和 **DCR2** 是不能用来公理化程序变量的有穷可变性的。为了公理化有穷可变性, [29] 引进了 ω - 规则。所以从这儿开始, 在本部分讨论中, 我们将用 ω - 规则代替 **DCR1** 和 **DCR2** 。在这里令 $\Omega = \Omega_{hdc}$, 其中:

$$\Omega_{hdc} \;\widehat{=}\; \{\exists x.(\llbracket V = x \rrbracket \vee \llbracket \rrbracket) \mid V \in PVar\}$$

定理 5.10 (可靠性) 用 ω - 规则替换 ***DCR1*** 和 ***DCR2*** 后的 ***HDC*** 的证明系统是可靠的, 即

$$\vdash_{\mathbf{HDC}} \phi \quad \text{蕴涵} \quad \models_{\mathbf{HDC}} \phi$$

证明: 由定理 **5.1**, 易证。 □

5.6 HDC 在抽象时间域上的完备性

在这部分，我们将运用已知的关于 $\boldsymbol{IL}_2$ 的完备性结果来证明 **HDC** 在抽象时间域上的完备性。为此目的，让我们先取一个 $\boldsymbol{IL}_2$ 的语言 $\mathcal{L}_{\boldsymbol{IL}_2}$ ，在它里面仅有一个一元函数符号 g 属于第二种类。我们也取一个 **HDC** 的语言 $\mathcal{L}_{hdc}$ 。我们的证明步骤如下：我们首先构造一个翻译函数 $dc2il$，它可以把 $\mathcal{L}_{hdc}$ 翻译到 $\mathcal{L}_{\boldsymbol{IL}_2}$ 中去；然后，我们证明如果一个 $\mathcal{L}_{hdc}$ 的公式集合 Γ 在 **HDC** 里是协调的，那么 $dc2il(\Gamma)\cup dc2il(Axiom_{hdc})$ 在 $\boldsymbol{IL}_2$ 里是协调的。其中，$Axiom_{hdc}$ 表示 **HDC** 的所有公理实例的集合；根据定理 **5.9**，我们可以得到一个模型 $<\mathcal{F},\mathcal{J}>$ 和一个区间 $[t_1,t_2]$ 使得 $<\mathcal{F},\mathcal{J}>,[t_1,t_2]\models_{\boldsymbol{IL}_2} dc2il(\Gamma)\cup dc2il(Axiom_{hdc})$；最后，根据这个模型和区间，我们可以构造一个 **HDC** 的模型 $<\mathcal{F}',\mathcal{I}>$ 和区间 $[t_1',t_2']$ 使得 $<\mathcal{F},\mathcal{I}>,[t_1',t_2']\models_{\boldsymbol{HDC}}\Gamma$ 。

我们首先在 $\mathcal{L}_{hdc}$ 中的程序变量和 $\mathcal{L}_{\boldsymbol{IL}_2}$ 第二种类的全局变量间建立一个双射 $V\rightarrow y_V$，在 $\mathcal{L}_{hdc}$ 中的谓词符号和 $\mathcal{L}_{\boldsymbol{IL}_2}$ 中除 $\ominus$，$\oplus$，f_l，和 f_r 外的第一种类柔性函数符号间建立一个双射 $R\rightarrow f_R$ 。为了下面能够构造一个从 $\mathcal{L}_{hdc}$ 到 $\mathcal{L}_{\boldsymbol{IL}_2}$ 的一个子集间的双射，我们对 $\mathcal{L}_{\boldsymbol{IL}_2}$ 的语法作适当的限制，即函数符号 f_R 仅能出现在 f_l 和 f_r 的参数里，且仅有那些具有形式 $f_R(\theta_1,\ldots,\theta_n)$，$t_1\ominus t_2$，$t_1\oplus t_2$ 的项能作为 $\ominus$ 和 $\oplus$ 的参数。易见这样的限制是不会影响 $\boldsymbol{IL}_2$ 的证明系统的完备性。

我们定义从 $\mathcal{L}_{hdc}$ 到 $\mathcal{L}_{\boldsymbol{IL}_2}$ 的翻译函数 $dc2il$ 如下：

$$
\begin{array}{lcl}
dc2il(c) & \widehat{=} & c\\
dc2il(x) & \widehat{=} & x\\
dc2il(V) & \widehat{=} & g(y_V)\\
dc2il(f(\vartheta_1,\ldots,\vartheta_n)) & \widehat{=} & f(dc2il(\vartheta_1),\ldots,dc2il(\vartheta_n))\\
dc2il(\overleftarrow{\vartheta}) & \widehat{=} & f_l(dc2il(\vartheta))\\
dc2il(\overrightarrow{\vartheta}) & \widehat{=} & f_r(dc2il(\vartheta))\\
dc2il(f(\theta_1,\ldots,\theta_n)) & \widehat{=} & f(dc2il(\theta_1),\ldots,dc2il(\theta_n))\\
dc2il(\int S) & \widehat{=} & \begin{cases}
f_0 & \text{当}\quad S=0\\
f_1 & \text{当}\quad S=1\\
f_R(dc2il(\vartheta_1),\ldots,dc2il(\vartheta_n)) & \text{当}\quad S=R(\vartheta_1,\ldots,\vartheta_n)\\
\ell-dc2il(\int S_1) & \text{当}\quad S=\neg S_1\\
dc2il(\int S_1)\ominus dc2il(\int S_2) & \text{当}\quad S=S_1\wedge S_2\\
dc2il(\int S_1)\oplus dc2il(\int S_2) & \text{当}\quad S=S_1\vee S_2
\end{cases}\\
dc2il(\phi) & \widehat{=} & \begin{cases}
X & \text{当}\quad \phi=X\\
R(dc2il(\theta_1),\ldots,dc2il(\theta_n)) & \text{当}\quad \phi=R(\theta_1,\ldots,\theta_n)\\
\neg dc2il(\psi) & \text{当}\quad \phi=\neg\psi\\
dc2il(\phi_1)\vee dc2il(\phi_2) & \text{当}\quad \phi=\phi_1\wedge\phi_2\\
dc2il(\phi_1)\wedge dc2il(\phi_2) & \text{当}\quad \phi=\phi_1\vee\phi_2\\
\exists x.dc2il(\psi) & \text{当}\quad \phi=\exists x.\psi\\
\exists y_V.dc2il(\psi) & \text{当}\quad \phi=\exists V.\psi
\end{cases}
\end{array}
$$

其中，$f_0=0$，$f_1=\ell$ 。

对称地，我们定义一个从 $\mathcal{L}_{\boldsymbol{IL}_2}$ 到 $\mathcal{L}_{hdc}$ 的部分翻译函数如下：

$$
\begin{array}{lcl}
il2dc(c) & \widehat{=} & c \\
il2dc(x) & \widehat{=} & x \\
il2dc(g(y_V)) & \widehat{=} & V \\
il2dc(f_l(\theta)) & \widehat{=} & il2\overleftarrow{dc}(\theta) \\
il2dc(f_r(\theta)) & \widehat{=} & il2\overrightarrow{dc}(\theta) \\
il2dc(f(\theta_1,\ldots,\theta_n)) & \widehat{=} & f(il2dc(\theta_1),\ldots,il2dc(\theta_n)) \\
il2dc(f_0) & \widehat{=} & \int 0 \\
il2dc(f_1) & \widehat{=} & \int 1 \\
il2dc(f_R(\theta_1,\ldots,\theta_n)) & \widehat{=} & \int R(il2dc(\theta_1),\ldots,il2dc(\theta_n))
\end{array}
$$

$$
\begin{array}{ll}
& il2dc(f_{R_1}(\theta_{11},\ldots,\theta_{1n_1}) * \ldots * f_{R_m}(\theta_{m1},\ldots,\theta_{mn_m})) \\
\widehat{=} & \int(R_1(il2dc(\theta_{11}),\ldots,il2dc(\theta_{1n_1}))\&\ldots\&R_m(il2dc(\theta_{m1}),\ldots,il2dc(\theta_{mn_m})))
\end{array}
$$

其中，$* \in \{\oplus,\ominus\}$, $\& \in \{\vee,\wedge\}$ 。如果 $* = \oplus$ 那么对应的 $\& = \vee$, 否则所对应的 $\& = \wedge$ 。

$$
il2dc(\phi) \quad \widehat{=} \quad \begin{cases}
X & \text{当} \quad \phi = X \\
R(il2dc(\theta_1),\ldots,il2dc(\theta_n)) & \text{当} \quad \phi = R(\theta_1,\ldots,\theta_n) \\
\neg il2dc(\psi) & \text{当} \quad \phi = \neg\psi \\
il2dc(\phi_1) \wedge il2dc(\phi_2) & \text{当} \quad \phi = \phi_1 \wedge \phi_2 \\
il2dc(\phi_1) \vee il2dc(\phi_2) & \text{当} \quad \phi = \phi_1 \vee \phi_2 \\
\exists x.il2dc(\psi) & \text{当} \quad \phi = \exists x.\psi \\
\exists V.il2dc(\psi) & \text{当} \quad \phi = \exists y_V.\psi
\end{cases}
$$

由上面的 $dc2il$ 和 $il2dc$ 的定义，我们可以得到下面一些性质。

引理 5.12 对任意 $\phi \in \mathcal{L}_{hdc}$, $il2dc(dc2il(\phi)) = \phi$ 。

证明: 由上面的定义，对 ϕ 的结构进行归纳即可。 □

定理 5.11 对任意 $\mathcal{L}_{hdc}$ 里的公式集合 Γ，Γ 在 ***HDC*** 是协调的当且仅当 $dc2il(\Gamma) \cup dc2il(Axiom_{hdc})$ 在 $\mathbf{IL}_2$ 里是协调的。此时，在 $\mathbf{IL}_2$ 证明系统中，取 $\Omega = dc2il(\Omega_{hdc})$ 。

证明: 根据 $dc2il$ 和 $il2dc$ 的定义及引理 **5.12**，易证。 □

以后，我们用 $[\![dc2il(S)]\!]$ 表示 $dc2il(S) = f_1 \wedge f_1 > 0$ 。我们用 “f_{eq}” 代表在 $\mathcal{L}_{hdc}$ 中的谓词 “=” 在 $\mathcal{L}_{IL_2}$ 里对应的第一种类里的柔性函数。

下面，我们将证明一些比较有用的引理。

引理 5.13 (I) 如果 $S_1 \Rightarrow S_2$, 其中 S_1, S_2 是两个状态表达式, 那么 $dc2il(Axiom_{hdc}) \vdash_{\mathbf{IL}_2} dc2il(S_1) \leq dc2il(S_2)$

(II) $dc2il(Axiom_{hdc}) \vdash_{\mathbf{IL}_2} \Phi \Rightarrow \Box\Phi$ 对所有 $\Phi \in dc2il(\Omega_{hdc})$ 。

证明: 我们只需用函数 $dc2il$ 翻译一下 **DCT1** 和 **DCT6** 的证明即可。 □

引理 5.14

(a) $dc2il(\int S) \oplus 0 = \ell \Rightarrow dc2il(\int S) = \ell$
(b) $0 \oplus dc2il(\int S) = \ell \Rightarrow dc2il(\int S) = \ell$
(c) $dc2il(\int S) \ominus \ell = \ell \Rightarrow dc2il(\int S) = \ell$
(d) $\ell \ominus dc2il(\int S) = \ell \Rightarrow dc2il(\int S) = \ell$

证明: 我们仅给出 (a) 的证明，其他可以类似地证明。

(1) $dc2il(\int S) \ominus dc2il(\int 0) = dc2il(\int S \wedge 0) = dc2il(\int 0) = 0$ (**Def., DC6**)
(2) $dc2il(\int S) + dc2il(\int 0) = dc2il(\int S) \ominus 0 + dc2il(\int S) \oplus 0$ (**DC4**)
(3) $dc2il(S) = dc2il(S) \oplus 0 = \ell$ ((1), (2))

引理 5.15 设 S 是一个状态表达式， $V_1, \ldots, V_n$ 是所有在它里面出现的程序变量。那么

$$dc2il(Axiom_{hdc}) \vdash_{\mathbf{IL}_2} \left(\begin{array}{c} [\![dc2il(V_1 = x_1)]\!] \wedge \ldots \wedge [\![dc2il(V_n = x_n)]\!] \\ \Rightarrow dc2il(\int S(V_1, \ldots, V_n)) = dc2il(\int S[x_1/V_1, \ldots, x_n/V_n]) \end{array} \right)$$

证明:

(1) $V_1 = x_1 \wedge \ldots \wedge V_n = x_n$
$\Rightarrow (S(V_1, \ldots, V_n) \Leftrightarrow S[x_1/V_1, \ldots, x_n/V_n])$ (**Ident**)
(2) $dc2il(\int V_1 = x_1) \ominus \ldots \ominus dc2il(\int V_n = x_n)$
$\leq dc2il(\int S(V_1, \ldots, V_n) \Leftrightarrow S[x_1/V_1, \ldots, x_n/V_n])$ ((1), 引理 **5.13**)
(3) $[\![f_{eq}(g(y_{V_1}), x_1)]\!] \wedge \ldots \wedge [\![f_{eq}(g(y_{V_n}), x_n)]\!]$
$\Rightarrow [\![f_{eq}(g(y_{V_1}), x_1) \ominus \ldots \ominus f_{eq}(g(y_{V_n}), x_n)]\!]$ (引理 **5.13, DC4**)
(4) $[\![f_{eq}(g(y_{V_1}), x_1)]\!] \wedge \ldots \wedge [\![f_{eq}(g(y_{V_n}), x_n)]\!]$
$\Rightarrow [\![dc2il(\int S(V_1, \ldots, V_n) \Leftrightarrow S[x_1/V_1, \ldots, x_n/V_n])]\!]$ ((3), **DC3**)
(5) $[\![dc2il(\int S(V_1, \ldots, V_n) \Leftrightarrow S[x_1/V_1, \ldots, x_n/V_n])]\!]$
$\Rightarrow \left(\begin{array}{l} [\![dc2il(\int S(V_1, \ldots, V_n)) \\ \quad \oplus dc2il(\int \neg S[x_1/V_1, \ldots, x_n/V_n])]\!] \\ \wedge [\![dc2il(\int \neg S(V_1, \ldots, V_n)) \\ \quad \oplus dc2il(\int S[x_1/V_1, \ldots, x_n/V_n])]\!] \end{array} \right)$ (引理 **5.13**)
(6) $\left(\begin{array}{l} [\![dc2il(\int \neg S[x_1/V_1, \ldots, x_n/V_n])]\!] \\ \wedge [\![dc2il(\int S(V_1, \ldots, V_n)) \\ \quad \oplus dc2il(\int \neg S[x_1/V_1, \ldots, x_n/V_n])]\!] \\ \wedge [\![dc2il(\int \neg S(V_1, \ldots, V_n)) \\ \quad \oplus dc2il(\int S[x_1/V_1, \ldots, x_n/V_n])]\!] \end{array} \right)$
$\Rightarrow [\![dc2il(\int \neg S(V_1, \ldots, V_n))]\!]$ (引理 **5.14**)

$$(7)\quad \left(\begin{array}{c} \llbracket dc2il(\int S[x_1/V_1,\ldots,x_n/V_n])\rrbracket \\ \wedge\llbracket dc2il(\int S(V_1,\ldots,V_n)) \\ \oplus dc2il(\int\neg S[x_1/V_1,\ldots,x_n/V_n])\rrbracket \\ \wedge\llbracket dc2il(\int\neg S(V_1,\ldots,V_n)) \\ \oplus dc2il(\int S[x_1/V_1,\ldots,x_n/V_n])\rrbracket \end{array}\right)$$

$\Rightarrow \llbracket dc2il(\int S(V_1,\ldots,V_n))\rrbracket$ (引理 **5.14**)

$$(8)\quad \left(\begin{array}{c} \llbracket dc2il(\int S(V_1,\ldots,V_n)) \\ \oplus dc2il(\int\neg S[x_1/V_1,\ldots,x_n/V_n])\rrbracket \\ \wedge\llbracket dc2il(\neg S(V_1,\ldots,V_n)) \\ \oplus dc2il(\int S[x_1/V_1,\ldots,x_n/V_n])\rrbracket \end{array}\right)$$

$\Rightarrow dc2il(\int S(V_1,\ldots,V_n)) = dc2il(\int S[x_1/V_1,\ldots,x_n/V_n])$ ((7), (8), **DC7**)

$$(9)\quad \llbracket f_{eq}(g(y_{V_1}),x_1)\rrbracket\wedge\ldots\wedge\llbracket f_{eq}(g(y_{V_n}),x_n)\rrbracket$$

$\Rightarrow dc2il(S(V_1,\ldots,V_n)) = dc2il(S[x_1/V_1,\ldots,x_n/V_n])$

引理 5.16 如果 c_1 和 c_2 是常量，那么 $\llbracket f_{eq}(g(y_V),c_1)\rrbracket\wedge\llbracket f_{eq}(g(y_V),c_2)\rrbracket\Rightarrow c_1=c_2$

证明:

(1)	$V=c_1\wedge V=c_2\Rightarrow c_1=c_2$	(**Ident**)
(2)	$f_{eq}(g(y_V),c_1)\ominus f_{eq}(g(y_V),c_1)\leq f_{eq}(c_1,c_2)$	(引理 **5.13**)
(3)	$\llbracket f_{eq}(g(y_V),c_1)\rrbracket\wedge\llbracket f_{eq}(g(y_V),c_2)\rrbracket\Rightarrow\llbracket f_{eq}(c_1,c_2)\rrbracket$	(**DC4,** 引理 **5.13**)
(4)	$(c_1\neq c_2)\Rightarrow\llbracket dc2il(c_1\neq c_2)\rrbracket$	(**DC7**)
(5)	$\llbracket dc2il(c_1\neq c_2)\rrbracket\Rightarrow\neg\llbracket f_{eq}(c_1,c_2)\rrbracket$	(**DC4, Def.**)
(6)	$\llbracket f_{eq}(c_1,c_2)\rrbracket\Rightarrow c_1=c_2$	((4), (5))
(7)	$\llbracket f_{eq}(g(y_V),c_1)\rrbracket\wedge\llbracket f_{eq}(g(y_V),c_2)\rrbracket\Rightarrow c_1=c_2$	((3), (6))

在下面的讨论中，我们固定一个 **HDC** 的公式集合 Γ 。

引理 5.17 设 $\Gamma\subset\mathcal{L}_{hdc}$ 是协调的，那么 $\Gamma_0=\{\ell=a\}^\frown\Gamma^\frown\{\ell=b\}$ 也在 ***HDC*** 里是协调的，其中 $a,b>0$ 。

证明: 反设此命题不成立，那么存在 Γ 的一个有穷子集 $\{\phi_1,\ldots,\phi_n\}$ 使得 $\vdash_{\boldsymbol{HDC}}(\ell=a^\frown\phi_1^\frown\ell=b)\wedge\ldots\wedge(\ell=a^\frown\phi_n^\frown\ell=b)\Rightarrow$ **false** 。根据 **ILT9** 可得 $\vdash_{\boldsymbol{HDC}}(\ell=a)^\frown(\phi_1\wedge\ldots\wedge\phi_n)^\frown(\ell=b)\Rightarrow$ **false** 。由邻接规则可得 $\vdash_{\boldsymbol{HDC}}\phi_1\wedge\ldots\wedge\phi_n\Rightarrow$ **false** 。从而与 Γ 是协调的矛盾。 □

而且，根据定理 **5.11,** Γ_0 在 **HDC** 里的协调性可以蕴涵 $dc2il(\Gamma_0)\cup dc2il(Axiom_{hdc})$ 在 $\boldsymbol{IL}_2$ 里的协调性，此时 $\Omega=dc2il(\Omega_{hdc})$ 。因而，根据定理 **5.9,** 存在一个 $\boldsymbol{IL}_2$ 模型 $\mathcal{M}=<\mathcal{F},\mathcal{J}>$，其中 $\mathcal{F}=<<T,\leq>,<D,+,0>,D_1,m>$ 是它的语义框架，和一个区间 $[t_1,t_2]\in\mathbf{Intv}(T)$ 使得 $<\mathcal{F},\mathcal{J}>,[t_1,t_2]\models_{\boldsymbol{IL}_2} dc2il(\Gamma_0)\cup dc2il(Axiom_{hdc})$，并且对任意 $\Phi\in dc2il(\Omega_{hdc})$，它在 $<\mathcal{F},\mathcal{J}>$ 上有有穷可变性。因而，存在一个真子区间 $[t_1',t_2']$ 使得 $t_1<t_1'\leq t_2'<t_2$，$t_1'=t_1+a$，$t_2'=t_2-b$，且 $<\mathcal{F},\mathcal{J}>,[t_1',t_2']\models_{\boldsymbol{IL}_2} dc2il(\Gamma)\cup dc2il(Axiom_{hdc})$ 。

从现在开始，我们将根据 $<\mathcal{F},\mathcal{J}>$ 构造一个 **HDC** 模型 $<\mathcal{F}',\mathcal{I}>$ 使得 $<\mathcal{F}',\mathcal{I}>,[t_1',t_2']\models_{\boldsymbol{HDC}}\Gamma$ 。

设 $\Im \widehat{=} \{\mathcal{J}' \mid < \mathcal{F}, \mathcal{J}' > \text{是 } \boldsymbol{IL}_2 \text{ 的一个模型且 } \mathcal{J}'[\![g]\!] = \mathcal{J}[\![g]\!]\}$ 。

对任意 $\mathcal{J}' \in \Im$，我们构造一个 $\mathcal{L}_{hdc}$ 的解释 $\mathcal{I}'$ 如下：

对于任意 $V \in PVar$，公式 $\Phi = dc2il(\exists x.\lceil V = x \rceil \vee \ell = 0) \in dc2il(\Omega_{hdc})$ 。根据定理 **5.8**，存在 $[t_1, t_2]$ 的一个分割 $t_1 = t_1'' \leq t_2'' \leq \ldots \leq t_n'' = t_2$ 使得 $< \mathcal{F}, \mathcal{J}' >, [t_i'', t_{i+1}''] \models_{\boldsymbol{IL}_2} dc2il(\exists x.\lceil V = x \rceil \vee \ell = 0)$，即对 $i = 1, \ldots, n-1$ $< \mathcal{F}, \mathcal{J}' >, [t_i'', t_{i+1}''] \models_{\boldsymbol{IL}_2} \exists x.\lceil f_{eq}(g(y_V), x) \rceil \vee \ell = 0$ 。这样，$\mathcal{I}'$ 可以定义如下：

$$
(\bullet\bullet)\quad
\begin{array}{lcl}
\mathcal{I}'[\![V]\!](t) & \widehat{=} & \begin{cases} \mathcal{J}''(x) & \begin{array}{l}\text{其中} t_i'' \leq t < t_{i+1}'', \\ \text{且} < \mathcal{F}, \mathcal{J}'' >, [t_i'', t_{i+1}''] \models_{\boldsymbol{IL}_2} \\ \lceil f_{eq}(g(y_V), x) \rceil \vee \ell = 0 \\ \text{而且} \mathcal{J}'' \text{ 是 } x\text{- 等价于 } \mathcal{J}' \end{array} \\ 0 & \text{其它} \end{cases} \\
\mathcal{I}'(x) & \widehat{=} & \mathcal{J}'(x) \\
\mathcal{I}'(c) & \widehat{=} & \mathcal{J}'(c) \\
\mathcal{I}'(f_i^n) & \widehat{=} & \mathcal{J}'(f_i^n) \\
\mathcal{I}'(X) & \widehat{=} & \mathcal{J}'(X) \\
\mathcal{I}'(R_i^n) & \widehat{=} & \mathcal{J}'(R_i^n)
\end{array}
$$

引理 5.18 设 $\mathcal{J}'$ 和 $\mathcal{I}'$ 有 $(\bullet\bullet)$ 式关系。则对任意状态表达式 S 和 $[t_1, t_2]$ 的任意子区间 $[c, d]$，我们有

$$\mathcal{I}'[\![\int S]\!][c, d] \quad = \quad \mathcal{J}'[\![dc2il(\int S)]\!][c, d]$$

证明： 设 $V_1, \ldots, V_n$ 是所有在 S 中出现的程序变量。因为对所有 $i = 1, \ldots, n$，$dc2il(\exists x_i.\lceil V_i = x_i \rceil \vee \ell = 0)$ 在 $[c, d]$ 上具有有穷可变性，所以我们可以构造 $[c, d]$ 的分割 $c = t_1'' \leq t_2'' \leq \ldots \leq t_m'' = d$ 使得对所有 $i = 1, \ldots, n$ 和 $j = 1, \ldots, m-1$ $< \mathcal{F}, \mathcal{J}' >, [t_j'', t_{j+1}''] \models_{\boldsymbol{IL}_2} dc2il(\exists x_i.\lceil V_i = x_i \rceil \vee \ell = 0)$ 。也就是说，对每个 $j = 1, \ldots, m-1$，存在 n 个常量 $c_{j,1}, \ldots, c_{j,n}$ 使得

$$(\mathbf{A}) \qquad < \mathcal{F}, \mathcal{J}' >, [t_j'', t_{j+1}''] \models_{\boldsymbol{IL}_2} dc2il(\lceil V_i = c_{ji} \rceil \vee \ell = 0)$$

由 $(\bullet\bullet)$ 和引理 **5.16**，可得对所有 $i = 1, \ldots, n$ 和 $j = 1, \ldots, m-1$ 有

$$(\mathbf{B}) \qquad < \mathcal{F}, \mathcal{I}' >, [t_j'', t_{j+1}''] \models_{\boldsymbol{IL}_2} \lceil V_i = c_{ji} \rceil \vee \ell = 0$$

由 **(A)** 和引理 **5.15** 可得对所有 $j = 1, \ldots, m-1$ 有

$$(\mathbf{C}) \qquad < \mathcal{F}, \mathcal{J}' >, [t_j'', t_{j+1}''] \models_{\boldsymbol{IL}_2} dc2il(\int S(V_1, \ldots, V_n)) = dc2il(\int S[c_{j,1}/V_1, \ldots, c_{j,n}/V_n])$$

根据 **(B)** 和 **DCT11** 可得对所有 $j = 1, \ldots, m-1$ 有

$$(\mathbf{D}) \qquad < \mathcal{F}, \mathcal{I}' >, [t_j'', t_{j+1}''] \models_{\boldsymbol{IL}_2} \int S(V_1, \ldots, V_n) = \int S[c_{j,1}/V_1, \ldots, c_{j,n}/V_n]$$

因为对任意 $j = 1, \ldots, m-1$，$S[c_{j,1}/V_1, \ldots, c_{j,n}/V_n]$ 是一个刚性状态表达式，所以根据公理 **DC7** 及 **(C)** 和 **(D)** 可得对所有 $j = 1, \ldots, m-1$ 有

$$(\mathbf{E}) \qquad \mathcal{I}'[\![\int S(V_1, \ldots, V_n)]\!][t_j'', t_{j+1}''] = \mathcal{J}'[\![dc2il(\int S(V_1, \ldots, V_n))]\!][t_j'', t_{j+1}'']$$

因此运用 **(E)** 和公理 **DC5** $m-2$ 次可得

$$\mathcal{I}'[\![\int S(V_1,\ldots,V_n)\,]\!][c,d] = \mathcal{J}'[\![\,dc2il(\int S(V_1,\ldots,V_n))\,]\!][c,d]。$$ □

由定理 **5.18,** 我们可以证明下面一个比较重要的定理。

定理 5.12 设 $\mathcal{J}' \in \Im$ 且 $\mathcal{J}'$ 和 $\mathcal{I}'$ 有 (••) 关系。那么对任意一个项 $\theta \in \mathcal{L}_{hdc}$ 和任意区间 $[c,d] \subseteq [t_1', t_2']$, 可得:

$$\mathcal{I}'[\![\,\theta\,]\!][c,d] = \mathcal{J}'[\![\,dc2il(\theta)\,]\!][c,d]。$$

证明:

Case 1: $\theta = x$

由 (••) 直接可证。

Case 2: $\theta = \int S$, 其中 S 是一个状态表达式

由引理 **5.18** 可以立证。

Case 3: $\theta = \overleftarrow{V}$

因为 $\mathcal{J}'[\![\,dc2il(\exists x.\lceil V = x\rceil \vee \ell = 0)\,]\!]$ 在 $[t_1,t_2]$ 上有穷可变, 所以存在 $[t_1,t_2]$ 的一个分割 $t_1 = t_1'' \le t_2'' \le \ldots \le t_n'' = t_2$ 使得对任意 $i = 1,\ldots,n-1$, $< \mathcal{F}, \mathcal{J}' >, [t_i'', t_{i+1}''] \models_{\boldsymbol{IL_2}} dc2il(\exists x.\lceil V = x\rceil \vee \ell = 0)$。因为 $t_1 < t_1' \le t_2' < t_2$, 因此存在 $1 \le i_0 < n$ 使得 $t_{i_0}'' < c \le t_{i_0+1}''$ 且 $< \mathcal{F}, \mathcal{J}' >, [t_{i_0}'', t_{i_0+1}''] \models_{\boldsymbol{IL_2}} dc2il(\exists x.\lceil V = x\rceil \vee \ell = 0)$。也就是说, 存在一个常量 e 使得

$$\mathcal{J}'[\![\,\lceil f_{eq}(v(y_V), e)\rceil \vee \ell = 0\,]\!]([t_{i_0}'', t_{i_0+1}'']) = tt。$$

令 $\delta = c - t_{i_0}''$, 从而 $\delta > 0$。由 (••), 引理 **5.16** 和公理 **(PV1)** 可得

$$\mathcal{J}'[\![\,dc2il(\overleftarrow{V})\,]\!][c,d] = e$$

且

$$< \mathcal{F}, \mathcal{I}' >, [c-\delta, c] \models_{\boldsymbol{HDC}} \lceil V = e\rceil$$

根据 **HDC** 的解释的定义, 即

$$\mathcal{I}'[\![\ \overleftarrow{V}\ ]\!][t_1', t_2'] = e \quad \text{当且仅当存在 } \delta > 0 \text{ 使得 } < \mathcal{F}, \mathcal{I}' >, [t_1' - \delta, t_1'] \models_{\boldsymbol{HDC}} \lceil V = e\rceil$$

因此可得 $\mathcal{I}'[\![\ \overleftarrow{V}\ ]\!][c,d] = e = \mathcal{J}'[\![\,dc2il(\overleftarrow{V})\,]\!][c,d]$。

Case 4: $\theta = \overleftarrow{\vartheta}$, 其中 ϑ 是刚性的

由定理 **4.1 (III)** 可得

$$\mathcal{I}'[\![\ \vartheta\ ]\!][c,d] = \mathcal{I}'[\![\ \overleftarrow{\vartheta}\ ]\!][c,d]$$

且

$$\mathcal{J}'[\![\ dc2il(\vartheta)\ ]\!][c,d] = \mathcal{J}'[\![\ dc2il(\overleftarrow{\vartheta})\ ]\!][c,d]。$$

因为根据语法定义, ϑ 也是一个项, 运用归纳假设可得

$$\mathcal{I}'[\![\ \overleftarrow{\vartheta}\ ]\!][c,d] = \mathcal{J}'[\![\ dc2il(\overleftarrow{\vartheta})\ ]\!][c,d]。$$

Case 5: $\theta = f(\vartheta, \overleftarrow{\ldots}, \vartheta_n)$ 。

根据定理 **4.1 (IV)** 可得

$$\mathcal{I}'[\![\ f(\vartheta_1, \overleftarrow{\ldots}, \vartheta_n)\ ]\!][c,d] = \mathcal{I}'[\![\ f\ ]\!](\mathcal{I}'[\![\ \overleftarrow{\vartheta_1}\ ]\!][c,d], \ldots, \mathcal{I}'[\![\ \overleftarrow{\vartheta_n}\ ]\!][c,d])。$$

根据归纳假设得

$$\mathcal{I}'[\![\ \overleftarrow{\vartheta_1}\ ]\!][c,d] = \mathcal{J}'[\![\ dc2il(\overleftarrow{\vartheta_1})\ ]\!][c,d], \ldots, \mathcal{I}'[\![\ \overleftarrow{\vartheta_n}\ ]\!][c,d] = \mathcal{J}'[\![\ dc2il(\overleftarrow{\vartheta_n})\ ]\!][c,d]。$$

因此由定理 **4.1 (IV)** 得

$$\begin{aligned} & \mathcal{I}'[\![\ f(\vartheta_1, \overleftarrow{\ldots}, \vartheta_n)\ ]\!][c,d] \\ = \ & \mathcal{J}'[\![\ f\ ]\!](\mathcal{J}'[\![\ dc2il(\overleftarrow{\vartheta_1})\ ]\!][c,d], \ldots, \mathcal{J}'[\![\ dc2il(\overleftarrow{\vartheta_n})\ ]\!][c,d]) \\ = \ & \mathcal{J}'[\![\ dc2il(f(\vartheta_1, \overleftarrow{\ldots}, \vartheta_n))\ ]\!][c,d] \end{aligned}$$

Case 6 $\theta = \overrightarrow{V}$ 。

类似于 **Case 3.**

Case 7 $\theta = \overrightarrow{\vartheta}$, 其中 ϑ 是刚性的

类似于 **Case 4.**

Case 8 $\theta = f(\vartheta_1, \overrightarrow{\ldots}, \vartheta_n)$ 。

类似于 **Case 5.**

Case 9 $\theta = f(\theta_1, \ldots, \theta_n)$ 。

$$\begin{aligned} & \mathcal{I}'[\![\ f(\theta_1, \ldots, \theta_n)\ ]\!][c,d] \\ = \ & \mathcal{I}'[\![\ f\ ]\!](\mathcal{I}'[\![\ \theta_1\ ]\!][c,d], \ldots, \mathcal{I}'[\![\ \theta_n\ ]\!][c,d]) && (\textbf{Def.}) \\ = \ & \mathcal{J}'[\![\ f\ ]\!](\mathcal{J}'[\![\ dc2il(\theta_1)\ ]\!][c,d], \ldots, \mathcal{J}'[\![\ dc2il(\theta_n)\ ]\!][c,d]) && (\text{归纳假设}) \\ = \ & \mathcal{J}'[\![\ dc2il(f(\theta_1, \ldots, \theta_n))\ ]\!][c,d] && (\textbf{Def.}) \end{aligned}$$

□

现在，我们可以给出满足 (••) 的两个解释 $\mathcal{I}'$ 和 $\mathcal{J}'$ 关于公式的对应关系。这种对应关系是：

定理 5.13 设 $\mathcal{J}' \in \Im$ 且 $\mathcal{J}'$ 和 $\mathcal{I}'$ 有 (••) 关系。那么对任意公式 $\phi \in \mathcal{L}_{hdc}$ 和 $[c,d] \subset [t'_1, t'_2]$，$< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\mathbf{HDC}} \phi$ 当且仅当 $< \mathcal{F}, \mathcal{J}' >, [c,d] \models_{\mathbf{IL}_2} dc2il(\phi)$ 。

证明： 我们通过对 ϕ 的结构归纳来进行证明。其证明如下：

Case 1: $\phi \widehat{=} X$ 。
由 (••) 立得。

Case 2: $\phi \widehat{=} R(\theta_1, \ldots, \theta_n)$ 。
由定理 **5.12** 立即可证。

Case 3 $\phi \widehat{=} \neg\psi$ 。
$< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\boldsymbol{HDC}} \neg\psi$ 当且仅当 $< \mathcal{F}, \mathcal{I}' >, [c,d] \not\models_{\boldsymbol{HDC}} \psi$ 。根据归纳假设，即当且仅当 $< \mathcal{F}, \mathcal{J}' >, [c,d] \not\models_{\boldsymbol{IL}_2} dc2il(\psi)$，根据 $dc2il$ 的定义，也就是当且仅当 $< \mathcal{F}, \mathcal{J}' >, [c,d] \models_{\boldsymbol{IL}_2} dc2il(\neg\psi)$ 。

Case 4: $\phi \widehat{=} \psi \vee \varphi$ 。
$< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\boldsymbol{HDC}} \psi \vee \varphi$ 当且仅当 $< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\boldsymbol{HDC}} \psi$ 或者 $< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\boldsymbol{HDC}} \varphi$ 。由归纳假设，即当且仅当 $< \mathcal{F}, \mathcal{J}' >, [c,d] \models_{\boldsymbol{IL}_2} dc2il(\psi)$ 或者 $< \mathcal{F}, \mathcal{J}' >, [c,d] \models_{\boldsymbol{IL}_2} dc2il(\varphi)$ 。根据 $dc2il$ 的定义，即当且仅当 $< \mathcal{F}, \mathcal{J}' >, [c,d] \models_{\boldsymbol{IL}_2} dc2il(\psi \vee \varphi)$ 。

Case 5: $\phi = \psi^\frown\varphi$ 。
$< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\boldsymbol{HDC}} \psi^\frown\varphi$ 当且仅当存在 $m \in [c,d]$ 使得 $< \mathcal{F}, \mathcal{I}' >, [c,m] \models_{\boldsymbol{HDC}} \psi$ 且 $< \mathcal{F}, \mathcal{I}' >, [m,d] \models_{\boldsymbol{HDC}} \varphi$ 。由归纳假设，即当且仅当存在 $m \in [c,d]$ 使得 $< \mathcal{F}, \mathcal{J}' >, [c,m] \models_{\boldsymbol{IL}_2} dc2il(\psi)$ 且 $< \mathcal{F}, \mathcal{J}' >, [m,d] \models_{\boldsymbol{IL}_2} dc2il(\varphi)$ 。由 $dc2il$ 的定义，即当且仅当 $< \mathcal{F}, \mathcal{J}' >, [c,d] \models_{\boldsymbol{IL}_2} dc2il(\psi^\frown\varphi)$ 。

Case 6: $\phi \widehat{=} \exists V.\psi$ 。
"$\Leftarrow$"
设 $< \mathcal{F}, \mathcal{J}' >, [c,d] \models_{\boldsymbol{IL}_2} dc2il(\exists V.\psi)$，即存在一个解释 $\mathcal{J}''$，它 y_V- 等价于 $\mathcal{J}'$，使得 $< \mathcal{F}, \mathcal{J}'' >, [c,d] \models_{\boldsymbol{IL}_2} dc2il(\psi)$ 。由归纳假设可得，$< \mathcal{F}, \mathcal{I}'' >, [c,d] \models_{\boldsymbol{HDC}} \psi$ 。明显地，$\mathcal{I}''$ 是 V- 等价于 $\mathcal{I}'$，因而 $< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\boldsymbol{HDC}} \phi$ 。
"$\Rightarrow$"
设 $< \mathcal{F}, \mathcal{I}' >, [c,d] \models_{\boldsymbol{HDC}} \phi$ 。那么存在一个 **HDC** 的解释 $\mathcal{I}''$，它是 V- 等价于 $\mathcal{I}'$ 且使得 $< \mathcal{F}, \mathcal{I}'' >, [c,d] \models_{\boldsymbol{HDC}} \psi$ 。令 $t''_0, t''_1, \ldots, t''_n, t''_{n+1} \in T$ 使得 $t''_0 < c = t''_1 \leq \ldots \leq t''_n = d < t''_{n+1}$ 且对 $i = 0, \ldots, n$，$\mathcal{I}''(V)$ 在 $[t'_i, t'_{i+1})$ 上是常量。假设这些 $n+1$ 个常量是 $c_0, \ldots, c_n$ 。因为 $< \mathcal{F}, \mathcal{I}'' >$ 是 **HDC** 的一个模型，所以上述假设是合理的。因为 $\mathcal{M} =< \mathcal{F}, \mathcal{J} >$ 是 $\boldsymbol{IL}_2$ 的一个模型，根据公理 **QV** 可得对任意 $i = 0, \ldots, n$，存在一个 $d_i \in D_1$ 使得在 $[t''_i, t''_{i+1}]$ 的任意子区间 $[b,e]$ 上，

$$(*) \quad \mathcal{J}[\![g]\!](d_i, [b,e]) = c_i \quad \text{如果} \quad \mathcal{I}''[\![V]\!](t) = c_i \text{ 对任意 } t \in [b,e]$$

运用公理 **HDC1-HDC3** n 次可得, 存在一个 $d \in D_1$ 使得对所有 $i = 0, \ldots, n+1$ 和 $t \in [t''_i, t''_{i+1})$

$$\mathcal{I}''(V)(t) = c_i \qquad \text{当且仅当} \qquad < \mathcal{F}, \mathcal{J}'' >, [t''_i, t''_{i+1} \models_{\boldsymbol{IL}_2} \ [\![f_{eq}(g(d), c_i)]\!]$$

令 $\mathcal{J}''(y_V) = d$, 其中 d 是上述构造的; 对于 $\mathcal{L}_{\boldsymbol{IL}_2}$ 中的其它符号 z, 令 $\mathcal{J}''(z) = \mathcal{J}'(z)$ 。因此 $\mathcal{J}''$ 是 y_V- 等价于 $\mathcal{J}'$ 且 $\mathcal{J}''$ 和 $\mathcal{I}''$ 有 (●●) 中所列的关系。由归纳假设, $\mathcal{J}'', [c, d] \models_{\boldsymbol{IL}_2} dc2il(\psi)$, 因而由 $dc2il$ 的定义得 $\mathcal{J}', [c, d] \models_{\boldsymbol{IL}_2} dc2il(\phi)$ 。

□

引理 5.19 如果 Γ 在 ***HDC*** 里是协调的, 那么 Γ 是可满足的。

证明: 根据上面的证明过程, 因为 $\mathcal{J} \in \Im$, 所以根据定理 **5.13** 可得 $< \mathcal{F}, \mathcal{I} >, [t'_1, t'_2] \models_{\boldsymbol{HDC}} \Gamma$, 其中 $\mathcal{F} = << T, \leq >, < D, +, 0 >, D_1, m >$ 。现在, 令 $\mathcal{F}' = << T, \leq > , < D, +, 0 >, m >$ 。易证 $< \mathcal{F}', \mathcal{I} >, [t'_1, t'_2] \models_{\boldsymbol{HDC}} \Gamma$ 。 □

定理 5.14 (完备性) 如果 $\models_{\mathbf{HDC}} \phi$ 那么 $\vdash_{\mathbf{HDC}} \phi$ 。

证明: 反设 $\models_{\boldsymbol{HDC}} \phi$, 但是 $\not\vdash_{\boldsymbol{HDC}} \phi$ 。因此, $\{\neg\phi\}$ 在 **HDC** 的证明系统里是协调的。由定理 **5.19** 知, 存在一个模型 $< \mathcal{F}, \mathcal{I} >$ 和一个区间 $[t_1, t_2]$ 使得 $< \mathcal{F}, \mathcal{I} >, [t_1, t_2] \models_{\boldsymbol{HDC}} \neg\phi$ 。从而与 $\models_{\boldsymbol{HDC}} \phi$ 矛盾。因此, $\vdash_{\boldsymbol{HDC}} \phi$ 。 □

5.7 关于 HDC 的完备性的讨论

在传统上, 人们喜欢讨论时段演算的两种不同的完备性: 一种是它在抽象时间域上的完备性 [29]; 另一种是以实数为时间域的相对完备性 [34] 。上面, 我们已经得到了高阶时段演算的第一类完备性。我们也可以得到高阶时段演算在 [34] 意义上的相对完备性, 但是为了保证区间时序逻辑和实数的永真公式翻译到 **HDC** 后仍为 **HDC** 的永真公式, 我们必须对区间时序逻辑的语法加上许多限制, 这样的相对完备性的意义不是太大。

在 [57] 中指出, 若我们考虑时段演算的另一种相对完备性, 即相对于所有关于实数的永真公式的完备性, 那么如果我们能够证明区间时序逻辑相对于所有关于实数的永真公式是完备的, 则我们可以用证明在抽象时间域上完备性的技巧来证明时段演算相对于所有关于实数的永真公式也是完备的。这个结论对于高阶时段演算及时段演算的变种也是成立的。但是如何证明区间时序逻辑相对于所有关于实数的永真公式的完备性仍旧是一个开问题。

第 六 章 实时语义

6.1 超稠密计算

在计算系统中，许多程序是顺序执行的，且执行它的一些语句所消耗的时间用其环境中大的时间粒度来度量的话显得非常小以至可以忽略不计。由一系列认为不消耗时间的操作构成的计算称为超稠密计算 [49] 。为了能够用时段演算来刻画程序的超稠密计算， [10] 引进了超稠密切割算子 $\bullet$ 。使用超稠密切割算子，我们可以用一个不可见的中间状态将两个公式连接起来。因此，对于 V 的任意初值 x，用 $x+2$ 作为中间状态，下面公式

$$(\exists x.\ (\overleftarrow{V}= x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V}= x+2)) \bullet (\exists x.\ (\overleftarrow{V}= x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V}= x+1))$$

将简化成

$$\exists x.\ (\overleftarrow{V}= x)\ \wedge\ [\![\ ]\!]\ \wedge\ (\overrightarrow{V}= x+3)$$

[45] 指出超稠密切割算子可以在高阶时段演算中定义。假设 $V_1, \ldots, V_n$ 是在 $\phi(V_1, ..., V_n)$ 和 $\psi(V_1, ..., V_n)$ 中出现的所有自由程序变量。我们可以定义超稠密切割算子如下：

$$\begin{array}{l}\phi(V_1, ..., V_n) \bullet \psi(V_1, ..., V_n)\\ \widehat{=}\ \ \exists x_1, ..., x_n, V_{11}, ..., V_{1n}, V_{21}, ..., V_{2n}.\\ (\phi(V_{11}, ..., V_{1n}) \wedge \bigwedge_{i=1}^{n}((\llbracket V_i = V_{1i} \rrbracket \vee [\![\ ]\!]) \wedge (\overleftarrow{V_i}=\overleftarrow{V_{1i}}) \wedge (\overrightarrow{V_{1i}}= x_i)))\\ \frown(\psi(V_{21}, ..., V_{2n}) \wedge \bigwedge_{i=1}^{n}((\llbracket V_i = V_{2i} \rrbracket \vee [\![\ ]\!]) \wedge (\overrightarrow{V_i}=\overrightarrow{V_{2i}}) \wedge (\overleftarrow{V_{2i}}= x_i)))\end{array}$$

定义的右边说明我们可以用 $(V_i = x_i)\ (i = 1, ..., n)$ 作为不可见的中间状态将公式 ϕ 和 ψ 连接起来。公式 ϕ 中的 V_i 的值 x_i 通过不可见的中间状态立即传递给公式 ψ 。这样，程序的顺序复合操作的语义可以定义为

$$[\![\mathcal{P}_1; \mathcal{P}_2]\!]\ \widehat{=}\ [\![\mathcal{P}_1]\!] \bullet [\![\mathcal{P}_2]\!]$$

根据定理 **4.1 (I)&(II), HDC1** 和 **HDC2**，我们可以建立关于程序变量在顺序复合操作中值的传递的定理。

定理 6.1 如果 $\leftarrow, \rightarrow \notin \phi, \psi$，那么

$$\begin{array}{ll} & ((\overleftarrow{V}= x_1) \wedge \phi \wedge (\overrightarrow{V}= x_2)) \bullet ((\overleftarrow{V}= x_3) \wedge \psi \wedge (\overrightarrow{V}= x_4))\\ \Leftrightarrow & (\phi \frown \psi) \wedge (\overleftarrow{V}= x_1) \wedge (\overrightarrow{V}= x_4) \wedge\ (x_2 = x_3)\end{array}$$

证明: 设 V 是唯一自由出现于 ϕ 和 ψ 中的程序变量。我们先证明定理的充分部分。

$$
\begin{array}{lll}
 & (\overleftarrow{V}= x_1) \wedge \phi \wedge (\overrightarrow{V}= x_2)) \bullet ((\overleftarrow{V}= x_3) \wedge \psi \wedge (\overrightarrow{V}= x_4)) & \\
\Rightarrow & \exists x, V_1, V_2. \left(\begin{array}{c} \left(\begin{array}{c} (\overleftarrow{V_1}= x_1) \wedge \phi[V_1/V] \wedge (\overrightarrow{V_1}= x_2) \\ \wedge(\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \\ \wedge(\overleftarrow{V}=\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}= x) \end{array} \right) \\ \frown \left(\begin{array}{c} (\overleftarrow{V_2}= x_3) \wedge \psi[V_2/V] \wedge (\overrightarrow{V_2}= x_4) \\ \wedge(\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \\ \wedge(\overrightarrow{V}=\overrightarrow{V_2}) \wedge (\overleftarrow{V_2}= x) \end{array} \right) \end{array} \right) & (\textbf{Def}\ \bullet) \\
\Rightarrow & \exists x, V_1, V_2. \left(\begin{array}{c} \left(\begin{array}{c} \phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \\ \wedge(\overleftarrow{V}=\overleftarrow{V_1}= x_1) \wedge (\overrightarrow{V_1}= x = x_2) \end{array} \right) \\ \frown \left(\begin{array}{c} \psi[V_2/V] \wedge (\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \\ \wedge\ \ (\overrightarrow{V}=\overrightarrow{V_2}= x_4) \wedge (\overleftarrow{V_2}= x = x_3) \end{array} \right) \end{array} \right) & (\textbf{PL}) \\
\Rightarrow & \left(\begin{array}{l} (\overleftarrow{V}= x_1) \wedge (\overrightarrow{V}= x_4) \wedge (x_2 = x_3) \\ \wedge \exists V_1, V_2. \left(\begin{array}{c} (\phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket)) \\ \frown\ \ (\psi[V_2/V] \wedge (\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket)) \end{array} \right) \end{array} \right) & \left(\begin{array}{l} \text{定理 } \textbf{4.1 (I)} \\ \textbf{\&(II), IL4} \end{array} \right) \\
\Rightarrow & (\overleftarrow{V}= x_1) \wedge (\overrightarrow{V}= x_4) \wedge\ \ (x_2 = x_3) \wedge \exists V_1, V_2.\ (\phi \frown \psi) & (\text{定理 } \textbf{4.2 (II)}) \\
\Rightarrow & (\overleftarrow{V}= x_1) \wedge (\overrightarrow{V}= x_4) \wedge\ \ (x_2 = x_3) \wedge (\phi \frown \psi) & (\textbf{PL})
\end{array}
$$

定理的必要部分可以证明如下。

$$
\begin{array}{lll}
 & (\phi \frown \psi) \wedge (\overleftarrow{V}= x_1) \wedge (\overrightarrow{V}= x_4) \wedge\ \ (x_2 = x_3) & \\
\Rightarrow & (x_2 = x_3) \wedge ((\phi \wedge (\overleftarrow{V}= x_1)) \frown (\psi \wedge (\overrightarrow{V}= x_4))) & (\text{定理 } \textbf{4.1 (I)\&(II)}) \\
\Rightarrow & \left(\begin{array}{l} (x_2 = x_3) \\ \wedge((\overleftarrow{V}= x_1) \wedge \phi \wedge (\llbracket V = V \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}=\overleftarrow{V})) \\ \frown(\psi \wedge (\overrightarrow{V}= x_4) \wedge (\llbracket V = V \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}=\overrightarrow{V})) \end{array} \right) & (\textbf{DCT5}) \\
\Rightarrow & \left(\begin{array}{l} (x_2 = x_3) \\ \wedge\ \exists V_1. \left(\begin{array}{c} (\overleftarrow{V_1}= x_1) \wedge \phi[V_1/V] \\ \wedge(\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}=\overleftarrow{V_1}) \end{array} \right) \\ \frown \exists V_2. \left(\begin{array}{c} \psi[V_2/V] \wedge (\overrightarrow{V_2}= x_4) \\ \wedge\ \ (\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}=\overrightarrow{V_2}) \end{array} \right) \end{array} \right) & (\mathrm{Q_V}) \\
\Rightarrow & \left(\begin{array}{l} (x_2 = x_3) \\ \wedge \exists V_1. \left(\begin{array}{c} (\overleftarrow{V_1}= x_1) \wedge \phi[V_1/V] \wedge (\overleftarrow{V}=\overleftarrow{V_1}) \\ \wedge(\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V_1}= x_2) \end{array} \right) \\ \frown \exists V_2. \left(\begin{array}{c} \psi[V_2/V] \wedge (\overrightarrow{V_2}= x_4) \wedge (\overrightarrow{V}=\overrightarrow{V_2}) \\ \wedge(\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V_2}= x_3) \end{array} \right) \end{array} \right) & (\textbf{HDC1\&2})
\end{array}
$$

$$
\begin{array}{lll}
\Rightarrow & \exists x, V_1, V_2. \left(\begin{array}{c} \left(\begin{array}{l} (\overleftarrow{V_1}= x_1) \wedge \phi[V_1/V] \wedge (\overrightarrow{V_1}= x_2) \\ \wedge(\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \\ \wedge(\overleftarrow{V}=\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}= x) \end{array} \right) \\ \frown \left(\begin{array}{l} (\overleftarrow{V_2}= x_3) \wedge \psi[V_2/V] \wedge (\overrightarrow{V_2}= x_4) \\ \wedge(\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \\ \wedge(\overrightarrow{V}=\overrightarrow{V_2}) \wedge (\overleftarrow{V_2}= x) \end{array} \right) \end{array} \right) & (\mathbf{PL}) \\
\Rightarrow & ((\overleftarrow{V}= x_1) \wedge \phi \wedge (\overrightarrow{V}= x_2)) \bullet ((\overleftarrow{V}= x_3) \wedge \psi \wedge (\overrightarrow{V}= x_4)) & (\mathbf{Def}\ \bullet)
\end{array}
$$

定理 **6.1** 的一个推论是如果 $\leftarrow, \rightarrow \notin \phi, \psi$, 那么

$$\phi \bullet \psi \ \Leftrightarrow\ \phi \frown \psi$$

使用定理 **6.1**, 我们易证程序的合成规则

$$\llbracket V := V + 2; V := V + 1 \rrbracket \ \Leftrightarrow\ \llbracket V := V + 3 \rrbracket$$

证明:

$$
\begin{array}{lll}
 & \llbracket V := V + 2; V := V + 1 \rrbracket & \\
\Leftrightarrow & \left(\begin{array}{l} \quad (\exists x.\ (\overleftarrow{V}= x)\ \wedge\ \llbracket\ \rrbracket\ \wedge\ (\overrightarrow{V}= x + 2)) \\ \bullet\ \ (\exists x.\ (\overleftarrow{V}= x)\ \wedge\ \llbracket\ \rrbracket \wedge\ (\overrightarrow{V}= x + 1)) \end{array} \right) & \\
\Leftrightarrow & \exists x_1, x_2. \left(\begin{array}{l} \quad ((\overleftarrow{V}= x_1)\ \wedge\ \llbracket\ \rrbracket \wedge\ (\overrightarrow{V}= x_1 + 2)) \\ \bullet\ \ ((\overleftarrow{V}= x_2)\ \wedge\ \llbracket\ \rrbracket\ \wedge\ (\overrightarrow{V}= x_2 + 1)) \end{array} \right) & (\mathbf{IL5}) \\
\Leftrightarrow & \exists x_1, x_2. \left(\begin{array}{l} \quad (\llbracket\ \rrbracket \frown \llbracket\ \rrbracket) \wedge (\overleftarrow{V}= x_1) \\ \wedge\ \ (\overrightarrow{V}= x_2 + 1) \wedge (x_2 = x_1 + 2) \end{array} \right) & (\text{定理 } \mathbf{6.1}) \\
\Leftrightarrow & \exists x.\ \llbracket\ \rrbracket \wedge (\overleftarrow{V}= x) \wedge (\overrightarrow{V}= x + 3) & (\mathbf{ILT1, PL}) \\
\Leftrightarrow & \llbracket V := V + 3 \rrbracket &
\end{array}
$$

程序的顺序复合操作具有许多漂亮的性质, 例如, 它满足结合律, 并且 skip 是它的单位元。

结合律

$$\llbracket \mathcal{P}_1; (\mathcal{P}_2; \mathcal{P}_3) \rrbracket \ \Leftrightarrow\ \llbracket (\mathcal{P}_1; \mathcal{P}_2); \mathcal{P}_3 \rrbracket$$

这可以由 $\bullet$ 满足结合律推导出。

$$\phi \bullet (\psi \bullet \varphi) \ \Leftrightarrow\ (\phi \bullet \psi) \bullet \varphi$$

$\bullet$ 满足结合律可以证明如下, 其中我们假设在 ϕ, ψ 和 φ 中仅有程序变量 V 自由出现。

$$
\begin{array}{ll}
 & \phi \bullet (\psi \bullet \varphi) \\
\Leftrightarrow & \exists x_1, V_1, V_2. \\
 & (\phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}{=}\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}{=} x_1)) \\
 & \frown((\psi \bullet \varphi)[V_2/V] \wedge (\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \\
 & \wedge(\overrightarrow{V}{=}\overrightarrow{V_2}) \wedge (\overleftarrow{V_2}{=} x_1)) & (\textbf{Def}\bullet) \\
\Leftrightarrow & \exists x_1, V_1, V_2. \\
 & (\phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}{=}\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}{=} x_1)) \\
 & \frown((\exists x_2, V_3, V_4.\ (\psi[V_3/V] \wedge (\llbracket V_2 = V_3 \rrbracket \vee \llbracket\ \rrbracket) \\
 & \wedge(\overleftarrow{V_2}{=}\overleftarrow{V_3}) \wedge (\overrightarrow{V_3}{=} x_2)) \\
 & \quad \frown(\varphi[V_4/V] \wedge (\llbracket V_2 = V_4 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V_2}{=}\overrightarrow{V_4}) \wedge (\overleftarrow{V_4}{=} x_2))) \\
 & \quad \wedge(\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}{=}\overrightarrow{V_2}) \wedge (\overleftarrow{V_2}{=} x_1)) & (\textbf{Def}\bullet) \\
\Leftrightarrow & \exists x_1, x_2, V_1, V_3, V_4. \\
 & (\phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}{=}\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}{=} x_1)) \\
 & \frown(((\psi[V_3/V] \wedge (\llbracket V = V_3 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V_3}{=} x_1) \wedge (\overrightarrow{V_3}{=} x_2)) \\
 & \quad \frown(\varphi[V_4/V] \wedge (\llbracket V = V_4 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}{=}\overrightarrow{V_4}) \wedge (\overleftarrow{V_4}{=} x_2))) & (\text{定理 } \textbf{4.3 (I)} \\
 & \quad \wedge\ \exists V_2.\ (\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}{=}\overrightarrow{V_2}) \wedge (\overleftarrow{V_2}{=} x_2)) & \textbf{\&(II)}) \\
\Leftrightarrow & \exists x_1, x_2, V_1, V_3, V_4. \\
 & (\phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}{=}\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}{=} x_1)) \\
 & \frown(((\psi[V_3/V] \wedge (\llbracket V = V_3 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V_3}{=} x_1) \wedge (\overrightarrow{V_3}{=} x_2)) \\
 & \quad \frown(\varphi[V_4/V] \wedge (\llbracket V = V_4 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}{=}\overrightarrow{V_4}) \wedge (\overleftarrow{V_4}{=} x_2))) \\
 & \quad \wedge\ \exists V_2.\ (\llbracket V = V_2 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}{=}\overrightarrow{V_2})) & (\textbf{HDC1}) \\
\Leftrightarrow & \exists x_1, x_2, V_1, V_3, V_4. \\
 & (\phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}{=}\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}{=} x_1)) \\
 & \frown((\psi[V_3/V] \wedge (\llbracket V = V_3 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V_3}{=} x_1) \wedge (\overrightarrow{V_3}{=} x_2)) & (\textbf{DCT5}, \\
 & \quad \frown(\varphi[V_4/V] \wedge (\llbracket V = V_4 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}{=}\overrightarrow{V_4}) \wedge (\overleftarrow{V_4}{=} x_2))) & \textbf{DCT1}) \\
\end{array}
$$

类似地，我们可以证明

$$
\begin{array}{ll}
 & (\phi \bullet \psi) \bullet \varphi \\
\Leftrightarrow & \exists x_1, x_2, V_1, V_3, V_4.\ ((\phi[V_1/V] \wedge (\llbracket V = V_1 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V}{=}\overleftarrow{V_1}) \wedge (\overrightarrow{V_1}{=} x_1)) \\
 & \frown(\psi[V_3/V] \wedge (\llbracket V = V_3 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overleftarrow{V_3}{=} x_1) \wedge (\overrightarrow{V_3}{=} x_2))) \\
 & \quad \frown(\varphi[V_4/V] \wedge (\llbracket V = V_4 \rrbracket \vee \llbracket\ \rrbracket) \wedge (\overrightarrow{V}{=}\overrightarrow{V_4}) \wedge (\overleftarrow{V_4}{=} x_2)) \\
\end{array}
$$

由公理 IL3 和上面的公式，我们可以得到

$$\phi \bullet (\psi \bullet \varphi) \ \Leftrightarrow\ (\phi \bullet \psi) \bullet \varphi$$

单位元

skip 除了接收初值，然后立即将接收到的值作为终值传递出去外，不干任何事情，也就是， skip 的语义可以形式化为

$$[\![\mathsf{skip}]\!] \;\widehat{=}\; (\llbracket\ \rrbracket \wedge (\overleftarrow{V}{=}\overrightarrow{V})) \quad \text{(简记为 } I\text{)}$$

其中，我们假设在 skip 的上下文环境中仅有 V 这个程序变量自由出现。为了要证

$$[\![\mathcal{P};\mathsf{skip}]\!] \;\Leftrightarrow\; [\![\mathcal{P}]\!] \;\Leftrightarrow\; [\![\mathsf{skip};\mathcal{P}]\!]$$

我们必须证明对于所有仅含有 V 作为自由程序变量的公式 ϕ，我们有

$$\phi \bullet I \;\Leftrightarrow\; I \bullet \phi \;\Leftrightarrow\; \phi$$

证明：

$$\begin{array}{lll}
& \phi \bullet I & \\
\Leftrightarrow & \exists x, V_1, V_2.\left(\begin{array}{c}\left(\begin{array}{c}\phi[V_1/V]\wedge(\llbracket V=V_1\rrbracket\vee\llbracket\ \rrbracket)\\ \wedge(\overleftarrow{V}{=}\overleftarrow{V_1})\wedge(\overrightarrow{V_1}{=}x)\end{array}\right)\\ \frown((\overleftarrow{V_2}{=}\overrightarrow{V_2})\wedge\llbracket\ \rrbracket\wedge(\overrightarrow{V}{=}\overrightarrow{V_2})\wedge(\overleftarrow{V_2}{=}x))\end{array}\right) & (\mathbf{Def}\bullet)\\
\Leftrightarrow & \exists x.\left(\begin{array}{c}\exists V_1.\left(\begin{array}{c}(\phi[V_1/V]\wedge(\llbracket V=V_1\rrbracket\vee\llbracket\ \rrbracket)\\ \wedge(\overleftarrow{V}{=}\overleftarrow{V_1})\wedge(\overrightarrow{V_1}{=}x)\end{array}\right)\\ \frown\exists V_2.(\overleftarrow{V_2}{=}\overrightarrow{V_2}{=}x\wedge\llbracket\ \rrbracket)\wedge\ (\overrightarrow{V}{=}x)\end{array}\right) & \left(\begin{array}{l}\text{定理 }\mathbf{4.1\ (II)},\\ \text{定理 }\mathbf{4.3}\end{array}\right)\\
\Leftrightarrow & \exists x, V_1.\left(\begin{array}{c}\phi[V_1/V]\wedge(\llbracket V=V_1\rrbracket\vee\llbracket\ \rrbracket)\\ \wedge(\overleftarrow{V}{=}\overleftarrow{V_1})\wedge(\overrightarrow{V_1}{=}x)\wedge(\overrightarrow{V}{=}x)\end{array}\right) & (\mathbf{M},\text{定理 }\mathbf{4.3\ (II)})\\
\Leftrightarrow & \exists V_1.\ \phi[V_1/V]\wedge(\llbracket V=V_1\rrbracket\vee\llbracket\ \rrbracket)\wedge(\overleftarrow{V}{=}\overleftarrow{V_1})\wedge(\overrightarrow{V}{=}\overrightarrow{V_1}) & (\mathbf{PL})\\
\Leftrightarrow & \exists V_1.\ \phi & (\text{定理 }\mathbf{4.2\ (I)})\\
\Leftrightarrow & \phi & (\mathbf{PL})
\end{array}$$

类似地，我们可以证明

$$I \bullet \phi \;\Leftrightarrow\; \phi$$

6.2 局部变量声明的形式描述

我们在引言中已指出，声明一个局部程序变量 V 在语义上相当于说存在 V 。因而，声明 V 为程序段 $\mathcal{P}$ 的局部变量的程序 (begin $V{:}\mathcal{P}$ end) 所对应的语义为

$$\exists V.\ [\![\mathcal{P}]\!]$$

其中， $[\![\mathcal{P}]\!]$ 为程序 $\mathcal{P}$ 的语义。

现在，我们重新考虑一下在引言中出现的延迟 k 个时间单位的例子。我们用 wait(k) 表示该过程，其过程体为 (begin V:$\mathcal{Q}(V,k)$ end)，其中 $\mathcal{Q}(V,k)$ 为

$$V := 0;\ \text{while } V < k \text{ do } (\text{tick};\ V := V + 1)$$

k 是一个正整数， tick 表示延迟一个时间单位的操作。下面，我们研究一下它的语义。因为 tick 是执行延迟一个时间单位的操作，它不改变局部变量 V 的值。因而其语义可表示为

$$[\![\text{tick}]\!] \ \widehat{=}\ \exists x.\ (\overleftarrow{V} = x) \wedge [\![V = x]\!] \wedge (\ell = 1) \wedge (\overrightarrow{V} = x)$$

while 语句的循环体的语义为

$$\begin{array}{ll} & [\![\mathbf{tick}; V := V + 1]\!] \\ \widehat{=} & (\exists x.\ (\overleftarrow{V} = x) \wedge [\![V = x]\!] \wedge (\ell = 1) \wedge (\overrightarrow{V} = x)) \\ \bullet & (\exists x.\ (\overleftarrow{V} = x) \wedge [\![\ ]\!] \wedge (\overrightarrow{V} = x + 1)) \\ \Leftrightarrow & \exists x.\ (\overleftarrow{V} = x) \wedge [\![V = x]\!] \wedge (\ell = 1) \wedge (\overrightarrow{V} = x + 1) \quad (\text{定理 } \mathbf{6.1}) \end{array}$$

$(V := 0)$ 表示接收一个初值，然后立即将 0 作为终值传递出去。也就是，

$$[\![V := 0]\!] \ \widehat{=}\ \exists x.\ (\overleftarrow{V} = x) \wedge [\![\ ]\!] \wedge (\overrightarrow{V} = 0)$$

while 表示反复执行循环体直到它的布尔条件不满足为止。循环体可以被执行无穷多次，即发散。然而，假定 V 的初值是 0，在 $\mathcal{Q}(V,k)$ 中的 while 语句的循环体仅能执行 k 次，即循环会终止。

$$\begin{array}{lll} [\![\text{wait}(k)]\!] & \widehat{=} & \exists V.\ [\![V := 0]\!] \bullet [\![\text{while}]\!] \\ & \Leftrightarrow & (\exists x.\ (\overleftarrow{V} = x) \wedge [\![\ ]\!] \wedge (\overrightarrow{V} = 0)) \\ & & \bullet (\exists x < k.\ (\overleftarrow{V} = x) \wedge [\![V = x]\!] \wedge (\ell = 1) \wedge (\overrightarrow{V} = x + 1))^k \end{array}$$

其中，

$$\begin{array}{lll} \phi^0 & \widehat{=} & I \\ \phi^{n+1} & \widehat{=} & \phi \bullet \phi^n \end{array}$$

使用定理 **6.1**, **HDC2** 和 **HDC3**, 我们可以将上述公式简化而得到

$$\begin{array}{lll} & [\![\text{wait}(k)]\!] & \\ \Leftrightarrow & \exists x, V.\ (\overleftarrow{V} = x) \wedge (\overrightarrow{V} = x + k) \wedge ((([\![V = 0]\!] \wedge (\ell = 1)) \frown \ldots & \\ & \frown([\![V = k - 1]\!] \wedge (\ell = 1))) & (\text{定理 } \mathbf{6.1}) \\ \Leftrightarrow & \exists V.\ ([\![V = 0]\!] \wedge (\ell = 1)) \frown \ldots & \\ & \frown([\![V = k - 1]\!] \wedge (\ell = 1)) & (\mathbf{HDC2}) \\ \Leftrightarrow & (\exists V.\ [\![V = 0]\!] \wedge (\ell = 1)) \frown \ldots & \\ & \frown(\exists V.\ [\![V = k - 1]\!] \wedge (\ell = 1)) & (\mathbf{HDC3}) \\ \Leftrightarrow & (\ell = k) & (\mathrm{Q_V}, \mathbf{M}, \mathbf{IL7}) \end{array}$$

在一个仅有 V 作为自由程序变量的程序环境里，当我们以 3 作为实参调用该过程时，该过程调用将使程序的执行延迟 3 个时间单位，而 V 的值保持不变。即，

$$\begin{aligned}[\![\text{call wait}(3)]\!] &\widehat{=} [\![\text{wait}(3)]\!] \wedge \exists x.\ (\overleftarrow{V}= x) \wedge ([\![V = x]\!] \vee [\![\ ]\!]) \wedge (\overrightarrow{V}= x) \\ &\Leftrightarrow \exists x.\ (\overleftarrow{V}= x) \wedge [\![V = x]\!] \wedge (\ell = 3) \wedge (\overrightarrow{V}= x)\end{aligned}$$

6.3 实时语义的分解

我们期望把程序的实时语义分解为两部分：与时间无关部分和与时间有关部分。例如，在上述过程调用的例子 $[\![\text{call begin } V{:}\mathcal{Q}(V,3) \text{ end}]\!]$ 中，它的语义可以重写为

$$\exists x.\ (\overleftarrow{V}{=}\overrightarrow{V}= x) \wedge ([\![V = x]\!] \wedge (\ell = 3))$$

其中，$(\overleftarrow{V}{=}\overrightarrow{V}= x)$ 是关于程序变量 V 的初值和终值的一个全局关系。它描述了过程调用中与时间无关的行为。而 $([\![V = x]\!] \wedge (\ell = 3))$ 描述了该过程调用中与时间有关的行为。其中，$V = x$ 是一个稳定状态，而 $(\ell = 3)$ 表示该状态持续的时间。全局变量 x 则将关于程序变量的初值，终值和状态的这两种关系联系起来。通过这种分解，我们期望得到程序的实时语义是它的与时间无关语义的一种保守扩张。因而，当我们综合分析程序的实时行为时，仍旧可以应用许多现存的关于处理非实时程序的技术。

[45] 给出并证明了一个关于这种分解的定理。下面的定理概括了他们的结果，并说明了这种分解的能行性。假设

$$\exists x_1, ..., x_n.\ \phi_1 \wedge G_1(\overleftarrow{V}, \overrightarrow{V})$$

和

$$\exists y_1, ..., y_n.\ \phi_2 \wedge G_2(\overleftarrow{V}, \overrightarrow{V})$$

分别刻画的是两个仅以 V 作为自由程序变量的程序段的语义。$\phi_i\ (i = 1, 2)$ 表示它们与时间有关的行为，且不含有 $\leftarrow$ 和 $\rightarrow$。$G_i(x, y)\ (i = 1, 2)$ 是刚性公式，$G_i(\overleftarrow{V}, \overrightarrow{V})\ (i = 1, 2)$ 分别描述这两个程序段的与时间无关的行为。$x_1, ..., x_n$ 和 $y_1, ..., y_n$ 是全局变量，分别用来连接这两个程序段的与时间有关的行为和与时间无关的行为。下面我们将证明顺序复合操作保持这种语义的分解。

定理 6.2

$$\begin{aligned}&(\exists x_1, ..., x_n.\ \phi_1 \wedge G_1(\overleftarrow{V}, \overrightarrow{V})) \bullet (\exists y_1, ..., y_n.\ \phi_2 \wedge G_2(\overleftarrow{V}, \overrightarrow{V})) \\ \Leftrightarrow\ &\exists x_1, ..., x_n, y_1, .., , y_n.\ \phi_1 \frown \phi_2\ \wedge\ \exists x.\ G_1(\overleftarrow{V}, x) \wedge G_2(x, \overrightarrow{V})\end{aligned}$$

证明:

$$
\begin{array}{lll}
 & (\exists x_1, ..., x_n.\ \phi_1 \wedge G_1(\overleftarrow{V}, \overrightarrow{V})) \bullet (\exists y_1, ..., y_n.\ \phi_2 \wedge G_2(\overleftarrow{V}, \overrightarrow{V})) & \\
\Leftrightarrow & \exists x_1, ..., x_n, y_1, ..., y_n, z_1, ..., z_4. & \\
 & (\phi_1 \wedge G_1(z_1, z_2) \wedge (\overleftarrow{V} = z_1) \wedge (\overrightarrow{V} = z_2)) & \\
 & \bullet(\phi_2 \wedge G_2(z_3, z_4) \wedge (\overleftarrow{V} = z_3) \wedge (\overrightarrow{V} = z_4)) & (\mathbf{PL}) \\
\Leftrightarrow & \exists x_1, ..., x_n, y_1, ..., y_n, z_1, ..., z_4. & \\
 & (\phi_1 \frown \phi_2) \wedge G_1(z_1, z_2) \wedge (\overleftarrow{V} = z_1) \wedge G_2(z_3, z_4) & \\
 & \wedge (\overrightarrow{V} = z_4) \wedge (z_2 = z_3) & (\text{定理 } \mathbf{6.1}, \mathbf{IL4}) \\
\Leftrightarrow & \exists x_1, ..., x_n, y_1, ..., y_n.\ (\phi_1 \frown \phi_2) \wedge \exists x.\ G_1(\overleftarrow{V}, x) \wedge G_2(x, \overrightarrow{V}) & (\mathbf{PL})
\end{array}
$$

第 七 章 总结

7.1 本文工作总结

本文首先以期限驱动调度算法为例说明如何用 **ITL** 和 **DC** 来形式描述实时系统并用它们的证明系统来验证实时系统的性质。为了建立基于 **DC** 的程序设计方法，特别是为了处理信息隐蔽（ **hiding**），例如变量和通道的局部化等，建立高阶时段演算是必要的。本文的主要工作是建立高阶时段演算理论，包括它的语法，语义及证明系统，并在假设所有程序变量均有有穷可变性的条件下，我们证明了所建立的高阶时段演算理论在抽象时间域上是完备的。本文最后也给出了 **HDC** 的几个应用。我们首先用它定义超稠密切割算子 $(\bullet)$，从而表明 **HDC** 可以刻画超稠密计算模型；其次，我们用 **HDC** 形式描述局部变量声明的语义；我们最后用 **HDC** 证明了一些实时程序的性质，例如我们证明了 **skip** 语句是程序复合操作 “;” 的单位元，并且复合操作满足结合律；我们也证明了可以把实时程序语义分解成两部分：实时部分和非实时部分，从而可以认为程序的实时语义是它的非实时语义的一个保守扩张。

7.2 相关工作

下面我们介绍一下与本文有关的相关工作。

7.2.1 区间时序逻辑

区间时序逻辑是用来描述实时系统需求并能够推导实时系统性质的一种模态逻辑。**ITL** 的研究始于哲学领域 **[39, 71, 52]** 。 **[2, 3]** 建立了在人工智能领域中时间和动作间关系的理论。此理论中引进了区间间的十三种不同关系，例如 $[b_1, e_1]$ 在 $[b_2, e_2]$ 的前面及 $[b_1, e_1]$ 和 $[b_2, e_2]$ 重叠等。在此理论中没有引进切割算子 “$\frown$”，实际上它应是区间上的一个三元关系。

在 **[74, 31]** 里， **ITL** 被用来描述协议和硬件部件。 **[74]** 使用的模态算子具有 $[I]\phi$ 的形式，可解释为：下次可以构造区间 I，并且公式 ϕ 在该区间上为真。其中的区间是由事件刻划。这些关于 **ITL** 的早期研究工作刺激了人们从理论和应用两个方面对 **ITL** 进行了大量的研究，例如 **[52, 53, 54, 32, 72, 51, 30, 78, 79, 26, 55, 21, 20]** 。

在 **[52]** 里给出了一些关于切割算子 “$\frown$” 的公理和规则；而 **[32]** 研究了带有能够表达在 **[3]** 中所有区间间关系的模态算子的命题区间逻辑的永真性和可满足性的复杂性问题。

[21, 20] 给出了 ITL 的一个完备的证明系统。 [12] 证明了 ITL 是不可判定的。

7.2.2 与时段演算有关的工作

时段演算 (Duration Calculus) 的研究始于 1989 年。当时 ESPRIT 的研究项目 ProCoS (可证明正确性的系统) 正寻求设计严格安全系统的形式技术, 应该项目的需要开始了时段演算的研究。该演算首先由周巢尘, C.A.R. Hoare 和 A.P. Ravn 提出 [13] 。

时段演算的扩展

在第二章中我们介绍的时段演算是以布尔函数为状态模型的。在实际实时系统设计中, 人们习惯在不同的设计阶段采用不同的模型, 这就需要我们考虑不同的函数作为 DC 的状态模型。针对不同的计算模型, 人们对 DC 做了不同的扩展。现将它们概述如下:

布尔状态模型 基本 DC [13] 是以布尔函数为状态模型。在假设所有状态均有穷可变的条件下, 它给出了关于状态时段的一些公理。基本 DC 使用了切割模态算子 ($\frown$) 。它可以描述实时系统的安全性。其它时段演算均是它的保守扩充。

实数状态模型 扩展时段演算 (Extended Duration Calculus) [17] 引入分段连续或可微实函数, 这样就可用来描述连续状态的性质。它已经用于拥有连续状态和离散状态的混成系统的设计。

布尔状态和事件模型 平均值演算 (Mean Value Calculus) [18, 83] 用状态在区间上的积分的平均值取代积分, 以 δ- 函数表示瞬时动作, 如通讯和赋值等事件。设 S 是一个状态表达式, 则它的平均值 $\bar{S}$ 可以定义为: 若 $b = e$ 则 $\bar{S}([b, e]) = S(e)$; 若 $b < e$ 则 $\bar{S}([b, e]) = \int_b^e S(t)dt/(e - b)$ 。从而状态的平均值可以表示状态在点上的行为。平均值演算可用来将基于状态的需求规范, 经状态和事件混合描述, 求精为以事件为基础的描述, 甚至最终成为可执行程序。

概率模型 概率时段演算 (Probabilistic Duration Calculus) [87, 88] 是为设计人员提供规则, 用来推导和计算一个不可靠系统, 满足时段演算所表达的需求的概率。这里, 不可靠系统是用概率自动机作为模型的。

有穷发散模型 状态和事件的有穷可变性规定在任意有穷时间内状态转换和事件仅能发生有穷多次。这种假设是适合软件系统的, 因为在它里面时间是按一定单位不断前进。但是在认为时间是连续的软件嵌入式系统内将不遵守这种假设。与有穷可变相对应的概念是有穷发散 (finite divergence), 即零现象 (Zeno phenomenon) 。 [38] 在 DC 中引进一些规则将在有穷发散模型中的状态时段值计算成它在有穷可变模型中的近似值的极限, 从而描述了有穷发散模型。

活性和公平性模型 时段演算和区间时序逻辑一样, 仅有一个原始模态算子 – "切割 " 算子 ($\frown$), 而 "切割 " 算子是一个内敛算子, 因此它不能够描述一些关于系统活性的性

质, 例如公平性, 活性和同步等。为了刻画系统这些行为, 人们对基本时段演算 [13] 进行了扩充。到目前为止, 主要有两种方法: 一种是引入无穷区间, 这种扩充后的时段演算称为无穷时段演算 (Infinite Duration Calculus) [14, 33]; 另一种方法是引入新的原始模态算子。邻接逻辑 (Neighbourhood Logic) [8] 引入了 "左邻 " 和 "右邻 " 模态算子。因为 "左邻算子 " 和 "右邻算子 " 可以表示当前区间外的性质, 因此邻接逻辑可以描述上述系统性质。并且, [35] 指出, 可以用 "左邻算子 " 和 "右邻算子 " 定义所有目前在时序逻辑中出现的模态算子。

超稠密计算模型 超稠密计算 [49] 是一个实时计算模型, 我们在上面已经知道用基本 DC [13] 是不能够描述它的。为了描述超稠密计算模型, 也有几种方法: 切割点理论 (Chopping a Point) [10] 。[10] 把宏观上的一点映射到微观上的一个区间, 所有在宏观同一点上同时发生的事件被认为是在微观上的这一区间上发生的; 二维邻接逻辑 (Two-Dimensional Neighbourhood Logic) [8, 89] 。 [8] 认为宏观上的一点对应于微观上的一个时间轴, 这样所有状态的变化都是 "之 " 字形, 并且状态在宏观和微观时间轴上的处理方法是一样的; [89] 认为宏观上的一点对应于微观上无穷多个离散点。这样, 状态的变化轨迹是若干点和线段。宏观上的轨迹 (线段) 用时段演算的方法处理; 微观上的行为 (若干离散点) 用线性时序逻辑的方法处理。这样就把时段演算和线性时序逻辑结合起来; 弱单调时间的时段演算 (Duration Calculus of Weakly Monotonic Time) [60] 。 [60] 认为状态在一个宏观时间点上可以改变多次, 它们对应到不同的微观上的时间点。所有宏观时间点和微观时间点的配对集合仍旧是一个全序集合。

高阶理论 为了刻画程序设计中信息隐蔽 (hiding), 建立高阶时段演算是必须的。 [61] 在平均值演算中引进了关于布尔状态的量词。[27] 引进了关于布尔状态的量词, 给出了几条关于它的公理, 并指出它的证明系统在抽象时间域上是完备的。 [9] 中引进了关于程序变量的量词, 并给出了一些关于高阶量词和程序中值传递的公理和规则。 [57] 证明 [9] 的证明系统在抽象时间域上是完备的。

不动点理论 为了描述实时程序设计中循环语句的语义, 建立不动理论是不可避免的。 [62] 提出了不动点算子; [43] 在 DC 中引进并行操作 "\\", 并且将区间分成连续区间和离散区间, 这两种区间通过并行操作来联系, 从而可以表示公式的复叠; [45] 建议用无穷析取和无穷合取来定义不动点算子; [40, 28] 在时段演算中引入循环操作。

时段演算的应用

时段演算已经有许多应用。例如, 它已经广泛应用于描述计算系统的实时需求, 其中包括软件嵌入式系统, 例如 [67, 76, 22, 68, 41, 63, 84, 66, 80, 81] 。同时, 时段演算也被用来定义程序设计语言的实时语义 [11, 46, 37, 73, 10, 15, 47] 。另外, 时段演算也用来形式描述调度算法并形式地证明调度算法的正确性, 例如 [86, 56, 75] 。

时段演算的工具

DC 工具的研究涉及到给出它的完备的证明系统, 判定过程和 **DC** 子集的模型检查算法。 [34] 证明若将关于实数和区间时序逻辑的永真公式作为 **DC** 的公理, 那么 **DC** 是完备的。 [83] 用 [34] 的方法证明了平均值演算的相对完备性。 [29] 用 ω - 规则替换 **DC** 中的归纳规则, 证明 **DC** 及其一些扩充在抽象时间域上是完备的。 [57] 用 [34] 和 [29] 相结合的方法证明了 **HDC** 在抽象时间域上的完备性。

[12] 给出时了 **DC** 的可判定和不可判定子集。 [83] 也给出了平均值演算的可判定性和不可判定性结果, 它类似于 [12] 中的结论。

[82] 基于 **PVS** 实现了一个用于时段演算可判定子集的自动模型检查器。对于线性时段不变式, [16] 应用线性规划技术给出了一个有效的模型检查算法。

7.3 未来工作

为了能够建立一套基于时段演算的程序设计理论, **HDC** 是必要的。为了引进关于程序变量的量词, 我们可能要付出一定的代价。但本文证明了我们可以将高阶时段演算通过翻译到一阶区间时序逻辑的方法来得到 **HDC** 的一个完备的证明系统。这是一个非常有意义的结果。另外, [61] 证明了 **HDC** 的一个可判定的子集。当然, 我们也可以用这种将高阶时段演算翻译到一阶区间时序逻辑的方法来研究 **HDC** 的判定问题, 或许我们能够从已有的关于一阶区间时序逻辑的判定结果中得到一些有趣的关于 **HDC** 的可判定性结果。

从上面的相关工作介绍我们也可以看到, 到目前为止, 人们已经建立了许多 **DC** 的扩展。如何把这些扩展统一到同一个逻辑框架内是今后的主要工作之一。 [42] 已经在这方面做了许多工作。[42] 已经将到目前为止除了概率 **DC** 外的所有扩展都统一到同一框架内。但是 [42] 没有研究统一后的逻辑基础。本文也在这方面做了一些工作, 例如 **HDC** 可以把基本 **DC** [13] , 扩充 **DC** [17] 和切割点理论 [10] 等统一起来, 并且统一后的逻辑系统仍旧是完备的。

致 谢

我首先感谢我的导师周巢尘教授. 周老师不仅学识渊博, 成绩蜚然, 是一位享誉国内外的著名计算机专家; 而且周老师治学严谨, 工作一丝不苟, 踏踏实实。我从周老师身上学到了许多东西, 这些对我的今后成长致关重要。这三年中, 在周老师的精心指导和严格要求下, 我逐渐掌握了治学方法并渐渐成长起来。我的博士论文也是在周老师的指导和帮助下完成的。周老师不仅在学习上指导我, 而且这三年里他和师母张于萍女士在生活上给了我许多关怀和帮助。在此, 也向师母表示真挚的谢意。

其次, 我要感谢我的朋友, 也是我的合作伙伴, **Dr. Dimitar P. Guelev** 。不仅因为我的论文的部分结果是我和他及周巢尘教授合作的结果, 而且与他的许多讨论也使我获益非浅。他还对我的论文提出了许多宝贵的修改意见。

我的论文中的大部分内容是我在澳门联合国大学国际软件技术研究所访问期间完成的。在这期间在学习和生活上得到了许多人的帮助, 在此, 对他们表示谢意。他们是: 何积丰教授, **Dr. Dang Van Hung**, 许启文博士, **Dr. Tomasz Janowski, Dr. Richard Moore**, 李晓山博士, 田思远先生, **Ms Margaret, Mr. Murray Singer** 等.

同时, 我要感谢林惠民教授多年来在学术上对我的指导及帮助。另外, 我也感谢实验室的张健博士, 蒋颖博士, 陈海明博士, 魏俊博士, 刘学慧博士, 王文成博士和赵琛博士等在学习和生活上给予的帮助。这三年期间, 实验室的郭菊卿老师, 胡永迁老师, 屠晓平老师和庄丽华老师为我们提供了舒适的工作环境和生活条件, 在此表示感谢。

人教处的李彩丽老师和钱军博士这三年给了我许多帮助和照顾, 在此表示谢意。

我也感谢和我一起在小楼生活的同学们, 是他们使我三年博士生活更加丰富多采。我们生活在一起, 一起讨论学术问题, 吃在一起, 玩在一起。他们是: 丁一强, 杜林, 梁海华, 黄涛, 朱军, 王栩, 李广元, 夏伟忠, 周桓, 郑新, 阮彤, 郭亮等。

最后, 我要感谢我的家人多年来对我学习和事业的支持和理解。特别是与我冷暖相依的妻子熊立俊女士, 自从我们相爱以来, 对我的事业给予了最大的支持和理解, 为此毫无怨言地承担起照顾家庭的重担。

Bibliography

[1] M. Abadi. The power of temporal proofs. *Theoretical Computer Science*, 65:35–83, 1989. Corrigendum in TCS 70 (1990), page 275.

[2] J.F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[3] J.F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.

[4] R. Alur, C. Courcoubetis, T. Henzinger, and P-H. Ho. Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems. In *Hybrid Systems, R.L. Grossman, A. Nerode, A.P. Ravn and H. Rischel (Eds)*, pages 209–229. LNCS 736, Springer-Verlag, 1993.

[5] R. Alur and D. Dill. The theory of timed automata. In *Real-Time: Theory in Practice, J.W. de Bakker, C. Huizing, W.P. de Roever and G. Rozenberg (Eds)*, pages 45 – 73. LNCS 600, Springer-Verlag, 1992.

[6] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:45 – 73, 1994.

[7] R. Alur and T.A. Henzinger. Real-time logics: Complexity and expressiveness. *Information and Computation*, 104(1):2–34, May 1993.

[8] Rana Barua and Zhou Chaochen. Neighbourhood logics : NL and NL^2. Technical report, UNU/IIST Report No. 120, UNU/IIST, P.O. Box 3058, Macau, August, 1997.

[9] Zhou Chaochen, Dimitar P. Guelev, and Zhan Naijun. A higher-order duration calculus. Technical report, UNU/IIST Report No. 167, UNU/IIST, P.O. Box 3058, Macau, July, 1999.

[10] Zhou Chaochen and Michael R. Hansen. Chopping a point. In *BCS-FACS 7th Refinement Workshop*. Electronic Workshops in Computing, Springer-Verlag, 1996.

[11] Zhou Chaochen, Michael R. Hansen, A.P. Ravn, and Hans Rischel. Duration specifications for shared processors. In J. Vytopil, editor, *Symposium on Formal Techniques in Real-Time and Fault Tolerant Systems*, volume 571 of *LNCS*, pages 21–32. Springer-Verlag, 1991.

[12] Zhou Chaochen, M.R. Hansen, and P. Sestoft. Decidability and undecidability results for duration calculus. In P. Enjalbert, A. Finkel, and K.W. Wagner, editors, *STACS'93*, volume 665 of *LNCS*, pages 58–68. Springer-Verlag, 1993.

[13] Zhou Chaochen, C.A.R. Hoare, and A.P. Ravn. A calculus of durations. *Information Processing Letters*, 40(5):269–276, 1991.

[14] Zhou Chaochen, Dang Van Hung, and Li Xiaoshan. A duration calculus with infinite intervals. In Horst Reichel, editor, *Fundamentals of Computation Theory*, volume 965 of *LNCS*, pages 16–41. Springer-Verlag, 1995.

[15] Zhou Chaochen, Wang Ji, and Anders P. Ravn. A formal description of hybrid systems. In R. Alur, T. Henzinger, and E. Sontag, editors, *Hybrid Systems III*, volume 1066 of *LNCS*, pages 511–530. Springer-Verlag, 1996.

[16] Zhou Chaochen, Zhang Jingzhong, Yang Lu, and Li Xiaoshan. Linear duration invariants. In H. Langmack, W.-P. de Roever, and J. Vytopil, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 863 of *LNCS*, pages 86–109. Springer-Verlag, 1994.

[17] Zhou Chaochen, A.P. Ravn, and M.R. Hansen. An extended duration calculus for hybrid systems. In R.L. Grossman, A. Nerode, A.P. Ravn, and H. Rischel, editors, *Hybrid Systems*, volume 736 of *LNCS*, pages 36–59. Springer-Verlag, 1993.

[18] Zhou Chaochen and Li Xiaoshan. A mean value calculus of durations. In Prentice Hall International, editor, *A Classical Mind: Essays in Honour of C.A.R. Hoare*, pages 431–451. Prentice Hall International, 1994.

[19] E.M. Clarke, E.A. Emerson, and A.P. Sistla. Automatic verification of finite state concurrent systems using temporal logic. *ACM Trans. on Programming Languages and Systems*, 8(2):244–263, 1986.

[20] B. Dutertre. Complete proof systems for first order interval temporal logic. In *Tenth Annual IEEE Symp. on Logic in Computer Science*, pages 36–43. IEEE Press, 1995.

[21] B. Dutertre. On first order interval temporal logic. Technical report, Report no. CSD-TR-94-3, Department of Computer Science, Royal

Holloway, University of London, Eghan, Surrey TW20 0EX, England, 1995.

[22] M. Engel, M. Kubica, J. Madey, D. L. Parnas, A. P. Ravn, and A. J. van Schouwen. A formal approach to computer systems requirements documentation. In R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, editors, *Hybrid Systems*, volume 736 of *LNCS*, pages 452–474, 1993.

[23] Richard L. Epstein. *The Semantic Foundations of Logic: Predicate Logic*. Oxford University Press, Oxford, UK, 1994.

[24] Editor F. Jay. *IEEE Standard Dictionary of Electrical and Electronics Terms*. The Insitute of Electrical and Electronics Engineers, Piscataway, New Jersey, foruth edition, 1988.

[25] J.W. Garson. Quantification in modal logic. In D. Gabbay and F. Guenther (Eds), editors, *Handbook of Philosophical Logic*, volume II, pages 249–307. Reidel, 1984.

[26] Asis Goswami, Michael Bell, and Mathai Joseph. Isl: An interval logic for the specification of real-time programs. In *Formal Techniques in Real-Time and Fault-Tolerant Systems, J. Vytopil (Ed.)*, pages 1–20. LNCS 571, Springer-Verlag, 1991.

[27] Dimitar P. Guelev. Quantification over state in duration calculus. Manuscript, August, 1998.

[28] Dimitar P. Guelev. Iteration of simple formulas in duration calculus. Technical report, UNU/IIST Report No. 141, UNU/IIST, P.O. Box 3058, Macau, June, 1998.

[29] Dimitar P. Guelev. A calculus of durations on abstract domains: Completeness and extensions. Technical report, UNU/IIST Report No. 139, UNU/IIST, P.O. Box 3058, Macau, May, 1999.

[30] R. Hale. *Temporal Logics and their applications*, chapter Temporal Logic Programming, pages 91–119. Academic Press, 1987.

[31] J. Halpern, B. Moskowski, and Z. Manna. A hardware semantics based on temporal intervals. In *ICALP'83*, volume 154 of *LNCS*, pages 278–291. Springer-Verlag, 1983.

[32] J.Y. Halpern and Y. Shoham. A propositional modal logic of time intervals. In *Proceedings of the First IEEE Symposium on Logic in Computer Science*, pages 279–292. IEEE Computer Society Press, 1986.

[33] Wang Hanpin and Xu Qiwen. Temporal logics over infinite intervals. Technical report, UNU/IIST Report No. 158, UNU/IIST, P.O. Box 3058, Macau, March, 1999.

[34] M.R. Hansen and Zhou Chaochen. Semantics and completeness of duration calculus. In J. W. de Bakker, C. Huizing, W.-P. de Roever, and G. Rozenberg, editors, *Real-Time: Theory in Practice, REX Workshop*, volume 600 of *LNCS*, pages 209–225. Springer-Verlag, 1992.

[35] M.R. Hansen and Zhou Chaochen. Lecture notes on the logical foundations of duration calculus. Technical report, Report no. ID-TR 1995-166, Department of Computer Science, Technical University of Denmark, 1995.

[36] M.R. Hansen and Zhou Chaochen. Duration calculus: Logical foundations. *Formal Aspects of Computing*, 9(3):283–33, 1997.

[37] M.R. Hansen, E.-R. Olderog, M. Schenke, M. Fränzle, B. von Karger, M. Müller-Olm, and H. Rischel. A duration semantics for real-time reactive systems. Technical report, Report no. OLD MRH 1/1, ProCoS ESPRIT BRA 7071, Oldenburg University, Germany, 1993.

[38] M.R. Hansen, P.K. Pandya, and Zhou Chaochen. Finite divergence. *Theoretical Computer Science*, 138:113–139, 1995.

[39] I.L. Humberstone. Interval semantics for tense logics: Some remarks. *Journal of Philosophical Logic*, 8:171–196, 1979.

[40] Dang Van Hung and Dimitar P. Guelev. Completeness and decidability of a fragment of duration calculus with iteration. In *Advances in Computing Science*, pages 139–150. LNCS 1742, Springer-Verlag, 1999.

[41] R. Inal. Modular specification of real-time systems. In *1994 Euromicro Workshop on Real-Time Systems*. IEEE Computer Society Press, 1994.

[42] He Jifeng. Integrating variants of dc. Technical report, UNU/IIST Report No. 172, UNU/IIST, P.O. Box 3058, Macau, August, 1999.

[43] He Jifeng. A behavioural model for co-design. Technical report, UNU/IIST Report No. 166, UNU/IIST, P.O. Box 3058, Macau, June, 1999. The report will appear in the Proceedings of the World Congress of Formal Methods, Toulouse, France, September, 1999.

[44] He Jifeng. *Provably Correct Systems: Modelling of Communication Languages and Design of Optimized Compilers*. McGraw-Hill, 1995.

[45] He Jifeng and Xu Qiwen. Advanced features of duration calculus and their applications. Technical report, UNU/IIST Report No. 171, UNU/IIST, P.O. Box 3058, Macau, August, 1999.

[46] He Jifeng and J. Bowen. Time interval semantics and implementation of a real-time programming language. In *1992 Euromicro Workshop on Real-Time Systems*. IEEE Computer Society Press, 1992.

[47] Li Li and He Jifeng. A denotational semantics of timed RSL using duration calculus. Technical report, UNU/IIST Report No. 169, UNU/IIST, P.O. Box 3058, Macau, July, 1999. To be presented at and published in the proceedings of The Sixth International Conference on Real-Time Computing Systems and Applications (RTCSA'99), part of the federated 1999 International Computer Congress, December 13 - 15, 1999, Hong Kong.

[48] C.L. Liu and J.W. Layland. Scheduling algorithm for multiprogramming in a hard real-time environment. *Journal of the ACM*, 20(1):46–61, 1973.

[49] Z. Manna and A. Pnueli. Models of reactitivity. *Acta Informatica*, 30(7):609–678, 1993.

[50] Zohar Manna and Amir Pnueli. The temporal framework for concurrent programs. In *The Correctness Problem in Computer Science*, pages 215–274. Academic Press, 1981.

[51] P.M. Melliar-Smith. Extending interval logic to real time systems. In *Temporal Logic in Specification, B. Banieqbal, H. Barringer and A. Pnueli (Eds)*, pages 224–242. LNCS 398, Springer-Verlag, 1987.

[52] B. Moszkowski. *Reasoning about Digital Circuits.* PhD thesis, Stanford University, 1983.

[53] B. Moszkowski. A temporal logic for multilevel reasoning about hardware. *IEEE Computer*, 18(2):10–19, 1985.

[54] B. Moszkowski. *Executing Temporal Logic Programs.* Cambridge University Press, Cambridge, UK, 1986.

[55] B. Moszkowski. Some very compositional temporal properties. In *Programming Concepts, Methods and Calculi, E.-R. Olderog (Ed.), IFIP Transactions, Vol. A-56*, pages 307–326. North-Holland, 1994.

[56] Zhan Naijun. Another formal proof for deadline driven scheduler. Technical report, UNU/IIST Report No. 169, UNU/IIST, P.O. Box 3058, Macau, August, 1999.

[57] Zhan Naijun. Completeness of higher-order duration calculus. Technical report, UNU/IIST Report No. 175, UNU/IIST, P.O. Box 3058, Macau, August, 1999.

[58] X. Nicollin, J.L. Richer, J. Sifakis, and J. Vorion. ATP : an algebra for timed processes. In *IFIP TC2 Working Conference on Programming Concepts and Methods, Sea of Gallilee, Israel*, April 1990.

[59] X. Nicollin and J. Sifakis. The algebra of timed processes ATP : theory and application. *Information and Computation*, 65:35–83, 1990.

[60] Paritosh K. Pandya and Dang Van Hung. Duration calculus with weakly monotonic time. In *Formal Techniques in Real-Time and Fault-Tolerant Systems 5th International Symposium*, pages 55–64. LNCS 1486, Springer-Verlag, Lyngby, Denmark, September 1998 (FTRTFT'98).

[61] P.K. Pandya. Some extensions to propositional mean value calculus: Expressiveness and decidability. Technical report, Computer Science Group, TIFR, Bombay, Technical Report TCS-95/9. To appear in proc of CSL'95, 1995.

[62] P.K. Pandya, Wang Hanping, and Xu Qiwen. Towards a theory of sequential hybrid programs. In *IFIP TC2/WG2.2,2.3 International Conference on Programming Concepts and Methods (PROCOMET'98), David Gries and Willem-Paul de Roever (Eds.)*, pages 366–384. Chapman & Hall, 1998.

[63] J.L. Petersen and H. Rischel. Formalizing requirements and design for a production cell system. In *Symposium ADPM '94: Automatisation des Processus Mixtes: Les Systemes Dynamiques Hybrides*, pages 37–46. Belgian Institute of Automatic Control, IBRA, 1994.

[64] A. Pnueli and E. Harel. Applications of temporal logic to the specification of real-time systems (extended abstract). In M. Joseph, editor, *Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems*, pages 84–98. LNCS 331, Springer Verlag, 1988.

[65] Amir Pnueli. The temporal logic of programs. In *18th IEEE Symp. on Foundations of Computer Science*, pages 46–77. IEEE Press, 1977.

[66] Anders P. Ravn, Hans Rischel, Michael Holdgaard, Thomas J. Eriksen, Finn Conrad, and Torben O. Andersen. Hybrid control of a robot - a case study. In P. Antsaklis, W. Cohn, A. Nerode, and S. Sastry, editors, *Hybrid Systems II*, volume 999 of *LNCS*, pages 391–404. Springer-Verlag, 1995.

[67] A.P. Ravn and H. Rischel. Requirements capture for embedded real-time systems. In *Proceedings of IMACS-MCTS'91 Symposium on Modelling and Control of Technological Systems, Villeneuve d'Ascq, France, May 7-10*, volume 2, pages 147–152. IMACS, 1991.

[68] A.P. Ravn, H. Rischel, and K.M. Hansen. Specifying and verifying requirements of real-time systems. *IEEE Trans. Softw. Eng.*, 1993.

[69] G.M. Reed and A.W. Roscoe. A timed model for communicating sequential processes. In *ICALP86: Automata, Language and Programming*. LNCS 266, Springer-Verlag, 1986.

[70] G.M. Reed and A.W. Roscoe. Metric spaces as models for real-time concurrency. In *Mathematical Foundations of Programming*, pages 331–343. LNCS 298, Springer-Verlag, 1987.

[71] P. Röper. Intervals and tenses. *Journal of Philosophical Logic*, 9:451–469, 1980.

[72] R. Rosner and A. Pnueli. A choppy logic. In *First Annual IEEE Symp. on Logic in Computer Science*, pages 306–313. IEEE Press, 1986.

[73] Michael Schenke and Ernst-Rüdiger Olderog. Requirements to programs: A development methodology for real time systems, part 1. Technical report, Fachbereich Informatik, Universität Oldenburg, 1995.

[74] R.L. Schwartz, P.M. Melliar-Smith, and F.H. Vogt. An interval logic for higher-level temporal reasoning. In *Second Annual ACM Symposium on Principles of Distributed Computing*, pages 173–186. ACM, 1983.

[75] Dong Shuzhen, Xu Qiwen, and Zhan Naijun. A formal proof for the rate monotonic scheduler. Technical report, UNU/IIST Report No. 174, UNU/IIST, P.O. Box 3058, Macau, August, 1999. To be presented at and published in the proceedings of The Sixth International Conference on Real-Time Computing Systems and Applications (RTCSA'99), part of the federated 1999 International Computer Congress, December 13 - 15, 1999, Hong Kong.

[76] J.U. Skakkebæk, A.P. Ravn, H. Rischel, and Zhou Chaochen. Specification of embedded, real-time systems. In *Proceedings of 1992 Euromicro Workshop on Real-Time Systems*. IEEE Computer Society Press, 1992.

[77] Moshe Vardi. Verification of concurrent programs - the automata-theoretic framework. In *the Second IEEE Symp. on Logic in Computer Science*, pages 167–176. IEEE Press, 1987.

[78] Y. Venema. Expressiveness and completeness of an interval tense logic. *Notre Dame Journal of Formal Logic*, 31(4):529–547, 1990.

[79] Y. Venema. A modal logic for chopping intervals. *J. Logic Computat.*, 1(4):453–476, 1991.

[80] He Weidong and Zhou Chaochen. A case study of optimization. *The Computer Journal*, 38(9):734–746, 1995.

[81] Belawati H. Widjaja, He Weidong, Chen Zongji, and Zhou Chaochen. A cooperative design for hybrid control systems. Technical report, UNU/IIST Report No. 36, UNU/IIST, P.O. Box 3058, Macau, 1995.

[82] Mao Xiaoguang, Xu Qiwen, Dang Van Hung, and Wang Ji. Towards a proof assistant for interval logics. Technical report, UNU/IIST Report No. 77, UNU/IIST, International Institute for Software Technology, P.O. Box 3058, Macau, 1996.

[83] Li Xiaoshan. *A Mean Value Calculus*. PhD thesis, Software Institute, Academia Sinica, 1993.

[84] Yu Xinyao, Wang Ji, Zhou Chaochen, and Paritosh K. Pandya. Formal design of hybrid systems. In H. Langmack, W.-P. de Roever, and J. Vytopil, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 863 of *LNCS*, pages 738–755. Springer-Verlag, 1994.

[85] Wang Yi. Real-time behaviour of asynchronous agents. In *CONCUR90: Theories of Concurrency: Unification and Extension*, pages 502–520. LNCS 458, Springer-Verlag, 1990.

[86] Zheng Yuhua and Zhou Chaochen. A formal proof of the deadline driven scheduler. In H. Langmack, W.-P. de Roever, and J. Vytopil, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 863 of *LNCS*, pages 756–775. Springer-Verlag, 1994.

[87] Liu Zhiming, A.P. Ravn, E.V. Sørensen, and Zhou Chaochen. A probabilistic duration calculus. In H. Kopetz and Y. Kakuda, editors, *Dependable Computing and Fault-Tolerant Systems Vol. 7: Responsive Computer Systems*, pages 30–52. Springer-Verlag, Wien, New York, 1993.

[88] Liu Zhiming, A.P. Ravn, E.V. Sørensen, and Zhou Chaochen. Towards a calculus of systems dependability. *High Integrity Systems*, 1(1):49–75, 1994.

[89] Qiu Zongyan and Zhou Chaochen. A combination of interval logic and linear temporal logic. In *IFIP TC2/WG2.2,2.3 International Conference on Programming Concepts and Methods (PROCOMET'98), 8-12*, pages 444–461. Chapman & Hall, June 1998, Shelter Island, New York, USA.