EXTENSÕES NÃO-LÓGICAS DO CÁLCULO RELACIONAL

Jorge Petrúcio Viana

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:

Prof$^{\underline{a}}$ Sheila Regina Murgel Veloso, D.Sc.

Prof. Paulo Augusto Silva Veloso, Ph.D.

Prof. Mario Roberto Folhadela Benevides, Ph.D.

Prof. Marcelo Finger, Ph.D.

Prof. Edward Hermann Haeusler, D.Sc.

Prof. Luiz Carlos Pinheiro Dias Pereira, Ph.D.

RIO DE JANEIRO, RJ – BRASIL
DEZEMBRO DE 2005

VIANA, JORGE PETRÚCIO

Extensões Não-lógicas do Cálculo Relacional [Rio de Janeiro] 2005

VIII, 172 p. 29,7 cm (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 2005)

Tese – Universidade Federal do Rio de Janeiro, COPPE

1 - Cálculo relacional

2 - Axiomatização

3 - Expressividade

I. COPPE/UFRJ II. Título (série)

I must invent my own system,
or be enslaved by another man's.

William Blake, *Jerusalem*, 1804.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

EXTENSÕES NÃO-LÓGICAS DO CÁLCULO RELACIONAL

Jorge Petrúcio Viana

Dezembro/2005

Orientadores: Sheila Regina Murgel Veloso
Paulo Augusto Silva Veloso

Programa: Engenharia de Sistemas e Computação

Cálculos relacionais são sistemas formais nos quais a informação é expressa em termos de propriedades das relações e a inferência é efetuada através do raciocínio sobre relações. Um de seus mais representativos exemplos é o *Cálculo das Relações Binárias*, RC. Este sistema, embora adequado para alguns propósitos, tem sérias limitações em seu poder de expressão e prova. Nesta tese, estudamos extensões de RC, por operadores não-lógicos, no sentido de A. Tarski. Em particular, os seguintes sistemas são abordados: BRC, o cálculo relacional com ligadores; FRC, o cálculo relacional com bifurcação; +RG, o cálculo relacional positivo com grafos; e LTL2, a lógica linear temporal bisortida. Investigamos o poder expressivo de cada sistema e provamos resultados de completude para BRC, FRC, e +RG.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

NON-LOGICAL EXTENSIONS OF THE RELATIONAL CALCULUS

Jorge Petrúcio Viana

December/2005

Advisors: Sheila Regina Murgel Veloso
Paulo Augusto Silva Veloso

Department: Systems Engineering and Computer Science

Relational calculi are formal systems in which information is expressed in terms of properties of relations and reasoning is performed through reasoning on relations. One of the most representative example of a relational calculus is the *Calculus of Binary Relations*, RC. Although being adequate for some purposes, this system has a very limited expressive and proof power. In this thesis we study extensions of RC by non-logical operators, in the sense of A. Tarski. In particular, the following systems are considered: BRC, the relational calculus with binders; FRC, the relational calculus with fork; +RG, the positive graph relational calculus, and LTL2, the two-sorted linear temporal logic. We investigate the expressive power of each system and provide completeness results for BRC, FRC, and +RG.

Sumário

Capítulo 1

Introdução

Nesta tese estudamos o *poder expressivo* de cálculos relacionais (binários) cujas semânticas são baseadas em estruturas relacionais (binárias). Na primeira parte, investigamos duas extensões não-lógicas do cálculo relacional de Tarski. Na segunda parte, examinamos o reduto positivo do cálculo relacional de Tarski e sua extensão na forma de um cálculo de grafos. Na terceira parte, mudamos a estrutura do cálculo relacional positivo expandindo-o com elementos tomados da lógica linear temporal e da lógica temporal bidimensional. A questão central em todas as partes é o *poder expressivo* das linguagens consideradas, embora *axiomatizações* e *decidibilidade* também sejam tratadas. Tópicos importantes que não são discutidos neste trabalho são *teoria da prova*, *teoria dos modelos* e *complexidade*.

O estudo do poder expressivo de cálculos relacionais é relevante para os campos da *lógica*, *lógica algébrica*, *teoria da computação* e *lógica temporal*. Possui também algumas conseqüências práticas importantes que serão apenas mencionadas.

Cálculos relacionais são sistemas formais nos quais a informação é expressa em termos de propriedades das relações e inferências são efetuadas através do raciocínio sobre relações. Como um pequeno exemplo, vamos comparar partes de formalizações de uma teoria: Uma em primeira ordem e a outra em uma linguagem relacional.

Exemplo 1.0.1 Dado um conjunto U, um *grafo* (não direcionado e sem laços) em U é uma relação binária irreflexiva e simétrica em U. Formalmente, uma relação binária R em U é um *grafo* se satisfaz aos seguintes axiomas de primeira-ordem: $\forall x(\neg xRx)$ e $\forall xy(xRy \to yRx)$.

Considere a proposição a seguir:

Proposição 1.0.1 *Se R, S são grafos em U, então $R \cup S, R \cap S$ são grafos em U.*

Uma prova formal detalhada da Proposição 1.0.1, na lógica de primeira ordem, é obtida por várias aplicações das regras usuais de eliminação e de introdução dos conectivos e quantificadores. O mesmo resultado pode ser expresso e demonstrado em um cálculo relacional de uma maneira completamente diferente. Primeiro, considerando *expressividade*: na linguagem das relações binárias, relações irreflexivas e simétricas podem ser definidas se estipulamos que uma relação R em U é um *grafo* quando satisfaz as duas seguintes identidades :

$$R \cap Id_U = \emptyset$$

e

$$R = R^{-1},$$

onde $\cap$ e $\emptyset$ são usuais, I_U é a relação identidade em U, e $^{-1}$ é a operação sobre relações que quando aplicada a uma relação R, reverte a ordem dos pares ordenados pertencentes a R. Agora, usando esta caracterização relacional podemos apresentar uma *prova* bastante simples da Proposição 1.0.1, usando identidades relacionais e raciocínio equacional:

PROVA. (a) Para provar que $R \cup S$ é irreflexiva: $(R \cup S) \cap I_U = (R \cap I_U) \cup (S \cap I_U) = \emptyset \cup \emptyset = \emptyset$. Para provar que $R \cup S$ é simétrica: $(R \cup S)^{-1} = R^{-1} \cup S^{-1} = R \cup S$.
(b) Para provar que $R \cap S$ é irreflexiva: $(R \cap S) \cap I_U = R \cap (S \cap I_U) = R \cap \emptyset = \emptyset$. Para provar que $R \cap S$ é simétrica: $(R \cap S)^{-1} = R^{-1} \cap S^{-1} = R \cap S$. ■

Apesar da sua simplicidade, o Exemplo 1.0.1 já evidencia alguns aspectos importantes da comparação entre formalizações em lógica de primeira ordem com seus correspondentes no cálculo relacional.

- Simbolizações nos cálulos relacionais parecem ser mais difícies de entender.

- Provas nos cálculos relacionais parecem ser mais simples.

O primeiro aspecto talvez seja uma conseqüência da ausência, nas linguagens relacionais, de variáveis individuais para referência a pontos. As vantagens dos mecanismos de inferência dos cálculos relacionais têm sido salientadas por vários autores, principalmente em consideração ao uso efetivo e à mecanização do raciocínio [49, 33]. Além disso, alguns autores sustentam que a prevalência da lógica de primeira ordem sobre outros formalismos é apenas uma questão de tradição e que para muitos propósitos o emprego da lógica de primeira ordem não é a melhor escolha [89]. O objetivo deste trabalho é contribuir nesta linha de desenvolvimento apresentando e estudando os poderes de expressão e prova de alguns cálculos relacionais que foram propostos como alternativas para a lógica de primeira ordem. De acordo com o que pudemos apurar, o estabelecimento de uma sistema *padrão* para raciocínio relacional é ainda uma matéria de investigação, embora alguns candidatos já tenham sido desenvolvidos [76, 2].

Nos próximos capítulos vamos estudar quatro sistemas diferentes relacionados a RC, o cálculo relacional das relações binárias: BRC, o cálculo relacional com ligadores; FRC, o cálculo relacional com bifurcação; +RG, o cálculo relacional positivo com grafos, e LTL2, a lógica linear temporal bisortida. Investigaremos o poder expressivo de cada sistema e provaremos resultados de completude para BRC, FRC e +RG, deixando a completude de LTL2 para um trabalho futuro. Sistemas relacionados a BRC têm sido aplicados na análise da semântica das linguagens naturais desde o trabalho seminal de P. Suppes [91]. Sistemas relacionados a FRC têm sido aplicados extensivamente no estudo da semântica de programas [37, 53, 40]. Sistemas de grafos como +RG também têm sido aplicados na semânica de programas e nos fundamentos da matemática [57, 15, 33, 18]. LTL2 tem fortes ligações com sistemas usados na investigação de aspectos práticos da linguagem XPath e suas sublinguagens [48, 47, 71, 72, 73, 6].

1.1 Cálculo relacional de Tarski

Um dos mais, possivelmente *o mais*, representativo exemplo de cálculo relacional surgiu a partir dos trabalhos de A. de Morgan [75], S.C. Peirce [78, 79, 80, 81], e E. Schröder [88]. Através do empenho de A. Tarski, seus estudantes e colaboradoes [92, 95], as idéias destes precursores foram desenvolvidas, levando à construção de um "bom sistema formal para o raciocínio relacional", cujo aspecto mais destacado é a economia de meios. De fato, embora Peirce e Schröder tenham também desenvolvido cálculos relacionais baseados no uso de relações heterogêneas infinitárias [21], o sistema de Tarski usa somente relações binárias homogêneas e, além dos operadores booleanos, operações bastante simples que levam em consideração a estrutura interna das relações.

Dado um *conjunto base* U, considerado como "o universo de discurso", uma *relação* (*binária*) em U é um subconjunto do produto cartesiano $U \times U$. No que segue, elementos de U e relações entre elementos de U são tipicamente denotados por a, b, c e X, Y, Z, respectivamente. A característica principal de RC é que nele as próprias relações, e não os elementos a partir dos quais elas são construídas, são vistas como indivíduos e colocadas em uma posição central. Sejam U e X, Y dados. As relações distinguidas básicas, *consideradas como indivíduos*, são:

Nome	Relação	Autor
Relação universal	$U \times U$	Peirce
Relação vazia	$\emptyset$	Peirce
Relação identidade	$I_U = \{(a, b) : a = b\}$	Peirce

As operações distinguidas básicas, unárias e binárias, sobre *relações consideradas como indivíduos* são:

Nome	Operação	Autor
Complementação	$X^c = \{(a, b) : (a, b) \notin X\}$	De Morgan
Reversão	$X^{-1} = \{(a, b) : (b, a) \in X\}$	De Morgan
Interseção	$X \cap Y = \{(a, b) : (a, b) \in X \text{ and } (a, b) \in Y\}$	Peirce
União	$X \cup Y = \{(a, b) : (a, b) \in X \text{ or } (a, b) \in Y\}$	Peirce
Composição	$X \mid Y = \{(a, b) : \exists c((a, c) \in X \text{ and } (c, b) \in Y)\}$	De Morgan

Além destas, as relações distinguidas básicas que podem, ou não, se estabelecer entre *relações consideradas como indivíduos* são:

Nome	Relação	Autor
Inclusão	$X \subseteq Y$ sse $\forall wv(\text{se } (w,v) \in X, \text{ então } (w,v) \in Y)$	De Morgan
Igualdade	$X = Y$ sse $\forall wv((w,v) \in X \text{ sse } (w,v) \in Y)$	De Morgan

RC é um *formalismo* que inclui símbolos constantes para denotar as relações distinguidas acima, símbolos para operações unárias e binárias para denotar as operações sobre relações acima, e símbolos de predicado binários, para denotar a inclusão e a igualdade de relações. Formalmente, a linguagem de RC contém três tipos de expressões: termos, inclusões e iqualdades, definidas como segue. Adaptamos a notação do livro [87], onde alguns aspectos do cálculo relacional de Tarski são completamente desenvolvidos.

Definição 1.1.1 Seja $\textsc{Rvar} = \{r_i : i \in \omega\}$ um conjunto de variáveis relacionais, tipicamente denotadas por r, s, t. Os *termos* de RC, tipicamente denotados por R, S, T, são definidos de acordo com a seguinte gramática:

$$R \ ::= \ \mathsf{E} \mid \mathsf{O} \mid \mathsf{I} \mid r \mid \overline{R} \mid R^{\mathsf{T}} \mid R \sqcap R \mid R \sqcup R \mid R \circ R.$$

Dados termos R, S de RC, expressões da forma $R \sqsubseteq S$, respectivamente $R \approx S$, são chamadas *inclusões*, respectivamente *igualdades*, de RC.

Dado um conjunto não-vazio arbitrário U, as variáveis relacionais irão denotar relações binárias em U, escolhidas arbitrariamente. A interpretação pretendida das operações acima é dada a seguir. Os termos E, O e I irão denotar, respectivamente, as relações $U \times U$, $\emptyset$ e I_U. Dado que os termos R, S denotam, respectivamente, as relações binárias X, Y em U, os termos $\overline{R}$, R^{T}, $R \sqcap S$, $R \sqcup S$ e $R \circ S$ denotarão, respectivamente, as relações X^c, X^{-1}, $X \cap Y$, $X \cup Y$ e $X \mid Y$. Formalmente, é útil introduzir uma notação para o significado de um termo em um modelo.

Definição 1.1.2 Um *modelo* para RC é uma estrutura $\mathfrak{M} = (M, r_i^{\mathfrak{M}})_{i \in \omega}$, onde M é um conjunto não-vazio e $r_i^{\mathfrak{M}} \subseteq M \times M$, para todo $i \in \omega$. Dado um modelo $\mathfrak{M}$, cada termo R de RC denota um subconjunto $[\![R]\!]_{\mathfrak{M}}$ de $M \times M$, conforme especificado na Tabela 1.1.

$$
\begin{array}{lcl}
[\![\mathsf{E}]\!]_{\mathfrak{M}} & ::= & M \times M, \\
[\![\mathsf{O}]\!]_{\mathfrak{M}} & ::= & \emptyset, \\
[\![\mathsf{I}]\!]_{\mathfrak{M}} & ::= & I_M, \\
[\![r]\!]_{\mathfrak{M}} & ::= & r^{\mathfrak{M}}, \\
[\![\overline{R}]\!]_{\mathfrak{M}} & ::= & [\![R]\!]^c_{\mathfrak{M}}, \\
[\![R^{\mathsf{T}}]\!]_{\mathfrak{M}} & ::= & [\![R]\!]^{-1}_{\mathfrak{M}}, \\
[\![R \sqcap S]\!]_{\mathfrak{M}} & ::= & [\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}}, \\
[\![R \sqcup S]\!]_{\mathfrak{M}} & ::= & [\![R]\!]_{\mathfrak{M}} \cup [\![S]\!], \\
[\![R \circ S]\!]_{\mathfrak{M}} & ::= & [\![R]\!]_{\mathfrak{M}} \mid [\![S]\!]_{\mathfrak{M}}.
\end{array}
$$

Tabela 1.1: Significado dos termos em um modelo.

Desse ponto em diante, em todo o texto, dado um termo R, um modelo $\mathfrak{M}$ e quaisquer elementos $a, b \in M$, adotamos a convenção, típica da lógica modal, de escrever:

$$\mathfrak{M}, a, b \Vdash R \text{ ou, simplesmente, } a, b \Vdash R,$$

ao invés da notação mais usual $(a, b) \in [\![R]\!]_{\mathfrak{M}}$. Utilizando esta notação, a semântica de todos os símbolos de RC pode ser dada como na Tabela 1.2. O significado de $a, b \nVdash R$ deve estar claro.

A valididade das inclusões e igualdades é definida de maneira usual.

Definição 1.1.3 Seja $\mathfrak{M}$ um modelo.
(1) Dizemos que uma inclusão $R \sqsubseteq S$ é *válida em* $\mathfrak{M}$ quando $a, b \Vdash R$ implica $a, b \Vdash S$, para quaisquer $a, b \in M$.
(2) Dizemos que uma igualdade $R \approx S$ é *válida em* $\mathfrak{M}$, ou que é uma *identidade em* $\mathfrak{M}$, quando $a, b \Vdash R$ sse $a, b \Vdash S$, para quaisquer $a, b \in M$.

$a, b \Vdash \mathsf{E}$	sse	sempre,
$a, b \Vdash \mathsf{O}$	sse	nunca,
$a, b \Vdash \mathsf{I}$	sse	$a = b$,
$a, b \Vdash r$	sse	$(a, b) \in r^{\mathfrak{M}}$,
$a, b \Vdash \overline{R}$	sse	$a, b \nVdash R$,
$a, b \Vdash R^{\mathsf{T}}$	sse	$a, b \Vdash R$,
$a, b \Vdash R \sqcap S$	sse	$a, b \Vdash R$ and $a, b \Vdash S$,
$a, b \Vdash R \sqcup S$	sse	$a, b \Vdash R$ or $a, b \Vdash S$,
$a, b \Vdash R \circ S$	sse	$\exists c(a, c \Vdash R$ and $c, b \Vdash S)$.

Tabela 1.2: Significado dos termos em um modelo, na notação modal.

(3) Dizemos que $R \sqsubseteq S$ é *válida*, denotado por $\Vdash R \sqsubseteq S$, quando é válida em todos os modelos.
(4) Dizemos que $R \approx S$ é *válida*, denotado por $\Vdash R \approx S$, quando é válida em todos os modelos.

A proposição seguinte e seu corolário são uma conseqüência imediata das definições.

Proposição 1.1.1 *Sejam R, S termos de* RC *e $\mathfrak{M}$ um modelo. São equivalentes:*
(a) *$R \approx S$ é válida em $\mathfrak{M}$.*
(b) *$R \sqsubseteq S$ e $S \sqsubseteq R$ são ambas válidas em $\mathfrak{M}$.*

Corolário 1.1.1 *Sejam R, S termos de* RC. *São equivalentes:*
(a) *$\Vdash R \approx S$.*
(b) *$\Vdash R \sqsubseteq S$ e $\Vdash S \sqsubseteq R$.*

Assim, em RC, a valididade de uma igualdade se reduz à validade de uma inclusão, e vice-versa. Outra maneira de interpretar este resultado é dizer que conjunções de inclusões podem ser expressas como igualdades. Como foi observado por Schröder [88], e enfatizado por Tarski [92, 95], um resultado bem mais forte é verdadeiro.

Proposição 1.1.2 (Schröder, 1896) *Qualquer combinação booleana de igualdades de* RC *pode ser expressa como uma igualdade de* RC. *Além disso, esta igualdade pode ser escolhida como uma da forma* $R \approx \mathsf{E}$.

PROVA. Para transformar uma combinação booleana de igualdades em uma igualdade da forma desejada, aplicam-se as seguintes equivalências.
(a) $R \approx S$ sse $(R \sqcap S) \sqcup (\overline{R} \sqcap \overline{S}) \approx \mathsf{E}$;
(b) $\neg R \approx \mathsf{E}$ sse $\mathsf{E} \circ \overline{R} \circ \mathsf{E} \approx \mathsf{E}$;
(c) $R \approx \mathsf{E} \wedge S \approx \mathsf{E}$ sse $R \sqcap S \approx \mathsf{E}$;
(d) $R \approx \mathsf{E} \vee S \approx \mathsf{E}$ sse $\mathsf{E} \circ \overline{(\mathsf{E} \circ \overline{R} \circ \mathsf{E}) \sqcap (\mathsf{E} \circ \overline{S} \circ \mathsf{E})} \circ \mathsf{E} \approx \mathsf{E}$. ■

O Corolário 1.1.1 garante que considerar RC como uma lógica de desigualdades ao invés de um formalismo equacional não altera o seu poder de expressão. A Proposição 1.1.2 garante que o mesmo acontece se considerarmos RC como uma lógica de termos. Neste trabalho vamos tirar proveito de todas estas possibilidades a fim de fornecer mecanismos para a extensão do poder de expressão de RC.

Terminamos esta seção apresentando alguns exemplos de como expressar informação utilizando igualdades (ou inclusões) de RC.

Exemplo 1.1.1 *Relações especiais.* Várias propriedades de relações podem ser especificadas através de inclusões e igualdades de RC. A Tabela 1.3 ilustra como isto pode ser feito no caso de propriedades relacionadas a equivalência e funcionalidade. Exemplos interessantes de sua utilização podem ser encontrados em [20, 30].

1.2 Lógica das relações binárias

Modelos $\mathfrak{M} = \langle M, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ de RC são estruturas relacionais [56] cujas relações distinguidas $r_i^{\mathfrak{M}}$ são relações binárias em M. O formalismo padrão para representar

Propriedade		Iqualdade
R é reflexiva	sse	$\mathsf{I} \sqsubseteq R$
R é simétrica	see	$R^{\mathsf{T}} \sqsubseteq R$
R é transitiva	sse	$R \circ R \sqsubseteq R$
R é funcional	sse	$R^{\mathsf{T}} \circ R \sqsubseteq \mathsf{I}$
R é injetiva	sse	$R \circ R^{\mathsf{T}} \sqsubseteq \mathsf{I}$
R é total	sse	$\mathsf{E} \leq R \circ \mathsf{E}$
R é sobrejetiva	sse	$\mathsf{E} \leq R^{\mathsf{T}} \circ \mathsf{E}$

Tabela 1.3: Relações de equivalência e funcionais em RC.

o discurso e o raciocínio sobre estruturas relacionais é a lógica de primeira ordem [32]. Em particular, para a classe dos modelos de RC, a seguinte versão da linguagem de primeira ordem, que vamos chamar de *lógica de primeira orden das relações binárias*, ou $\mathsf{FOL}(\mathrm{R})$, parace ser a mais adequada.

Definição 1.2.1 Sejam $\textsc{Ivar} = \{x_i : i \in \omega\}$ um conjunto de variáveis individuais, tipicamente denotadas por x, y, z, e $\textsc{Rvar} = \{r_i : i \in \omega\}$ o conjunto de variáveis relacionais, tipicamente denotadas por r, s, t. As *fórmulas* de $\mathsf{FOL}(\mathrm{R})$, tipicamente denotadas por φ, ψ, são definidas de acordo com a seguinte gramática:

$$\varphi \quad ::= \quad xry \mid x \approx y \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \exists x\varphi \mid \forall x\varphi.$$

Usaremos livremente, em toda a tese, as noções, propriedades e convenções, sintáticas ou semânticas, relativas aos formalismos de primeira ordem, quando restritas a $\mathsf{FOL}(\mathrm{R})$. Em particular, seguiremos [32], exceto em algumas poucas convenções notacionais que serão explicadas quando utilizadas.

FOL(R) é um formalismo muito expressivo que tem uma complexidade bastante alta. Em linhas gerais, vale o seguinte resultado. Uma análise mais fina pode ser encontrada em [32] e em [41].

Teorema 1.2.1 (a) *Toda a matemática usual pode ser desenvolvida em* FOL(R).
(b) *O problema de* model checking *para* FOL(R) *é PSPACE-completo.*
(c) *O problema da satisfabilidade para* FOL(R) *é indecidível.*

Não é necessário enfatizar a importância que as linguagens de primeira ordem possuem na formalização de conceitos e teorias. Avaliar a escolha de um formalismo para a formalização do conhecimento sobre (uma classe de) estruturas relacionais consiste, essencialmente, em analisar o inter-relacionamento de dois parâmetros:

1. *Expressividade*: que tipo de informação pode ser expressa usando a linguagem do formalismo e como isto pode ser feito.

2. *Inferência*: que tipo de raciocínio pode ser formalizado usando o aparato dedutivo do formalismo e como isto pode ser feito.

O Teorema 1.2.1 mostra que a alta expressividade de FOL(R) tem um preço. Ele motiva a investigação de alternativas que adequem melhor, de uma maneira ou de outra, o inter-relacionamento entre os dois importantes parâmetros: expressividade e complexidade. Não seguiremos esta linha de investigação aqui. Alternativamente, vamos nos concentrar na questão um pouco mais vaga da adequabilidade do cálculo relacional como um veículo efetivo para formalizar enunciados e inferências em algumas partes da matemática e da ciência da computação. Assim, na próxima seção vamos discutir os resultados clássicos sobre a comparação do poder expressivo de RC com o de FOL(R). Como é bem conhecido, o primeiro formalismo é apenas um fragmento próprio do segundo.

1.3 O problema da expressividade

Como vimos nas Seções 1.1 e 1.2, uma característica importante de RC é que por seu intermédio é possível desenvolver alguma matemática sem se fazer referências a indivíduos. De fato, como vimos, algumas propriedades das relações binárias usualmente expressas em FOL(R) podem ser reescritas como inclusões ou igualdades de RC. Além de satisfazerem o requerimento de concisão, as traduções dos enunciados em primeira ordem para o formalismo relacional trazem algumas vantagens:

- *O relacionamento estrutural entre os enunciados tende a ser mais fácil de ser percebido.* Por exemplo, examinando as inclusões na Tabela 1.3, é fácil notar a reciprocidade entre funcionalidade e injetividade de um lado e entre a totalidade e a sobrejetividade do outro.

- *O raciocínio relacional tende a ser mais direto.* Alguns autores [87] sustentam que em algumas situações o raciocínio efetuado por meio de um aparato relacional é muito mais adequado do que o raciocínio efetuado por outros meios.

Infelizmente, nem todas as sentenças de FOL(R) podem ser expressas desta maneira e o Teorema 1.3.1 caracteriza precisamente quais sentenças podem ser. Sejam x, y, z as três primeiras variáveis individuais de FOL(R), consideradas como sendo fixadas. Denotamos por $\mathsf{FOL}(\mathrm{R})^{xyz}$ o conjunto de todas as fórmulas φ de FOL(R) satisfazendo $\mathsf{var}\varphi \subseteq \{x, y, z\}$. Denotamos por $\mathsf{FOL}(\mathrm{R})_z^{xy}$ o conjunto de todas as fórmulas φ de $\mathsf{FOL}(\mathrm{R})^{xyz}$ satisfazendo $\mathsf{free}\varphi \subseteq \{x, y\}$. Isto é, obtemos $\mathsf{FOL}(\mathrm{R})^{xyz}$ restringindo a linguagem de FOL(R) a fórmulas tendo no máximo três variáveis, sendo estas, por escolha, x, y, z e obtemos $\mathsf{FOL}(\mathrm{R})_z^{xy}$ restringindo ainda mais $\mathsf{FOL}(\mathrm{R})^{xyz}$ a fórmulas tendo no máximo ocorrências livres de x, y.

Teorema 1.3.1 (Tarski, 1941) (a) *Existe uma tradução de termos de* RC *para fórmulas de* $\mathsf{FOL}(\mathrm{R})_z^{xy}$*, nas quais exatamente* x, y *ocorrem livres, que preserva a definibilidade de relações em modelos. Como uma conseqüência, a cada equação de* RC *corresponde uma sentença de* $\mathsf{FOL}(\mathrm{R})^{xyz}$ *que define a mesma classe de modelos.* (b) *Existe uma tradução de fórmulas de* $\mathsf{FOL}(\mathrm{R})_z^{xy}$ *para termos de* RC *que preserva a definibilidade de relações. Como uma conseqüência, a cada sentença de* $\mathsf{FOL}(\mathrm{R})^{xyz}$ *corresponde uma igualdade de* RC *que define a mesma classe de modelos.*

Assim, RC é mais fraco que FOL(R) em meios de expressão. Após provar este resultado, Tarski começou a investigar se esta *inequipolência* era uma conseqüência de ele ter definido um cálculo relacional inadequado, i.e., se era possível mudar os operadores escolhidos de modo a obter *equipolência* em meios de expressão com $\mathsf{FOL}(\mathrm{R})$. Segue de um raciocínio devido a S. Givant [45] que existe uma maneira trivial de aumentar o poder expressivo de sistemas relacionados a RC de modo a obter sistemas tão expressivos quanto $\mathsf{FOL}(\mathrm{R})$. Mas, como o próprio autor afirma, as extensões propostas estão muito longe de serem elegantes.

Motivado por seu trabalho sobre o significado dos *conceitos lógicos*, Tarski impôs que os operadores admitidos no enriquecimento do poder expressivo do seu cálculo relacional satisfizessem uma restrição bastante geral. Ele queria que os operadores escolhidos fossem "lógicos" no seguinte sentido. Uma operação sobre relações em um conjunto U é *lógica sobre* U se ela é invariante sob todas as permutações de U. (Uma *permutação* de U é uma função bijetiva $\rho : U \to U$.) Um símbolo de operação de um cálculo relacional RC é *lógico* se em todo modelo de RC ele denota uma operação lógica. Para fazer estas observações precisas, precisamos dos seguintes conceitos.

Definição 1.3.1 Sejam U um conjunto não vazio, ρ uma permutação de U e $X \subseteq U \times U$. A relação ρX obtida pela aplicação de ρ a X é definida por:

$$(a, b) \in \rho X \text{ sse } (\rho a, \rho b) \in X,$$

para todos $a, b \in U$.

Por exemplo, dado U e uma permutação ρ de U, temos que $\rho X = X$ sse $X \in \{\emptyset, U \times U, I_U, D_U\}$, onde D_U é a relação $\{(x, y) \in U \times U : x \neq y\}$.

Definição 1.3.2 Seja op uma operação n-ária sobre relações. Dizemos que op é *lógica em* U se a seguinte condição é verdadeira, para todas as permutações ρ de U:

$$(a, b) \in \mathsf{op}(\rho X_1, \ldots, \rho X_n) \text{ sse } (\rho a, \rho b) \in \mathsf{op}(X_1, \ldots, X_n).$$

Dizemos que op é *lógica* se ela é lógica em todos os universos U.

Por exemplo, como vimos acima, $\emptyset$, $U \times U$, I_U e D_U são as únicas relações em U que são lógicas. Segue que todos os operadores de RC são lógicos. Estes exemplos podem ser generalizados como segue. Dada uma fórmula φ e um modelo $\mathfrak{M}$ de $\mathsf{FOL}(\mathrm{R})$, e $X_1, \ldots, X_n$ relações em M, denotamos por:

– $\mathsf{free}\varphi$ o conjunto das variáveis livres de φ;

– $\textsc{Rvar}\varphi$ o conjunto das variáves relacionais que ocorrem em φ;

– $\mathfrak{M}_{X_1,\dots,X_n}$ o modelo tal que $r_i^{\mathfrak{M}_{X_1,\dots,X_n}} = X_i$, para todo i, $1 \leq i \leq n$ e $r_j^{\mathfrak{M}_{X_1,\dots,X_n}} = r_j^{\mathfrak{M}}$, para todo $j \neq 1, \dots, n$.

Definição 1.3.3 Seja op uma operação n-ária sobre relações. Dizemos que op é *definível em primeira ordem* se existe uma fórmula de primeira ordem $\varphi(x,y)$ de $\mathsf{FOL}(\mathrm{R})$ tal que:
(1) $\mathsf{free}\varphi(x,y) = \{x,y\}$.
(2) $\textsc{Rvar}\varphi(x,y) = \{r_1, \dots, r_n\}$.
(3) Para todo conjunto não-vazio U, relações binárias $X_1, \dots, X_n$ em U e $a, b \in U$:

$$\mathfrak{M}_{X_1,\dots,X_n} \models \varphi(x,y)\ [a,b] \text{ sse } (a,b) \in \mathsf{op}(X_1,\dots,X_n).$$

Proposição 1.3.1 *Se* op *é definível em primeira ordem, então* op *é lógica.*

Em torno de 1940, Tarski provou o seguinte resultado negativo [95] mostrando que a escolha dos operadores na definição de RC não era equivocada. Um operador de um cálculo relacional RC é *lógico* se denota operações lógicas em todos os modelos de RC. Caso contrário, é *não-lógico.*

Teorema 1.3.2 (Tarski, ± 1940) *Dado qualquer cálculo relacinal RC, obtido de* RC *pelo acréscimo de um número finito de operadores lógicos, existem sentenças de* $\mathsf{FOL}(\mathrm{R})$ *que não são equivalentes a nenhuma igualdade de RC. Além disso, existe um número natural* m *tal que, para todo* $n \geq m$*, a sentença que expressa a existência de ao menos* n *elementos não é equivalente a nenhuma igualdade de RC.*

De uma maneira direta: nenhum formalismo finitário similar a RC tem o mesmo poder expressivo que $\mathsf{FOL}(\mathrm{R})$. Assim, surge o importante problema de estender RC em meios de expressão:

Questão 1.3.1 (Problema da Expressividade) *Encontrar um conjunto infinito de operadores lógicos naturais e intrinsicamente interessantes para estender* RC *de modo a alcançar equipolência com* $\mathsf{FOL}(\mathrm{R})$ *em meios de expressão.*

Mesmo parecendo um objetivo um tanto vago, o preenchimento de todas as condições no enunciado do Problema da Expressividade é um requerimento extremamente importante para que um possível candidato a solução seja considerado como *uma solução* [54]. Apesar disso, consideramos que existem algumas soluções interessantes para o Problema da Expressividade mesmo se permitirmos a extensão de RC pela adição de alguns operadores não-lógicos. De fato, é possível reformular FOL(R) de modo a considerar cada variável individual como um operador zero-ário, cujo significado pode mudar de um modelo para o outro, juntamente com os quantificadores, de maneira que FOL(R), ela mesma, seja uma solução degenerada para o Problema da Expressividade pela introdução de operadores não-lógicos em RC.

Nesta tese investigamos algumas abordagens que são apresentadas como *soluções não-lógicas* para o Problema da Expressividade.

Capítulo 2

Principais resultados

Neste capítulo, sumarizamos os principais temas tratados e resultados obtidos durante a nossa pesquisa. Também indicamos alguns trabalhos relacionados. Os conteúdos aqui apresentados encontram-se desenvolvidos nos apêndices.

2.1 Cálculo relacional com ligadores

Nos Apêndices A e B investigamos as chamadas *extensões não-lógicas* por meio dos seus exemplos mais representativos: *ligadores*, *projeções* e *bifurcação*. Iniciamos investigando BRC, o cálculo relacional com ligadores (*downarrow binders*). BRC é uma extensão de RC por meio de um número infinito de operadores não-lógicos unários. Foi originalmente introduzido por M. Marx em [70], onde sua *equipolência* com a lógica de primeira ordem das relações binárias em meios de expressão e conseqüência semântica foi provada. Análogos dos teoremas de interpolação de Craig e definabilidade de Beth seguem como corolários. Em [97], B. ten Cate provou que estes resultados são os melhores possíveis, no sentido que qualquer extensão de RC que possua interpolação deve ser ao menos tão expressiva quanto BRC.

A estes resultados acrescentamos completude [27] e [25]. Como um bônus da axiomatização apresentada, obtemos uma maneira bastante natural de provar as inclusões e igualdades válidas de BRC.

Finalmente, apresentamos uma prova detalhada da tradução em [70] entre termos de BRC e fórmulas de FOL(R) que preserva a definibilidade de relações em modelos.

2.2 Cálculo relacional com bifurcação

No Apêndice B estudamos PRC e FRC, o cálculo relacional com projeções e o cálculo relacional com bifurcação (*fork*), respectivamente. Estes sistemas são interdefiníveis mas vamos considerar PRC como sendo o sistema mais básico. PRC é uma extensão de RC por meio de dois operadores não-lógicos constantes. FRC estende RC com apenas um operador não-lógico binário. Projeções foram introduzidas por A. Tarski em [93], onde resultados de completude e expressividade foram anunciados. Provas detalhadas destes resultados apareceram em [95] e uma prova alternativa foi dada em [66] (cf. [67] para a correção de alguns erros). A forma mais restrita de projeções que usamos aqui tem uma longa história. Aparecem em vários artigos sobre a semântica algébrica de programas [22, 87, 14]. Foram investigadas, embora numa perspectiva diferente da nossa, por R. Maddux em [68]. Alguns resultados de M. van de Vel [98, 99] podem ser interpretados como a afirmação de fatos sobre projeções *generalizadas*. FRC é um cálculo relacional análogo à versão definitiva de álgebras com bifurcação proposta por M. Frias, A. Haeberer e P.A.S. Veloso [39]. O operador de bifurcação tem uma longa história que pode ser recontada a partir de [36, 39, 52, 53, 40]. Estes trabalhos também contém uma extensa bibliografia que cobre tanto aspectos teóricos quanto práticos do operador de bifurcação.

A estes resultados acrescentamos uma investigação sobre a definibilidade de operadores e um resultado de completude para PRC. Como um corolário deste último, apresentamos o resultado simples de completude para FRC que aparece em [39, 52]. Finalmente, *apresentamos uma tradução entre termos de* FRC *e fórmulas de uma extensão de* FOL(R) *que preserva a definibilidade de relações em modelos*, mostrando que estes dois sistemas têm o mesmo poder de expressão.

2.3 Cálculo relacional positivo e grafos

Após investigar duas extensões de RC, no Apêndice C nos movemos para a outra direção e restringimos a linguagem de RC para o fragmento positivo. Em particular, definimos +RC, o cálculo relacional positivo. Como uma ferramenta para o estudo de +RC, definimos +RG, o cálculo relacional com grafos (positivo). A base para o cáculo relacional com grafos foi introduzida por S. Curtis e G. Lowe [23]. Variantes de +RG aparecem, em várias formas, na literatura sobre a semântica de programas e sobre os fundamentos da matemática. Em particular, D. Cantone et al. [18, 33] lidam com algumas questões referentes a poder de expressão. Ambos D. Doughery e C. Gutiérrez [31, 51], e P.J. Freyd e A. Scedrov [34] lidam com aspectos categóricos. C. Brown e G. Hutton [15, 57] contém uma abordagem para a introdução de *projeções* e *paralelismo* no cálculo com grafos. Gostaríamos de enfatizar que nenhum destes trabalhos trata o cálculos com grafos como um sistema lógico, de maneira apropriada.

Após uma apresentação formal de +RG*, apresentamos um conjunto correto e completo de regras para a derivação de grafos a partir de grafos e mostramos como ele pode ser usado para decidir as inclusões e igualdade válidas de* +RC. Finalmente, *provamos que a linguagem do cálculo relacional com grafos tem o mesmo poder de expressão que o fragmento existencial positivo da lógica de primeira ordem das relações binárias, sendo, de fato, apenas uma variante notacional dele.*

2.4 Lógica temporal linear bisortida

No Apêndice D, estudamos LTL2, a lógica temporal linear bisortida. LTL2 é definida pela combinação de elementos tomados da lógica temporal linear [42], lógica temporal bidimensional [106], lógica dinâmica proposicional [61] e do cálculo relacional positivo ou, de uma maneira mais geral, do cálculo relacional com grafos. LTL2 tem dois sortes: um para conjuntos e outro para relações. Investigamos o poder expressivo de LTL2, comparando-o com o de FOL$(<, \mathrm{P})$, a lógica de primeira ordem da ordem com predicados monádicos. Esta investigação é uma parte principal de nosso trabalho, pois nela fazemos uso da experiência adquirida previamente no estudo das extensões não-lógicas como uma base para algumas intuições e métodos.

Provamos que LTL2 *e* FOL$(<, \mathrm{P})$ *têm o mesmo poder de expressão quando restritas à classe das ordens linerares discretas.* Fazemos uma abordagem indireta através de TAL$\flat$, uma extensão não-lógica de RC introduzida por Y. Venema [106]. FOL$(<, \mathrm{P})$ está diretamente relacionada a TAL$\flat$ que, por sua vez, está diretamente relacionada a LTL2. Usamos estas proximidades para investigar o relacionamento entre estes dois últimos sistemas. Em particular, provamos que *o sorte para conjuntos tem o mesmo poder de expressão que a lógica temporal linear e que, sobre a classe das ordens lineares discretas, o sorte relacional tem o mesmo poder de expressão que* TAL$\flat$ *e, conseqüentemente, que* $\mathsf{FOL}(<, \mathrm{P})$.

Capítulo 3

Perspectivas

Nesta tese, estudamos os poderes de *expressão* e *prova* de cálculos relacionais (binários) munidos de *operadores não-lógicos* na especificação de estruturas relacionais. Quatro sistemas diferentes, relacionados a RC, o cálculo relacional das relações binárias, foram considerados: BRC, o cálculo relacional com ligadores (*downarrow binders*); FRC, o cálculo relacional com bifurcação (*fork*); +RG, o cálculo relacional positivo com grafos; e LTL2, a lógica temporal linear bisortida. Investigamos o poder de expressão de cada sistema e apresentamos resultados de completude para BRC, FRC, e +RG. Neste capítulo final, listamos alguns temas específicos para futuras investigações.

BRC é uma extensão do cálculo das relações binárias com as ferramentas do ligador. Esta extensão aumenta tanto o poder de expressão quanto o poder de prova de RC. Apresentamos uma axiomática correta e completa, contendo três axiomas de *quantificação* e uma regra de inferência *não-ortodoxa* [25, 27]. Revimos o resultado de [70] de que BRC possui o poder de expressão de FOL(R), a lógica de primeira ordem das relações binárias. Quanto ao sistema axiomático, o próximo passo será a reformulação dos axiomas e regras de modo a enfraquecer o papel dos axiomas de quantificação e da regra não-ortodoxa na formulação do sistema. Uma solução para esta questão parece ser adaptável de [94] e [97]. Uma questão mais interessante é o desenvolvimento da teoria da prova de BRC, a fim de obter provas construtivas para os análogos dos teoremas de interpolação de Craig e de definibilidade de Beth. Sabe-se que BRC é a menor extensão de RC que possui estas propriedades [97]. Assim, seria interessante que este aspecto de BRC fosse completamente estudado. A tradução de

FOL(R) para BRC que reproduzimos aqui é perfeitamente adequada para traduzir fórmulas em forma normal prenex. Neste sentido, ela pode ser considerada apenas como uma mudança de notação. Uma comparação entre os resultados obtidos pela aplicação desta tradução aos exemplos elaborados em [70] e os resultados obtidos naquele artigo, através de um procedimento diferente do aqui descrito, mostra que, na prática, parece ser mais proveitoso utilizar uma tradução que respeite a estrutura original da fórmula e faça uso da tradução do fragmento de três variáveis de FOL(R) para RC, como uma subrotina.

FRC é uma extensão do cálculo relacional das relações binárias com alguns operadores que possuem uma contraparte natural na semântica algébrica de programas. Esta extensão também aumenta tanto o poder de expressão quanto o poder de prova de RC, mas com uma semântica diferente [26, 100]. Apresentamos FRC como uma extensão por definição de PRC, o cálculo relacional com projeções. Esta visão nos permitiu apresentar um sistema de axiomas para FRC cuja corretude e completude foram provadas diretamente a partir de um sistema de axiomas para PRC e do fato de que FRC estende PRC por definição. Esta estratégia foi utilizada implicitamente nos artigos originais [39, 52] e contribui para esclarecer o papel exato desempenhado por FRC entre a miríade de extensões não-lógicas introduzidas com base em operadores que possuem contrapartes na semântica algébrica de programas. Finalmente, provamos que FRC possui o poder de expressão de FOL$(\mathrm{R}, \star)$, a lógica de primeira ordem das relações binárias com codificação. Como uma questão geral, deixamos o problema do desenvolvimento de uma extensão do cálculo relacional com grafos para FRC, que seja mais adequada do que aquela apresentada em [57].

+RG é um cálculo relacional baseado em grafos, para a prova das igualdades e inclusões válidas de +RC, o cálculo relacional positivo. Como o estudo do sistema com grafos foi apenas iniciado, existem vários problemas a serem considerados. O primeiro, principal, é a extensão do c álculo com grafos para o raciocínio a partir de hipóteses. Um problema relacionado é o de modificar +RG para o tratamento de alguma forma de negação. Outro tópico interessante é fazer uma comparação exata entre o trabalho apresentado em [23] e o nosso. Em particular, examinar o papel desempenhado pelas regras de transformação na derivação de algumas inclusões específicas deve mostrar a relação exata entre os dois sistemas. Como mostramos, o cálculo com grafos é apenas uma variante notacional da lógica de primeira ordem das relações binárias existencial positiva. Esta característica deixa em aberto a possibilidade de desenvolver a lógica de primeira ordem existencial positiva como um cálculo com grafos. Este desenvolvimento deve ser interessante, dado o aspecto lúdico das regras de inferência do cálculo com grafos, em contraste com o aspecto mais sério das regras da lógica de primeira ordem.

Problemas sobre LTL2 aparecem, principalmente, em duas linhas de investigação. Em primeiro lugar, a extensão do resultado apresentado, sobre a *completude funcional* do sistema para a classe das ordens lineares discretas, para a classe das ordens Dedekind-completas. Em segundo, a extensão da linguagem de LTL2 de modo a obter um sistema adequado para o estudo de estruturas ordenadas mais gerais, tais como as *árvores*. Esta extensão tornaria o sistema mais próximo de XPath e enfatizaria sua aplicabilidade. Resultados nesta linha de investigação já aparecem nos artigos [71, 72, 73] que serviram de inspiração para a definição de LTL2. Finalmente, uma questão interessante é a de estender o sistema com o operador de fecho transitivo da lógica dinâmica proposicional.

Com exceção apenas de +RG e LTL2, cada um dos sistemas tratados foi investigado isoladamente. Assim, consideramos que o principal problema deixado em aberto é o de iniciar uma investigação unificada das *extensões não-lógicas do cálculo relacional*. Como todos os sistemas tratados, e outros não considerados aqui, possuem importância prática, este estudo deve contribuir tanto para o entendimento quanto para a utilização dos cálculos relacionais, no futuro próximo.

Apêndice A

Relational calculus with binders

In this chapter, we shall investigate BRC, the relational calculus with binders. BRC is an extension of RC with an infinite number of unary non-logical operators, inherited from hybrid logic [3, 10, 4]. It was originally introduced by M. Marx in [70], where its equipollence with the first-order logic of binary relations in means of expression and semantical consequence is proved. Interpolation and Beth definability follow as corollaries. In [97], B. ten Cate proved that these results are optimal in the sense that any extension of RC that has interpolation should be at least as expressive as BRC. To these results we add completeness [25, 27].

In defining BRC one should make two main changes in RC:

1. First, adding a new sort of *individual variables*. Hence, the system is two-sorted. Symbols in this new sort will play a double role. At the syntactical side they will be considered as relational terms and will be combined with the other relational terms through Boolean and Peircean operators. Semantically, their denotation will be restricted to points, in a manner similar to the individual variables of first-order logic.

2. Second, introducing *new operators* for handling the now available information given by the use of individual variables. In the literature of hybrid logic, many operators have been investigated for this purpose [8, 9, 28, 50, 46, 44, 5]. The results referred above show that, for hybridizing RC, the natural candidate is a version of the downarrow binder $\downarrow$ [11].

To motivate the definition of BRC, we start reviewing some known results about the introduction of state variables, accessibility operators, and downarrow binders in RC. In particular, we exemplify two related approaches to the introduction of the downarrow binders. One, adopted in the original version of the system, uses individual variables to denote singleton relations of the form $\{(a, a)\}$. The other, studied in [27], takes advantage of the two-dimensionality of models and uses pairs of variables to denote singleton relations of the form $\{(a, b)\}$. We do not go further in this difference. Pair-variables will be introduced as a definitional extension of the single ones and used to present a very natural set of axioms and rules for BRC, adapting some ideas from [12, 13].

Theorem 1 BRC *is axiomatizable by a finite set of axioms and finitary rules.*

As a bonus from this axiomatics we obtain a very natural way of proving the valid equalities and inclusions of BRC.

Finally, we present a detailed proof of the simple translation between terms of BRC and formulas of FOL(R) that preserves meaning on models, presented in [70].

The chapter is organized as follows. In Section A.1 we present an introductory account of how nominals, accessibility operators, and binders can be introduced in RC. In Section A.2 we present BRC, the binder relational calculus. In Section A.3, we present a sound and complete axiomatization for (the valid equalities of) BRC. Finally, Section A.4 contains the main result of [70] that BRC and FOL(R) are equally expressive.

A.1 State variables, @ operators, and ↓ binders

One general source of ideas for how to extend RC is the symbolization of statements about binary relations. By examining some characteristics present in FOL(R) that are missing in RC, some natural candidates to experiment with will appear. For instance, individual variables and some less restrictive form of quantification. An adequate formalism motivated by this process is the language, introduced in [70], obtained from RC by the addition of some hybrid machinery. But, as it happens to be, some of the hybrid mechanisms are already present in RC.

A.1.1 State variables in RC

The basic step to hybridization is the introduction of special atomic formulas to refer to elements in the universe of discourse. This can be done by the use of *nominals* or *state variables*. The difference between these two categories of symbols is semantical and analogous to that in first-order logic. In RC, state variables are already present in the sense that, by the inclusion of some restrictive conditions, some relational variables can be forced to be interpreted as special relations that, in some way or another, behave as elements. In fact, this can be done in many ways, for instance, by the use of *subdiagonal elements*, *singleton sets*, or *point relations*.

Subdiagonal elements

In this view —which will be the one adopted in this work— given an universe U, each element $a \in U$ is represented by the subdiagonal relation $\{(a, a)\}$.

Definition A.1.1 Let R be a term of RC. We say that R is a *subdiagonal element relation* if the following inclusions are valid:
(1) $\mathsf{E} \sqsubseteq \mathsf{E} \circ R \circ \mathsf{E}$,
(2) $R \circ \mathsf{E} \circ R \sqsubseteq \mathsf{I}$.

Proposition A.1.1 *For every term R of* RC*, we have that R is a subdiagonal element relation iff in each model $\mathfrak{M}$, there is an $a \in M$ such that $[\![R]\!]_{\mathfrak{M}} = \{(a, a)\}$.*

Applying Proposition 1.1.2, we have an equational characterization of elements as subdiagonal element relations.

Singleton sets

Recall that, given an universe U, a relation $X \subseteq U \times U$ is a set relation iff there is a set A such that $X = A \times U$. In this view, which is adopted, for instance, in [87], each element $a \in U$ is represented by the singleton set relation $\{a\} \times U$. A relation is a set relation iff it is characterized by its domain. A relational algebraic way of writing this is $R \circ \mathsf{E} \approx R$. To warrant that the domain of X consists of just one element, we say that it is non-empty and that X is injective.

Definition A.1.2 Let R be a term of RC. We say that R is an *unitary set relation* if the following equality and inclusions are valid:

(1) $\mathsf{E} \sqsubseteq \mathsf{E} \circ R \circ \mathsf{E}$,

(2) $R \circ \mathsf{E} \approx R$,

(3) $R \circ R^{\mathsf{T}} \sqsubseteq \mathsf{I}$.

Proposition A.1.2 *For every term R of RC, we have that R is a singleton set relation iff for every model $\mathfrak{M}$, there is an $a \in M$ such that $[\![R]\!]_{\mathfrak{M}} = \{a\} \times M$.*

Applying Proposition 1.1.2, we have an equational characterization of elements as singleton set relations.

Point relations

Characterizations above are useful [66, 87], but the nicest way to represent elements by relations may be through the *difference operator*. This operator has played an important role in the development of hybrid languages in general [28, 44, 104, 96]. Its role in RC and systems close to it has been investigated as well [104, 69, 74].

Definition A.1.3 For each term R of RC, we define $\mathsf{D}R ::= (\bar{\mathsf{I}} \circ R \circ \mathsf{E}) \sqcup (\mathsf{E} \circ R \circ \bar{\mathsf{I}})$.

Proposition A.1.3 *Let R be a term of RC. Then, we have:*

$$\mathfrak{M}, a, b \Vdash \mathsf{D}R \textit{ iff } \exists c, d \in U : (a, b) \neq (c, d) \textit{ and } \mathfrak{M}, c, d \Vdash R,$$

for every model $\mathfrak{M}$ and $a, b \in M$.

Using D, we can define the *only operator*.

Definition A.1.4 For each term R of RC, we define $\mathsf{O}R ::= R \sqcap \overline{\mathsf{D}R}$, i.e., $\mathsf{O}R$ is the term $\overline{\bar{\mathsf{I}} \circ R \circ \mathsf{E}} \sqcap R \sqcap \overline{\mathsf{E} \circ R \circ \bar{\mathsf{I}}}$.

Proposition A.1.4 *Let R be a term of RC. Then, we have:*

$$\mathfrak{M}, a, b \Vdash \mathsf{O}R \textit{ iff } (a, b) \textit{ is the only ordered pair in } [\![R]\!]_{\mathfrak{M}},$$

for every model $\mathfrak{M}$ and $a, b \in M$.

Definition A.1.5 Let R be a term of RC. We say that R is an *atom relation* if $\mathsf{E} \sqsubseteq \mathsf{E} \circ \mathsf{O}R \circ \mathsf{E}$.

Proposition A.1.5 *For every term R of* RC*, we have that R is an atom relation iff for every model $\mathfrak{M}$, there are $a, b \in M$ such that $[\![R]\!]_{\mathfrak{M}} = \{(a, b)\}$.*

Definition A.1.6 Let R be a term of RC. We say that R is a *point relation* if R is a subdiagonal atom relation.

Corollary A.1.1 *For every term R of* RC*, we have that R is a point relation iff for every model $\mathfrak{M}$, there is an $a \in M$ such that $[\![R]\!]_{\mathfrak{M}} = \{(a, a)\}$.*

Hence, applying Proposition 1.1.2, we have an equational characterization of elements as point relations.

A.1.2 Accessibility operators in RC

As it is seen from above, the extension of RC by the introduction of terms to denote points is just a matter of choice of an adequate formalism. But in doing this, using any of the mechanisms above, we have to work with a distinguished set of relational terms satisfying side conditions. A more natural way is to introduce from the beginning a new set of variables whose denotation is restricted to point relations of the form $\{(a, a)\}$, or even to singleton set relations of the form $\{a\} \times U$, where U is the universe of discourse. Following this direction, we shall consider BRC as a sorted system whose set of variables is partitioned in two categories: individual variables $x_1, x_2, x_3, \ldots$, intended to denote points, ranging on point relations and relational variables ranging on all relations.

Once we have a way to denote points using variables, it is natural to introduce an operator that allows us to jump to a point denoted by some variable and verify if some given condition holds there. Such an operator works as a bridge between syntax and semantics permitting, in a sense, the internalization of the satisfaction relation [9]. In RC with individual variables, operators like this can be introduced by definition.

Let x, y be individual variables whose denotation is restricted to point relations and R be a relational term. We can define the *accessibility operators*:

(1) $@_x^1 R ::= \mathsf{E} \circ x \circ R$;

(2) $@_x^2 R ::= R \circ x \circ \mathsf{E}$;

(3) $@_{xy} R ::= @_x^1 @_y^2 R$, that is, $\mathsf{E} \circ x \circ R \circ y \circ \mathsf{E}$.

We can see that these operators have the intended meaning. In fact, for all models $\mathfrak{M}$ in which x, y denote the point relations $\{(c, c)\}$ and $\{(d, d)\}$, respectively, and for all $a, b \in M$, we should have:

(1) $\mathfrak{M}, a, b \Vdash @_x^1 R$ iff $\mathfrak{M}, c, b \Vdash R$;

(2) $\mathfrak{M}, a, b \Vdash @_x^2 R$ iff $\mathfrak{M}, a, d \Vdash R$;

(3) $\mathfrak{M}, a, b \Vdash @_{xy} R$ iff $\mathfrak{M}, c, d \Vdash R$.

A.1.3 Binders and RC

Finally, we see two natural ways to introduce downarrow binders in RC. In the first, one extends the relational calculus language with individual variables x to denote the point relations $\{(a, a)\}$, together with operators $\downarrow_x^1$ and $\downarrow_x^2$ to form the new terms $\downarrow_x^1 R$ and $\downarrow_x^2 R$ from the individual variable x and the relational term R. The ordered pair (a, b) belongs to the meaning of $\downarrow_x^1 R$ if, and only if, after *denoting the element* $\{(a, a)\}$ *by* x *we see that the ordered pair* (a, b) *belongs to the meaning of* R. Analogously, the ordered pair (a, b) belongs to the meaning of $\downarrow_y^2 R$ if, and only if, after *denoting the element* $\{(b, b)\}$ *by* y *we see that the ordered pair* (a, b) *belongs to the meaning of* R.

As it is proved in [70], in the system obtained from RC by these extensions we can symbolize any elementary statement about binary relations, in a neat way.

Example A.1.1 As an example, consider the 'sibling' relation, whose intuitive meaning is given by:

siblings are two persons **that** *share both parents.*

Let the relations 'is a sibling of' and 'is a parent of' be symbolized by s and p, respectively. In the first-order language of binary relations, we can symbolize the relation 'is a sibling of' in the following standard way:

$$xsy ::= x \not\approx y \wedge \exists zw(z \not\approx w \wedge zpx \wedge wpx \wedge zpy \wedge wpy). \qquad \text{(A.1)}$$

In the extended relational language we have the following symbolization for the sibling relation that is just a restatement of (A.1):

$$S ::= \bar{\mathsf{I}} \sqcap \downarrow^1_x(\mathsf{E} \circ \downarrow^1_z @^1_x(\mathsf{E} \circ \downarrow^1_w @^1_x(@_{zw}\bar{\mathsf{I}} \sqcap @^2_z p^{\mathsf{T}} \sqcap @^2_w p^{\mathsf{T}} \sqcap @^1_z p \sqcap @^1_w p))). \qquad \text{(A.2)}$$

But with some more effort, one can obtain a shorter symbolization [70]:

$$S ::= \bar{\mathsf{I}} \sqcap \downarrow^2_w((p \sqcap ((\bar{\mathsf{I}} \sqcap (p \circ w \circ p^{\mathsf{T}})) \circ p)^{\mathsf{T}} \circ \mathsf{E}). \qquad \text{(A.3)}$$

Alternatively, we can introduce binders in RC, by first introducing pair-variables xy to denote the atom relations $\{(a, b)\}$ representing the ordered pairs of elements of the domain of discourse. Then, we add downarrow operators to form the new term $\downarrow_{xy} R$ from the pair-variable xy and the relational term R. Now, the ordered pair (a, b) belongs to the meaning of $\downarrow_{xy} R$ if, and only if, after *denoting the ordered pair* $\{(a, b)\}$ *by* xy, *of course denoting* a *by* x *and* b *by* y, *we see that the ordered pair* (a, b) *belongs to the meaning of* R.

Example A.1.2 The sibling relation from Example A.1.1 can be symbolized using this alternative machinery. Realizing that in the definition of the sibling relation the relative pronoun **that** refers to a pair of individuals, we have the following short symbolization [27]:

$$S ::= \bar{\mathsf{I}} \sqcap \downarrow_{zw}(p^{\mathsf{T}} \circ (\bar{\mathsf{I}} \sqcap (p \circ wz \circ p^{\mathsf{T}})) \circ p). \qquad \text{(A.4)}$$

Here, we assume that wz is a convenient shorthand for zw^{T}.

This second approach was investigated in [27]. There we provide a strongly sound and complete axiomatization for the system. Moreover, we prove that both systems —the one with single variables and the other one with pair-variables— are equally expressive. So, the latter is also adequate to the symbolization of any elementary statement about binary relations.

A.2 Relational calculus with binders

In this section, we shall extend the syntax and semantics of RC to obtain the relational calculus BRC and state some simple facts that will be useful afterwards.

The basic ideas underlying syntax is that BRC is an extension of RC by the introduction of the hybrid $\downarrow$ apparatus. While RC has just one sort of relational variables ranging over binary relations, the language of BRC is two-sorted, having a new sort of *individual variables* $x_1, x_2, x_3, \ldots$. Note that, despite of their name, syntactically, individual variables act as a second sort of relational variables. The downarrow operator $\downarrow$ binds variables belonging to the individual sort.

Definition A.2.1 Starting with the language of RC, we add the following, to constitute the language of BRC:

(1) a set $\textsc{Ivar} = \{x_i : i \in \omega\}$ of *individual variables*, typically denoted by x, y, z;

(2) the *downarrow binder* $\downarrow$;

(3) to the generating grammar, the following rule:

$$R ::= x \mid \downarrow_x R.$$

Following the first-order logic paradigm, a distinction between free and bound occurrences of individual variables in terms is drawn, and substitutions of individual variables by individual variables in terms are performed. These definitions are mostly as in first-order logic. We will repeat them here for the sake of completeness.

Definition A.2.2 Let x be an individual variable and R be a term of BRC.

(1) We say that R is in the *scope* of the binder expression $\downarrow_x$ in the term $\downarrow_x R$.

(2) An occurrence of x in R is *bound* if either it is the occurrence of x in $\downarrow_x$ or it occurs in the scope of a binder expression $\downarrow_x$ in R. Otherwise, it is *free.*

(3) The set of free individual variables occurring in R is denoted by $\mathsf{free}R$.

(4) R is *closed* if $\mathsf{free}R = \emptyset$.

Definition A.2.3 Let x, y be individual variables and R be a term of BRC. The *term obtained by the substitution of x by y in R*, denoted by $R\frac{y}{x}$ is defined by the following rules:

1. First, we have:
$$z\tfrac{y}{x} ::= \begin{cases} y & \text{if } x = z, \\ z & \text{otherwise;} \end{cases}$$

2. $*\tfrac{y}{x} = *$, for $* \in \textsc{Rvar} \cup \{\mathsf{O}, \mathsf{E}, \mathsf{I}\}$;

3. $(R * S)\tfrac{y}{x} = R\tfrac{y}{x} * S\tfrac{y}{x}$, for $* \in \{\sqcap, \sqcup, \circ\}$;

4. $R^*\tfrac{y}{x} = (R\tfrac{y}{x})^*$, for $* \in \{^-, ^\mathsf{T}\}$;

5. Finally, we have:
$$(\downarrow_z R)\tfrac{y}{x} ::= \begin{cases} \downarrow_z R\tfrac{y}{x} & \text{if } z \neq x, \\ \downarrow_z R & \text{otherwise.} \end{cases}$$

When one makes substitutions for logical purposes, one has to guard against accidental binding. Once more the hybrid paradigm is followed, maintaining the intuition behind this essentially first-order notion.

Definition A.2.4 Let x, y be individual variables and R be a term of BRC. The variable y *being substitutable for x in R* is defined by the following rules:

1. y is substitutable for x in $*$, for $* \in \textsc{Ivar} \cup \textsc{Rvar} \cup \{\mathsf{O}, \mathsf{E}, \mathsf{I}\}$;

2. y is substitutable for x in $R * S$ iff it is substitutable for in both R and S, for $* \in \{\sqcap, \sqcup, \circ\}$;

3. y is substitutable for x in R^* iff it is substitutable for in R, for $* \in \{^-, ^\mathsf{T}\}$;

4. y is substitutable for x in $\downarrow_z R$ iff $x \notin \mathsf{free}\downarrow_z R$, or $z \neq y$ and y is substitutable for x in R.

Given a non-empty set U, each term of BRC denotes a binary relation on U. The denotation of relational variables is unrestricted but that of an individual variable x is restricted to a point relation $\{(a, a)\}$. This is an indirect way of using x to denote the point a. To handle the fact that individual variables may become bound, their denotations are given by a separate assignment function in a manner familiar from first-order logic.

$$
\begin{aligned}
[\![\mathsf{E}]\!]^{\beta}_{\mathfrak{M}} &::= M \times M,\\
[\![\mathsf{O}]\!]^{\beta}_{\mathfrak{M}} &::= \emptyset,\\
[\![\mathsf{I}]\!]^{\beta}_{\mathfrak{M}} &::= I_M,\\
[\![x]\!]^{\beta}_{\mathfrak{M}} &::= \{(\beta x, \beta x)\},\\
[\![r]\!]^{\beta}_{\mathfrak{M}} &::= r^{\mathfrak{M}},\\
[\![\overline{R}]\!]^{\beta}_{\mathfrak{M}} &::= ([\![R]\!]^{\beta}_{\mathfrak{M}})^{c},\\
[\![R^{\mathsf{T}}]\!]^{\beta}_{\mathfrak{M}} &::= ([\![R]\!]^{\beta}_{\mathfrak{M}})^{-1},\\
[\![R \sqcap S]\!]^{\beta}_{\mathfrak{M}} &::= [\![R]\!]^{\beta}_{\mathfrak{M}} \cap [\![S]\!]^{\beta}_{\mathfrak{M}},\\
[\![R \sqcup S]\!]^{\beta}_{\mathfrak{M}} &::= [\![R]\!]^{\beta}_{\mathfrak{M}} \cup [\![S]\!]^{\beta}_{\mathfrak{M}},\\
[\![R \circ S]\!]^{\beta}_{\mathfrak{M}} &::= [\![R]\!]^{\beta}_{\mathfrak{M}} \mid [\![S]\!]^{\beta}_{\mathfrak{M}},\\
[\![\downarrow_x R]\!]^{\beta}_{\mathfrak{M}} &::= \{(a,b) \in M \times M : (a,b) \in [\![R]\!]^{\beta\frac{a}{x}}_{\mathfrak{M}}\}.
\end{aligned}
$$

Table A.1: Meaning of terms of BRC at models.

Definition A.2.5 Let $\mathfrak{M} = \langle M, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ be a model.
(1) An *assignment* in $\mathfrak{M}$ is a mapping $\beta : \text{IVAR} \to M$.
(2) Given a model $\mathfrak{M}$ and an assignment β, each term R of BRC denotes a subset $[\![R]\!]^{\beta}_{\mathfrak{M}}$ of $M \times M$, as specified in Table A.1. Of course, $\beta\frac{a}{x}$ is the *a variant of* β *in* x, defined in the usual way:

$$\beta\tfrac{a}{x}y ::= \begin{cases} a & \text{if } y = x,\\ \beta y & \text{otherwise,}\end{cases}$$

for any $a \in M$ and individual variables x, y.

Definition A.2.5 ensures that the individual variables range over the set of all relations of the form $\{(a,a)\}$, where $a \in M$. Moreover, given a term $\downarrow_x R$ of BRC, it is interpreted as that the operator $\downarrow_x$ binds the individual variable x by storing

$\mathfrak{M}, \beta, a, b \Vdash x$	iff	$\beta x = a = b,$
$\mathfrak{M}, \beta, a, b \Vdash \downarrow_x R$	iff	$\mathfrak{M}, \beta\frac{a}{x}, a, b \Vdash R.$

Table A.2: Meaning of terms at models in a modal logic like notation.

the value a of the first coordinate of the pair (a, b) where belonging to R is being evaluated. These observations are clarified if we state the meaning of terms using the modal logic like notation referred to in Section 1.1. We omit the clauses referring to Boolean and Peircean operators. Using this notation, the semantics of the new symbols in BRC can be given as in Table A.2.

The following basic propositions hold directly.

Proposition A.2.1 (Agreement Lemma) *Let $\mathfrak{M}$ be a model and β_1, β_2 assignments in $\mathfrak{M}$ that agree on* free$R$. *Then:*

$$\mathfrak{M}, \beta_1 \Vdash R \textit{ iff } \mathfrak{M}, \beta_2 \Vdash R.$$

Proposition A.2.2 (Substitution Lemma) *Let x, y be individual variables and R be a term of* BRC. *If y is substitutable for x in R, then:*

$$\mathfrak{M}, \beta\frac{\beta y}{x}, a, b \Vdash R \textit{ iff } \mathfrak{M}, \beta, a, b \Vdash R\frac{y}{x},$$

for any model $\mathfrak{M}$, assignment β, and points $a, b \in M$.

As we noticed in Section A.1 one of the tools in the basic hybrid machinery is the accessibility operator that, in the presence of individual variables to denote points, allows us to jump to a point denoted by a variable and to see whether some formula is true there. In BRC accessibility operators are definable. Besides, since in BRC we have the individual variables ranging on the coordinates of ordered pairs of points, it is natural to introduce accessibility and binder operators for each coordinate as well as for pair of variables.

Definition A.2.6 For all individual variables x, y and term R of BRC, we define:
(1) $xy ::= x \circ \mathsf{E} \circ y$;

(2) $\downarrow^1_x R ::= \downarrow_x R$;

(3) $\downarrow^2_x R ::= (\downarrow_x R^{\mathsf{T}})^{\mathsf{T}}$;

(4) $\downarrow_{xy} R ::= \downarrow^1_x \downarrow^2_y R$;

(5) $@^1_x R ::= \mathsf{E} \circ x \circ R$;

(6) $@^2_x R ::= R \circ x \circ \mathsf{E}$;

(7) $@_{xy} R ::= \mathsf{E} \circ x \circ R \circ y \circ \mathsf{E}$.

Finally, we can prove that these operators have the intended meaning.

Proposition A.2.3 *Let x, y be individual variables and R be a term of* BRC. *Then the following hold:*

(a) $[\![xy]\!]^\beta_{\mathfrak{M}} = \{(\beta x, \beta y)\}$;

(b) $[\![\downarrow^1_x R]\!]^\beta_{\mathfrak{M}} = \{(a, b) \in M \times M : (a, b) \in [\![R]\!]^{\beta\frac{a}{x}}_{\mathfrak{M}}\}$;

(c) $[\![\downarrow^2_x R]\!]^\beta_{\mathfrak{M}} = \{(a, b) \in M \times M : (a, b) \in [\![R]\!]^{\beta\frac{b}{x}}_{\mathfrak{M}}\}$;

(d) $[\![\downarrow_{xy} R]\!]^\beta_{\mathfrak{M}} = \{(a, b) \in M \times M : (a, b) \in [\![R]\!]^{\beta\frac{a}{x}\frac{b}{y}}_{\mathfrak{M}}\}$;

(e) $[\![@^1_x R]\!]^\beta_{\mathfrak{M}} = \{(a, b) \in M \times M : (\beta x, b) \in [\![R]\!]^\beta_{\mathfrak{M}}\}$;

(f) $[\![@^2_x R]\!]^\beta_{\mathfrak{M}} = \{(a, b) \in M \times M : (a, \beta x) \in [\![R]\!]^\beta_{\mathfrak{M}}\}$;

(g) $[\![@_{xy} R]\!]^\beta_{\mathfrak{M}} = \{(a, b) \in M \times M : (\beta x, \beta y) \in [\![R]\!]^\beta_{\mathfrak{M}}\}$.

Rewritten in the modal logic like notation, conditions above can be formulated as in Table A.3.

A.3 Axiomatizing BRC

In this section, we shall present an axiomatic system, adapted from [27], for proving the valid equalities and inclusions of BRC. One difference between the previous approach and the one adopted here is a more effective use of pair-variables. This leads to a certain simplification of the system. As before, we want to use multi-modal and hybrid logic ideas to formulate our axioms and rules, and to prove completeness. So, we present a deductive system for proving *full terms* and deriving *local consequences*.

Definition A.3.1 Let $\Gamma \cup \{R\}$ be a set of terms of BRC.

(1) We say that R is *full*, denoted by $\Vdash R$, when $\mathfrak{M}, \beta, a, b \Vdash R$, for any model $\mathfrak{M}$, assignment β, and $a, b \in M$.

$\mathfrak{M}, \beta, a, b \Vdash xy$	iff	$\beta x = a, \beta y = b,$
$\mathfrak{M}, \beta, a, b \Vdash \downarrow^2_x R$	iff	$\mathfrak{M}, \beta\frac{b}{y}, a, b \Vdash R,$
$\mathfrak{M}, \beta, a, b \Vdash \downarrow_{xy} R$	iff	$\mathfrak{M}, \beta\frac{a}{x}\frac{b}{y}, a, b \Vdash R,$
$\mathfrak{M}, \beta, a, b \Vdash @^1_x R$	iff	$\mathfrak{M}, \beta, \beta x, b \Vdash R,$
$\mathfrak{M}, \beta, a, b \Vdash @^2_x R$	iff	$\mathfrak{M}, \beta, a, \beta x \Vdash R,$
$\mathfrak{M}, \beta, a, b \Vdash @_{xy} R$	iff	$\mathfrak{M}, \beta, \beta x, \beta y \Vdash R.$

Table A.3: Meaning of definable operators at models in a modal logic like notation.

(2) We say that R is a *consequence* of Γ, denoted by $\Gamma \Vdash R$, when:

$$\mathfrak{M}, \beta, a, b \Vdash S \text{ for all } S \in \Gamma, \text{ implies } \mathfrak{M}, \beta, a, b \Vdash R,$$

for any model $\mathfrak{M}$, assignment β and $a, b \in M$.

The use of our system as an axiomatization of the valid equalities and inclusions is based on the following Boolean equivalences:

$$R \sqsubseteq S \quad \text{iff} \quad (R \sqcap S) \sqcup \overline{R} \approx \mathsf{E}$$

$$R \approx S \quad \text{iff} \quad ((R \sqcap S) \sqcup \overline{R}) \sqcap ((S \sqcap R) \sqcup \overline{S}) \approx \mathsf{E}$$

Equivalences above show that an inclusion $R \sqsubseteq S$ is valid if, and only if, the term $(R \sqcap S) \sqcup \overline{R}$ is full. Likewise for equalities.

We shall now present a strongly sound and complete axiom system for the consequence relation $\Vdash$. I.e., we shall define a deductive relation $\vdash$ that coincides with the semantical one.

A.3.1 Axioms and rules

Now, we present axioms and rules. Theoremhood and derivability are defined as usual. Some explanations follow.

Definition A.3.2 (1) The *axioms* and the *inference rules* of BRC are given in Tables A.4 and A.5, respectively.

(2) Let $\Gamma \cup \{R\}$ be a set of terms of BRC.

(2.a) We say that R is a *theorem* of BRC if there exists a sequence $R_1, \ldots, R_n = R$ of terms of BRC such that each R_i, $1 \leq i \leq n$, is an axiom or obtained from previous ones by one application of one of the rules of inference.

(2.b) We say that R is *derivable* from Γ in BRC, denoted by $\Gamma \vdash R$, if there exists a sequence $R_1, \ldots, R_n = R$ of terms of BRC such that each R_i, $1 \leq i \leq n$, is an axiom or a member of Γ or a theorem or obtained from previous ones by one application of the rule MP.

Axiom CT and rule MP, in addition to the fact that all the other rules are restricted just to theorems, assure the nice feature that every instance of the classical propositional reasoning can be performed inside the system. We shall make reference to this fact by using PR, from propositional reasoning. Axioms CT and Dual, and rules MP, Nec and Usub are usual for hybrid logic formalisms. In particular, rule USub refers to substitutions σ that uniformly replace state variables by state variables and relational variables by terms. A notion of replacement of εquivalents follows.

Definition A.3.3 Let R, S, T be terms of BRC. A term obtained from R by *replacement* of some occurrences of S in R by occurrences of T, denoted by $R^{\mathcal{E}}$, is obtained by application of the following rules:

1. If $S = R$, then $R^{\mathcal{E}} = R$ or T, your choice.

2. If $S \neq *$, then $*^{\mathcal{E}} = *$, for $* \in \text{IVAR} \cup \text{RVAR} \cup \{\mathsf{E}, \mathsf{O}, \mathsf{I}\}$.

3. If $S \neq R_1 * R_2$, then $(R_1 * R_2)^{\mathcal{E}} = R_1^{\mathcal{E}} * R_2^{\mathcal{E}}$, for $* \in \{\sqcap, \sqcup, \circ\}$.

4. If $S \neq R^*$, then $(R^*)^{\mathcal{E}} = (R^{\mathcal{E}})^*$, for $* \in \{^-, ^\mathsf{T}\}$.

5. If $S \neq @_{xy}R$, then $(@_{xy}R)^{\mathcal{E}} = @_{xy}(R^{\mathcal{E}})$.

6. If $S \neq \downarrow_{xy}R$, then $(\downarrow_{xy}R)^{\mathcal{E}} = \downarrow_{xy}(R^{\mathcal{E}})$.

CT) All the classical tautologies for the Booleans.

Dual) Dual axioms for the Peirceans.

Name) $@_{xy}xy$

Nom) $@_{xy}uv \to (@_{xy}R \to @_{uv}R)$

Elimination) $(xy \sqcap @_{xy}R) \to R$

Scope) $@_{xy}@_{uv}R \leftrightarrow @_{uv}R$

SelfDual@) $@_{xy}R \leftrightarrow \overline{@_{xy}\overline{R}}$

SelfDual↓) $\downarrow_{xy}R \leftrightarrow \overline{\downarrow_{xy}\overline{R}}$

Bridge I) $@_{xx}yy \leftrightarrow @_{xy}\mathsf{I}$

Bridge$^{\mathsf{T}}$) $@_{xy}R \leftrightarrow @_{yx}(R^{\mathsf{T}})$

Bridge ∘) $(@_{xz}R \sqcap @_{zy}S) \to @_{xy}(R \circ S)$

Q1) $\downarrow_{xy}R \leftrightarrow R$, if $x, y \notin \mathsf{free}R$.

Q2) $\downarrow_{xy}R \to (uv \to R\frac{u}{x}\frac{v}{y})$, if uv is substitutable for xy in R.

Q3) $\downarrow_{xy}(xy \to R) \to \downarrow_{xy}R$, if $x \neq y$.

Table A.4: Axioms for BRC.

MP) $\dfrac{R, R \to S}{S}$

Nec) Necessitation rules for the dual of all operators.

USub) $\dfrac{R}{\sigma R}$

Paste) $\dfrac{(@_{xz} R \sqcap @_{zy} S) \to T}{@_{xy}(R \circ S) \to T}$, where z is new.

Table A.5: Inference rules for BRC.

Note that Definition A.3.3 is not deterministic. Replacement may produce different outputs for the same input. In what follows, the notation $R^{\mathcal{E}}$ will be used ambiguously to denote any term obtained from R by replacement. In this sense, Replacement of Equivalents, referred to as REq, is better interpreted as a schema.

Lemma A.3.1 (REq) *If* $\Gamma \vdash R$ *and* $\vdash S \leftrightarrow T$, *then* $\Gamma \vdash R^{\mathcal{E}}$.

Axioms Name, Nom, Elimination, and Scope express that pair-variables denote pair of points. Axioms SelfDual@ and SelfDual$\downarrow$ state that the hybrid operators $@_{xy}$ and $\downarrow_{xy}$ are self-dual. Axioms Bridge I, Bridge$^{\mathsf{T}}$ and Bridge $\circ$, and rule Paste are particularizations, for the multi-modal case, of axioms and rules of hybrid logic [12, 13]. Rule Paste acts as a kind of reverse of axiom Bridge $\circ$. Axioms Bridge I, Bridge$^{\mathsf{T}}$, Bridge $\circ$, and rule Paste reduce the Peircean operators to the hybrid logic operators, when they occur under the scope of a satisfaction operator. Axioms Q1, Q2, and Q3 ensure that the downarrow operator behaves as a quantifier.

A notion of αlphabetic variant can be introduced in BRC, as in FOL.

Definition A.3.4 Let R be a term and x, y, u, v be individual variables of BRC. (1) We say that *uv is substitutable for xy in R* when u is substitutable for x in R and v is substitutable for y in $R\frac{u}{x}$.

(2) When uv is substitutable for xy in R, the *alphabetic variant* of R with respect to x, y, u, v, denoted by R^α, is obtained by application of the following rules:

1. $*^\alpha = *$, for $* \in \text{IVAR} \cup \text{RVAR} \cup \{\mathsf{E}, \mathsf{O}, \mathsf{I}\}$.

2. $(R_1 * R_2)^\alpha = R_1^\alpha * R_2^\alpha$, for $* \in \{\sqcap, \sqcup, \circ\}$.

3. $R^{*\alpha} = (R^\alpha)^*$, for $* \in \{, ^\mathsf{T}\}$.

4. $(@_{zw}R)^\alpha = @_{zw}(R^\alpha)$, for all $z, w \in \text{IVAR}$.

5. $$(\downarrow_{zw}R)^\alpha = \begin{cases} \downarrow_{z'w'}(R^\alpha \frac{z'}{z} \frac{w'}{w}) & \text{if } z = u, w = v, u \neq x, v \neq y, \\ \downarrow_{zw'}(R^\alpha \frac{w'}{w}) & \text{if } w = v, v \neq y, (z \neq u \text{ or } z = x), \\ \downarrow_{z'w}(R^\alpha \frac{z'}{z}) & \text{if } z = u, u \neq x, (w = y \text{ or } w \neq v), \\ \downarrow_{zw}(R^\alpha) & \text{otherwise,} \end{cases}$$

   where z' and w' are the two next new variables.

Lemma A.3.2 (Alphabetic Variant) *Let R be a term of* BRC *and x, y, u, v be individual variables such that uv is substitutable for xy in R. Then:*
(a) $\mathsf{free}R = \mathsf{free}R^\alpha$.
(b) *uv is substitutable for xy in R^α.*
(c) $\vdash R \leftrightarrow R^\alpha$.

Proof. By induction on terms, using REq and axioms Name, Nom, Elimination, SelfDual@, Q1, and Q2.

A.3.2 Theorems and derived rules

Propositions below contain some theorems and rules of BRC that will be useful in what follows. In particular, derived rule R$*$ conduces to a very interesting process that can be used to prove theorems of BRC.

Proposition A.3.1 *For any terms R, S and individual variables x, y, u, v, the terms in Table A.6 are theorems of* BRC.

Proof. By REq, using axioms Name, Nom, Elimination, SelfDual@, and Q2. ■

Negation) $@_{xy}\overline{R} \leftrightarrow \overline{@_{xy}R}$
Conjunction) $@_{xy}(R \sqcap S) \leftrightarrow (@_{xy}R \sqcap @_{xy}S)$
Swap) $@_{xy}uv \leftrightarrow @_{uv}xy$
Bridge E) $@_{xy}\mathsf{E} \leftrightarrow \mathsf{E}$
Bridge O) $@_{xy}\mathsf{O} \leftrightarrow \mathsf{O}$
Bridge ↓) $@_{xy}R^{x\,y}_{u\,v} \leftrightarrow @_{xy}{\downarrow_{uv}}R$, if uv is substitutable for xy in R.

Table A.6: Some useful theorems of BRC.

Theorems Negation and Conjunction assure that the satisfiability operator distributes over the Booleans. This fact, together with the self-duality of @ conduces to the following strategy in proving theorems, using all the strenght of the hybrid apparatus available in BRC.

Proposition A.3.2 *The following rule is derived in* BRC.

$$\mathsf{R}{*})\ \ \frac{@_{xy}R}{R},\ \ \textit{if } x \neq y \textit{ and } x, y \notin \mathsf{free}R.$$

PROOF. By axioms Elimination, Q1 and Q3, and theorem Negation, and REq. ■

To prove any term R, the recipe is to take a new pair of variables x, y and prove $@_{xy}R$. Then, using R∗, we obtain R. As a particular case, this recipe is applied in the next proposition, where the use of pair-variables gives the system the power to derive the missing relation algebraic axioms for the Peircean operators.

Proposition A.3.3 *The* RA *axioms* [19, 104] *are theorems of* BRC.

PROOF. We just prove the more evolved axiom $R \circ \overline{R^{\mathsf{T}} \circ S} \to \overline{S}$. Let x, y, z be three new individual variables. The following sequence of terms is a proof in BRC:

Bridge$\circ$	1.	$(@_{zx}R^{\mathsf{T}} \sqcap @_{xy}S) \to @_{zy}(R^{\mathsf{T}} \circ S)$
SelfDual@	2.	$(@_{zx}R^{\mathsf{T}} \sqcap \overline{@_{xy}\overline{S}}) \to \overline{@_{zy}\overline{R^{\mathsf{T}} \circ S}}$
CT	3.	$(@_{zx}R^{\mathsf{T}} \sqcap @_{zy}\overline{R^{\mathsf{T}} \circ S}) \to @_{xy}\overline{S}$
Bridge$^{\mathsf{T}}$	4.	$(@_{xz}R \sqcap @_{zy}\overline{R^{\mathsf{T}} \circ S}) \to @_{xy}\overline{S}$
Paste	5.	$@_{xy}(R \circ \overline{R^{\mathsf{T}} \circ S}) \to @_{xy}\overline{S}$
Negation, Conjunction	6.	$@_{xy}(R \circ \overline{R^{\mathsf{T}} \circ S} \to \overline{S})$
R$*$	7.	$R \circ \overline{R^{\mathsf{T}} \circ S} \to \overline{S}$

The other axioms can be proved in an entirely similar way. ∎

A.3.3 Completeness theorem

Soundness, as usual, is a simple exercise.

Theorem A.3.1 *If* $\Gamma \vdash R$, *then* $\Gamma \Vdash R$.

PROOF. An exhaustive case analysis shows that the axioms are full and that the inference rules derive full terms when applied to full terms. ∎

Now that we know the system is sound, we can move to the completeness result. We use the canonical model construction, together with the technique of extending consistent sets to consistent sets with witnesses. This mixture is usual in hybrid logic [12, 13, 97]. It makes things easy but also has its drawbacks. In particular, it makes strong use of the axioms Q1, Q2 and Q3 as well as of the rule Paste. These, due to their side conditions, give the system a strong first-order appearance.

Our proof uses an extension of the techniques presented in [12]. Consistent and maximal consistent sets (MCS) are defined as usual. Besides, we shall use the following notions.

Definition A.3.5 Let Γ be a set of terms.

(1) Γ is *named* when $xy \in \Gamma$, for some $x, y \in \textsc{Ivar}$.

(2) Γ has *witnesses* when for any $x, y \in \textsc{Ivar}$ and terms R, S of BRC, $@_{xy}(R \circ S) \in \Gamma$ implies $@_{xz}R \sqcap @_{zy}S \in \Gamma$, for some $z \in \textsc{Ivar}$.

(3) In case Γ is an MCS, we define $x \sim y$ iff $@_{xx}yy \in \Gamma$, for any $x, y \in \textsc{Ivar}$.

Next proposition assures that $\sim$ is an equivalence relation on the set of all individual variables.

Proposition A.3.4 *If* Γ *is an* MCS, *then the following hold for any individual variables* x, y, z*:*

(a) $@_{xx}xx \in \Gamma$.

(b) *If* $@_{xx}yy \in \Gamma$*, then* $@_{yy}xx \in \Gamma$.

(c) *If* $@_{xx}yy, @_{yy}zz \in \Gamma$*, then* $@_{xx}zz \in \Gamma$.

Proof. By applying REq, using axioms Name, Swap, and Nom. ∎

Definition A.3.6 Let Γ be an MCS.

(1) The *canonical model* yielded by Γ is $\mathfrak{M}^\Gamma = \langle M^\Gamma, r^{\mathfrak{M}^\Gamma} \rangle$, where $M^\Gamma ::= \{\widetilde{x} : x \in \textsc{Ivar}\}$ and $r^{\mathfrak{M}^\Gamma} ::= \{(\widetilde{x}, \widetilde{y}) : @_{xy}r \in \Gamma\}$.

(2) The *canonical assignment* in $\mathfrak{M}^\Gamma$ is $\beta^\Gamma : \textsc{Ivar} \to M^\Gamma$ such that $\beta^\Gamma(x) = \widetilde{x}$.

By the following proposition, $r^{\mathfrak{M}^\Gamma}$ is well-defined.

Proposition A.3.5 *Let* Γ *be an* MCS. *Then, if* $@_{xx}uu, @_{yy}vv, @_{xy}r \in \Gamma$*, then also* $@_{uv}r \in \Gamma$.

Proof. Using axioms Name, Nom, $\mathsf{Bridge\,I}$, and $\mathsf{Bridge} \circ$, and Proposition A.3.3. ∎

As usual, we use the canonical model yielded by Γ to prove a model existence theorem via a satisfiability lemma.

Lemma A.3.3 *If* Γ *is an* MCS *with witnesses, then for any individual variables* x, y*, we have:*

$$@_{xy}R \in \Gamma \text{ iff } \mathfrak{M}^\Gamma, \beta^\Gamma, \widetilde{x}, \widetilde{y} \Vdash R.$$

PROOF. By induction on terms. The Boolean cases follow from Negation and Conjunction; the Peircean cases from Bridge O, Bridge E, Bridge$^{\mathsf{T}}$, and Bridge ∘ ; the pair-variable case from Proposition A.3.5; the satisfaction operator case from Scope; the downarrow binder case from Bridge ↓ and the Substitution Lemma, using alphabetic variants. ■

Corollary A.3.1 (Satisfiability Lemma) *If* Γ *is an* MCS *with witnesses named with* xy*, then* $R \in \Gamma$ *iff* $\mathfrak{M}^{\Gamma}, \beta^{\Gamma}, \widetilde{x}, \widetilde{y} \Vdash R$*.*

PROOF. Applying Lemma A.3.3, and using the fact that Γ is named together with axiom Elimination. ■

Now we will show that every consistent set of formulas can be extended to an MCS named and with witnesses, as required by Lemma A.3.1. Let BRC^N be the auxiliary system obtained from BRC by enriching the language with a countable set of new variables NIvar.

Lemma A.3.4 (Extended Lindenbaum Lemma) *Every consistent set of terms of* BRC *can be extended to a named* MCS *with witnesses in* BRC^N*.*

PROOF. Let $\Gamma_0 = \Gamma \cup \{x_0 x_1\}$ with x_0, x_1 being the first two new variables in an enumeration of NIvar. Γ_0 is consistent, from axioms Q1, Q3, and PR. Let $R_0, R_1, R_2, \ldots$ be an enumeration of the terms of BRC^N. Let Θ_0 be Γ_0. For $m \geq 0$, define:

$$\Theta_{m+1} ::= \begin{cases} \Theta_m & \text{if } \Theta_m \cup \{R_m\} \text{ is inconsistent} \\ \Theta_m \cup \{R_m\} & \text{if } \Theta_m \cup \{R_m\} \text{ is consistent} \\ & \text{and } R_m \neq @_{xy}(R \circ S) \\ \Theta_m \cup \{@_{xy}(R \circ S), @_{xz}R \sqcap @_{zy}S\} & \text{otherwise,} \end{cases}$$

with $R_m = @_{xy}(R \circ S)$ and z being the first variable in NIvar not occurring in Θ_m neither in $@_{xy}(R \circ S)$.

Let $\Theta = \bigcup_{n \geq 0} \Theta_n$. By rule Paste and PR, Θ is consistent. ■

Lemmas A.3.1 and A.3.4 together yield the Model Existence Theorem and give, as a corollary, strong completeness.

Theorem A.3.2 (Model Existence) *Every consistent set of terms is satisfiable.*

Theorem A.3.3 (Completeness) *If* $\Gamma \Vdash R$, *then* $\Gamma \vdash R$.

PROOF. Suppose $\Gamma \nvdash R$. Hence, by PR, $\Gamma \cup \{\overline{R}\}$ is consistent. So, by the Model Existence Theorem, we have both $\mathfrak{M}, \beta, a, b \Vdash \Gamma$ and $\mathfrak{M}, \beta, a, b \Vdash \overline{R}$, for some $\mathfrak{M}, \beta, a, b$. But the later is the same as $\mathfrak{M}, \beta, a, b \nVdash R$. Hence, we conclude $\Gamma \nVdash R$. ■

A.4 Expressive power of BRC

In this section, we shall present the results from [70], showing that BRC and FOL(R) are equally expressive on models. To this end, we assume that x, y, the first and the second individual variables of FOL(R), respectively, do not occur neither in any term of BRC nor bounded in any formula of FOL(R). First, we shall present a forward translation from terms of BRC into formulas of FOL(R). Second, we shall describe the backward translation from formulas of FOL(R) having at most x, y as free variables —but having any number of bounded variables— into terms of BRC.

A.4.1 Translating from BRC into FOL(R)

As usual, the forward translation is just an adequate rephrasing of the definition of meaning of terms.

Definition A.4.1 Let R be a term of BRC.
(1) Let u, v be two distinct individual variables of FOL(R) not occurring in R. The *forward translation* of R associated to u, v is the formula $\mathrm{FT}_{uv}R$ defined recursively in Table A.7. In the fourth and last lines, w is distinct from u and v. In the tenth line, w is the first individual variable distinct from u and v that does not occur neither in R nor in S.
(2) The *forward translation of* R is the formula $\mathrm{FT}R ::= \mathrm{FT}_{xy}R$, where x is the first and y is the second individual variable of FOL(R). We assume no term of BRC has occurrences of x or y, so that FT is well-defined.

The next lemma assures that FT_{uv} behaves syntactically as expected.

Term R	Formula $\mathrm{FT}_{uv}R$	
O	$u \not\approx u \wedge v \not\approx v$	
I	$u \approx v$	
E	$u \approx u \vee v \approx v$	
w	$w \approx u \wedge u \approx v,$	$w \neq u, v$
r	urv	
$\overline{R}$	$\neg \mathrm{FT}_{uv}R$	
R^{T}	$\mathrm{FT}_{vu}R$	
$R \sqcap S$	$\mathrm{FT}_{uv}R \wedge \mathrm{FT}_{uv}S$	
$R \sqcup S$	$\mathrm{FT}_{uv}R \vee \mathrm{FT}_{uv}S$	
$R \circ S$	$\exists w(\mathrm{FT}_{uw}R \wedge \mathrm{FT}_{wv}S),$	w is new
$\downarrow_w R$	$\exists w(w \approx u \wedge FT_{uv}R),$	$w \neq u, v$

Table A.7: Translating from BRC to FOL(R).

Lemma A.4.1 *Let R be a term of* BRC *and u, v be two distinct individual variables of* FOL(R) *not occurring in R. Then,* $\mathsf{free}\mathrm{FT}_{uv}R = \{u, v\} \cup \mathsf{free}R$.

Proof. By induction on terms, using Definition A.2.2. ∎

The next lemma assures that the semantical behavior of FT_{uv} is also as expected.

Lemma A.4.2 *Let R be a term of* BRC *and u, v be two distinct individual variables of* FOL(R) *not occurring in R. Then, we have:*

$$\mathfrak{M}, \beta, \beta u, \beta v \Vdash R \textit{ iff } \mathfrak{M}, \beta \models \mathrm{FT}_{uv}R,$$

for any model $\mathfrak{M}$ and assignment β in $\mathfrak{M}$.

Proof. By induction on terms, using the Agreement Lemma. ∎

Now, we can put the two lemmas together to obtain:

Theorem A.4.1 *Let R be a closed term of* BRC. *Then, we have:*
(a) $\mathsf{free}\mathrm{FT}R = \{x, y\}$.
(b) $(a, b) \in [\![R]\!]_{\mathfrak{M}}$ *iff* $\mathfrak{M} \models \mathrm{FT}R\,[a, b]$, *for any model $\mathfrak{M}$ and points $a, b \in M$.*

Proof. (a) By Definition A.4.1 and Lemma A.4.1, since R is closed.
(b) Given a model $\mathfrak{M}$ and points $a, b \in M$, take an assignment β such that $\beta x = a$ and $\beta y = b$. ∎

A.4.2 Translating from FOL(R) into BRC

As it is expected, the translation from formulas of FOL(R) into terms of BRC is a little more evolved than the forward translation presented in Section A.4.1. But thanks to the presence of the accessibility operators in BRC and to the fact that binders behave like quantifiers, the translation presented in [70] is just as evolved as it needs to be. Again, we consider that individual variables x, y are distinguished. We assume that they do not have bounded occurrences in any formula of FOL(R), but may occur free.

Formula φ	Term BTφ	
$x \approx x$	E	
$x \approx y$	I	
$x \approx z$	$z \circ \mathsf{E}$	for $z \neq x, y$
$y \approx x$	I	
$y \approx y$	E	
$y \approx z$	$\mathsf{E} \circ z$	for $z \neq x, y$
$z \approx x$	$@_z^2\mathsf{I}$	for $z \neq x, y$
$z \approx y$	$@_z^1\mathsf{I}$	for $z \neq x, y$
$z \approx w$	$@_{zw}\mathsf{I}$	

Table A.8: Translating equalities from FOL(R) into BRC.

Formula φ	Term BTφ	
xrx	$(r \sqcap \mathsf{I}) \circ \mathsf{E}$	
xry	r	
xrz	$@^2_z r$	for $z \neq x, y$
yrx	r^{T}	
yry	$\mathsf{E} \circ (\mathsf{I} \sqcap r)$	
yrz	$@^1_z r^{\mathsf{T}}$	for $z \neq x, y$
zrx	$@^2_z r^{\mathsf{T}}$	for $z \neq x, y$
zry	$@^1_z r$	for $z \neq x, y$
zrw	$@_{zw} r$	for $z, w \neq x, y$

Table A.9: Translating atomic formulas from $\mathsf{FOL}(\mathrm{R})$ into BRC.

Formula φ	Term BTφ
$\neg\psi$	$\overline{\mathrm{BT}\psi}$
$\psi \wedge \theta$	$\mathrm{BT}\psi \sqcap \mathrm{BT}\theta$
$\psi \vee \theta$	$\mathrm{BT}\psi \sqcup \mathrm{BT}\theta$
$\exists z\psi$	$\downarrow_x(\mathsf{E} \circ \downarrow_z @^1_x \mathrm{BT}\psi)$
$\forall z\psi$	$\downarrow_x\overline{(\mathsf{E} \circ \downarrow_z \overline{@^1_x \mathrm{BT}\psi})}$

Table A.10: Translating non-atomic formulas from $\mathsf{FOL}(\mathrm{R})$ into BRC.

Definition A.4.2 Let φ be a formula of $\mathsf{FOL}(\mathrm{R})$ such that $\mathsf{free}\varphi \subseteq \{x, y\}$ and such that φ does not have any bounded occurrences of x or y. The *backward translation* of φ, is the term $\mathrm{BT}\varphi$, defined recursively in Tables A.8, A.9, and A.10.

The next lemma assures that BT behaves syntactically well.

Lemma A.4.3 *Let φ be a formula of $\mathsf{FOL}(\mathrm{R})$ such that $\mathsf{free}\varphi \subseteq \{x, y\}$ and such that φ do not have any bound occurrences of x or y. Then,* $\mathsf{free}\mathrm{BT}\varphi = \mathsf{free}\varphi \setminus \{x, y\}$.

Proof. By induction of formulas. Just note that, in the base cases, the occurrences of variables x, y are dropped when translating, and all the other occurrences of variables are maintained; in the quantifier cases, the bound occurrences of variables are translated to bound occurrences of the same variables, and the x introduced is bounded by the $\downarrow_x$. ■

No, we examine the semantical behavior of BT.

Lemma A.4.4 *Let φ be a formula of $\mathsf{FOL}(\mathrm{R})$ such that $\mathsf{free}\varphi \subseteq \{x, y\}$ and such that φ do not have any bound occurrences of x or y. Then the following hold:*

$$\mathfrak{M}, \beta \models \varphi \textit{ iff } \mathfrak{M}, \beta, \beta x, \beta y \Vdash \mathrm{BT}\varphi,$$

for any model $\mathfrak{M}$, and assignment β.

Proof. By induction of formulas. ■

Finally, we put the two lemmas together.

Theorem A.4.2 *Let φ be a formula of $\mathsf{FOL}(\mathrm{R})$ such that $\mathsf{free}\varphi \subseteq \{x, y\}$ and such that φ does not have any bound occurrences of x or y. Then, we have:*

(a) $\mathrm{BT}\varphi$ *is closed.*

(b) $\mathfrak{M} \models \varphi\ [a, b]$ *iff* $(a, b) \in [\![\mathrm{BT}\varphi]\!]_{\mathfrak{M}}$*, for any model $\mathfrak{M}$ and points $a, b \in M$.*

Apêndice B

Relational calculus with fork

In the previous chapter we brought some characteristics from contemporary modal logic to increase the expressive and deductive power of RC up to the level of FOL(R). In this chapter we shall prove a similar result but staying on the algebraic side of RC. In particular, we shall define PRC, the relation calculus with projection relations, and $\mathsf{FOL}(\mathrm{R}, \star)$, the first-order language of binary relations with coding. As we shall see, $\mathsf{FOL}(\mathrm{R}, \star)$ is the natural first-order counterpart of the relational formalism PRC. We shall prove that PRC has the same expressive power as $\mathsf{FOL}(\mathrm{R}, \star)$.

In defining PRC, we do not need to assign variables to points. So, in contrast with BRC, the system has just one sort for relations. But, to define PRC from RC, we have to change the latter in at least two aspects:

1. Restrict the class of models to *structured models.* These are models whose domains are closed under a function behaving like ordered pair formation from set theory.

2. Introduce exactly two *projection relations.* These are two new non-logical constant relations behaving as the set-theoretical projections on the first and second coordinates of ordered pairs.

This time, the operators introduced have natural counterparts in the algebraic semantics of programs [22, 87, 14]. So, besides projections we shall investigate some other operators linked to the introduction of concurrence and parallelism [38, 103] into RC.

The systems presented in this chapter are relational calculi versions of some known algebraic formalisms. More specifically, we shall investigate PRC, a relational calculus associated to a class of relation algebras appearing in [68], and FRC, a relational calculus analogous to the version of fork algebras defined in [39].

Our analysis of these systems leads to the results described as follows.

By presenting a carefully analysis of a set of operators having natural counterparts in the algebraic semantic of programs [22, 87, 14], we are to consider, besides projections and fork, also other operators linked to the introduction of concurrence and parallelism [38, 103] into RC. In particular, we consider the constant operator *double* and the binary operator *parallel*. Our analysis closes with a result summarizing the interdefinability of these operators.

Theorem 2 *The following hold on the interdefinability of projections, fork, double, and parallel operators:*
(a) *Double, parallel, and fork are definable from projections.*
(b) *Projections, double, and parallel, are definable from fork.*
(c) *Fork and projections are definable from double and parallel.*
(d) *Double is not definable from parallel. So, neither projections nor fork are.*
(e) *Parallel is not definable from double. So, neither projections nor fork are.*
(f) *Neither double nor parallel are definable from the Booleans and Peirceans. So, none of the other operators are.*

Although systems PRC and FRC are definitionally equivalent, PRC should be considered as the most basic system. We present some evidence for this claim by presenting a very natural set of equational axioms for PRC and by building FRC, very naturally, as a definitional extension of PRC. As a bonus from this analysis we obtain very simple axiomatizations, together with their respective completeness proofs, to both systems.

Theorem 3 (a) PRC *is axiomatizable by a finite set* Γ *of equations;*
(b) FRC *is axiomatized by extending* Γ *with a finite number of equations, one of them being just an equational definition of fork from projections.*

Finally, we characterize the expressive power of PFRC, a common extension of PRC and FRC, in terms of first-order logic. Given the special nature of our new non-logical operators, semantics is restricted to a special class of models. Accordingly, we define a special first-order language $\mathsf{FOL}(\mathrm{R}, \star)$ that is the natural language to be used as a yard-stick to measure the expressive power of PRC.

Theorem 4 *Let $\mathfrak{M}$ be a model of* $\mathsf{FOL}(\mathrm{R}, \star)$ *and X be binary relation on M. Then, X is definable by a term of* PRC *iff X is definable by a formula of* $\mathsf{FOL}(\mathrm{R}, \star)$ *having occurrences of the free variables x, y.*

To prove this result, we first define a forward translation from terms of PRC to the formulas of $\mathsf{FOL}(\mathrm{R}, \star)$, having exactly two free variables, that preserves the meaning of terms in the intended class of models. Then, we present the more evolved backward translation from formulas of $\mathsf{FOL}(\mathrm{R}, \star)$, having exactly two free variables, to terms of PRC. To be translated, each formula must be first rewritten in a normal form that restricts the order of the variables and the type of the atomic sub-formulas occurring in it. Moreover, to define the translation recursively we first define an auxiliary translation saying how formulas having more than two free variables should be translated. We deal with these details, presenting the translation in a series of steps and proving some intermediate results showing that it has the expected behavior.

The chapter is structured as follows. Section B.1 motivates the new semantics and defines the new non-logical operators whose addition to the machinery of RC seems to be necessary if one wants to deal with some kind of parallelism through a relational calculus extending RC. Section B.2 presents the relational calculi PRC and FRC. We present syntax and semantics and discuss inter-definability of operators. In Section B.3, we present sound sets of axioms for both systems and prove the respective completeness results. Section B.4 characterizes the expressive power of PFRC, a common extension of PRC and FRC. It is divided into three parts. First, in Sub-section B.4.1, we define the first-order language $\mathsf{FOL}(\mathrm{R}, \star)$. This is the natural language to be used as a yard-stick to measure the expressive power of PFRC. After that, in Section B.4.2, we define a forward translation from terms of PFRC to the formulas of $\mathsf{FOL}(\mathrm{R},\star)$, having exactly two free variables, which preserves the meaning of terms in structured models. In Section B.4.3 we present the backward translation from formulas of $\mathsf{FOL}(\mathrm{R},\star)$, having exactly two free variables, to terms of PFRC.

B.1 Parallelism, coding, projections, and fork

Given a term R of RC and a model $\mathfrak{M} = \langle M, r_i^{\mathfrak{M}} \rangle_{i\in\omega}$, the meaning $[\![R]\!]_{\mathfrak{M}}$ of R in $\mathfrak{M}$ is a binary relation on M. A basic idea of *relational semantics* [22] is to let the meaning of a program statement P be the binary relation X_P, which connects inputs (initial machine states) to outputs (final machine states). A natural setting for the manipulation of program specifications and meaning is to regard each term R of RC as the specification of a program and $[\![R]\!]_{\mathfrak{M}}$ as its input-output behavior. As a matter of fact, variants of RC have been used for this purpose for a long time [24, 7]. Here, we just drive attention to the fact that some features seem absent from RC for the proper treatment of parallelism, synchronization of processes, etc. Both practical [57] and theoretical [103] considerations suggest regarding parallel processes as manipulating ordered pairs of machine states. To this end, it seems necessary to introduce new operators into the RC apparatus.

The natural operators to manipulate ordered pairs in the relational framework are the direct product and its associated projections.

Definition B.1.1 Let A be a set. The *direct product* of A is the set:

$$A \times A = \{(a, b) : a, b \in A\}.$$

The associated *first* and *second projections* are the relations:

$$\mathsf{p} = \{((a, b), a) : (a, b) \in A \times A\}$$

and

$$\mathsf{q} = \{((a, b), b) : (a, b) \in A \times A\},$$

respectively.

We are interested in the case when p and q are restricted to equivalence relations on a base set $U \neq \emptyset$ considered as the *universe of discourse.*

Definition B.1.2 Let U be a non-empty set and $X \subseteq U \times U$ be an equivalence relation on U. The *first* and *second projections* associated to X are the relations:

$$\mathsf{p}^X = \{((a,b),a) : (a,b) \in X\}$$

and

$$\mathsf{q}^X = \{((a,b),b) : (a,b) \in X\},$$

respectively.

In the rest of this section, we shall write simply p and q in lieu of p^X and q^X when no ambiguity seems to arise.

Proposition B.1.1 *For any non-empty set U and equivalence relation $X \subseteq U \times U$, the following hold:*

$$\mathsf{p}^{-1} \mid \mathsf{p} = I_U; \tag{B.1}$$

$$\mathsf{q}^{-1} \mid \mathsf{q} = I_U; \tag{B.2}$$

$$\mathsf{p}^{-1} \mid \mathsf{q} = X; \tag{B.3}$$

$$(\mathsf{p} \mid \mathsf{p}^{-1}) \cap (\mathsf{q} \mid \mathsf{q}^{-1}) = I_X; \tag{B.4}$$

$$\mathsf{p} \mid X = \mathsf{q} \mid X. \tag{B.5}$$

Equations (B.1)–(B.5) in Proposition B.1.1 state that p and q are functional relations, sharing the same domain, covering the equivalence relation X, and satisfying a certain unicity condition on the pairs of elements of U belonging to X. To obtain a general approach in introducing projection relations into RC, we will make two simplifications. The first, non-essential, consists in taking only the sufficient halves of the equations (B.1)–(B.5). The second, which will imply a restriction on semantics, consists in to collapse the difference of types displayed, for instance, in equations (B.1) and (B.4). We shall reach this simplification by codifying pairs of elements of U as single elements of U. Hence, assuming that pairs of elements of U are coded as elements of U, we have:

Proposition B.1.2 *Let U be a non-empty set and X be an equivalence relation on U. Then,* p, q *satisfy equalities* (B.1)–(B.5) *iff they satisfy the following weaker equalities:*

$$\mathsf{p}^{-1} \mid \mathsf{p} \subseteq I_U; \tag{B.6}$$

$$\mathsf{q}^{-1} \mid \mathsf{q} \subseteq I_U; \tag{B.7}$$

$$X \subseteq \mathsf{p}^{-1} \mid \mathsf{q}; \tag{B.8}$$

$$(\mathsf{p} \mid \mathsf{p}^{-1}) \cap (\mathsf{q} \mid \mathsf{q}^{-1}) \subseteq I_X; \tag{B.9}$$

$$\mathsf{p} \mid X = \mathsf{q} \mid X. \tag{B.10}$$

Example B.1.1 below shows that equalities above do not provide a characterization of the set-theoretical projections p^X and q^X. In fact, there are many pairs of relations, essentially different from p^X and q^X, satisfying the equalities given in Proposition B.1.1 or B.1.2.

Example B.1.1 Let $\mathbb{N}$ be the set of natural numbers. Consider the Gödel numbering $\star : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ given by $m \star n ::= 2^m \cdot (2n+1)$. Observe that $\star$ is a binary injective function coding the pair (m, n) of natural numbers by the single natural number $m \star n \in \mathbb{N}$. From it we define $\mathsf{p}, \mathsf{q} : \mathbb{N} \to \mathbb{N}$ by setting $\mathsf{p}(m \star n) = m$ and $\mathsf{q}(m \star n) = n$. Taking $X = \mathbb{N} \times \mathbb{N}$, it is a simple exercise to prove that p, q satisfy equalities (B.6)–(B.10).

In general, to construct a pair of relations satisfying equalities (B.6)–(B.10), one just needs to take any binary function $\star : U \times U \to U$ satisfying a condition ensuring that $\star$ behaves in a manner close to the set-theoretical ordered-pair formation, that is, we require $\star$ to be injective in the sense that:

$$a \star b = c \star d \text{ iff } a = c \text{ and } b = d,$$

for any $a, b, c, d \in U$.

The freedom to define many different projections is what makes this approach interesting from an algebraic program point of view. One does not need actual Cartesian-like records; some coding for them is enough. The intuition behind this approach is that, in practical terms, as long as one can recover the given arguments from the coded pair, the particular coding schema adopted is not essential. So, we adopt the following definition.

Definition B.1.3 (1) A *pair-coding* on U is an injective function $\star : U \times U \to U$, coding each pair (a, b) of elements of U by the single element $a \star b \in U$.
(2) A *structured universe* is a structure $\langle U, \star \rangle$, where U is a non-empty set and $\star$ is a pair-coding on U.

Proposition B.1.3 *Let $\langle U, \star \rangle$ be a structured universe. Then, U is either a singleton or an infinite set.*

Restricting semantics to structured universes allows us to define new means of combining relations, taking advantage of the fact that its individual elements may have some structure given by $\star$. The following operators are typical and have been used extensively in the framework of relational semantics of programs [57, 103].

Definition B.1.4 Let $\langle U, \star \rangle$ be a structured universe and X, Y be binary relations on it. First, we define new distinguished relations.
(1) The *projections* on U are the relations $\mathsf{p}^\star$ and $\mathsf{q}^\star$, defined by:

$$\mathsf{p}^\star ::= \{(a \star b, a) : a, b \in U\}$$

and

$$\mathsf{q}^\star ::= \{(a \star b, b) : a, b \in U\}.$$

(2) The *duplication* on U is the relation defined by:

$$2^\star ::= \{(a, a \star a) : a \in U\}.$$

Second, we define new binary operators:
(3) The *parallel product* (or simply *parallel*) of X, Y is the relation $X \;\mathrm{II}_\star\; Y$, defined by:

$$X \;\mathrm{II}_\star\; Y ::= \{(a \star b, c \star d) : (a, c) \in X \text{ and } (b, d) \in Y\}.$$

(4) The *fork* of X, Y is the relation $X \angle_\star Y$, defined by:

$$X \angle_\star Y ::= \{(a, b \star c) : (a, b) \in X \text{ and } (a, c) \in Y\}.$$

The projections correspond to the programs for extracting components from structured data. In a picture, we have:

$$a \quad \xleftarrow{\mathsf{p}^\star} \quad \begin{pmatrix} a \\ \star \\ b \end{pmatrix} \quad \xrightarrow{\mathsf{q}^\star} \quad b$$

The duplication corresponds to a program to produce a structured data containing two copies of a data. In a picture, we have:

$$a \quad \xrightarrow{2^\star} \quad \begin{pmatrix} a \\ \star \\ a \end{pmatrix}$$

The parallel product $\mathrm{II}_\star$ corresponds to the parallel execution of two programs. In a picture, we have:

$$\begin{pmatrix} a \\ \star \\ b \end{pmatrix} \quad \xrightarrow{X \,\mathrm{II}_\star\, Y} \quad \begin{pmatrix} c \\ \star \\ d \end{pmatrix} \quad \Leftrightarrow \quad \begin{pmatrix} a \xrightarrow{X} c \\ \text{and} \\ b \xrightarrow{Y} d \end{pmatrix}$$

The fork operator $\angle_\star$ comes from the idea of feeding a common input to two processes. In a picture:

$$a \quad \xrightarrow{X \,\angle_\star\, Y} \quad \begin{pmatrix} b \\ \star \\ c \end{pmatrix} \quad \Leftrightarrow \quad \begin{pmatrix} a \xrightarrow{X} b \\ \text{and} \\ a \xrightarrow{Y} c \end{pmatrix}$$

The comments above confirm that in order to be used as a formalism for the manipulation of specifications and meaning of programs involving features related to parallelism, synchronization, etc., the relational calculus RC should be extended by introducing new operators interpreted on structured universes. To this end, the universe U should have coded pairs of its elements, i.e., it is to be closed under a coding function $\star$. So, we no longer have U as an unstructured set of elements. Instead, we will be dealing with a universe with structure, having besides the usual objects $a, b, c, \ldots$, also structured objects as $a \star a$, $a \star b$, $c \star c$, $a \star (a \star a)$, $a \star (a \star b), \ldots$. Thanks to this structured character of the universe, we are able to restrict attention to some new, structural, operations on relations that appear to be convenient and natural on the relational framework of programs. Some papers choose to take fork as the primitive concept since the others are relationally definable from it [40, 37, 53]. In the next two sections we shall see that by choosing projections as primitives we can make a smoother transition from RC to the new formalism.

B.2 Projective relation calculi

Let $2, \pi, \rho$ be zero-ary operators, and $\|, \nabla$ binary ones. In this section, we extend the syntax and semantics of RC to obtain a relation calculi XRC, for every $\mathsf{X} \subseteq \{\pi, \rho, 2, \|, \nabla\}$. When $\mathsf{X} = \emptyset$, of course, we have that XRC is the same as RC. We close the section investigating the inter-definability of these operators.

Definition B.2.1 (1) To each operator in $\{\pi, \rho, 2, \|, \nabla\}$ we have associated a generating grammar "rule" as follows:
(1.a) to each zero-ary operator op the rule $R ::= \mathsf{op}$;
(1.b) to each binary operator op the rule $R ::= R \;\mathsf{op}\; R$.
(2) Compared with the syntax of RC, we add the following to constitute the language of XRC:
(2.a) the operators in X;
(2.b) to the generating grammar the "rules" corresponding to the elements of X.

To interpret the operators in X as the formal counterparts of the operators in Definition B.1.4, we need to restrict semantics to a proper class of models. This is obtained by considering just the models of RC whose universes are structured sets according to Definition B.1.3.

Definition B.2.2 (1) A *structured model* is a triple $\mathfrak{M} = \langle M, \star, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$, where $\langle M, \star \rangle$ is an structured universe and $r_i^{\mathfrak{M}} \subseteq M \times M$ for any $i \in \omega$.
(2) Starting with the semantics of RC, we add to the definition of meaning the clauses, chosen from below, corresponding to the operators in X, to constitute the language of XRC:

$$\begin{aligned}
[\![\pi]\!]_{\mathfrak{M}} &::= \{(a \star b, a) : a, b \in M\}, \\
[\![\rho]\!]_{\mathfrak{M}} &::= \{(a \star b, b) : a, b \in M\}, \\
[\![2]\!]_{\mathfrak{M}} &::= \{(a, a \star a) : a \in M\}, \\
[\![R \| S]\!]_{\mathfrak{M}} &::= \{(a \star b, c \star d) : a, b, c, d \in M, (a, c) \in [\![R]\!]_{\mathfrak{M}}, \text{ and } (b, d) \in [\![S]\!]_{\mathfrak{M}}\}, \\
[\![R \nabla S]\!]_{\mathfrak{M}} &::= \{(a, b \star c) : a, b, c \in M, (a, b) \in [\![R]\!]_{\mathfrak{M}}, \text{ and } (a, c) \in [\![S]\!]_{\mathfrak{M}}\}.
\end{aligned}$$

For any set X, the meaning $[\![R]\!]_{\mathfrak{M}}$ of a term R of XRC in a structured model $\mathfrak{M}$ is a binary relation on M. But note that, since we are now restricted to structured models, the elements of $[\![R]\!]_{\mathfrak{M}}$ are not necessarily ordered pairs of *atomic elements* as $(a,b),(c,a),\ldots$, but they may also include ordered pairs of *structured elements* such as $(a\star a, a\star b)$, $(c,(a\star a)\star a)$, $(a\star(a\star b), c\star a)\ldots$

All the other syntactical and semantical basic notions of RC are extended to XRC without trouble, modulo the operators belonging to X. Let us call PRC the relational calculus $\{\pi,\rho\}\mathsf{RC}$, defined as above.

From Definition B.2.2, given a structured model $\mathfrak{M}$, the meanings of π and ρ in $\mathfrak{M}$ are, respectively, the projections $\mathsf{p}^\star$ and $\mathsf{q}^\star$ of $\langle M,\star\rangle$. What seems to be lacking to PRC are the other useful operators $2^\star$, $\amalg_\star$ and $\angle_\star$, as encountered at the end of Section B.1. But, as we shall see below, thanks to the injectivity of $\star$ these operators are already definable in PRC.

Definition B.2.3 Let $\mathsf{X}\subseteq\{\pi,\rho,2,\|,\nabla\}$ and op be an m-ary operator on binary relations. We say that op is *term-definable* in XRC if there is a term $R(r_1,\ldots,r_m)$ of XRC containing occurrences of exactly the relational variables $r_1,\ldots,r_m$ and no occurrences of op, such that:

$$\mathfrak{M},a,b\Vdash R(r_1,\ldots,r_m) \text{ iff } (a,b)\in\mathsf{op}(X_1,\ldots,X_m),$$

for every structured model $\mathfrak{M}$ and binary relations $X_1,\ldots,X_m$ on M for which $r_1^{\mathfrak{M}}=X_1,\ldots,r_m^{\mathfrak{M}}=X_m$.

Let us call FRC the relational calculus $\{\nabla\}\mathsf{RC}$, as defined above.

Proposition B.2.1 *The following hold:*

(a) 2*,* $\|$*, and* ∇ *are term-definable in* PRC*.*

(b) π *and* ρ *are term-definable in* FRC*. As a consequence,* 2 *and* $\|$ *are also term-definable in* FRC*.*

(c) ∇ *is term-definable in* $\{2,\|\}\mathsf{RC}$*. As a consequence,* π *and* ρ *are also term-definable in* $\{2,\|\}\mathsf{RC}$*.*

(d) 2 *is not term-definable in* $\{\|\}\mathsf{RC}$*. As a consequence, neither* π *nor* ρ *simultaneously, nor* ∇ *are term-definable in* $\{\|\}\mathsf{RC}$*.*

(e) $\|$ *is not term-definable in* $\{2\}$RC. *As a consequence, neither* π *nor* ρ *simultaneously, nor* ∇ *are term-definable in* $\{2\}$RC.
(f) *Neither* 2 *nor* $\|$ *are term-definable in* RC. *So neither* π *nor* ρ *simultaneously, nor* $\|$, *nor* ∇ *are term-definable in* RC.

PROOF. (a) Given relational variables r, s define $2 ::= \pi^{\mathsf{T}} \sqcap \rho^{\mathsf{T}}$, $r\|s ::= (\pi \circ r \circ \pi^{\mathsf{T}}) \sqcap (\rho \circ s \circ \rho^{\mathsf{T}})$, and $r \nabla s ::= (r \circ \pi^{\mathsf{T}}) \sqcap (s \circ \rho^{\mathsf{T}})$.
(b) Define $\pi ::= (\mathsf{I} \nabla \mathsf{E})^{\mathsf{T}}$ and $\rho ::= (\mathsf{E} \nabla \mathsf{I})^{\mathsf{T}}$.
(c) Given relational variables r and s, define $r \nabla s ::= 2 \circ (r\|s)$.
(d) Consider the structured model $\mathfrak{M} = \langle \mathbb{N}, \star, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ of $\{\|\}$RC whose $\star$ is defined by $m \star n = 2^m(2n+1)$ for all $m, n \in \mathbb{N}$ and whose set $\{r_i^{\mathfrak{M}} : i \in \omega\}$ corresponds to the set of all binary relations definable in $\mathbb{N}$ by formulas of the first-order language with 0 (zero) and S (successor). It can be shown by induction on terms that for every term R of $\{\|\}$RC the range of $[\![R]\!]_{\mathfrak{M}}$ is a finite or co-finite set. But we have that the range of the relation $2^\star = \{(m, 2^m(2m+1)) : m \in \mathbb{N}\}$ is an infinite set whose complement is also infinite. So, 2 is not term-definable in $\{\|\}$RC.
(e) Consider the structured model $\mathfrak{M} = \langle \mathbb{N}, \star, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ and $\mathfrak{M}' = \langle \mathbb{N}, \star', r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ of $\{2\}$RC with $\star$ and $\star'$ defined, respectively, by $m \star n = 2^m(2n+1)$ and $m \star' n = 2^n(2m+1)$ for all $m, n \in \mathbb{N}$, and with $r_0^{\mathfrak{M}} = \{(0,0)\}$, $r_1^{\mathfrak{M}} = \{(0,1)\}$, and $r_i^{\mathfrak{M}} = \emptyset$, for every $i > 1$. It can be shown by induction on terms that every term of $\{2\}$RC has the same meaning on $\mathfrak{M}$ and $\mathfrak{M}'$. But the meaning of $r_0\|r_1$ on $\mathfrak{M}$ is $\{(1,3)\}$ and on $\mathfrak{M}'$ is $\{(1,2)\}$. So, $\|$ is not term-definable in $\{2\}$RC.
(f) Consider a structured model $\mathfrak{M} = \langle \mathbb{N}, \star, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ of RC whose $\star$ is defined by $m \star n = 2^m(2n+1)$ for all $m, n \in \mathbb{N}$, and for which $r_i^{\mathfrak{M}} = \emptyset$ for every $i \in \omega$. It can be shown by induction on terms that $[\![R]\!]_{\mathfrak{M}} \in \{\emptyset, I_{\mathbb{N}}, (I_{\mathbb{N}})^c, \mathbb{N} \times \mathbb{N}\}$ for every term R of RC. But we have that both relations $2^\star$ and $(I_{\mathbb{N}} \mathrm{II}_\star I_{\mathbb{N}})$ are different from $\emptyset, I_{\mathbb{N}}, (I_{\mathbb{N}})^c$, and $\mathbb{N} \times \mathbb{N}$. So, 2 and $\|$ are not term-definable in RC. ∎

B.3 Axiomatizing projective relation calculi

Our main objective is to characterize the expressive power of the system obtained by introducing the two new operators π and ρ, and consequently, 2, $\|$, and ∇ to the language of RC. This will be the subject of Section B.4. In this section, we shall study questions related to axiomatizability. Considering π and ρ as primitives, we present proofs in an uniform, modular, way. In fact, the results in this section can be generalized to: over structured domains, any operator term-definable from π and ρ is axiomatized by the addition of its definition to the axioms of projective relation algebras.

First, we define projective relation algebra, an algebraic counterpart to PRC. Second, we prove that projective relation algebra relates to PRC in the same way RA relates to RC. Hence, some questions about PRC can be answered through algebraic tools. In particular, we shall employ an usual algebraic technique to prove completeness of sets of equations for classes of proper algebras: the definition of an abstract class of algebras by a set of equations and the proof that every algebra in the abstract class is isomorphic to a proper one. This *representation theorem* implies that the set of equations defining the abstract class is complete for the class of proper algebras.

B.3.1 Axiomatizing projective relation algebras

Definition B.3.1 An algebraic structure $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I}, \pi, \rho \rangle$ is a *projective relation algebra* if its relational reduct $\langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I} \rangle$ is a relation algebra and axioms (B.11)–(B.15) below are true in $\mathfrak{A}$.

$$\pi^{\mathsf{T}} \circ \pi \sqsubseteq \mathsf{I}, \tag{B.11}$$

$$\rho^{\mathsf{T}} \circ \rho \sqsubseteq \mathsf{I}, \tag{B.12}$$

$$\mathsf{E} \sqsubseteq \pi^{\mathsf{T}} \circ \rho, \tag{B.13}$$

$$(\pi \circ \pi^{\mathsf{T}}) \sqcap (\rho \circ \rho^{\mathsf{T}}) \sqsubseteq \mathsf{I}, \tag{B.14}$$

$$\pi \circ \mathsf{E} \approx \rho \circ \mathsf{E}. \tag{B.15}$$

If $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I}, \pi, \rho \rangle$ is a projective relation algebra whose relational reduct $\langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I} \rangle$ is a proper relation algebra with base set U and biggest relation E, then π and ρ are binary relations on U. Axioms (B.11) and (B.12) state that π and ρ are both functional. Axiom (B.13) states that any pair of elements that are related by E can be coded as a pair. Axiom (B.14) states that pairs having the same first and second coordinates are equal. Axiom (B.15) states that both projections have the same domain.

Lemma B.3.1 *The relational reduct of a projective relation algebra is a representable relation algebra.*

Proof. Let $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I}, \pi, \rho\rangle$ be a projective relation algebra. By Axioms (B.11)– (B.13), we have that $\pi, \rho \in A$ satisfy:

$$(R^{\mathsf{T}} \circ R \sqsubseteq \mathsf{I}) \wedge (S^{\mathsf{T}} \circ S \sqsubseteq \mathsf{I}) \wedge (\mathsf{E} \sqsubseteq R^{\mathsf{T}} \circ S),$$

when R is replaced by π and S by ρ, respectively. So, the relational reduct of $\mathfrak{A}$ satisfies:

$$\exists R \exists S((R^{\mathsf{T}} \circ R \sqsubseteq \mathsf{I}) \wedge (S^{\mathsf{T}} \circ S \sqsubseteq \mathsf{I}) \wedge (\mathsf{E} \sqsubseteq R^{\mathsf{T}} \circ S)),$$

and, since Theorem VII of [93] warrants the representability of relations algebras satifying this condiction, it is representable. ∎

Definition B.3.2 Let $\mathfrak{A} = \langle A, \cup, \cap, |, ^{c}, ^{-1}, \emptyset, E, I, \mathsf{p}, \mathsf{q}\rangle$ be an algebraic structure. We say that $\mathfrak{A}$ is a *proper projective relation algebra* if there is a structured universe $\langle U, \star\rangle$ such that:

(1) The relational reduct $\langle A, \cup, \cap, |, ^{c}, ^{-1}, \emptyset, E, I\rangle$ of $\mathfrak{A}$ is a proper relation algebra with base set U and biggest relation E. So, in particular, E is an equivalence relation, A is a subset of 2^{E}, and I is the identity on U.

(2) p and q are, respectively, the first and the second projections associated to $\star$.

A proper projective relation algebra is not exactly a proper algebra [52], since the operations in a proper projective relation algebra are not completely determined by its universe, but may depend on the choice of $\star$. A proper projective relation algebra is a *quasi-proper* algebra: its operations are completely determined by the structured universe $\langle U, \star\rangle$. Hence, for projective relation algebras (as well as for fork algebras), we have a *weak* representation result: every projective relation algebra (fork algebra) is isomorphic to a quasi-proper projective relation algebra (fork algebra). Next result is immediate from definitions:

Proposition B.3.1 *Every proper projective relation algebra is a projective relation algebra.*

More interesting is to establish its converse: the weak representation result for projective relation algebras. To this end, we prove the following lemma, which was in the background for both proofs of (weak) representation of fork algebras, presented in [39] and [52].

Lemma B.3.2 *Let $\mathfrak{A}$ be a projective relation algebra with relational reduct $\mathfrak{B}$. Let $\mathfrak{B}'$ be a proper relation algebra and $\phi : A \to B'$ be a homomorphism from $\mathfrak{B}$ to $\mathfrak{B}'$. Then, there is an expansion $\mathfrak{A}'$ of $\mathfrak{B}'$ by distinguished relations p, q such that:*
(a) *$\mathfrak{A}'$ is a proper projective relation algebra.*
(b) *There is a homomorphism $\phi' : A \to A'$ from $\mathfrak{A}$ to $\mathfrak{A}'$.*

Proof. Let $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I}, \pi, \rho \rangle$ be a projective relation algebra with relational reduct $\mathfrak{B} = \langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I} \rangle$.

Suppose $\mathfrak{B}' = \langle B', \cup, \cap, \mid, ^{c}, ^{-1}, E, \emptyset, I \rangle$ is a proper relation algebra with base set U and greater relation E. Let $\phi : A \to B'$ be a homomorphism from $\mathfrak{B}$ to $\mathfrak{B}'$.

Since $\pi, \rho \in A$, we have that $\phi\pi, \phi\rho \in B'$. Since π and ρ satisfy Axioms (B.11)–(B.15), we have that $\phi\pi, \phi\rho$ are functional relations having the same domain, covering E, and warranting unicity of ordered pairs.

We shall prove that conditions above suffice to define an injective function $\star : E \to U$ for which $\phi\pi = \mathsf{p}^{\star}$ and $\phi\rho = \mathsf{q}^{\star}$.

Define $f \subseteq (U \times U) \times U$ in the following way, for all $a, b, c \in U$:

$$((a, b), c) \in f \text{ iff } (c, a) \in \phi\pi \text{ and } (c, b) \in \phi\rho.$$

We shall prove that the domain of f contains E and that f_E, the restriction of f to E, is an injective functional relation. Note that, by definition, $f_E \subseteq E \times U$ is such that:

$$((a, b), c) \in f_E \text{ iff } (a, b) \in E \text{ and } ((a, b), c) \in f.$$

All these facts together allow us to define $\star : E \to U$ by setting:

$$a \star b = f_E(a, b),$$

for any pair $(a, b) \in E$. From this definition it is obvious that

$$a \star b = c \text{ iff } (c, a) \in \phi\pi \text{ and } (c, b) \in \phi\rho,$$

for any $(a, b) \in E$ and $c \in U$. Now, we expand $\mathfrak{B}'$ with the two distinguished relations:

$$\mathsf{p}^\star = \{(a \star b, a) : (a, b) \in E\}$$

and

$$\mathsf{q}^\star = \{(a \star b, b) : (a, b) \in E\},$$

and we are ready to prove items (a) and (b) from the lemma.

To prove (a): by definition, $\mathfrak{A}' = \langle B', \cup, \cap, \mid, {}^c, {}^{-1}, \emptyset, E, I, \mathsf{p}^\star, \mathsf{q}^\star \rangle$ is a proper projective relation algebra.

To prove (b): since $\mathfrak{A}$ and $\mathfrak{B}$, respectively, $\mathfrak{A}'$ and $\mathfrak{B}'$, have the same universe A, respectively B', we just need to prove that $\phi\pi = \mathsf{p}^\star$ and $\phi\rho = \mathsf{q}^\star$. Let $(a, b) \in \phi\pi$. Hence, $a \in \mathsf{Dom}\phi\pi$. Now, by Axiom (B.15), we have that $\mathsf{Dom}\phi\pi = \mathsf{Dom}\phi\rho$. So, $a \in \mathsf{Dom}\phi\rho$ as well, and there is some $c \in U$ for which $(a, c) \in \phi\rho$. From $(a, b) \in \phi\pi$ and $(a, c) \in \phi\rho$, we have $a = b \star c$. But this implies that $(a, b) = (b \star c, b)$ that, in its turn, implies $(a, b) \in \mathsf{p}^\star$. To prove the other inclusion, suppose $(a, b) \in \mathsf{p}^\star$. Hence, $a = b \star c$ and $(b, c) \in E$. So, $(a, b) \in \phi\pi$.

The proof that $\phi\rho = \mathsf{q}^\star$ is entirely analogous.

To conclude the proof we just need the following simple claims.

Claim 1. $E \subseteq \mathsf{Dom}f$. Let $(a, b) \in E$. By Axiom (B.13), we have $(a, b) \in \phi\pi^{-1} \mid \phi\rho$. Hence, $\exists c \in U : (c, a) \in \phi\pi$ and $(c, b) \in \phi\rho$. Taking such an element c, we have $((a, b), c) \in f$. So, $(a, b) \in \mathsf{Dom}f$.

Claim 2. *f_E is functional.* Let $(a, b) \in E$. Suppose $c, c' \in U$ are such that $((a, b), c) \in f_E$ and $((a, b), c') \in f_E$. Hence, $(c, a), (c', a) \in \phi\pi$ and $(c, b), (c', b) \in \phi\rho$. But this gives us $(c, c') \in (\phi\pi \mid (\phi\pi)^{-1}) \cap (\phi\rho \mid (\phi\rho)^{-1})$. So, by Axiom (B.14), we have $(c, c') \in \phi\mathsf{I}$, that is, $c = c'$.

Claim 3. *f_E is injective.* Let $(a_1, a_2), (b_1, b_2) \in E$ and $c \in U$ be such that $((a_1, a_2), u), ((b_1, b_2), c) \in f_E$. Hence, $(c, a_1), (c, b_1) \in \phi\pi$ and $(c, a_2), (c, b_2) \in \phi\rho$. This gives us $(a_1, b_1) \in (\phi\pi)^{-1} \mid \phi\pi$ and $(a_2, b_2) \in (\phi\rho)^{-1} \mid \phi\rho$. Now, from these and Axioms (B.11) and (B.12), we have $(a_1, b_1), (a_2, b_2) \in \phi\mathsf{I}$, that is, $a_1 = b_1$ and $a_2 = b_2$. So, $(a_1, a_2) = (b_1, b_2)$. ■

Corollary B.3.1 (Representation Theorem) *Each projective relation algebra is isomorphic to a proper projective relation algebra.*

Proof. Let $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I}, \pi, \rho \rangle$ be a projective relation algebra. By Lemma B.3.1, the relational reduct $\mathfrak{B}$ of $\mathfrak{A}$ is a representable relation algebra. Let $\mathfrak{B}'$ be a proper relation algebra and $\phi : A \to B'$ be an isomorphism from $\mathfrak{B}$ into $\mathfrak{B}'$. By Lemma B.3.2, $\mathfrak{B}'$ can be expanded to a proper projective relation algebra $\mathfrak{A}'$ in a such way that $\phi : A \to A'$ is a homomorphism from $\mathfrak{A}$ to $\mathfrak{A}'$. Since $\phi : A = B \to B' = A'$ is bijective, we have that ϕ is, in fact, an isomorphism from $\mathfrak{A}$ onto $\mathfrak{A}'$. ■

The Representation Theorem has the Completeness Theorem as a corollary.

Corollary B.3.2 (Completeness Theorem) *The class of proper projective relation algebras is axiomatized by the equalities* (B.11)–(B.15) *plus axioms for* RA.

B.3.2 Axiomatizing fork algebras

We can use Lemma B.3.2 to obtain very simple completeness proofs for axioms of fork algebras, following the route started in [39, 52].

Definition B.3.3 [Frias et al, 1997; Gyuris, 1997] We say that an algebraic structure $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, \nabla, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I} \rangle$ is a *fork algebra* when its relational reduct $\langle A, \sqcup, \sqcap, \circ, ^{-}, ^{\mathsf{T}}, \mathsf{O}, \mathsf{E}, \mathsf{I} \rangle$ is a relation algebra and the axioms (B.16)–(B.18) below are true in $\mathfrak{A}$.

$$r \nabla s \approx (r \circ (\mathsf{I} \nabla \mathsf{E})) \sqcap (s \circ (\mathsf{E} \nabla \mathsf{I})), \tag{B.16}$$

$$(r \nabla s) \circ (t \nabla u)^{\mathsf{T}} \approx (r \circ t^{\mathsf{T}}) \sqcap (s \circ u^{\mathsf{T}}), \tag{B.17}$$

$$(\mathsf{I} \nabla \mathsf{E})^{\mathsf{T}} \nabla (\mathsf{E} \nabla \mathsf{I})^{\mathsf{T}} \sqsubseteq \mathsf{I}. \tag{B.18}$$

At a first sight, the complete meaning of Axioms (B.16)–(B.18) is not so easy to grasp. They have been subject of extensive investigation as reported in [35, 101, 102]. Here, we move to a more suggestive set of equations, linking the notion of a fork algebra to the former notion of a projective relation algebra.

Lemma B.3.3 *Let* $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, \nabla, ^-, ^\mathsf{T}, \mathsf{O}, \mathsf{E}, \mathsf{I}\rangle$ *be an algebraic structure whose relational reduct* $\langle A, \sqcup, \sqcap, \circ, ^-, ^\mathsf{T}, \mathsf{O}, \mathsf{E}, \mathsf{I}\rangle$ *is a relation algebra. Then the following are equivalent:*

(a) $\mathfrak{A}$ *is a fork algebra.*

(b) *Axioms* (B.19)–(B.24) *below are true in* $\mathfrak{A}$.

$$(\mathsf{I}\nabla\mathsf{E}) \circ (\mathsf{I}\nabla\mathsf{E})^\mathsf{T} \sqsubseteq \mathsf{I}, \tag{B.19}$$

$$(\mathsf{E}\nabla\mathsf{I}) \circ (\mathsf{E}\nabla\mathsf{I})^\mathsf{T} \sqsubseteq \mathsf{I}, \tag{B.20}$$

$$\mathsf{E} \sqsubseteq (\mathsf{I}\nabla\mathsf{E}) \circ (\mathsf{E}\nabla\mathsf{I})^\mathsf{T}, \tag{B.21}$$

$$(((\mathsf{I}\nabla\mathsf{E})^\mathsf{T} \circ (\mathsf{I}\nabla\mathsf{E})) \sqcap (((\mathsf{E}\nabla\mathsf{I})^\mathsf{T} \circ (\mathsf{E}\nabla\mathsf{I})) \approx \mathsf{I}, \tag{B.22}$$

$$(\mathsf{I}\nabla\mathsf{E})^\mathsf{T} \circ \mathsf{E} \approx (\mathsf{E}\nabla\mathsf{I})^\mathsf{T} \circ \mathsf{E}, \tag{B.23}$$

$$r\nabla s \approx (r \circ (\mathsf{I}\nabla\mathsf{E})) \sqcap (s \circ (\mathsf{E}\nabla\mathsf{I})). \tag{B.24}$$

Using the fact that $\Vdash R^{\mathsf{T}^\mathsf{T}} = R$, we see that Axioms (B.19)–(B.23) are no more than the Axioms (B.11)–(B.15) rewritten by replacing every occurrence of π and ρ by $(\mathsf{I}\nabla\mathsf{E})^\mathsf{T}$ and $(\mathsf{E}\nabla\mathsf{I})^\mathsf{T}$, respectively. So Axioms (B.19)–(B.23) just say that $(\mathsf{I}\nabla\mathsf{E})^\mathsf{T}, (\mathsf{E}\nabla\mathsf{I})^\mathsf{T}$ is a pair of projective relations. Axiom (B.24) is exactly Axiom (B.16) which, in this context, defines the fork operator ∇ from the relational operators $\sqcap, \circ$ and the converses of the projective relations $(\mathsf{I}\nabla\mathsf{E})^\mathsf{T}$ and $(\mathsf{E}\nabla\mathsf{I})^\mathsf{T}$.

Proof. Let $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, \nabla, ^-, ^\mathsf{T}, \mathsf{O}, \mathsf{E}, \mathsf{I}\rangle$ be an algebraic structure whose relational reduct satisfies the relation algebra axioms. To prove that (a) implies (b), suppose that $\mathfrak{A}$ satisfies Axioms (B.16)–(B.18).

To prove Axiom (B.19), we reason as follows:

$$\begin{aligned}(\mathsf{I}\nabla\mathsf{E}) \circ (\mathsf{I}\nabla\mathsf{E})^\mathsf{T} &\approx (\mathsf{I} \circ \mathsf{I}^\mathsf{T}) \sqcap (\mathsf{E} \circ \mathsf{E}^\mathsf{T}) && \text{(Ax (B.17))}\\ &\sqsubseteq \mathsf{I} && \text{(RA)}\end{aligned}$$

To prove Axiom (B.20), we reason in a similar way.

To prove Axiom (B.21), we reason as follows:

$$\begin{aligned}\mathsf{E} &\sqsubseteq (\mathsf{I} \circ \mathsf{E}^\mathsf{T}) \sqcap (\mathsf{E} \circ \mathsf{I}^\mathsf{T}) && \text{(RA)}\\ &\approx (\mathsf{I}\nabla\mathsf{E}) \circ (\mathsf{I}\nabla\mathsf{E})^\mathsf{T} && \text{(Ax (B.17))}\end{aligned}$$

To prove Axiom (B.22), we reason as follows:

$$\begin{aligned}
&((\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}} \circ (\mathsf{I}\nabla\mathsf{E})) \sqcap ((\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \circ (\mathsf{E}\nabla\mathsf{I})) \\
&\quad\approx\ ((\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}} \circ (\mathsf{I}\nabla\mathsf{E}))^{\mathsf{T}^{\mathsf{T}}} \sqcap ((\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \circ (\mathsf{E}\nabla\mathsf{I}))^{\mathsf{T}^{\mathsf{T}}} && (\mathsf{RA}) \\
&\quad\approx\ ((\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}}\nabla(\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}}) \circ ((\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}}\nabla(\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}})^{\mathsf{T}} && (\text{Ax (B.17)}) \\
&\quad\sqsubseteq\ \mathsf{I} \circ \mathsf{I}^{\mathsf{T}} && (\text{Ax (B.18)}) \\
&\quad\approx\ \mathsf{I} && (\mathsf{RA})
\end{aligned}$$

To prove Axiom (B.23), we prove two inequalities. The first one is as follows:

$$\begin{aligned}
(\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}} \circ \mathsf{E} &\approx ((\mathsf{I} \circ (\mathsf{I}\nabla\mathsf{E})) \sqcap (\mathsf{E} \circ (\mathsf{E}\nabla\mathsf{I})))^{\mathsf{T}} \circ \mathsf{E} && (\text{Ax (B.16)}) \\
&\approx ((\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}} \sqcap (\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \circ \mathsf{E})) \circ \mathsf{E} && (\mathsf{RA}) \\
&\sqsubseteq ((\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \circ \mathsf{E}) \circ \mathsf{E} && (\mathsf{RA}) \\
&\approx (\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \circ \mathsf{E} && (\mathsf{RA})
\end{aligned}$$

The second, that $(\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \circ \mathsf{E} \sqsubseteq (\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}} \circ \mathsf{E}$, is proved in a entirely similar way.

To prove that (b) implies (a), we assume that Axioms (B.19)–(B.23) are true on $\mathfrak{A}$. To prove Axiom (B.17), we just have to apply Axiom (B.24) together with Theorem 4.1(viii) of [95], which states:

$$\text{Axs } (B.19), (B.20), B.21) \ \Vdash \ \begin{array}{c} ((r \circ \pi^{\mathsf{T}}) \sqcap (s \circ \rho^{\mathsf{T}})) \circ ((t \circ \pi^{\mathsf{T}}) \sqcap (u \circ \rho^{\mathsf{T}})) \\ \approx \\ (r \circ t^{\mathsf{T}}) \sqcap (s \circ u^{\mathsf{T}}). \end{array}$$

To prove Axiom (B.18), we reason as follows:

$$\begin{aligned}
&(\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}}\nabla(\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \\
&\quad\approx\ ((\mathsf{I}\nabla\mathsf{E})^{\mathsf{T}} \circ (\mathsf{I}\nabla\mathsf{E})) \sqcap ((\mathsf{E}\nabla\mathsf{I})^{\mathsf{T}} \circ (\mathsf{E}\nabla\mathsf{I})) && (\text{Ax (B.24)}) \\
&\quad\sqsubseteq\ \mathsf{I} \sqcap \mathsf{I} && (\text{Axs (B.19),(B.20)}) \\
&\quad\sqsubseteq\ \mathsf{I} && (\mathsf{RA})
\end{aligned}$$

This finishes the proof. ∎

Definition B.3.4 Let $\mathfrak{A} = \langle A, \cup, \cap, \mid, \angle, ^c, ^{-1}, \emptyset, E, I\rangle$ be an algebraic structure. We say that $\mathfrak{A}$ is a *proper fork algebra* if there is a structured universe $\langle U, \star\rangle$ such that:

(1) The relational reduct $\langle A, \cup, \cap, \mid, ^c, ^{-1}, \emptyset, E, I\rangle$ of $\mathfrak{A}$ is a proper relation algebra with base set U and biggest relation E. So, in particular, E is an equivalence relation, A is a subset of 2^E, and I is the identity on U.

(2) $\star : E \to U$ is an injective function.

(3) $\angle$ is the fork operator associated to $\langle U, \star\rangle$, restricted to A.

Next result is immediate from definitions:

Proposition B.3.2 *Every proper fork algebra is a fork algebra.*

The converse can be easily established by applying Lemma B.3.3 and Corollary B.3.1.

Corollary B.3.3 (Representation Theorem) *Each fork algebra is isomorphic to a proper fork algebra.*

Proof. Let $\mathfrak{A} = \langle A, \sqcup, \sqcap, \circ, \nabla, ^-, ^\mathsf{T}, \mathsf{O}, \mathsf{E}, \mathsf{I}\rangle$ be a fork algebra with relational reduct $\langle A, \sqcup, \sqcap, \circ, ^-, ^\mathsf{T}, \mathsf{O}, \mathsf{E}, \mathsf{I}, \rangle$. It follows, from Lemma B.3.3, that the algebraic structure $\mathfrak{B} = \langle A, \sqcup, \sqcap, \circ, ^-, ^\mathsf{T}, \mathsf{O}, \mathsf{E}, \mathsf{I}, \pi, \rho\rangle$, where π and ρ denote, respectively, the relations $(\mathsf{I}\nabla\mathsf{E})^\mathsf{T}$ and $(\mathsf{E}\nabla\mathsf{I})^\mathsf{T}$ is a projective relation algebra. So, by Corollary B.3.1, it is representable. Now, let $\mathfrak{B}' = \langle A', \cup, \cap, \mid, ^c, ^{-1}, \emptyset, E, I, \mathsf{p}^\star, \mathsf{q}^\star\rangle$ be the proper projective relation algebra on the structured set $\langle U, \star\rangle$ and having biggest relation E, given in the proof of Lemma B.3.1.

We know that there is an isomorphism $\phi : A \to A'$ from $\mathfrak{B}$ to $\mathfrak{B}'$ and, since ϕ preserves conversion, that $\star : E \to U$ is defined by setting, for all pair $(a, b) \in E$ and $c \in U$:

$$a \star b = c \text{ iff } (a, c) \in \phi(\mathsf{I}\nabla\mathsf{E}) \text{ and } (b, c) \in \phi(\mathsf{E}\nabla\mathsf{I}).' \tag{B.25}$$

Let $\mathfrak{A}'$ be the proper fork algebra $\langle A', \cup, \cap, \mid, \angle_\star, ^c, ^{-1}, \emptyset, E, I\rangle$. To complete the proof we show that ϕ is an isomorphism from $\mathfrak{A}$ to $\mathfrak{A}'$. In fact:

$$
\begin{aligned}
\phi(r \nabla s) &= \phi((r \circ \pi^{\mathsf{T}}) \sqcap (s \circ \rho^{\mathsf{T}})) && \text{(Ax (B.24))} \\
&= (\phi r \mid \phi\pi^{-1}) \cap (\phi s \mid \phi\rho^{-1}) && (\phi \text{ is isomorphism}) \\
&= \phi r \angle_{\star} \phi s && (B.25).
\end{aligned}
$$

Now, we expand $\mathfrak{B}'$ with the new operator $\angle_{\star}$ defined in the following way for any $X, Y \in A'$:

$$X \angle_{\star} Y = \{(a, b \star c) : a, b, c \in U, (a, b) \in X, \text{ and } (a, c) \in Y\}.$$

This completes the proof. ■

The Representation Theorem has the Completeness Theorem as a corollary.

Corollary B.3.4 (Completeness Theorem) *The class of proper fork algebras is axiomatized by the equalities* (B.19)– (B.24) *plus axioms for* RA.

B.4 The expressive power of PFRC

Definition B.4.1 Let us call PFRC the relational calculus $\{\pi, \rho, \nabla\}$RC, given by Definitions B.2.1 and B.2.2.

In this section, we shall characterize the expressive power of PFRC in terms of that of first-order logic. First, we shall define a first-order language FOL(R, $\star$) where the terms of PFRC will be translated. Besides the binary predicate symbols to translate the relational variables, the language also has a binary function symbol to translate the coding function. We shall use the same notation for the $\star$ symbol and for its realization, when no ambiguity seems to arise. Second, we define a forward translation from terms of PFRC to the formulas of FOL(R, $\star$), having exactly the two free variables x, y, that preserves the meaning of terms in structured models. Finally, we define a more evolved backward translation from formulas of FOL(R, $\star$), having exactly the two free variables x, y but having any number of bounded variables, to terms of PFRC. To be translated, each formula of FOL(R, $\star$) must be first rewritten in a normal form that restricts the kind of the atomic sub-formulas occurring in it. Moreover, to define the translation recursively we define first an auxiliary translation saying how formulas having any number of free variables should be translated. We deal with these details presenting the translation in a series of steps and proving some intermediate results showing that it has the expected behavior.

A version of these results will appear in [100].

According to Proposition B.2.1, PFRC is not minimal. As a matter of fact, we can make two choices in using simpler definitional equivalent systems: keep the projection operators as primitives and define ∇ from them or, alternatively, keep fork as primitive and define π and ρ from it. Working inside PFRC (having both operators as primitives) we have a more natural system, given the exact role each one of these operators play in the translation process.

B.4.1 The first-order language $\mathsf{FOL}(\mathrm{R},\star)$

First, we extend the syntax and semantics of $\mathsf{FOL}(\mathrm{R})$ to obtain $\mathsf{FOL}(\mathrm{R},\star)$.

Definition B.4.2 Starting with the syntax of $\mathsf{FOL}(\mathrm{R})$, we add the following to constitute the language of $\mathsf{FOL}(\mathrm{R},\star)$:
(1) the binary function symbol $\star$;
(2) the following generating grammar to *terms*:

$$t ::= x \mid t_1 \star t_2.$$

(3) to the generating grammar to formulas the following "rule":

$$\alpha ::= t_1 \approx t_2 \mid t_1 r t_2.$$

As in the RC case, the models of PFRC and that of $\mathsf{FOL}(\mathrm{R},\star)$ are exactly the same. All the other semantical concepts are defined as usual. We shall now establish some familiar concepts and notations from first-order languages when applied to $\mathsf{FOL}(\mathrm{R},\star)$. We shall make strong use of these in defining our back translation from $\mathsf{FOL}(\mathrm{R},\star)$ to PFRC.

Definition B.4.3 Let v be an individual variable, t be a term, and φ be a formula of $\mathsf{FOL}(\mathrm{R},\star)$. We denote by:
(1) $\mathsf{var}t$ the set of all variables occurring in t;

(2) $\mathsf{var}\varphi$ the set of all variables occurring in φ;

(3) $\mathsf{free}\varphi$ the set of variables with free occurrences in φ;

(4) $\varphi\frac{t}{v}$ the formula obtained by replacing each free occurrence of individual variable v by t in φ.

We shall consider the individual variables of $\mathsf{FOL}(\mathrm{R},\star)$ as being totally ordered in an arbitrary way. This ordering shall remain fixed throughout this section, with x and y being distinguished to denote the first two individual variables.

Definition B.4.4 Let $u, v, v_1, \ldots, v_m$ be individual variables and $\underline{v}$ be a finite non-empty set of individual variables of $\mathsf{FOL}(\mathrm{R},\star)$. We shall use:

(1) $u < v$ and $v_1 < v_2 < \ldots < v_m$ with their obvious meaning.

(2) $\mathsf{min}\underline{v}$ to denote the *first* variable in $\underline{v}$;

(3) $\mathsf{Max}\underline{v}$ to denote the *last* variable in $\underline{v}$;

(4) $\mathsf{nxt}\underline{v}$ to denote the first variable above all those variables in $\underline{v}$.

We also will need a special alphabetic variant of existential formulas, defined as follows.

Definition B.4.5 Let $\exists z\varphi$ be an existential formula of $\mathsf{FOL}(\mathrm{R},\star)$. The *ordered alphabetic variant* of $\exists z\varphi$ is the formula $\exists z'\varphi\frac{z'}{z}$, where $z' = \mathsf{nxt}(\mathsf{var}\varphi)$.

It is a well-known fact that every first-order formula is equivalent to one in term-reduced form having no "nested" occurrences of function symbols, i.e., with atomic sub-formulas of the form $v \approx w$, $f(v_1, \ldots, v_n) \approx w$, or $R(v_1, \ldots, v_m)$ [32]. In our case, with just binary predicate symbols and one binary operation, we have an analogous result, considering sets $\underline{x}$ and $\underline{y}$ of individual variables as distinguished. These will play the role of input and output sets of variables, respectively.

In what follows, let $\underline{x}$ and $\underline{y}$ be disjoint sets of individual variables.

Definition B.4.6 A formula of $\mathsf{FOL}(\mathrm{R},\star)$ is *$\underline{x}\underline{y}$-term-reduced* if each one of its atomic sub-formulas do not have repeated occurrences of a variable and no variable in $\underline{x}$ nor in $\underline{y}$ occur in formulas of the form $u \star v \approx w$.

Lemma B.4.1 *For every formula of* $\mathsf{FOL}(\mathrm{R},\star)$ *there is a logically equivalent* $\underline{x}\underline{y}$*-term-reduced formula of* $\mathsf{FOL}(\mathrm{R},\star)$ *with the same set of free variables.*

PROOF. By Theorem VIII.1.2 of [32], assume φ is in term-reduced normal form. So, we just need to replace each atomic sub-formula not $\underline{x}\underline{y}$-term-reduced by a formula having sub-formulas as required. To this end, apply repeatedly the following equivalences, where $w \neq u, v$, $w \notin \underline{x} \cup \underline{y}$, and $z \in \underline{x} \cup \underline{y}$.

$$\begin{array}{lcl}
u \approx u & \mathrel{=\!\!|\!\!\models} & \exists w(w \approx u) \\
u \star u \approx v & \mathrel{=\!\!|\!\!\models} & \exists w(u \star w \approx v \wedge w \approx u) \\
u \star v \approx u & \mathrel{=\!\!|\!\!\models} & \exists w(u \star v \approx w \wedge w \approx u) \\
v \star u \approx u & \mathrel{=\!\!|\!\!\models} & \exists w(v \star u \approx w \wedge w \approx u) \\
z \star u \approx v & \mathrel{=\!\!|\!\!\models} & \exists w(w \star u \approx v \wedge w \approx z) \\
u \star z \approx v & \mathrel{=\!\!|\!\!\models} & \exists w(u \star w \approx v \wedge w \approx z) \\
u \star v \approx z & \mathrel{=\!\!|\!\!\models} & \exists w(u \star v \approx w \wedge w \approx z) \\
uru & \mathrel{=\!\!|\!\!\models} & \exists w(urw \wedge w \approx u)
\end{array}$$

This completes the proof. ∎

B.4.2 Translating from PFRC to $\mathsf{FOL}(\mathrm{R},\star)$

Definition B.4.7 Let v, w be any two distinct individual variables of $\mathsf{FOL}(\mathrm{R},\star)$. The *forward translation function,* FT_{vw}, associated to v, w, from terms of PFRC into formulas of $\mathsf{FOL}(\mathrm{R},\star)$ is defined recursively by the following rules. We omit the rules corresponding to the operators of RC.

$$\begin{array}{rcl}
FT_{vw}\pi & ::= & \exists z(v \approx w \star z), \\
FT_{vw}\rho & ::= & \exists z(v \approx z \star w), \\
FT_{vw}(R \nabla S) & ::= & \exists zu(w \approx z \star u \wedge FT_{vz}R \wedge FT_{vu}S),
\end{array}$$

where u, v, w, z are pairwise distinct.

The next lemma states that FT_{vw} behaves syntactically as expected.

Lemma B.4.2 *Let variables v, w be given and let R be a term of* PFRC*. Then, the free variables of $FT_{vw}R$ are exactly v, w, i.e.,* $\mathsf{free}(FT_{vw}R) = \{v, w\}$.

The next lemma states that the semantical behavior of FT_{vw} is also as expected.

Lemma B.4.3 *Let R be a term of* PFRC *and v, w be any two distinct individual variables of* $\mathsf{FOL}(\mathrm{R}, \star)$*. Then, we have:*

$$\mathfrak{M}, \beta v, \beta w \Vdash R \text{ iff } \mathfrak{M}, \beta \models FT_{vw}R,$$

for any structured model $\mathfrak{M}$ and assignment β in $\mathfrak{M}$.

Definition B.4.8 Let R be a term of PFRC. The *forward translation of R* is the formula $FTR ::= FT_{xy}R$.

As a immediate consequence of Lemmas B.4.2 and B.4.3, we have:

Theorem B.4.1 *Let R be a term of* PFRC*. Then, we have:*
(a) $\mathsf{free}(FTR) = \{x, y\}$.
(b) $(a, b) \in [\![R]\!]_{\mathfrak{M}}$ *iff* $\mathfrak{M} \models FTR\,[a, b]$*, for any structured model $\mathfrak{M}$.*

B.4.3 Translating from $\mathsf{FOL}(\mathrm{R}, \star)$ to PFRC

One of the basic ideas underlying our translation procedure is to use the projections and fork operators to codify sequences of individual variables occurring in the formulas. As a first step, we shall need to define how to code nonempty finite sets $\underline{v} = \{v_1, \ldots, v_m\}$ of individual variables by terms, using the $\star$ operation applied to the variables in order.

Definition B.4.9 Let $\underline{v} = \{v_1, \ldots, v_m\}$ be a finite non-empty set of individual variables. The *variable-code* of $\underline{v}$ is the term $\mathsf{K}\underline{v}$ of $\mathsf{FOL}(\mathrm{R}, \star)$ defined by recursion on m as follows. Here, we denote $\mathsf{Max}\underline{v}$ by u.

$$\mathsf{K}\underline{v} ::= \begin{cases} v_1 & \text{if } m = 1, \\ \mathsf{K}(\underline{v} \setminus \{u\}) \star u & \text{if } m \neq 1. \end{cases}$$

Note that $v_1 < v_2 < \cdots < v_m$ iff $\mathsf{K}\underline{v} = (\cdots(v_1 \star v_2) \star \cdots) \star v_{m-1}) \star v_m$. We shall omit parenthesis, writing $v_1 \star \cdots \star v_m$ instead of $(\cdots(v_1 \star v_2) \star \cdots) \star v_m$ as well as $a_1 \star \cdots \star a_m$ instead of $(\cdots(a_1 \star a_2) \star \cdots) \star a_m$. Now, given $\underline{v} = \{v_1, \ldots, v_m\}$ with $v_1 < v_2 < \cdots < v_m$, the variable code $\mathsf{K}\underline{v}$ of $\underline{v}$ is a tree-like term from which we can extract each portion by means of a term of PFRC. In fact, we can use compositions and projections to extract single variables.

Definition B.4.10 Let $\underline{v} = \{v_1, \ldots, v_m\}$ with $v_1 < v_2 < \cdots < v_m$ and $v_i \in \underline{v}$. The *variable-extracting term*, associated to $\underline{v}$ and v_i, is the term $\underline{v} \hookrightarrow v_i$ of PFRC defined by the following clauses:

$$\underline{v} \hookrightarrow v_i ::= \begin{cases} \mathsf{I} & \text{if } 1 = i = m, \\ \rho & \text{if } 1 < i = m, \\ \pi \circ (\underline{v} \setminus \{v_m\} \hookrightarrow v_i) & \text{if } 1 < i < m. \end{cases}$$

The following result just renders Definition B.4.10 into a more intuitive way.

Proposition B.4.1 *Let $\underline{v} = \{v_1, \ldots, v_m\}$ with $v_1 < v_2 < \cdots < v_m$ and $v_i \in \underline{v}$. The following hold:*

(a) $\{v_1\} \hookrightarrow v_1 = \mathsf{I}$;

(b) *If $1 < m$, then $\underline{v} \hookrightarrow v_m = \rho$;*

(c) *If $1 < m$, then $\underline{v} \hookrightarrow v_1 = \underbrace{\pi \circ \cdots \circ \pi}_{m-1 \text{ times}} \circ \mathsf{I}$;*

(d) *If $1 < m$ and $1 < i < m$, then $\underline{v} \hookrightarrow v_i = \underbrace{\pi \circ \cdots \circ \pi}_{m-i \text{ times}} \circ \rho$.*

It also can be seen that this variable-extracting term $\underline{v} \hookrightarrow v_i$ has the desired meaning.

Proposition B.4.2 *Let $\underline{v} = \{v_1, \ldots, v_m\}$ with $v_1 < v_2 < \cdots < v_m$ and $v_i \in \underline{v}$. Then the following hold:*

(a) *$\mathfrak{M}, a_1 \star \cdots \star a_m, b \Vdash \underline{v} \hookrightarrow v_i$ iff $a_i = b$, for any model $\mathfrak{M}$ and $a_1, \ldots, a_m, b \in M$.*

(b) *$\mathfrak{M}, a, b \Vdash \underline{v} \hookrightarrow v_i$ iff there are $a_1, \ldots, a_m \in M$ such that $a = a_1 \star \ldots \star a_m$, and $a_i = b$, for any model $\mathfrak{M}$ and $a, b \in M$.*

By applying fork to variable-extracting terms, we can extract the code of subsets.

Definition B.4.11 Let $\underline{v} = \{v_1, \ldots, v_m\}$ with $v_1 < v_2 < \cdots < v_m$ and $\underline{u} = \{v_{i1}, v_{i2}, \ldots, v_{ik}\} \subseteq \underline{v}$ with $v_{i1} < v_{i2} < \ldots < v_{ik}$. The *subset-extracting term*, associated to $\underline{v}$ and $\underline{u}$, is the term $\underline{v} \hookrightarrow \underline{u}$ of PFRC defined by:

$$\underline{v} \hookrightarrow \underline{u} ::= (\underline{v} \hookrightarrow v_{i1}) \nabla (\underline{v} \hookrightarrow v_{i2}) \nabla \cdots \nabla (\underline{v} \hookrightarrow v_{ik}).$$

For instance, take $\underline{v'} = \{z, u, v, w\}$ with $z < u < v < w$ and $\underline{u'} = \{z, v, w\}$. From the above, it follows that:

$$\underline{v'} \hookrightarrow \underline{u'} = (\underline{v'} \hookrightarrow z) \nabla (\underline{v'} \hookrightarrow v) \nabla (\underline{v'} \hookrightarrow w) = (\pi \circ \pi \circ \pi \circ \mathsf{I}) \nabla (\pi \circ \rho) \nabla \rho.$$

It also can be seen that this subset-extracting term $\underline{v} \hookrightarrow \underline{u}$ has the desired meaning.

Proposition B.4.3 *Let* $\underline{v} = \{v_1, \ldots, v_m\}$ *with* $v_1 < v_2 < \cdots < v_m$ *and* $\underline{u} = \{v_{i1}, v_{i2}, \ldots, v_{ik}\} \subseteq \underline{v}$ *with* $v_{i1} < v_{i2} < \ldots < v_{ik}$. *Then the following hold:*
(a) $\mathfrak{M}, a_1 \star \cdots \star a_m, b \Vdash \underline{v} \hookrightarrow \underline{u}$ *iff* $a_{i1} \star a_{i2} \star \ldots \star a_{ik} = b$, *for any model* $\mathfrak{M}$ *and* $a_1, \ldots, a_m, b \in M$.
(b) $\mathfrak{M}, a, b \Vdash \underline{v} \hookrightarrow \underline{u}$ *iff there exist* $a_1, \ldots, a_m \in M$, *such that* $a = a_1 \star \cdots \star a_m$, *and* $a_{i1} \star a_{i2} \star \ldots \star a_{ik} = b$, *for any model* $\mathfrak{M}$ *and* $a, b \in M$.

Now, we define a general translation GT from $\underline{xy}$-term-reduced formulas of $\mathsf{FOL}(\mathrm{R}, \star)$ to terms of PFRC. GT is defined without any restriction on the number of occurrences of variables in the formulas being translated. Since GT is a little evolved, we proceed in a step by step manner.

First, for each $\underline{xy}$-term-reduced formula φ of $\mathsf{FOL}(\mathrm{R}, \star)$, we classify its free variables as input and output. Then we show how to translate φ to a term of PFRC that has the same meaning as φ, modulo variable coding. Our input-output conventions and the general translation are given next.

In what follows, we assume that $\underline{x}, \underline{y}$ are disjoint sets of individual variables and that the set of all individual variables are ordered in a manner that variables in $\underline{x}$ come first and are followed immediately by variables in $\underline{y}$. Besides, we adopt the conventions that:

– $x \in \underline{x}$ and $y \in \underline{y}$;

Formula φ	input set $\mathsf{i}\varphi$	output set $\mathsf{o}\varphi$	
$z \approx z'$	$\{z\}$	$\{z'\}$	
$y \approx z'$	$\{z'\}$	$\{y\}$	
$z \approx x$	$\{x\}$	$\{z\}$	
$u \star u' \approx u''$	$\{u, u'\}$	$\{u''\}$	
zrz'	$\{z\}$	$\{z'\}$	
yrz'	$\emptyset$	$\{y, z'\}$	
zrx	$\{x, z\}$	$\emptyset$	
yrx	$\{x\}$	$\{y\}$	
$\neg\varphi$	$\mathsf{i}\varphi$	$\mathsf{o}\varphi$	
$\varphi\frac{v'}{v}$	$(\mathsf{i}\varphi \setminus \{v\}) \cup \{v'\}$	$(\mathsf{o}\varphi \setminus \{v\}) \cup \{v'\}$	when $v' = \mathsf{nxt}(\mathsf{free}\varphi)$
$\varphi \wedge \psi$	$\mathsf{i}\varphi \cup \mathsf{i}\psi$	$\mathsf{o}\varphi \cup \mathsf{o}\psi$	
$\exists v\varphi$	$\mathsf{i}\varphi \setminus \{v\}$	$\mathsf{o}\varphi \setminus \{v\}$	when $v \in \mathsf{free}\varphi$

Table B.1: Classifying free variables as input or output.

- $z' \notin \underline{x}$ and $z \notin \underline{y}$ are two distinct variables;
- $u, u', u'' \notin \underline{x} \cup \underline{y}$ are pairwise distinct individual variables;
- v, v' are any individual variables.

Definition B.4.12 Let φ be an $\underline{xy}$-term-reduced formula of $\mathsf{FOL}(\mathrm{R}, \star)$. The *input set* $\mathsf{i}\varphi$ and the *output set* $\mathsf{o}\varphi$ of φ are defined according to Table B.1.

We classify the free variables of a formula considering that variables in $\underline{x}$ are always classified as input, variables in $\underline{y}$ are always classified as output, and the

$GT(v \approx v')$	$::=$	I
$GT(u \star u' \approx u'')$	$::=$	I
$GT(zrz')$	$::=$	r
$GT(yrz')$	$::=$	$\mathsf{E} \circ (\mathsf{I} \nabla r)$
$GT(zrx)$	$::=$	$(r \nabla \mathsf{I})^{\mathsf{T}} \circ \mathsf{E}$
$GT(yrx)$	$::=$	r^{T}
$GT\neg\varphi$	$::=$	$\mathsf{neg}\varphi$
$GT(\varphi \wedge \psi)$	$::=$	$\mathsf{conj}(\varphi, \psi) \sqcap \mathsf{conj}(\psi, \varphi)$
$GT\exists z\varphi$	$::=$	$\mathsf{exis}(z, \varphi)$

Table B.2: The general translation of formulas.

other free variables are classified according to where they occur in atomic formulas: variables at the left are classified as input, variables at the right as output.

Definition B.4.13 Let φ be an $\underline{xy}$-term-reduced formula of $\mathsf{FOL}(\mathrm{R}, \star)$. The *general translation of* φ is the term $GT\varphi$ of PFRC, defined recursively according to Table B.2.

Explanations follow.

Translating equalities presents no problem. In fact, an atomic formula $v \approx v'$ is satisfied when v and v' are assigned the same value. Considering v as input and v' as output or the other way around, this behavior can be described by I_U. So, we translate $v \approx v'$ into I. Moreover, an atomic formula $u \star u' \approx u''$, with u, u', u'' pairwise distinct, is satisfied by values a, b, c assigned to u, u', u'', respectively, exactly when $a \star b = c$. Consider variables u, u' as input and u'' as output, the input variables will be coded by $\mathsf{K}\{u, u'\} = u \star u''$. Since we wish to test whether $a \star b$ and c are equal, this input-output behavior can be described by I_U. Thus, we also translate $u \star u' \approx u''$ into I.

Translation of an atomic formula wrw' depends on the way its variables are classified as input or output. An atomic formula zrz' is satisfied by values a, b assigned, respectively, to z, z' exactly when $(a, b) \in [\![r]\!]_{\mathfrak{M}}$. Considering variable z as input and variable z' as output, its input-output behavior can be described by $[\![r]\!]_{\mathfrak{M}}$. So we translate zrz' to r. To translate yrz', note that variable y is classified always as output, no matter where it occurs at the formula, and $y < z'$, since $z' \neq x$. Hence, the output variables of formula yrz' will be coded by $\mathsf{K}\{y, z'\} = y \star z'$. Also, formula zry' has no variable classified as input. Hence, its input-output behavior should be a relation on structured elements, corresponding to $[\![r]\!]_{\mathfrak{M}}$ and not depending on inputs. So, we translate yrz' to $\mathsf{E} \circ (\mathsf{I} \nabla r)$. Analogously, the input-output behavior of zrx is a relation on structured elements corresponding to $[\![r]\!]_{\mathfrak{M}}$ and not depending on outputs. We translate zrx to $(r \nabla \mathsf{I})^{\mathsf{T}} \circ \mathsf{E}$. Finally, an atomic formula yrx is satisfied by values a, b assigned, respectively, to x, y exactly when $(b, a) \in [\![r]\!]_{\mathfrak{M}}$. Since x is classified as input and y as output, no matter where they occurs at the formula, its input-output behavior can be described by $[\![r]\!]_{\mathfrak{M}}{}^{-1}$. So we translate yrx to r^{T}.

We shall refer to this process of corresponding a relation to another that does not depend on its inputs or outputs by *cylindrification*.

The translation of non-atomic formulas is defined recursively. A formula $\neg\varphi$ has the same set of free variables as φ and the same input and output variables, say $\underline{v}$ and $\underline{w}$. When there is some variable classified as input and some classified as output, the input-output behavior of $\neg\varphi$ consists exactly of those pairs $(\underline{a}, \underline{b})$ outside that of φ, i.e., $[\![\neg\varphi]\!]_{\mathfrak{M}} = \overline{[\![\varphi]\!]_{\mathfrak{M}}}$. So we translate $\neg\varphi$ into $\overline{GT\varphi}$. When it is not the case, the translation proceeds by applying the process of cylindrification in an obvious way. Next definition explicits the way we obtain term $\mathsf{neg}\varphi$ from formula φ, according to the presence or absence of variables classified as input or output.

Definition B.4.14 Let φ be an $\underline{xy}$-term-reduced formula of $\mathsf{FOL}(\mathrm{R},\star)$. The term $\mathsf{neg}\varphi$ is defined as follows:

$$\mathsf{neg}\varphi ::= \begin{cases} \overline{GT\varphi} & \text{if } \mathsf{i}\varphi \neq \emptyset, \mathsf{o}\varphi \neq \emptyset, \\ \overline{GT\varphi} \circ \mathsf{E} & \text{if } \mathsf{i}\varphi \neq \emptyset, \mathsf{o}\varphi = \emptyset, \\ \mathsf{E} \circ \overline{GT\varphi} & \text{if } \mathsf{i}\varphi = \emptyset, \mathsf{o}\varphi \neq \emptyset, \\ \mathsf{E} \circ \overline{GT\varphi} \circ \mathsf{E} & \text{otherwise.} \end{cases}$$

The translation of conjunction requires that the formulas have the same free variables. To achieve this equality of sets of free variables, we will use the extracting terms defined above. Again, we apply cylindrification to deal with the absence of input or output variables.

Definition B.4.15 Let φ and ψ be $\underline{xy}$-term-reduced formulas of $\mathsf{FOL}(\mathrm{R},\star)$. The term $\mathsf{conj}(\varphi,\psi)$ is defined as follows:

$$\mathsf{conj}(\varphi,\psi) ::= \begin{cases} (\mathsf{i}\varphi \cup \mathsf{i}\psi \hookrightarrow \mathsf{i}\varphi) \circ GT\varphi \circ (\mathsf{o}\varphi \cup \mathsf{o}\psi \hookrightarrow \mathsf{o}\varphi) & \text{if } \mathsf{i}\varphi \neq \emptyset, \mathsf{o}\varphi \neq \emptyset, \\ \overline{\overline{(\mathsf{i}\varphi \cup \mathsf{i}\psi \hookrightarrow \mathsf{i}\varphi) \circ GT\varphi} \circ \mathsf{E}} & \text{if } \mathsf{i}\varphi \neq \emptyset, \mathsf{o}\varphi = \emptyset, \\ \overline{\mathsf{E} \circ \overline{GT\varphi \circ (\mathsf{o}\varphi \cup \mathsf{o}\psi \hookrightarrow \mathsf{o}\varphi)}} & \text{if } \mathsf{i}\varphi = \emptyset, \mathsf{o}\varphi \neq \emptyset, \\ \overline{\mathsf{E} \circ \overline{GT\varphi} \circ \mathsf{E}} & \text{otherwise.} \end{cases}$$

To translate an existential formula $\exists z\varphi$, when variable z occurs free in φ, we resort to the ordered alphabetic variant $\exists z\varphi\frac{z'}{z}$, that is logically equivalent to $\exists z\varphi$, using the new variable z' to store a value of z that satisfy the formula φ. Translation of a vacuous quantifier is immediate.

Definition B.4.16 Let $\underline{x}$ and $\underline{y}$ be disjoint sets of individual variables, φ be an $\underline{xy}$-term-reduced formula of $\mathsf{FOL}(\mathrm{R},\star)$, and z be an individual variable. The term $\mathsf{exis}(z,\varphi)$ is defined as follows. When $z \notin \mathsf{free}\varphi$, $\mathsf{exis}(z,\varphi)$ is simply $GT\varphi$. Otherwise, it is given by:

$$\mathsf{exis}(z,\varphi) ::= \begin{cases} (\mathsf{i}\varphi \cup \mathsf{i}\psi \hookrightarrow \mathsf{i}\varphi) \circ GT\varphi\frac{z'}{z} \circ (\mathsf{o}\varphi \cup \mathsf{o}\psi \hookrightarrow \mathsf{o}\varphi) & \text{if } \mathsf{i}\varphi \neq \emptyset, \mathsf{o}\varphi \neq \emptyset, \\ \overline{\overline{(\mathsf{i}\varphi \cup \mathsf{i}\psi \hookrightarrow \mathsf{i}\varphi) \circ GT\varphi\frac{z'}{z}} \circ \mathsf{E}} & \text{if } \mathsf{i}\varphi \neq \emptyset, \mathsf{o}\varphi = \emptyset, \\ \overline{\mathsf{E} \circ \overline{GT\varphi\frac{z'}{z} \circ (\mathsf{o}\varphi \cup \mathsf{o}\psi \hookrightarrow \mathsf{o}\varphi)}} & \text{if } \mathsf{i}\varphi = \emptyset, \mathsf{o}\varphi \neq \emptyset, \\ \overline{\mathsf{E} \circ \overline{GT\varphi\frac{z'}{z}} \circ \mathsf{E}} & \text{otherwise,} \end{cases}$$

where $z' = \mathsf{nxt}(\mathsf{var}\varphi \cup \{z\})$.

Lemma B.4.4 *Let $\underline{x}$ and $\underline{y}$ be disjoint sets of individual variables and φ be an $\underline{x}\underline{y}$-term-reduced formula of $\mathsf{FOL}(\mathrm{R},\star)$. For every model $\mathfrak{M}$ and every assignment β to the individual variables, we have:*

(a) *If $\mathsf{i}\varphi \neq \emptyset$ and $\mathsf{o}\varphi \neq \emptyset$, then:*

$$\mathfrak{M}, \beta \models \varphi \textit{ iff } \mathfrak{M}, \beta\mathsf{Ki}\varphi, \beta\mathsf{Ko}\varphi \Vdash GT\varphi.$$

(b) *If $\mathsf{i}\varphi \neq \emptyset$ and $\mathsf{o}\varphi = \emptyset$, then:*

$$\mathfrak{M}, \beta \models \varphi \textit{ iff, for every } b \in M : \mathfrak{M}, \beta\mathsf{Ki}\varphi, b \Vdash GT\varphi.$$

(c) *If $\mathsf{i}\varphi = \emptyset$ and $\mathsf{o}\varphi \neq \emptyset$, then:*

$$\mathfrak{M}, \beta \models \varphi \textit{ iff, for every } a \in M : \mathfrak{M}, a, \beta\mathsf{Ko}\varphi \Vdash GT\varphi.$$

(d) *If $\mathsf{free}\varphi = \emptyset$, then:*

$$\mathfrak{M} \models \varphi \textit{ iff } \mathfrak{M} \Vdash GT\varphi.$$

PROOF. By induction on formulas. ■

Definition B.4.17 Given a formula φ of $\mathsf{FOL}(\mathrm{R},\star)$ with m free variables classified as input $\mathsf{i}\varphi = \underline{x}$ and output $\mathsf{o}\varphi = \underline{y}$, a model $\mathfrak{M} = \langle M, r_i^{\mathfrak{M}}\rangle_{i\in\omega}$, we define:

(1) The *general input-output behavior* of φ in $\mathfrak{M}$ is the m-ary relation on M, defined by:

$$[\![\varphi]\!]^g_{\mathfrak{M}} ::= \{(\underline{a},\underline{b}) \in M^m : \mathfrak{M}, \beta \models \varphi(\underline{x},\underline{y}), \text{ for some } \beta \text{ with } \beta\underline{x} = \underline{a} \text{ and } \beta\underline{y} = \underline{b}\}.$$

(2) The *coded input-output behavior* of φ in $\mathfrak{M}$ is the binary relation on M, defined by:

$$[\![\varphi]\!]^{\mathsf{K}}_{\mathfrak{M}} ::= \{(\mathsf{K}\underline{a}, \mathsf{K}\underline{b}) \in M \times M : (\underline{a},\underline{b}) \in [\![\varphi]\!]^g_{\mathfrak{M}}\}.$$

Proposition B.4.4 *Given a formula φ of* $\mathsf{FOL}(\mathrm{R},\star)$ *with at least one input variable and one output variable, consider a model $\mathfrak{M}$ for* $\mathsf{FOL}(\mathrm{R},\star)$. *Then, the coded input-output behavior of φ in $\mathfrak{M}$ is the meaning of $GT\varphi$, i.e., $\mathfrak{M} \models \varphi\ [a,b]$ iff $(a,b) \in [\![GT\varphi]\!]^{\mathsf{K}}_{\mathfrak{M}}$, for all $a, b \in M$.*

We can now define the backward translation BT from formulas of $\mathsf{FOL}(\mathrm{R},\star)$ having exactly the two free variables $x < y$ to formulas of PFRC and show that it achieves its aim: both formulas have the same meaning.

Definition B.4.18 Let φ be a formula of $\mathsf{FOL}(\mathrm{R},\star)$ with exactly two distinct free variables x, y. The *backward translation* of φ is the term $BT\varphi ::= GT_{x,y}\varphi$ when $\mathsf{i}\varphi = \{x\}$ and $\mathsf{o}\varphi = \{y\}$.

Theorem B.4.2 *Let φ be a formula of* $\mathsf{FOL}(\mathrm{R},\star)$ *with exactly the two free variables $x < y$ classified as input and output, respectively. Then:*

$$\mathfrak{M}, \beta \models \varphi \ \textit{iff}\ \mathfrak{M}, \beta x, \beta y \Vdash BT\varphi,$$

for any structured model $\mathfrak{M}$ and assignment β in $\mathfrak{M}$.

Proof. Since $\mathsf{i}\varphi = \{x\}$ and $\mathsf{o}\{\varphi\} = y$, by Lemma B.4.4, we have:

$$\begin{aligned} \mathfrak{M}, \beta \models \varphi(x,y) \quad &\text{iff} \quad \mathfrak{M}, \beta\mathsf{K}x, \beta\mathsf{K}y \Vdash GT\varphi \\ &\text{iff} \quad \mathfrak{M}, \beta x, \beta y \Vdash BT\varphi. \end{aligned}$$

This finishes the proof. ■

Apêndice C

Positive relational calculus and graphs

In this chapter we will investigate the usage of (directed arc-labeled pseudo multi) graphs as a tool for proving inclusions and equalities of the Tarski's relational calculus. The application of the graph calculus will be restricted to the (axiomatizable but non-finitely axiomatizable) positive part of Tarski's system.

The basis for the graph relational calculus was introduced by S. Curtis and G. Lowe [23]. They exemplified its strong expressive power, claimed soundness of their inference rules and left completeness as a problem. In this chapter we will provide a more accurate formulation of the graph calculus. This will lead to the following improvements: a proper treatment of the union operator by the introduction of the notion of a component of a graph; a more elaborated definition of homomorphism enabling both precise formulation and use of the homomorphism rule in proofs; a proof that our set of rules is sound and weakly complete for the valid positive relational inclusions; a normal form for proofs resembling Gentzen's Hauptsatz in classical propositional logic; an analysis establishing the precise relationship among the positive relational calculus, the graph calculus and the positive existential fragment of the first-order language of binary relations.

Our main object of study is +RC, the positive part of RC —this is essentially RC without complementation. After defining +RC and reviewing some of its basic properties, we shall present +RG, the positive relational calculus based on graphs. +RG is our main tool to prove the valid inclusions and equalities of +RC. We will describe inference rules for transforming graphs. Since proofs are made inside the graph calculus, each term R of $+\mathsf{RC}$ will correspond to a graph G_R. Transformation rules will be called upon to change G_R into a graph SNFG_R, having a simple normal form. We will decide the validity of a inclusion $R \sqsubseteq S$ of $+\mathsf{RC}$ by testing the existence of (a certain kind of) homomorphism from SNFG_S to SNFG_R. This approach will lead us to the following result, where $\vdash R \sqsubseteq S$ means provability in the graph calculus and $\Vdash R \sqsubseteq S$ means validity in the positive relational calculus:

Theorem 5 *Let R, S be terms of $+\mathsf{RC}$. Then $\vdash R \sqsubseteq S$ iff $\Vdash R \sqsubseteq S$.*

The proof of Theorem 5 is based on a folklore result established at Section C.1. As corollaries, we obtain the finite model property and decidability for $+\mathsf{RC}$. Finally, in Section C.5 we characterize the expressive power of $+\mathsf{RC}$ and that of $+\mathsf{RG}$ in relation to the expressive power of the positive existential first-order language of binary relations. To this goal, we proceed as follows. There is a correspondence between the models of the relational formalisms $+\mathsf{RC}$ and $+\mathsf{RG}$ and those of the positive existential first-order language of binary relations. As a consequence, given any model $\mathfrak{M}$, the relational terms, the graphs, and the first-order formulas having occurrences of exactly two free individual variables, define binary relations on M, the domain of $\mathfrak{M}$.

Theorem 6 *Let $\mathfrak{M}$ be a model and X be binary relation on M. Then:*
(a) *X is definable by a term of $+\mathsf{RC}$ iff X is definable by a positive existential formula having occurrences of up to three variables being two free;*
(b) *X is definable by a graph of $+\mathsf{RG}$ iff X is definable by a positive existential formula having occurrences of up to two free variables.*

To prove item (a), we adapt the proof of the analogous result for RC, outlined in [2] and [104]. To prove item (b) we explore a normal form for graphs and first-order formulas.

This chapter is structured as follows. In Section C.1 we present a folklore theorem that will provide the core of our completeness result. In Section C.2 we will define $+\mathsf{RC}$, the positive part of RC, and will conclude that $+\mathsf{RC}$ is axiomatizable but not finitely axiomatizable by applying two capital results due to B. M. Schein [85] and H. Andréka [1]. In Sections C.3 and C.4, relying on the work in [23], we will present a relational calculus, +RG, based on graphs to prove the valid +RC inclusions and equalities. Finally, in Section C.5, we compare the expressive power of both the positive and the graph language in terms of the first-order language of binary relations.

C.1 A folklore theorem

In this section, we shall prove a simple result that will provide the core of our completeness result. We do not think that Theorem C.1.1 is original, but we were unable to find any references to it in the literature. Roughly speaking, it states that any deductive system that can be used both to derive normal forms and to prove logical consequences in normal form from formulas in normal form is simply complete. We say that a deductive system $\mathcal{D} = \langle L, \vdash \rangle$ is *simply complete* for a logical system $\mathcal{L} = \langle L, \models \rangle$ iff $\varphi \vdash \psi$ whenever $\varphi \models \psi$ for any formulas φ, ψ in L. We assume acquaintance with the basics on logical systems (for instance, we will make essential use of the assumption that $\vdash$ is transitive). To formulate our result rigorously, we also need some particular concepts.

In what follows, $\mathcal{L} = \langle L, \models \rangle$ is a logical system, NF is a subset of formulas of L, both φ, ψ are formulas of L, and $\mathcal{D} = \langle L \vdash \rangle$ is a deductive system for $\mathcal{L}$.

Definition C.1.1 (1) We say that NF *provides a normal form for* $\mathcal{L}$ if NF contains at least one element from each equivalence class of the quotient set of L by the logical equivalence relation $=\!\mid\!\models$.

(2) If $\varphi =\!\mid\!\models \psi$ and ψ is in NF, then ψ is called a *normal form of* φ.

(3) We say that φ *is in normal form* if it belongs to NF.

Any normal form of a formula φ is ambiguously denoted by NFφ.

Definition C.1.2 We say that $\mathcal{D}$ is an *NF-deductive system* when:

(1) $\varphi \vdash$ NFφ and NF$\varphi \vdash \varphi$, for every formula φ;

(2) $\varphi \vdash \psi$ whenever $\varphi \models \psi$, for any formulas φ, ψ in normal form.

We can now state the Folklore Theorem: every *NF*-deductive system is weakly complete.

Theorem C.1.1 (Folklore) *If $\mathcal{L} = \langle L, \models \rangle$ is a logical system, NF is a normal form for $\mathcal{L}$ and $\mathcal{D} = \langle L \vdash \rangle$ is an NF-deductive system for $\mathcal{L}$, then $\varphi \vdash \psi$ whenever $\varphi \models \psi$, for any formulas φ, ψ of L.*

Theorem C.1.1 provides a standard form for the proofs of the logically valid inferences $\varphi \models \psi$ of $\mathcal{L}$. In fact, given such an inference, one of its formal proofs has the form:

1.	φ,
2 to m.	$\varphi \vdash \mathrm{NF}\varphi$,
$m+1$.	$\mathrm{NF}\varphi$,
$m+2$.	$\mathrm{NF}\psi$,
$m+3$ to $m+n$.	$\mathrm{NF}\psi \vdash \psi$,
$m+n+1$.	ψ.

C.2 Syntax and semantics of +RC

We begin by describing the positive relational calculus +RC. It is both our main object of study and the basis underlying the construction of the graph relational language. +RC is essentially RC without complementation. Its language contains three kinds of expressions: terms, inclusions, and equalities, defined as in RC by excluding all references to the relational operators O and $^{-}$.

Definition C.2.1 (1) The *terms* of +RC, typically denoted by R, S, T, are generated from the set of relational variables $\text{RVAR} = \{r_i : i \in \omega\}$ by applying the relational operators E, I, $^{\mathsf{T}}$, $\sqcap$, $\sqcup$, and $\circ$.
(2) The *inclusions* and the *equalities* of +RC are the expressions of the forms $R \sqsubseteq S$ and $R \approx S$, respectively.

The class of models and the meaning $[\![R]\!]_{\mathfrak{M}}$ of a term R in a model $\mathfrak{M}$ are defined as in RC by, once more, excluding all references to O and $^{-}$. We omit the obvious definitions. Again, we write $\mathfrak{M}, a, b \Vdash R$ or just $a, b \Vdash R$ instead of $(a, b) \in [\![R]\!]_{\mathfrak{M}}$. So, as in RC, given a model $\mathfrak{M}$ any term of +RC defines a binary relation on M. We will characterize the relations that can be so defined.

Definition C.2.2 Given a model $\mathfrak{M}$ and $X \subseteq M \times M$, we say that X *is* +RC *definable in* $\mathfrak{M}$ if $X = [\![R]\!]_{\mathfrak{M}}$ for some +RC term R.

For example, for any model $\mathfrak{M}$, the relation $M \times M$ is definable by E.

C.2.1 Validity and definability in +RC

Valid inclusions and equalities are defined as usual and are related in the obvious way:

$$\Vdash R \sqsubseteq S \text{ iff } \Vdash R \sqcap S \approx R \text{ iff } \Vdash R \sqcup S \approx S, \tag{C.1}$$

$$\Vdash R \approx S \text{ iff } \Vdash R \sqsubseteq S \text{ and } \Vdash S \sqsubseteq R. \tag{C.2}$$

Equivalence (C.2) permits us to dispense with equalities and concentrate on the proof of inclusions (compare this with the similar result showing that equalities, inclusions, and terms are, in a certain sense, equivalent in RC). Typical examples of valid equations of +RC are:

Proposition C.2.1 *For any terms R, S, T of* +RC*, the following are valid:*

Laws of a distributive lattice with a top element.

(a) $(R \sqcap S) \sqcap T \approx R \sqcap (S \sqcap T)$ *and* $(R \sqcup S) \sqcup T \approx R \sqcup (S \sqcup T)$.

(b) $R \sqcap S \approx S \sqcap R$ *and* $R \sqcup S \approx S \sqcup R$.

(c) $R \sqcap R \approx R$ *and* $R \sqcup R \approx R$.

(d) $R \sqcap (S \sqcup T) \approx (R \sqcap S) \sqcup (R \sqcap T)$ *and* $R \sqcup (S \sqcap T) \approx (R \sqcup S) \sqcap (R \sqcup T)$.

(e) $R \sqcap \mathsf{E} \approx R$ *and* $R \sqcup \mathsf{E} \approx \mathsf{E}$.

Monoid laws.

(f) $(R \circ S) \circ T \approx R \circ (S \circ T)$.

(g) $R \circ \mathsf{I} \approx R$ *and* $\mathsf{I} \circ R \approx R$.

Anti-involution laws.

(h) $(R \sqcap S)^{\mathsf{T}} \approx S^{\mathsf{T}} \sqcap R^{\mathsf{T}}$ *and* $(R \sqcup S)^{\mathsf{T}} \approx S^{\mathsf{T}} \sqcup R^{\mathsf{T}}$.

(i) $(R \circ S)^{\mathsf{T}} \approx S^{\mathsf{T}} \circ R^{\mathsf{T}}$.

(j) $R^{\mathsf{T}^{\mathsf{T}}} \approx R$.

Laws relating intersection, union, and composition.

(k) $R \circ (S \sqcap T) \sqsubseteq (R \circ S) \sqcap (R \circ T)$. (*Semi-distributivity*)

(l) $R \circ (S \sqcup T) \approx (R \circ S) \sqcup (R \circ T)$. (*Distributivity*)

(m) $(R \circ S) \sqcap T \sqsubseteq (R \sqcap (T \circ S^{\mathsf{T}})) \circ S$. (*Modularity*)

Some notable missing laws can be easily proved from the basic laws:

Proposition C.2.2 *For any terms R, S, T of* $+\mathsf{RC}$*, the following hold:*

(a) $\Vdash \mathsf{E} \sqcap R \approx R$ *and* $\Vdash \mathsf{E} \sqcup R \approx R$.

(b) $\Vdash R \sqsubseteq S$ *iff* $\Vdash R^{\mathsf{T}} \sqsubseteq S^{\mathsf{T}}$.

(c) $\Vdash (R \sqcap S) \circ T \sqsubseteq (R \circ T) \sqcap (S \circ T)$.

(d) *Law* (c) *is equivalent to the semi-distributivity law, which is also equivalent to the Horn sentences in* (e).

(e) *If* $\Vdash R \sqsubseteq S$, *then* $\Vdash R \circ T \sqsubseteq S \circ T$ *and* $\Vdash T \circ R \sqsubseteq T \circ S$.

(f) $(R \sqcup S) \circ T \approx (R \circ T) \sqcup (S \circ T)$.

(g) *Law* (f) *is equivalent to the distributivity law.*

(h) $(R \circ S) \sqcap T \sqsubseteq (R \circ (S \sqcap (R^{\mathsf{T}} \circ T))$.

(i) *Law* (h) *is equivalent to the modularity law, which is also equivalent to* $(R \circ S) \sqcap T \sqsubseteq (R \sqcap (T \circ S^{\mathsf{T}})) \circ (S \sqcap (R^{\mathsf{T}} \circ T))$.

Proposition C.2.1 exhibits a nice set of valid inclusions and identities for derivation in $+\mathsf{RC}$. It cannot be complete since the non-finite axiomatizability of $+\mathsf{RC}$ is a consequence of a general result of H. Andréka [1].

Proposition C.2.3 $+\mathsf{RC}$ *is not finitely axiomatizable.*

This does not exclude infinite axiomatizations and, in fact, their existence follows from a general result of B. M. Schein [85].

Proposition C.2.4 +RC *is axiomatizable.*

To the best of our knowledge, no explicit infinite set of axioms to +RC has been exhibited. Paralleling the results of R. Lyndon [64, 65] and B. Jónsson [59], we believe that no such a set should be simple. So, our work in Section C.4 is justified. The quest for axiomatizability of +RC and some of its subreducts is one of the problems stated in [86].

C.3 Syntax and semantics of +RG

In this section, we present a relational language +RG, based on graphs. +RG is strongly inspired in the language defined in [23]. The main differences are that we have a proper treatment of the union operator by the introduction of the notion of a component of a graph; a more elaborated definition of homomorphism enabling both precise formulation and use of the homomorphism rule in proofs; a proof that our set of rules is sound and weakly complete for the valid positive relational inclusions; a normal form for proofs resembling Gentzen's Hauptsatz in classical propositional logic.

+RG is designed to *represent* relations using a very special kind of graph. Its language contains two kinds of expressions: *components* and *graphs*. Components are (directed arc-labeled pseudo multi) graphs having distinguished nodes and arcs labeled by terms of +RC. The label of an arc represents a restriction associated to the relation defined by the label. A path from a node to another represents a restriction associated to the composition of the corresponding relations. Two paths with the same start and ending points represent a restriction associated to the intersection of the corresponding relations. A graph is a set of components. Each graph represents a restriction associated to the union of the relations corresponding to its components.

We assume the existence of a set $\text{INOD} = \{x_n : n \in \omega\}$ of *individual nodes*, typically denoted by x, y, z, u, v, w.

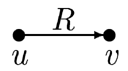
Definition C.3.1 A *component* is a structure $C = (N, A, x, y)$, where:

1. N is a non-empty set of nodes.

2. $A \subseteq N \times \mathfrak{T}^+ \times N$ is a set of labeled *arcs*. Here $\mathfrak{T}^+$ is the set of all +RC terms.

3. x, y are, not necessarily distinct, distinguished nodes from N.

Given a term R of +RC and nodes u, v, we denote the arc (u, R, v) by uRv or, pictorially, by:



Note that components need not be connected in the sense of graph theory, e.g., the graph formed just by two non-adjacent distinguished nodes x and y is not connected but can be a component in the sense of +RG:

Definition C.3.2 A *positive relational graph*, or simply a *graph*, is a finitely indexed non-empty set $G = (N_j, A_j, x_j, y_j)_{j \in J}$ of components.

We identify a component and a graph having only this component.

Given a base set U, considered as *universe*, a graph defines a binary relation on U, according to some conditions on its components. The basic pieces of semantics are the models as in +RC but to use the expressive power given by nodes, we need a notion of valuation.

Definition C.3.3 Given a graph G, a component C_j of G and a model $\mathfrak{M}$, an *assignment* for C_j in $\mathfrak{M}$ is a function $g_j : N_j \to M$.

Formally, each graph G denotes a binary relation $[\![G]\!]_{\mathfrak{M}}$ on M, as specified below.

Definition C.3.4 Let G be a graph and $\mathfrak{M}$ be a model. The *meaning of G in $\mathfrak{M}$* is the subset $[\![G]\!]_{\mathfrak{M}}$ of $M \times M$ defined in the following way:

$(a, b) \in [\![G]\!]_{\mathfrak{M}}$
iff
there exist a component $C = (N, A, x, y)$ of G and an assignment g for C in $\mathfrak{M}$ such that $gx = a$, $gy = b$, and $\mathfrak{M}, gu, gv \Vdash R$ whenever $uRv \in A$.

Note that, when x and y are the same node, we have $a = gx = gy = b$.

Hence, given a graph G, a model $\mathfrak{M}$ and points $a, b \in M$, to prove that (a, b) belongs to $[\![G]\!]_{\mathfrak{M}}$ one should exhibit some component $C = (N, A, x, y)$ of G and an assignment g for C in $\mathfrak{M}$ satisfying the condition in Definition C.3.4. When this is the case, we say that the pair C, g *witnesses* that $(a, b) \in [\![G]\!]_{\mathfrak{M}}$.

C.3.1 Validity, equivalence, and definability in +RG

Our objective is not to explore +RG as a formal system by itself. Here, it is used just as an auxiliary system for reasoning about +RC inclusions (and equalities). We do not define general notions of inclusion and equality for graphs, but we use related notions as far as they are helpful in studying the corresponding positive relational notions. For instance, the following definitions will be essential.

Definition C.3.5 Let G, H be graphs of +RG. We say that:
(1) *G implies H*, denoted $G \models H$, when $[\![G]\!]_{\mathfrak{M}} \subseteq [\![H]\!]_{\mathfrak{M}}$, for every model $\mathfrak{M}$;
(2) *G and H are equivalent*, denoted $G \mathrel{=\!\!|\!\!\models} H$, when $[\![G]\!]_{\mathfrak{M}} = [\![H]\!]_{\mathfrak{M}}$, for every model $\mathfrak{M}$.

Moreover, given a model $\mathfrak{M}$, every graph defines a binary relation on M. We shall characterize the relations that can be so defined.

Definition C.3.6 Given a model $\mathfrak{M}$ and $X \subseteq M \times M$, we say that *X is +RG definable in $\mathfrak{M}$* if $X = [\![G]\!]_{\mathfrak{M}}$ for some graph G.

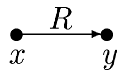
For example, for any model $\mathfrak{M}$, the universal relation $M \times M$ is definable by the graph:

Definability in +RG subsumes definability in +RC. To prove this it suffices to associate to each term R a graph G_R that defines the same relation as the term.

Definition C.3.7 Let R be a term of $+\mathsf{RC}$. The *graph associated to R* is defined as:

$$G_R ::= (\{x, y\}, \{xRy\}, x, y),$$

or simply:



Proposition C.3.1 *For every term R of $+\mathsf{RC}$, we have $[\![R]\!]_{\mathfrak{M}} = [\![G_R]\!]_{\mathfrak{M}}$, for any model $\mathfrak{M}$.*

Corollary C.3.1 *Let $\mathfrak{M}$ be a model and $X \subseteq M \times M$. If X is $+\mathsf{RC}$ definable, then X is $+\mathsf{RG}$ definable.*

The exact relationship between $+\mathsf{RC}$ and $+\mathsf{RG}$ definabilities is investigated in Section C.5.

Corollary C.3.2 *Let R, S be terms of $+\mathsf{RC}$. Then the following are equivalent:*
(a) $\Vdash R \sqsubseteq S$.
(b) $[\![R]\!]_{\mathfrak{M}} \subseteq [\![S]\!]_{\mathfrak{M}}$, *for all model $\mathfrak{M}$.*
(c) $[\![G_R]\!]_{\mathfrak{M}} \subseteq [\![G_S]\!]_{\mathfrak{M}}$, *for all model $\mathfrak{M}$.*
(d) $G_R \models G_S$.

C.4 Weak derivation system for $+\mathsf{RG}$

In this section, we shall present a graph relational calculus, i.e., a set of rules for deriving a relational graph from another. The graph calculus will be used for, given an inclusion $R \sqsubseteq S$ of $+\mathsf{RC}$, to decide whether $\Vdash R \sqsubseteq S$ or not. We shall prove that the rules are sound and weakly complete for $+\mathsf{RC}$.

Formal proofs will not be made directly in the relational algebraic formalism but rather in the graph relational one. The main idea is to derive an inclusion $R \sqsubseteq S$ by using the corresponding graphs G_R and G_S. To this end, we have a set of *graph transformation rules* whose application to G_R is expected to produce a finite sequence of graphs ending in G_S. Some rules do not change the corresponding relation, when applied to a graph, and one rule, when applied to a graph, alters the corresponding relation, transforming it into a larger one.

The main idea behind the choice of the rules is to use the Folklore Theorem from Section C.1. So, we shall consider a normal form, introduced in Definition C.4.3 for the graph language, and use it to prove that $\Vdash R \sqsubseteq S$ by executing the following two major steps. First, reduce the graphs G_R and G_S to their simple normal forms SNFG_R and SNFG_S, respectively. Second, verify whether or not the graph SNFG_R is a homomorphic image of the graph SNFG_S, according to Definition C.4.1. To obtain the weakly completeness result we just need to show that our rules can execute the two major steps described.

C.4.1 Definition of the graph relational calculus

The transformation rules are of three kinds, as presented on the following.

Rules of the first kind are rules of the form:

$$\frac{C_1}{C_2},$$

where C_1, C_2 denote components such that C_1 satisfies a computable condition and C_2 is the image of C_1 by some computable operation.

Given graphs G, H, we say that *H is obtained from G by applying a certain rule of the first kind* if H is obtained from G by replacing non-deterministically one component C_1 of G by a new component C_2 and leaving untouched all the other components of G. Of course, C_1 should satisfy the condition given in the rule and C_2 is obtained from C_1 by applying the respective operation.

Rules of the second kind are rules of the form:

$$\frac{C_1}{C_2 + C_3},$$

where C_1, C_2, C_3 denote components such that C_1 satisfies a computable condition and C_2, C_3 are images of C_1 by computable operations.

Given graphs G, H, we say that *H is obtained from G by applying a certain rule of the second kind* if H is obtained from G by replacing non-deterministically one component C_1 of G by two new components C_2, C_3 and leaving untouched all the other components of G. Of course, C_1 should satisfy the condition given in the rule and C_2, C_3 are obtained from C_1 by applying the respective operations.

The system will have just one *rule of the third kind*. Explanations will be given afterwards.

To state the transformation rules, we adopt the following conventions:

- If N is a set of nodes and $x_1, \ldots, x_n \in$ INOD, we denote by $N + x_1 + \ldots + x_n$ the set $N \cup \{x_1, \ldots, x_n\}$ and by $N - x_1 - \ldots - x_n$ the set $N \setminus \{x_1, \ldots, x_n\}$.
- Analogously, if A is a set of arcs and $x_1 R_1 y_1, \ldots, x_n R_n y_n$ are arbitrary arcs, we denote by $A + x_1 R_1 y_1 + \ldots + x_n R_n y_n$ the set $A \cup \{x_1 R_1 y_1, \ldots, x_n R_n y_n\}$ and by $A - x_1 R_1 y_1 - \ldots - x_n R_n y_n$ the set $A \setminus \{x_1 R y_1, \ldots, x_n R y_n\}$.
- Finally, if G is a graph, we denote by $C_1 + \ldots + C_n$ the set $\{C_1, \ldots, C_n\}$ of distinguished components of G.
- The first six rules below can be applied in both directions. In the downward direction, a rule eliminates its correspondent operator, in the upward direction the rule introduces it.

The *transformation rules* are given in Table C.1. Explanations follows.

Rule Univ states that we do not change the meaning of a graph by erasing an arc labeled by E from a component of a graph where it occurs, leaving all the rest of the graph untouched.

Rule Iden makes use of the function $\texttt{ren}_u^v$ as described by the following definitions.

Let u, v be arbitrary nodes. The *function renaming u to v* is defined by:

$$\texttt{ren}_u^v w = \begin{cases} v & \text{if } w = u, \\ w & \text{otherwise.} \end{cases}$$

Let N and A be arbitrary set of nodes and arcs, respectively. Define $\texttt{ren}_u^v N = \{\texttt{ren}_u^v w : w \in N\}$ and $\texttt{ren}_u^v A = \{\texttt{ren}_u^v w R \texttt{ren}_u^v w' : w R w' \in A\}$. Observe that:

$$\texttt{ren}_u^v N = \begin{cases} N & \text{if } u = v, \\ N - u + v & \text{otherwise.} \end{cases}$$

Hence, $u \notin \texttt{ren}_u^v N$, when $u \in N$ and $u \neq v$.

Univ	$\dfrac{N, A + u\mathsf{E}v, x, y}{N, A, x, y}$
Iden	$\dfrac{N, A + u\mathsf{I}v, x, y}{\mathtt{ren}_u^v N, \mathtt{ren}_u^v A, \mathtt{ren}_u^v x, \mathtt{ren}_u^v y}$
Conv	$\dfrac{N, A + uR^{\mathsf{T}}v, x, y}{N, A + vRu, x, y}$
Int	$\dfrac{N, A + uR \sqcap Sv, x, y}{N, A + uRv + uSv, x, y}$
Uni	$\dfrac{N, A + uR \sqcup Sv, x, y}{(N, A + uRv, x, y) + (N, A + uSv, x, y)}$
Comp	$\dfrac{N, A + uR \circ Sv, x, y}{N + w, A + uRw + wSv, x, y}$, if $w \notin N$
Hom	$\dfrac{G}{H}$, if $\Phi : H \to G$

Table C.1: Rules for transforming graphs.

Rule Iden states that we do not change the meaning of a graph by erasing an arc $u\mathsf{I}v$ and renaming the component where it occurs, replacing all the nodes called u by nodes called v. Observe that possibly the distinguished nodes may change after the end of this procedure.

Rule Conv states that we do not change the meaning of a graph by replacing an arc $uR^{\mathsf{T}}v$ by another vRu, inside a component where it occurs, leaving all the rest of the graph untouched.

Rule Int states that we do not change the meaning of a graph by replacing an arc $uR \sqcap Sv$ by two other uRv, uSv, inside a component where it occurs, leaving all the rest of the graph untouched.

Rule Uni states that we do not change the meaning of a graph by replacing a component C_1 having occurrence of an arc $uR \sqcup Sv$, by two other components C_2 and C_3, each one of them obtained from C_1 by replacing the arc $uR \sqcup Sv$ by a new arc: uRv for C_2 and uSv for C_3, leaving all the rest of the graph untouched.

Finally, rule Comp states that we do not change the meaning of a graph by replacing an arc $uR \circ Sv$ by two other uRw, wSv, for a new node w, inside a component where it occurs, leaving all the rest of the graph untouched.

Reading the above explanations in the other way around, we see that all of the first six rules can be applied in both directions.

Our capital rule, Hom, states that if there is a homomorphism Φ from H to G, then from G we can infer H. The notion of homomorphism used here is a little more evolved than the usual one for direct arc labeled graphs.

Definition C.4.1 (1) Let $C = (N, A, x, y)$ and $C' = (N', A', x', y')$ be components. A *homomorphism* from C' to C is a function $\phi : N' \to N$ preserving distinguished nodes and labeled arcs, in the following sense: $\phi x' = x$, $\phi y' = y$, and $\phi u R \phi v \in A$ for all $uRv \in A'$. We write $\phi : C' \to C$ when ϕ is a homomorphism from C' to C.
(2) Let G, H be graphs. We say that *there is a homomorphism from H to G* when, for each component C of G, there is a component C' of H and a homomorphism $\phi : C' \to C$. We write $\Phi : H \to G$ when there is a homomorphism from H to G.

The notion of proof is standard.

Definition C.4.2 (1) Let G, H be graphs of $+\mathsf{RG}$. We say that H is *derivable* from G, denoted, $G \vdash H$, if there is a sequence $G_1, \ldots, G_n$ of graphs such that:

1. $G_1 = G$;
2. $G_n = H$;
3. For each $i, 1 < i \leq n$, the graph G_i is obtained from the graph G_{i-1} by application of one of the transformation rules Univ, Iden, Conv, Int, Uni, Comp, and Hom.

(2) Let $R \sqsubseteq S$ be a inclusion of $+\mathsf{RC}$. We say that $R \sqsubseteq S$ is a *theorem* of $+\mathsf{RG}$ (or simply, a *theorem*), denoted by $\vdash R \sqsubseteq S$, if $G_R \vdash G_S$.

In [64, 65], R. Lyndon showed that inclusion:

$$T \sqcap (((U \circ V) \sqcap W) \circ (X \sqcap (Y \circ Z)))$$
$$\sqsubseteq$$
$$U \circ ((((U^\mathsf{T} \circ T) \sqcap (V \circ X)) \circ Z^\mathsf{T}) \sqcap (V \circ Y) \sqcap (U^\mathsf{T} \circ ((T \circ Z^\mathsf{T}) \sqcap (W \circ Y)))) \circ Z$$

althought valid is not derived in RC, from the Tarski's axioms [92]. Within the graph calculus, this inclusion can be proved. Take R, S to be the terms in the left and right sides of inclusion above. Applying rules of $+\mathsf{RG}$ on G_R and G_S we eliminate all occurrences of operators in the labels of the arcs, obtaining the graphs in Figure C.1 and C.2, respectively. The derivation is completed by *one* application of Hom, allowed by the homomorphism in the next table:

n	x	d	e	f	g	h	i	j	l	m	y
ϕn	x	b	c	y	a	x	x	a	y	a	y

C.4.2 Soundness and completeness of the graph relational calculus

Now, we prove soundness and completeness of the graph relational calculus. First, we claim that when applying the rules given in Table C.1 to a graph G, we obtain a graph that is implied by G. For the rules of the first or second kinds, this is established in Lemma C.4.1. This result is just a formal rendering of the quite intuitive property that the meaning of a graph is preserved in both directions by the application of a rule of the first or second kind.

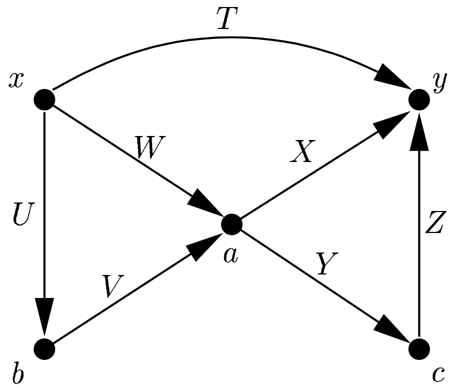


Figure C.1: G_R

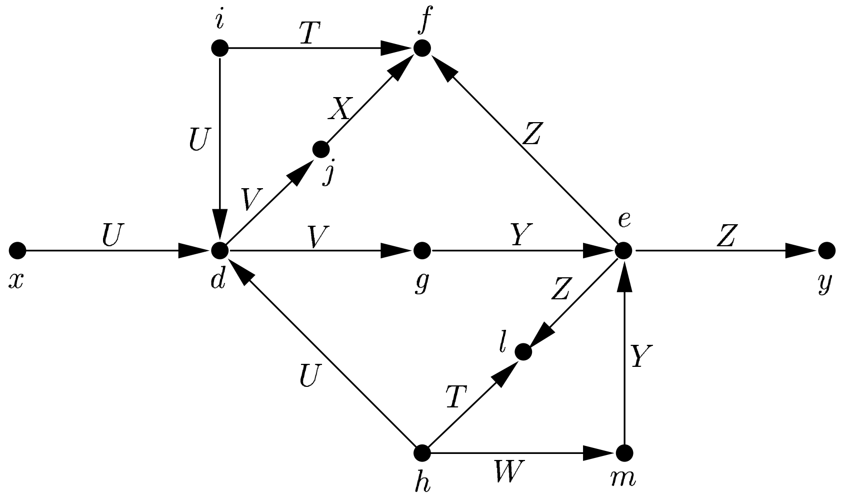


Figure C.2: G_S

Lemma C.4.1 *Let H be obtained from G by applying one of the rules* Univ*,* Iden*,* Conv*,* Int*,* Uni*, and* Comp*. Then G and H are equivalent.*

Now, we prove the analogous result for the not so intuitive rule Hom.

Lemma C.4.2 *Let $\Phi : H \to G$. Then, $[\![G]\!]_{\mathfrak{M}} \subseteq [\![H]\!]_{\mathfrak{M}}$, for any model $\mathfrak{M}$.*

PROOF. Suppose $(a, b) \in [\![G]\!]_{\mathfrak{M}}$ is witnessed by C, g, where $C = (N, A, x, y)$. Take a component $C' = (N', A', x', y')$ in H and a homomorphism $\phi : C' \to C$, guaranteed to exist since $\Phi : H \to G$. Define the assignment $g' : N' \to M$ by $g'u = g\phi u$, for any $u \in N'$. We have $g'x' = g\phi x' = gx = a$ and $g'y' = g\phi y' = gy = b$. Moreover, if $uRv \in A'$, we have $\phi uR\phi v \in A$. Hence, $\mathfrak{M}, g\phi u, g\phi v \Vdash R$ and $\mathfrak{M}, g'u, g'v \Vdash R$. So, $(a, b) \in [\![H]\!]_{\mathfrak{M}}$ is witnessed by C', g'. ∎

Example C.4.1 shows that Hom cannot be applied upwards.

Example C.4.1 Consider graphs $G = (\{x\}, xrx, x, x)$ and $H = (\{x\}, \emptyset, x, x)$, and the function identity ϕ on $\{x\}$. So, $\phi : H \to G$ is a homomorphism and, hence, $\Phi : G \to H$. But given $\mathfrak{M} = \langle M, r^{\mathfrak{M}} \rangle$ defined by $M = \{a, b\}$ and $r^{\mathfrak{M}} = \{(a, a)\}$, and $g : \{x\} \to M$ defined by $gx = a$, we have $[\![G]\!]_{\mathfrak{M}} = (a, a)$ and $[\![H]\!]_{\mathfrak{M}} = M \times M$.

From Lemmas C.4.1 and C.4.2, we have:

Corollary C.4.1 *Let G, H be graphs of* +RG*. If $G \vdash H$, then $G \models H$.*

Soundness is an immediate consequence.

Theorem C.4.1 (Soundness) *Let R, S be terms of* +RC*. Then $\vdash R \sqsubseteq S$ implies $\Vdash R \sqsubseteq S$.*

To prove completeness, we implement the following strategy, based on the Folklore Theorem. First, we prove that every graph can be proved equivalent to one having a *simple normal form*, by successive applications of transformation rules of first or second types. Second, we show that the inclusion of graphs in simple normal form can be decided by testing the existence or not of a homomorphism from one graph to another. The combination of these steps will give us the completeness result.

Definition C.4.3 Let G be a graph of $+\mathsf{RG}$.
(1) We say that G is in *simple normal form*, SNF, if all arcs of G are labeled by relational variables.
(2) A *simple normal form of* G is a graph H of $+\mathsf{RG}$ such that:

1. H is in SNF;

2. G and H are equivalent.

In this case we write, ambiguously, $H = \text{SNF}G$.

Theorem C.4.2 (SNF) *Every graph of* $+\mathsf{RG}$ *has a simple normal form.*

PROOF. By induction on the total number of occurrences of relational operators appearing in the labels of the arcs of G. ∎

Lemma C.4.3 *Let* G, H *be graphs in* SNF *Then* $G \models H$ *implies* $G \vdash H$.

PROOF. Let G, H be simple graphs. Suppose $G \models H$. Hence, $[\![G]\!]_{\mathfrak{M}} \subseteq [\![H]\!]_{\mathfrak{M}}$, for any model $\mathfrak{M}$. We prove that $G \vdash H$ by proving the stronger result that H can be obtained from G by exactly one application of the rule Hom.

Assume, for a contradiction, that it is not the case that $\Phi : H \to G$. By definition, there is a component $C = (N, A, x, y)$ of G forbidding any tentative of defining an homomorphism from H to G. That is, for C we have:

(*) for any component $C' = (N', A', x', y')$ of H and any function $\phi : N' \to N$ satisfying $\phi x' = x$ and $\phi y' = y$, there exists $arb \in A'$ such that $\phi a r \phi b \notin A$.

Take such a C and define $\mathfrak{M}_C = (N, r_i^{\mathfrak{M}_C})_{i \in I}$ by taking $r_i^{\mathfrak{M}_C} = \{(a, b) \in N \times N : ar_ib \in A\}$. Obviously, $\mathfrak{M}_C$ is a model. Also, we have $(x, y) \in [\![G]\!]_{\mathfrak{M}_C}$. In fact, taking the assignment g defined by $ga = a$ for any $a \in N$, we have $gx = x$ and $gy = y$. Moreover, if $ar_ib \in A$ then $(a, b) \in r_i^{\mathfrak{M}_C}$. Hence, $\mathfrak{M}_C, ga, gb \Vdash r_i$, since $ga = a$ and $gb = b$.

Now, since $[\![G]\!]_{\mathfrak{M}} \subseteq [\![H]\!]_{\mathfrak{M}}$ for all models $\mathfrak{M}$, we have $(x, y) \in [\![H]\!]_{\mathfrak{M}_C}$. Hence, by definition, there is a component $C' = (N', A', x', y')$ of H and an assignment $\phi : N' \to N$ such that $\phi x' = x$, $\phi y' = y$, and $\mathfrak{M}_C, \phi a, \phi b \Vdash r_i$, for any $ar_ib \in A'$.

By (*), we take an arc $arb \in A'$ such that $\phi a r \phi b \notin A$. Since $arb \in A'$, we have $\mathfrak{M}_C, \phi a, \phi b \Vdash r$. Hence, $(\phi a, \phi b) \in r^{\mathfrak{M}_C}$ giving us $\phi a r \phi b \in A$, a contradiction.

Hence, $\Phi : H \to G$. ■

From the Folklore Theorem, SNF, and Lemma C.4.3, we have:

Corollary C.4.2 *Let G, H be graphs of* +RG*. Then, $G \models H$ implies $G \vdash H$.*

Theorem C.4.3 (Completeness) *Let R, S be terms of* +RC*. Then $\Vdash R \sqsubseteq S$ implies $\vdash R \sqsubseteq S$.*

PROOF. Suppose $[\![R]\!]_{\mathfrak{M}} \subseteq [\![S]\!]_{\mathfrak{M}}$ for any model $\mathfrak{M}$. Then, by Proposition C.3.1, we also have $[\![G_R]\!]_{\mathfrak{M}} \subseteq [\![G_S]\!]_{\mathfrak{M}}$, for any model $\mathfrak{M}$. So, by Corollary C.4.2, $G_R \vdash G_S$. But this, by Definition C.4.2, is the same as $\vdash R \sqsubseteq S$. ■

From the remarks at the end of Section C.1, we know that Theorem C.4.3 provides a normal form for proofs. From the remarks about the transformation rules, we know that, except for the rule Hom, each one of the rules can be applied in both directions. One, bottom-up, shows the introduction rule aspect of the rule. The other, top-down, emphasizes its elimination rule aspect. By combining both observations, we arrive at Table C.2 that displays a more accurate normal form for proofs of the valid inclusions $R \sqsubseteq S$ in +RG. This can be considered as an analogue of Gentzen's Hauptsatz in classical propositional logic.

One application of the above normal form is the reduction of the proof of an inclusion to the test of whether or not a graph is a homomorphic image of another graph. Of course, Homomorphism is a decidable problem and as a corollary we also obtain the decidability of the Validity Problem for inclusions and equalities of +RC.

C.5 Expressive power of +RC and +RG

In this section we compare the expressive powers of +RC and +RG with that of first-order logic. In order to compare with +RC, as well as with +RG, the following version of first-order language appears as the most adequate.

1.	G_R	
2 to m.	steps deriving SNFG_R from G_R	applications of Univ, Iden, Conv, Int, Uni, Comp eliminating operators
	SNFG_R	one application of Hom
$m+1$.	SNFG_S	
$m+2$ to $m+n$.	steps deriving G_S from SNFG_S	applications of Univ, Iden, Conv, Int, Uni, Comp introducing operators
	G_S	

Table C.2: Gentzen's like Hauptsatz for +RG.

Definition C.5.1 Let IVAR $= \{x_i : i \in \omega\}$ be a set of individual variables, typically denoted by x, y, z, and RVAR $= \{r_i : i \in \omega\}$ be a set of relational symbols, typically denoted by r, s, t. The *formulas* of $+\exists$FOL(R), typically denoted by φ, ψ, are defined according to the following grammar:

$$\varphi \quad ::= \quad xry \mid x \approx y \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x \varphi.$$

We freely use all the syntactic notions, properties and conventions of FOL(R) when restricted to $+\exists$FOL(R). We denote by $\underline{x}, \underline{y}, \underline{z}$ arbitrary finite sequences of individual variables. In particular, any of $\underline{x}, \underline{y}, \underline{z}$ can be empty. We use set theoretical notation when referring to $\underline{x}, \underline{y}, \underline{z}$, in an usual way.

The semantics for $+\exists$FOL(R) is just the first-order semantics restrict to the positive language. Hence, the models for +RC, +RG and $+\exists$FOL(R) are the same. This makes easy to compare the expressive powers of these formalisms.

In this section, we present both a characterization of the expressive power of +RC in terms of a three-variable fragment of $+\exists$FOL(R) as well as a characterization of the expressive power of +RG in terms of the whole $+\exists$FOL(R). The former result is a consequence of the analogous result for RC. The latter follows from the fact that the disjunctive normal form of formulas of $+\exists\mathsf{FOL}(\mathrm{R})$ are very close to graphs of $+\mathsf{RG}$ in simple normal form.

C.5.1 Translating between $+\mathsf{RC}$ and $+\exists\mathsf{FOL}(\mathrm{R})$

Let $+\exists\mathsf{FOL}(\mathrm{R})^{xy}_{z}$ be the set of formulas of $+\exists\mathsf{FOL}(\mathrm{R})$ having up to the three individual variables x, y, z, being at most x, y free.

Theorem C.5.1 *Terms of $+\mathsf{RC}$ can be translated to formulas of $+\exists\mathsf{FOL}(\mathrm{R})^{xy}_{z}$ in which exactly x, y occur free.*

Theorem C.5.2 *Formulas of $+\exists\mathsf{FOL}(\mathrm{R})^{xy}_{z}$ can be translated as terms of $+\mathsf{RC}$.*

To prove these results, just note the modularity present in both the forward and backward translations from RC to $\mathsf{FOL}(\mathrm{R})^{xyz}$.

C.5.2 Translating between $+\mathsf{RG}$ and $+\exists\mathsf{FOL}(\mathrm{R})$

Now, we show that the graph language and the positive existential first-order fragment define the same relations inside any model $\mathfrak{M}$.

Let $+\exists\mathsf{FOL}(\mathrm{R})^{xy}$ be the $+\exists\mathsf{FOL}(\mathrm{R})$ restricted to formulas having exactly x, y free. We assume neither x nor y are nodes of the graph language.

Theorem C.5.3 *Let G be a graph of $+\mathsf{RG}$. Then there exists a formula φ of $+\exists\mathsf{FOL}(\mathrm{R})^{xy}$ such that:*

$$(a, b) \in [\![G]\!]_{\mathfrak{M}} \text{ iff } \mathfrak{M}, \beta \models \varphi,$$

for every model $\mathfrak{M}$, points $a, b \in M$, and assignments β for the individual variables satisfying $\beta x = a$ and $\beta y = b$.

Proof. Let G be a graph of $+\mathsf{RG}$. Consider the graph $\mathrm{SNF}G = (N_j, A_j, x_j, y_j)_{j \in J}$, equivalent to G, and define the formula φ in the following way. Take:

$$\varphi_j ::= \begin{cases} \exists N_j - x_j (x \approx y \wedge \bigwedge_{urv \in A_j} urv \frac{x}{x_j}) & \text{if } x_j = y_j, \\ \\ \exists N_j - x_j - y_j (\bigwedge_{urv \in A_j} urv \frac{xy}{x_j y_j}) & \text{otherwise,} \end{cases}$$

for each $j \in J$, and put $\varphi = \bigvee_{j \in J} \varphi_j$. Here, both $N_j - x_j$ and $N_j - x_j - y_j$ are viewed as ordered sequences of individual variables. Let $\mathfrak{M}$ be a model, $a, b \in M$, and β be an assignment to the individual variables such that $\beta x = a$ and $\beta y = b$. We shall prove $(a, b) \in [\![G]\!]_{\mathfrak{M}}$ iff $\mathfrak{M}, \beta \models \varphi$. An exhaustive case analysis shows that φ is a formula of $+\exists\mathsf{FOL}(\mathrm{R})^{xy}$ satisfying the required conditions. ∎

Theorem C.5.4 *Let φ be a formula of $+\exists\mathsf{FOL}(\mathrm{R})^{xy}$. Then there exists a graph $G = (N_j, A_j, x, y)_{j \in J}$ of $+\mathsf{RG}$, and:*

$$\mathfrak{M}, \beta \models \varphi \textit{ iff } (a, b) \in [\![G]\!]_{\mathfrak{M}},$$

for every model $\mathfrak{M}$, points $a, b \in M$, and assignment to the individual variables β satisfying $\beta x = a$ and $\beta y = b$.

Proof. Let φ be a formula of $+\exists\mathsf{FOL}(\mathrm{R})^{xy}$. We may assume that φ is:

$$\bigvee_{i=1}^{m} \exists \underline{x_i} \varphi_i,$$

where each $\exists \underline{x_i} \varphi_i$, $1 \leq i \leq m$, satisfies the following conditions:

- φ_i is a conjunction of pairwise distinct atomic formulas;
- all variables in $\underline{x_i}$ occur in φ_i;
- If $u \approx v$ is a sub-formula of φ_i, then $u, v \in \{x, y\}$;
- $x, y \notin \mathsf{bound}\varphi$.

Now, for each i, $1 \leq i \leq m$, define $N_i = \mathrm{IVAR}\varphi_i$,

$$A_i = \{u \mathsf{l} v : u \approx v \text{ occurs in } \varphi_i\} \cup \{urv : urv \text{ occurs in } \varphi_i\}, \text{ and}$$

$$C_i ::= \begin{cases} (N_i + x + y, A_i, x, y) & \text{if } \mathsf{free}\exists \underline{x_i} \varphi = \emptyset, \\ (N_i + y, A_i, x, y) & \text{if } \mathsf{free}\exists \underline{x_i} \varphi = \{x\}, \\ (N_i + x, A_i, x, y) & \text{if } \mathsf{free}\exists \underline{x_i} \varphi = \{y\}, \\ (N_i, A_i, x, y) & \text{if } \mathsf{free}\exists \underline{x_i} \varphi = \{x, y\}. \end{cases}$$

Take $G = (C_i)_{i \in \{1,\dots,m\}}$. Let $\mathfrak{M}$ be a model, $a, b \in M$, and β be an assignment to the individual variables such that $\beta x = a$ and $\beta y = b$. An exhaustive case analysis shows that φ is a formula of $+\exists\mathsf{FOL}(\mathrm{R})^{xy}$ satisfying the required conditions. ■

Apêndice D

Two-sorted linear temporal logic

In this chapter we shall investigate LTL2, the two-sorted linear temporal logic. This is a relational calculus obtained by combining elements from linear temporal logic [42], two-dimensional temporal logic [106], propositional dynamic logic [61], and (positive) relational calculus.

In studying LTL2 we focus on expressive power. Our main objective is to compare the expressive power of LTL2 with those of linear temporal logic, LTL, and flat two-dimensional temporal logic, TAL$\flat$. This study follows the line of investigation of the following works on the expressive power of fragments and extensions of XPath [71, 72, 6, 73].

LTL was introduced by A. Pnueli [82] as a temporal logic for computer science applications and it has been widely applicable. For this reason, since the seminal work of D. Gabbay et. al. [43] —where a sound and complete axiomatic system was presented and the logic was shown to be decidable, and *expressively complete*— all of its logic theoretical aspects have been investigated in a outstanding series of works covering, mainly, axiomatics [16, 107, 105], expressive power [60, 55, 62], decidability [43, 17], decision procedures [63], and complexity [77, 90, 83, 84].

TAL$\flat$ was introduced by Y. Venema in [106]. There, a way to derive the valid formulas of TAL$\flat$ was described, based on the axiomatization of a weaker system, using an irreflexivity rule. Concerning expressive power, it was proved that TAL$\flat$ is expressively complete with respect to first-order logic with monadic predicates over the class of linear orders. Results above were combined to eliminate the use of the irreflexivity rule to obtain a weakly sound and complete axiomatic system for well orderings in general and, in particular, for the order of the natural numbers.

TAL$\flat$ can be described as a non-logical extension of the relational calculus as follows. Syntactically, TAL$\flat$ is just RC with a new constant relational symbol $<$. The bigger differences are in semantics. Now, a *model* is a structure $\mathfrak{M} = \langle M, <^{\mathfrak{M}}, r_i^{\mathfrak{M}} \rangle$, where $M \neq \emptyset$ and $<^{\mathfrak{M}}$ is a binary relation on M, and each $r_i^{\mathfrak{M}}$ is a subset of M. A *flow of time* is a model in which $<^{\mathfrak{M}}$ is irreflexive and transitive. A flow of time is *linear* if $<^{\mathfrak{M}}$ is linear. Given a model $\mathfrak{M} = \langle M, <^{\mathfrak{M}}, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$, each term R of TAL$\flat$ denotes a binary relation $[\![R]\!]_{\mathfrak{M}}$ on M. The intended meaning of the usual atomic relation symbols and operators is as in RC. The meaning of relational variables is restricted to relations of the form $X \times M$, where X is a subset of M:

$$(a, b) \in [\![r]\!]_{\mathfrak{M}} \text{ iff } a \in r^{\mathfrak{M}}, \text{ for all } a, b \in M.$$

As we saw in Section A.1, this is an indirect way to refer to the set X in a relational framework. The interpretation of the constant relation $<$ is given by specifying that:

$$(a, b) \in [\![<]\!]_{\mathfrak{M}} \text{ iff } a <^{\mathfrak{M}} b,$$

for all $a, b \in M$.

The expressive power of TAL$\flat$ can be measured against an adequate first-order logic with one *binary* predicate $<$ and *monadic* predicates P_i corresponding to relational variables r_i, $i \in \omega$. The formulas of the *first-order language of order*, $\mathsf{FOL}(<, \mathrm{P})$, are generated by the following grammar:

$$\alpha ::= Px \mid x < y \mid x \approx y \mid \neg\alpha \mid \alpha \vee \alpha \mid \exists x\alpha,$$

where, x, y are individual variables and P is a monadic predicate. Models of TAL$\flat$ and $\mathsf{FOL}(<, \mathrm{P})$ correspond to each other in a direct way. Given a model $\mathfrak{M} = \langle M, <^{\mathfrak{M}}, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ for TAL$\flat$, associate the model $\mathfrak{M}' = \langle M, I \rangle$, where $I(<) = <^{\mathfrak{M}}$ and $I(P_i) = \{a \in M : \exists b \in M \text{ such that } (a, b) \in r_i^{\mathfrak{M}}\}$, and vice-versa. In [106], Venema proved that TAL$\flat$ and $\mathsf{FOL}(<, \mathrm{P})$ are equipollent in means of expression over the class of all linear flows of time. More precisely, there are computable functions, one forward from terms of TAL$\flat$ into formulas of $\mathsf{FOL}(<, \mathrm{P})$, and another backward, from formulas of $\mathsf{FOL}(<, \mathrm{P})$ into terms of TAL$\flat$, that preserve the meaning of terms and formulas in linear flows of time.

This functional completeness result [42] is a direct consequence of the two fundamental results that, over the class of linear flows of time, TAL♭ is equipollent in means of expression with the three-variable fragment of $\mathsf{FOL}(<, \mathrm{P})$ [106] and every formula of $\mathsf{FOL}(<, \mathrm{P})$ is equivalent to a formula with at most 3 bound variables [58].

The results in this chapter also say something about the expressive power of TAL♭ on models, in general, when restricted to linear, and when restricted to discrete flows of time.

We leave a detailed description of LTL2 for the next section. For immediate purposes let us just say that the system has two sorts of terms: one whose intended meaning are subsets of the universe of discourse and another whose intended meaning are binary relations on this universe. Both sorts are, mainly, interpreted on flows of time, as TAL♭ or $\mathsf{FOL}(<, \mathrm{P})$ are. But the differences between TAL♭ and LTL2 are significant. Whereas the former has all the Booleans and Peirceans in its language, in the relational type, the latter is restricted just to positive relational operators. The strongness of its expressive power comes from the fact that its atomic relations are in fact constructed from the set terms by using some carefully chosen operators. In particular, we shall start investigating the question of how far is the expressive power of LTL2 from that of TAL♭ on different classes of models. The investigation pursued here culminates in the following results.

Theorem 7 *Let $\mathfrak{M}$ be a model, X be a subset of M, and Y be a binary relation on M. Then, the following hold:*

(a) *X is definable by a set term of* LTL2 *iff X is definable by a formula of* LTL*;*

(b) *If $\mathfrak{M}$ is linearly ordered, then Y is definable by a relational term of* LTL2 *iff Y is definable by a term of* TAL♭ *having no occurrences of negation.*

(c) *If $\mathfrak{M}$ is discretely linear ordered, then Y is definable by a relation term of* LTL2 *iff Y is definable by a term of* TAL♭.

To prove item (a), we present translations in both directions. The first, from LTL to LTL2 just replaces inside formulas of LTL some occurrences of temporal operators by compositions of the analogue ones, present in LTL2. The second, a little more evolved, explores a normal form for the set terms of LTL2. When put in this normal form, a set term is clearly just a formula of LTL rewritten in an adequate way.

To prove item (b), we also present translations in both directions. The first, from LTL2 to TAL$\flat$ consists in defining all the LTL2 operators in the relational calculus language. It is very simple and may just be realized from the fact that all operators of LTL2 can be defined with three variables. The hard part of the proof is the implication from right to left. It also explores a normal form for relation terms of LTL2, but in this case resembling the separation property of [42].

Showing item (c) is the most difficult part. Implication from left to right is already given in the proof of item (b). The converse implication is much more evolved and, by far, more complex then all the previous translations we have presented. It makes use of the normal form previously introduced and of the relationship between LTL2 and a positive fragment of FOL$(<, \mathrm{P})$ presented in the form of a relational graph calculus.

The chapter is structured as follows. In Section D.1 we define LTL2, the two-sorted linear temporal logic. In Section D.2 we show that LTL2 and LTL are equally expressive on sets. In section D.3, we show that LTL2 and TAL$\flat$ are equally expressive over discrete flows of time. The proof proceeds in series of steps. First we show how to define the relational operators $\mathsf{O}, \mathsf{I}, \mathsf{E}$ and $^{\mathsf{T}}$. The three first are defined directly by terms and to define $^{\mathsf{T}}$ we use a weaker form of definability. Second, we show how to define $\sqcap$. To this end, we introduce terms in a *directed normal form*, resembling the notion of separation form [42]. We prove that, over flows of time, every term of LTL2 can be rewritten in direct normal form. Using this normal form, we show how to define $\sqcap$ over the class of flows of time. Finally, we treat complementation. We use two tools. The normal form introduced previously and the positive existential first-order language over relation terms of LTL2. This auxiliary language is presented and used in the form of a graph calculus.

LTL2 also can be used as a tool for studying other formalisms. For instance, from the work in the next section, we can obtain the nice result that in any class of models for which LTL2 and FOL$(<, \mathrm{P})$ are equally expressive in relations, we also have that LTL2 and FOL$(<, \mathrm{P})$ are equally expressive on sets. This can provide an alternative proof of Kamp's Theorem [60] if we establish that LTL2 and FOL$(<, \mathrm{P})$ are equally expressive on relations over the class of Dedekind complete linear orders.

D.1 Syntax and semantics of LTL2

In this section, we present the syntax and semantics of LTL2. The language is two-sorted, with a sort for points and another for ordered pairs of points. As we will see in Section D.2, the sort for points is just a notational variant for LTL. The sort for ordered pairs contains tests —well known from PDL— binaries since and until —generalized from LTL— and it is closed under union and composition —borrowed from the positive relational calculus.

Definition D.1.1 Let $\textsc{Pvar} = \{p_i : i \in \omega\}$ be a set of propositional variables, typically denoted by p, q, r. The *set terms*, typically denoted by A, B, C, and the *relation terms*, typically denoted by R, S, T, are defined by double recursion according to the following grammar:

$$A \ ::= \ \top \mid p \mid \neg A \mid A \wedge A \mid A \vee A \mid \mathsf{dom}(R),$$

$$R \ ::= \ ?A \mid \mathsf{until}(A, A) \mid \mathsf{since}(A, A) \mid R \sqcup R \mid R \circ R.$$

Both set and relation terms, will be interpreted on ordered sets. Given a fixed, non-empty, ordered set U, propositional variables will range on subsets of U. The intended interpretation of the above connectives and operators is as follows. Suppose that set terms A, B denote the subsets V, W of U and that relation terms R, S denote the binary relations X, Y on U. Then, the set terms $\neg A$, $A \wedge B$, $A \vee B$, and $\mathsf{dom}(R)$ denote, respectively, the complement of V, the intersection of V and W, the union of V and W, and the domain of the relation X. Besides, the relation terms $?A$, $\mathsf{until}(A, B)$, $\mathsf{since}(A, B)$, $R \sqcup S$, and $R \circ S$ denote, respectively, the relations "test-for-V", "W has been uninterruptedly true until V", "since V has W been uninterruptedly true", the union of X and Y, and the composition of X and Y. Formally, it is useful to introduce a notion of meaning of a term in a model as follows.

Definition D.1.2 (1) A *model* for LTL2 is a structure $\mathfrak{M} = (M, <^{\mathfrak{M}}, p_i^{\mathfrak{M}})_{i \in \omega}$, where M is a non-empty set, $<^{\mathfrak{M}}$ is a binary relation on M, and $p_i^{\mathfrak{M}} \subseteq M$, for every $i \in \omega$. (2) Given a model $\mathfrak{M}$, each set term denotes a subset of M and each relation term a subset of $M \times M$ as specified in Table D.1.

$[\![p_i]\!]_{\mathfrak{M}}$	$::=$	$p_i^{\mathfrak{M}}$
$[\![\top]\!]_{\mathfrak{M}}$	$::=$	M
$[\![\neg A]\!]_{\mathfrak{M}}$	$::=$	$[\![A]\!]_{\mathfrak{M}}^c$
$[\![A \wedge B]\!]_{\mathfrak{M}}$	$::=$	$[\![A]\!]_{\mathfrak{M}} \cap [\![B]\!]_{\mathfrak{M}}$
$[\![A \vee B]\!]_{\mathfrak{M}}$	$::=$	$[\![A]\!]_{\mathfrak{M}} \cup [\![B]\!]_{\mathfrak{M}}$
$[\![\mathsf{dom}(R)]\!]_{\mathfrak{M}}$	$::=$	$\{a \in M : \exists b \in M((a,b) \in [\![R]\!]_{\mathfrak{M}})\}$
$[\![?A]\!]_{\mathfrak{M}}$	$::=$	$\{(a,a) \in M \times M : a \in [\![A]\!]_{\mathfrak{M}}\}$
$[\![\mathsf{until}(A,B)]\!]_{\mathfrak{M}}$	$::=$	$\{(a,b) \in M \times M : a < b,\ b \in [\![A]\!]_{\mathfrak{M}},$ and $\forall c(\text{if } a < c < b, \text{ then } c \in [\![B]\!]_{\mathfrak{M}})\}$
$[\![\mathsf{since}(A,B)]\!]_{\mathfrak{M}}$	$::=$	$\{(a,b) \in M \times M : b < a,\ b \in [\![A]\!]_{\mathfrak{M}},$ and $\forall c(\text{if } b < c < a, \text{ then } c \in [\![B]\!]_{\mathfrak{M}})\}$
$[\![R \sqcup S]\!]_{\mathfrak{M}}$	$::=$	$[\![R]\!]_{\mathfrak{M}} \sqcup [\![S]\!]_{\mathfrak{M}}$
$[\![R \circ S]\!]_{\mathfrak{M}}$	$::=$	$[\![R]\!]_{\mathfrak{M}} \mid [\![S]\!]_{\mathfrak{M}}$.

Table D.1: Meaning of terms of LTL2.

We are mainly interested in viewing LTL2 as a temporal logic. So, its *standard* models are those in which $<^{\mathfrak{M}}$ is a linear ordering of M.

Definition D.1.3 Let $\mathfrak{M} = (T, <, p_i^{\mathfrak{M}})_{i\in\omega}$ be a model.
(1) We say that $\mathfrak{M}$ is a *flow of time* when $<^{\mathfrak{M}}$ is irreflexive and transitive.
(2) A flow of time is *linear* when $<^{\mathfrak{M}}$ is linear.
(3) It is *discrete* when it is linear and $<^{\mathfrak{M}}$ is discrete.

To maintain the system closer to temporal logic and emphasize its two-sorted aspects, we will depart from using the "belongs to" notation and will adopt the following conventions:

1. We shall write $\mathfrak{M}, a \Vdash A$, or simply $a \Vdash A$, in lieu of $a \in [\![A]\!]_{\mathfrak{M}}$. Similarly, $\mathfrak{M}, a, b \Vdash R$, or $a, b \Vdash R$, in lieu of $(a, b) \in [\![R]\!]_{\mathfrak{M}}$.

2. We shall write $(a, b) \Vdash A$ in lieu of $\forall c \in M$: if $a < c < b$, then $c \Vdash A$.

Under this convention, given a model $\mathfrak{M}$ and points $a, b \in M$, we use (a, b) to denote the ordered pair with first coordinate a and second coordinate b as well as the open interval containing all points $c \in M$ satisfying $a <^{\mathfrak{M}} c <^{\mathfrak{M}} b$. We left to the context indications of which of these senses is being adopted. Using these conventions, for instance, the definition of meaning can be restated in a modal logic-like notation, as displayed in Table D.2.

Equalities, inclusions, and validities are defined as usual.

Definition D.1.4 Let t_1, t_2 be either both set or both relation terms and K be a class of models.
(1) An *equality* is an expression of the form $t_1 \approx t_2$.
(2) An equality $t_1 \approx t_2$ is *valid* over K, or is an *identity* over K, when $[\![t_1]\!]_{\mathfrak{M}} = [\![t_2]\!]_{\mathfrak{M}}$, for every model $\mathfrak{M} \in \mathsf{K}$.
(3) An equality $t_1 \approx t_2$ is *valid*, or is an *identity*, when it is valid over the class of all models.

Sometimes we shall make use of inclusions, treating them as abbreviations, as usual.

$a \Vdash \top$	iff	always,
$a \Vdash p$	iff	$a \in p^{\mathfrak{M}}$,
$a \Vdash \neg A$	iff	$a \nVdash A$,
$a \Vdash A \wedge B$	iff	$a \Vdash A$ and $a \Vdash B$,
$a \Vdash A \vee B$	iff	$a \Vdash A$ or $a \Vdash B$,
$a \Vdash \mathsf{dom}(R)$	iff	$\exists b \in M(a, b \Vdash R)$,
$a, b \Vdash ?A$	iff	$a = b$ and $a \in A$,
$a, b \Vdash \mathsf{until}(A, B)$	iff	$a < b, b \Vdash A$ and $(a, b) \Vdash B$,
$a, b \Vdash \mathsf{since}(A, B)$	iff	$b < a, b \Vdash A$ and $(b, a) \Vdash B$,
$a, b \Vdash R \sqcup S$	iff	$a, b \Vdash R$ or $a, b \Vdash S$,
$a, b \Vdash R \circ S$	iff	$\exists c \in M(a, c \Vdash R$ and $c, b \Vdash S)$.

Table D.2: Meaning of terms at models in a modal logic like notation.

D.2 The expressive power of LTL2 on sets

LTL2 has two levels of expressivity: on sets and on relations. In this section, we characterize the expressive power of LTL2 on sets. We show that, at this level, it has the same expressive power as LTL, the linear temporal logic [42], on any class of models. The proof consists in exhibiting two translation functions T_1 and T_2, both preserving meaning on models. The former is from LTL formulas to LTL2 set terms and the latter the other way around.

We recall just the syntax and semantics of LTL. Many aspects of this important formalism are exposed in [42].

Definition D.2.1 Let $\textsc{Pvar} = \{p_i : i \in \omega\}$ be a set of propositional variables, typically denoted by p, q, r.
(1) The *formulas* of LTL, typically denoted by φ, ψ, are defined according to the following grammar:

$$\varphi \ ::= \ p \mid \top \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathsf{Kuntil}(\varphi, \varphi) \mid \mathsf{Ksince}(\varphi, \varphi).$$

(2) Given a model $\mathfrak{M}$ and a point $a \in M$, the relation "φ is satisfied at a in $\mathfrak{M}$", denoted $\mathfrak{M}, a \models \varphi$, is defined by as specified in Table D.3.

Observe the close resemblance between the temporal operators of LTL —that are evaluated on points— and those of LTL2 —that are evaluated on pairs of points. We shall use analogous conventions to both systems. In particular, we shall write $(a, b) \Vdash \varphi$ in lieu of $\forall c \in M(\text{if } a < c < b, \text{ then } c \Vdash \varphi)$.

D.2.1 Translating from LTL to LTL2

Translation T_1 just replaces inside LTL formulas each occurrence of Kuntil or Ksince by an application of dom succeeded by since or until, respectively.

Proposition D.2.1 *For every formula φ of* LTL*, there exists a set term $T_1\varphi$ of* LTL2 *such that:*

$$\mathfrak{M}, a \models \varphi \textit{ iff } \mathfrak{M}, a \Vdash T_1\varphi,$$

for any model $\mathfrak{M}$ and point $a \in M$.

$\mathfrak{M}, a \models p$	iff	$a \in p^{\mathfrak{M}}$
$\mathfrak{M}, a \models \top$	iff	always
$\mathfrak{M}, a \models \neg\varphi$	iff	$\mathfrak{M}, a \not\models \varphi$
$\mathfrak{M}, a \models \varphi \wedge \psi$	iff	$\mathfrak{M}, a \models \varphi$ and $\mathfrak{M}, a \models \psi$
$\mathfrak{M}, a \models \varphi \vee \psi$	iff	$\mathfrak{M}, a \models \varphi$ or $\mathfrak{M}, a \models \psi$
$\mathfrak{M}, a \models \mathsf{Kuntil}(\varphi, \psi)$	iff	$\exists b \in M(a < b, \mathfrak{M}, a \models \varphi$ and $\forall c \in M(\text{if } a < c < b, \text{then } \mathfrak{M}, c \models \psi)$
$\mathfrak{M}, a \models \mathsf{Ksince}(\varphi, \psi)$	iff	$\exists b \in T(b < a, \mathfrak{M}, b \models \varphi$ and $\forall c \in T(\text{if } b < c < a, \text{then } \mathfrak{M}, c \models \psi)$

Table D.3: Satisfaction of formulas of LTL2.

PROOF. Define a translation from formulas of LTL to set terms of LTL2 by stipulating that T_1 is the identity on $\top$ and p, commutes with the Booleans, and:

$$T_1\mathsf{Kuntil}(\varphi, \psi) \quad ::= \quad \mathsf{dom}(\mathsf{until}(T_1\varphi, T_1\psi)),$$

$$T_1\mathsf{Ksince}(\varphi, \psi) \quad ::= \quad \mathsf{dom}(\mathsf{since}(T_1\varphi, T_1\psi)),$$

for any formulas φ, ψ of LTL.

Given a model $\mathfrak{M}$ and a point $a \in T$, we prove by induction on formulas of LTL that $a \models \varphi$ iff $a \Vdash T_1\varphi$. The base cases are immediate. Suppose the result is true for all points a and formulas φ, ψ. The Boolean cases are routine. The case $\mathsf{Kuntil}(\varphi, \psi)$ is as follows:

$$
\begin{aligned}
& a \models \mathsf{Kuntil}(\varphi, \psi) \\
& \quad \text{iff} \quad \exists b \in M : a < b, b \models \varphi, \text{ and } (a,b) \models \psi \\
& \quad \text{iff} \quad \exists b \in M : a < b, b \Vdash T_1\varphi, \text{ and } (a,b) \Vdash T_1\psi \quad (\mathsf{IH}) \\
& \quad \text{iff} \quad \exists b \in M : a, b \Vdash \mathsf{until}(T_1\varphi, T_1\psi) \\
& \quad \text{iff} \quad a \Vdash \mathsf{dom}(\mathsf{until}(T_1\varphi, T_1\psi)) \\
& \quad \text{iff} \quad a \Vdash T_1(\mathsf{Kuntil}(\varphi, \psi)).
\end{aligned}
$$

The case $\mathsf{Ksince}(\varphi, \psi)$ is entirely similar. ■

D.2.2 Translating from LTL2 to LTL

Now, we move to the other direction, defining a translation from terms of LTL2 to formulas of LTL. Translation T_2 is a little more evolved and will be presented in two steps. First, we shall define a normal form for set terms of LTL2. Then, we will show how to translate set terms having this normal form.

Definition D.2.2 We say that a set term of LTL2 is in *dom normal form* (domNF) if all occurrences of dom in it has either a since or an until atom as an immediate component.

To put set terms in domNF we use an auxiliary operation D on LTL2 terms. For set terms D is defined as follows. We stipulate that D is the identity on $\top$ and p and commutes with Booleans. For a set term of the form $\mathsf{dom}(R)$ it is defined according to the structure of R, as given by:

$$
\begin{aligned}
D\mathsf{dom}(?A) \quad & ::= \quad DA, \\
D\mathsf{dom}(\mathsf{since}(A,B)) \quad & ::= \quad \mathsf{dom}(\mathsf{since}(DA, DB)), \\
D\mathsf{dom}(\mathsf{until}(A,B)) \quad & ::= \quad \mathsf{dom}(\mathsf{until}(DA, DB)), \\
D\mathsf{dom}(R \sqcup S) \quad & ::= \quad D\mathsf{dom}(R) \vee D\mathsf{dom}(S),
\end{aligned}
$$

$$D\mathsf{dom}(?A \circ R) ::= DA \wedge \mathsf{dom}(DR),$$

$$D\mathsf{dom}(\mathsf{since}(A,B) \circ R) ::= \mathsf{dom}(\mathsf{since}(DA \wedge D\mathsf{dom}(R), DB)),$$

$$D\mathsf{dom}(\mathsf{until}(A,B) \circ R) ::= \mathsf{dom}(\mathsf{until}(DA \wedge D\mathsf{dom}(R), DB)),$$

$$D\mathsf{dom}((R \sqcup S) \circ T) ::= D\mathsf{dom}(R \circ T) \vee D\mathsf{dom}(S \circ T),$$

$$D\mathsf{dom}((R \circ S) \circ T) ::= D\mathsf{dom}(R \circ (S \circ T)).$$

For relation terms, D is defined by:

$$D?A ::= ?DA,$$

$$D\mathsf{since}(A,B) ::= \mathsf{since}(DA, DB),$$

$$D\mathsf{until}(A,B) ::= \mathsf{until}(DA, DB),$$

$$D(R \sqcup S) ::= DR \sqcup DS,$$

$$D(R \circ S) ::= DR \circ DS.$$

Now we prove that every LTL2 formula has an equivalent one in domNF.

Lemma D.2.1 *$D\chi$ is in* domNF *and is equivalent to χ, for any term χ of* LTL2.

Proof. By double recursion on set terms and set relations. ∎

Now, we show how to translate set terms from LTL2 into formulas LTL. To this end we define a translation T_2 in two stages. First, T_2 transforms a set term A into a set term B which is in domNF. Second, it replaces, inside B, each occurrence of an operator dom followed immediately by an until or by a since by one occurrence of Kuntil or Ksince, respectively.

Proposition D.2.2 *For every set term A of* LTL2*, there exists a formula T_2A of* LTL*, such that:*

$$\mathfrak{M}, a \Vdash A \text{ iff } \mathfrak{M}, a \Vdash T_2A,$$

for any model $\mathfrak{M}$ and point $a \in M$.

Proof. First, observe that set terms of LTL2 in domNF can be defined by the following grammar:

$$A \;\; ::= \;\; \top \mid p \mid \neg A \mid A \wedge A \mid A \vee A \mid \mathsf{dom}(\mathsf{until}(A, A)) \mid \mathsf{dom}(\mathsf{since}(A, A)).$$

This fact allows us to define an auxiliary function T from LTL2 set terms in domNF into formulas of LTL by stipulating that T is the identity on $\top$ and p, commutes with the Booleans and treat dom in the following way:

$$T\mathsf{dom}(\mathsf{until}(A, B)) \;\; ::= \;\; \mathsf{Kuntil}(TA, TB),$$

$$T\mathsf{dom}(\mathsf{since}(A, B)) \;\; ::= \;\; \mathsf{Ksince}(TA, TB).$$

Given a model $\mathfrak{M}$ and a point $a \in M$, we prove by induction on set terms in domNF that:

$$a \Vdash TA \text{ iff } a \Vdash A. \tag{D.1}$$

The cases $\top$ and p are immediate. Suppose the result is true for all points $a \in M$ and set terms A, B of LTL2 in domNF. The Boolean cases are routine. The case $\mathsf{dom}(R)$ is proved by observing that since $\mathsf{dom}(R)$ is in domNF, there are set formulas A, B such that $R = \mathsf{until}(A, B)$ or $R = \mathsf{since}(A, B)$. In the first case, we have:

$$
\begin{aligned}
& a \Vdash T\mathsf{dom}(\mathsf{until}(A, B)) \\
& \quad \text{iff} \;\; a \Vdash \mathsf{Kuntil}(TA, TB) \\
& \quad \text{iff} \;\; \exists t \in T(s < t, t \Vdash TA, \text{ and} \\
& \quad\qquad \forall u \in T : \text{if } s < u < t, \text{ then } u \Vdash TB) \\
& \quad \text{iff} \;\; \exists t \in T(s < t, t \Vdash A, \text{ and} \\
& \quad\qquad \forall u \in T : \text{if } s < u < t, \text{ then } u \Vdash B) \qquad (\mathsf{IH}) \\
& \quad \text{iff} \;\; \exists t \in T : s, t \Vdash \mathsf{until}(A, B) \\
& \quad \text{iff} \;\; a \Vdash \mathsf{dom}(\mathsf{until}(A, B)).
\end{aligned}
$$

The other case is analogous.

Finally, we obtain a translation $T_2 = TD$ from LTL2 into LTL by composing the two auxiliary functions D and T in that order. It follows from Lemma D.2.1 and equivalence (D.1) that T_2 preserves meaning on models. ■

We close this section by showing the relationship between translations T_1 and T_2, defined above. Propositions below display the smooth relationship between them and show that, in a sense, the intermediate step of using domNF to define T_2 is a very natural one.

Proposition D.2.3 *$T_2 T_1 \varphi = \varphi$, for all formula φ of* LTL.

PROOF. Directly from the definition of D, by induction on φ. ∎

Proposition D.2.4 *$T_1 T_2 A = A$, for all set formula in* domNF *of* $\mathsf{LTL2}$.

PROOF. The proof is by induction on the length of A. To the base case, $A = p$, we have:

$$\begin{aligned} T_1 T_2 p &= T_1 p \\ &= p. \end{aligned}$$

Suppose the result is true for any set term of length smaller than $n \in \mathbb{N}$. Since both T_1 and T_2 commute with the Booleans, the cases $\top$, $\neg$, $\wedge$, and $\vee$ are routine. Now, suppose $A = \mathsf{dom}(R)$, where R is a relation term. Since A is in domNF, we have just to cases to consider.

- If $R = \mathsf{until}(B, C)$, where B and C are set terms. As the lengths of B and C are shorter than the length of A, we have:

$$\begin{aligned} T_1 T_2 \mathsf{dom}(R) &= T_1 T_2 \mathsf{dom}(\mathsf{until}(B, C)) \\ &= T_1 \mathsf{Kuntil}(T_2 B, T_2 C) \\ &= \mathsf{dom}(\mathsf{until}(T_1 T_2 B, T_1 T_2 C)) \\ &= \mathsf{dom}(\mathsf{until}(B, C)) \qquad (\mathsf{IH}) \end{aligned}$$

- If $R = \mathsf{since}(B, C)$, the proof is entirely analogous.

This completes the proof. ∎

D.3 The expressive power of LTL2 on relations

In this section, we compare the expressive powers of $\mathsf{LTL2}$ and TAL♭, showing that both languages are equally expressive on the class of all discrete linear flows of time. Since the latter is equally expressive as $\mathsf{FOL}(\mathrm{R}, <)$ on the class of all flows of time, we are, in fact, comparing $\mathsf{LTL2}$ with this formalism.

D.3.1 Translating from LTL2 to TAL$\flat$

As we mentioned previously, the fact that LTL2 is at least as expressive as TAL$\flat$ is just a consequence of the fact that the meaning of all operators of LTL2 can be defined with up to three variables. A more direct approach follows.

To set right some small syntactical and semantical differences, we consider two correspondences. One, $p \leftrightarrow r_p$, between PVAR and RVAR. The other, $\mathfrak{M} \leftrightarrow \mathfrak{M}^\flat$, that corresponds to a model $\mathfrak{M} = (M, <^{\mathfrak{M}}, p_i^{\mathfrak{M}})_{i\in\omega}$ of LTL2, the corresponding model $\mathfrak{M}^\flat = (M, <^{\mathfrak{M}}, r_{p_i}^{\mathfrak{M}})_{i\in\omega}$ of TAL$\flat$ such that $r_{p_i}^{\mathfrak{M}} = p_i^{\mathfrak{M}} \times M$, for every $i \in \omega$.

Theorem D.3.1 *Let R be a relation term of* LTL2*. Then, there exists a term $R^\flat$ of* TAL$\flat$ *such that:*

$$\mathfrak{M}, a, b \Vdash R \textit{ iff } \mathfrak{M}^\flat, a, b \Vdash R^\flat,$$

for every model $\mathfrak{M}$ and points $a, b \in M$.

PROOF. We define a translation from terms of LTL2 to terms of TAL$\flat$ as given in Table D.4. Given a model $\mathfrak{M}$ for LTL2, we prove by simultaneous recursion on terms that $\mathfrak{M}, a \Vdash A$ iff $\forall b \in M$: $\mathfrak{M}, a, b \Vdash A^\flat$ and $\mathfrak{M}, a, b \Vdash R$ iff $\mathfrak{M}, a, b \Vdash R^\flat$. ■

D.3.2 Translating from TAL$\flat$ to LTL2

Now, we move to the harder direction. We prove that all the relational operators of TAL$\flat$ are already present in the language of LTL2, when interpreted on discrete flows of time.

LTL2 is closed under O, I, $<$, and E

First, we investigate the easy definability questions. The next result shows that the operators O, I, $<$, and E are term definable and gives some useful macros.

$\top^\flat$	$::=$	E
$p^\flat$	$::=$	r_p
$(A \wedge B)^\flat$	$::=$	$A^\flat \sqcap B^\flat$
$(A \vee B)^\flat$	$::=$	$A^\flat \sqcup B^\flat$
$\mathsf{dom}(R)^\flat$	$::=$	$R^\flat \circ \mathsf{E}$
$(?A)^\flat$	$::=$	$\mathsf{I} \sqcap \overline{\overline{A^\flat} \circ \mathsf{E}}$
$\mathsf{until}(A, B)^\flat$	$::=$	$(< \circ (?A)^\flat) \sqcap \overline{< \circ \overline{(?B)^\flat} \circ <}$
$\mathsf{since}(A, B)^\flat$	$::=$	$(<^\mathsf{T} \circ ?A^\flat) \sqcap \overline{<^\mathsf{T} \circ (?\neg B)^\flat \circ <^\mathsf{T}}$
$(R \sqcup S)^\flat$	$::=$	$R^\flat \sqcup S^\flat$
$(R \circ S)^\flat$	$::=$	$R^\flat \circ S^\flat$

Table D.4: Translating from LTL2 to TAL♭.

Definition D.3.1 We define the following abbreviations:

$$\begin{aligned}
\mathsf{I} &::= ?\top \\
< &::= \mathsf{until}(\top,\top) \\
> &::= \mathsf{since}(\top,\top) \\
\mathsf{E} &::= > \sqcup \mathsf{I} \sqcup < \\
\emptyset &::= ?\neg\top \\
\leq &::= < \sqcup \mathsf{I} \\
\geq &::= > \sqcup \mathsf{I}
\end{aligned}$$

On the class of linear flows of time, *all* the abbreviations above have the intended meaning.

Lemma D.3.1 *Let* $\mathfrak{M} = (M, <^{\mathfrak{M}}, p_i^{\mathfrak{M}})_{i\in\omega}$ *be a model and* $a, b \in M$. *Then, the following hold:*

(a) $a, b \Vdash \mathsf{I}$ *iff* $a = b$,

(b) $a, b \Vdash <$ *iff* $a <^{\mathfrak{M}} b$,

(c) $a, b \Vdash >$ *iff* $b <^{\mathfrak{M}} a$,

(d) $a, b \Vdash \mathsf{E}$ *iff* $\mathfrak{M}$ *is a linear flow of time.*

(e) $a, b \Vdash \emptyset$ *iff never,*

(f) $a, b \Vdash \leq$ *iff* $a <^{\mathfrak{M}} b$ *or* $a = b$,

(g) $a, b \Vdash \geq$ *iff* $b < a$ *or* $b = a$.

PROOF. To prove D.3.1(d): Suppose $\mathfrak{M}$ is a flow of time. Then, $<^{\mathfrak{M}}$ is connected and given $a, b \in M$, we have:

$$\begin{aligned}
a, b \Vdash \mathsf{E} \quad &\text{iff} \quad a, b \in M \times M \\
&\text{iff} \quad a <^{\mathfrak{M}} b \text{ or } a = b, \text{ or } b <^{\mathfrak{M}} a \\
&\text{iff} \quad a, b \Vdash > \text{ or } a, b \Vdash \mathsf{I}, \text{ or } a, b \Vdash < \\
&\text{iff} \quad a, b \Vdash > \sqcup \mathsf{I} \sqcup <.
\end{aligned}$$

The other cases are immediate. ■

Thus, a part of the question is answered: I and $\emptyset$ are definable on the class of all models, and E is definable on the class of all linear flows of time. Now we prove that conversion is also definable on the class of all models.

LTL2 is closed under conversion

Differently from O, I, $<$, and E that have direct definitions given by terms of LTL2, we prove $^{\mathsf{T}}$ is definable using a slight weaker form of definability.

Definition D.3.2 Let op be an n-ary operator on binary relations and M be a class of models. We say that op *is definable on* M *in* LTL2 if for given relation terms $R_1, \ldots, R_n$ of LTL2 there exists a relation term R of LTL2 such that:

$$(a, b) \in \mathsf{op}([\![R_1]\!]_{\mathfrak{M}}, \ldots, [\![R_n]\!]_{\mathfrak{M}}) \text{ iff } \mathfrak{M}, a, b \Vdash R,$$

for every model $\mathfrak{M}$ in K and points $a, b \in M$.

Note that this notion is weaker than being in the sense that the definition of $\mathsf{op}(R_1, \ldots, R_n)$ might depend on $R_1, \ldots, R_n$. It is, however, general: the definition works for every model in the class.

Lemma D.3.2 *Conversion is definable in* LTL2 *on the class of all models.*

Proof. According to Definition D.3.2, we have to show that for every relation term S, there exists a relation term R such that $[\![R]\!]_{\mathfrak{M}}^{-1} = [\![S]\!]_{\mathfrak{M}}$, for any model $\mathfrak{M}$. First, we define an operation $^{\mathsf{rev}}$ on relation terms, as follows:

$$\begin{array}{lll}
(?A)^{\mathsf{rev}} & ::= & ?A, \\
\mathsf{until}(A, B)^{\mathsf{rev}} & ::= & ?A \circ \mathsf{since}(\top, B), \\
\mathsf{since}(A, B)^{\mathsf{rev}} & ::= & ?A \circ \mathsf{until}(\top, B), \\
(R \sqcup S)^{\mathsf{rev}} & ::= & R^{\mathsf{rev}} \sqcup S^{\mathsf{rev}}, \\
(R \circ S)^{\mathsf{rev}} & ::= & S^{\mathsf{rev}} \circ R^{\mathsf{rev}}.
\end{array}$$

Now, we prove by induction on R that $(a,b) \in [\![R]\!]^{-1}_{\mathfrak{M}}$ iff $a,b \Vdash R^{\mathsf{rev}}$, for any model $\mathfrak{M}$ and points $a,b \in M$. The base cases are:

$$\begin{aligned}
(a,b) \in [\![?A]\!]^{-1}_{\mathfrak{M}} \quad &\text{iff} \quad (b,a) \in [\![?A]\!]_{\mathfrak{M}} \\
&\text{iff} \quad b = a \text{ and } a \Vdash A \\
&\text{iff} \quad a = b \text{ and } b \Vdash A \\
&\text{iff} \quad a,b \Vdash ?A.
\end{aligned}$$

$$\begin{aligned}
(a,b) \in [\![\mathsf{until}(A,B)]\!]^{-1}_{\mathfrak{M}} \quad &\text{iff} \quad (b,a) \in [\![\mathsf{until}(A,B)]\!]_{\mathfrak{M}} \\
&\text{iff} \quad b < a,\ a \Vdash A \text{ and } (b,a) \Vdash B \\
&\text{iff} \quad a \Vdash A,\ b < a,\ b \Vdash \top \text{ and } (b,a) \Vdash B \\
&\text{iff} \quad a \Vdash A, \text{ and } a,b \Vdash \mathsf{since}(\top, B) \\
&\text{iff} \quad a \Vdash ?A \circ \mathsf{since}(\top, B)
\end{aligned}$$

The since case is entirely analogous. Now, assume the result holds for any relation terms R, S and any points $s, t \in T$. The remaining cases are just routine:

$$\begin{aligned}
(a,b) \in [\![R \cup S]\!]^{-1}_{\mathfrak{M}} \quad &\text{iff} \quad (b,a) \in [\![R \cup S]\!]_{\mathfrak{M}} \\
&\text{iff} \quad (b,a) \in [\![R]\!]_{\mathfrak{M}} \text{ or } (b,a) \in [\![S]\!]_{\mathfrak{M}} \\
&\text{iff} \quad (a,b) \in [\![R]\!]^{-1}_{\mathfrak{M}} \text{ or } (a,b) \in [\![S]\!]^{-1}_{\mathfrak{M}} \\
&\text{iff} \quad a,b \Vdash R^{\mathsf{rev}} \text{ or } a,b \Vdash S^{\mathsf{rev}} \qquad (\mathsf{IH}) \\
&\text{iff} \quad a,b \Vdash R^{\mathsf{rev}} \cup S^{\mathsf{rev}}.
\end{aligned}$$

$$\begin{aligned}
(a,b) \in [\![R \circ S]\!]^{-1}_{\mathfrak{M}} \quad &\text{iff} \quad (b,a) \in [\![R \circ S]\!]_{\mathfrak{M}} \\
&\text{iff} \quad \exists c \in M : (b,c) \in [\![R]\!]_{\mathfrak{M}} \text{ and } (c,a) \in [\![S]\!]_{\mathfrak{M}} \\
&\text{iff} \quad \exists c \in M : (c,b) \in [\![R]\!]^{-1}_{\mathfrak{M}} \text{ and } (a,c) \in [\![S]\!]^{-1}_{\mathfrak{M}} \\
&\text{iff} \quad \exists c \in M : a,c \Vdash S^{\mathsf{rev}} \text{ and } c,b \Vdash R^{\mathsf{rev}} \qquad (\mathsf{IH}) \\
&\text{iff} \quad a,b \Vdash R^{\mathsf{rev}} \circ S^{\mathsf{rev}}.
\end{aligned}$$

The proof is complete. ∎

Another useful operator is defined next. It will be useful in the sequel and, together with dom, it can be used to establish a close relationship between $\mathsf{LTL2}$ and the two important formalisms of dynamic modal logic and Peirce algebras [29]. But we will not go into these matters here. Observe that, in this case, we are considering operators that associate relations to sets much as does the dom operator.

Definition D.3.3 Let op be an n-ary operator on binary relations that produces a set as output, and M be a class of models. We say that op *is definable on* M *in* $\mathsf{LTL2}$ if for given relation terms $R_1, \ldots, R_n$ of $\mathsf{LTL2}$ there exists a set term A of $\mathsf{LTL2}$ such that:

$$a \in \mathsf{op}([\![R_1]\!]_{\mathfrak{M}}, \ldots, [\![R_n]\!]_{\mathfrak{M}}) \text{ iff } \mathfrak{M}, a \Vdash A,$$

for every model $\mathfrak{M}$ in K and point $a \in M$.

Let us denote by ran the operator that associates a relation to its range.

Lemma D.3.3 ran *is definable in* $\mathsf{LTL2}$ *on the class of all models.*

Proof. According to Definition D.3.3, we have to prove that, for any relation term R there exists a set term A such that $\mathsf{ran}([\![R]\!]_{\mathfrak{M}}) = [\![A]\!]_{\mathfrak{M}}$, for any model $\mathfrak{M}$. Given R, we associate by recursion on R, a set term R^{ran} of $\mathsf{LTL2}$, according to the rules below:

$$\begin{array}{lcl}
(?A)^{\mathsf{ran}} & := & A \\
\mathsf{until}(A, B)^{\mathsf{ran}} & := & \mathsf{dom}(?A \circ \mathsf{since}(\top, B)) \\
\mathsf{since}(A, B)^{\mathsf{ran}} & := & \mathsf{dom}(?A \circ \mathsf{until}(\top, B)) \\
(R \sqcup S)^{\mathsf{ran}} & := & R^{\mathsf{ran}} \sqcup S^{\mathsf{ran}} \\
(R \circ S)^{\mathsf{ran}} & := & \mathsf{dom}(S^{\mathsf{T}} \circ ?R^{\mathsf{ran}}).
\end{array}$$

By Lemma D.3.2, S^{T} can be replaced by a relation term, and the definition is complete. Now, given a model $\mathfrak{M}$ and a time point $a \in T$, we shall prove, by induction on R, that $b \in \mathsf{ran}([\![R]\!]_{\mathfrak{M}})$ iff $b \Vdash R^{\mathsf{ran}}$. The base cases are:

$$\begin{array}{rcl}
b \in \mathsf{ran}([\![?A]\!]_{\mathfrak{M}}) & \text{iff} & \exists a \in M : a, b \Vdash ?A \\
& \text{iff} & \exists a \in M : a = b \text{ and } a \Vdash A \\
& \text{iff} & b \Vdash A.
\end{array}$$

$$\begin{aligned}
b \in \mathsf{ran}(\mathsf{until}(A,B)) \quad & \text{iff} \quad \exists a \in M : a < b,\ b \Vdash A,\ \text{and } (a,b) \Vdash B \\
& \text{iff} \quad \exists a \in M : b \Vdash A,\ a < b,\ a \Vdash \top,\ \text{and } (a,b) \Vdash B \\
& \text{iff} \quad \exists a \in M : b \Vdash A \text{ and } b, a \Vdash \mathsf{since}(\top, B) \\
& \text{iff} \quad \exists a \in M : b, a \Vdash ?A \circ \mathsf{since}(\top, B) \\
& \text{iff} \quad b \Vdash \mathsf{dom}(?A \circ \mathsf{since}(\top, B)).
\end{aligned}$$

$$\begin{aligned}
b \in \mathsf{ran}([\![\mathsf{since}(A,B)]\!]_{\mathfrak{M}}) \quad & \text{iff} \quad \exists a \in M : a, b \Vdash \mathsf{since}(A,B) \\
& \text{iff} \quad \exists a \in M : b < a,\ b \Vdash A,\ \text{and } (b,a) \Vdash B \\
& \text{iff} \quad \exists a \in M : (b \Vdash A,\ b < a,\ a \models \top \text{ and } (b,a) \Vdash B \\
& \text{iff} \quad \exists a \in M : b \Vdash A \text{ and } b, a \models \mathsf{until}(\top, B) \\
& \text{iff} \quad \exists a \in M : b, a \Vdash ?A \circ \mathsf{until}(\top, B) \\
& \text{iff} \quad b \Vdash \mathsf{dom}(?A \circ \mathsf{until}(\top, B)).
\end{aligned}$$

Now, assume the result holds for relation terms R and S. The case $R \sqcup S$ is routine. To the case $R \circ S$, we have:

$$\begin{aligned}
b \in \mathsf{ran}([\![R \circ S]\!]_{\mathfrak{M}}) \quad & \text{iff} \quad \exists a, c \in M : a, c \Vdash R \text{ and } c, b \Vdash S \\
& \text{iff} \quad \exists a, c \in M : b, c \Vdash S^{\mathsf{T}} \text{ and } a, c \Vdash R \\
& \text{iff} \quad \exists c \in M : b, c \Vdash S^{\mathsf{T}} \text{ and } \exists a \in M : a, c \Vdash R \\
& \text{iff} \quad \exists c \in M : b, c \Vdash S^{\mathsf{T}} \text{ and } c \in \mathsf{ran}([\![R]\!]_{\mathfrak{M}}) \\
& \text{iff} \quad \exists c \in M : b, c \Vdash S^{\mathsf{T}} \text{ and } c \Vdash R^{\mathsf{ran}} \qquad \text{(IH)} \\
& \text{iff} \quad \exists c \in M : b, c \Vdash S^{\mathsf{T}} \circ ?R^{\mathsf{ran}} \\
& \text{iff} \quad b \models \mathsf{dom}(S^{\mathsf{T}} \circ ?R^{\mathsf{ran}}).
\end{aligned}$$

This completes the proof. ∎

LTL2 is closed under intersection

Now, we shall prove that intersection is definable in LTL2, on the class of linear flows of time. The proof will be split in two parts. First, we shall prove that every relational term of LTL2 can be rewritten as a union of terms having a normal form, resembling the *separated formulas* of [42]. More specifically, terms in normal form will be unions whose components are relation terms having one of the three forms defined below.

Definition D.3.4 Let R be a relation term of LTL2. We say that R is:
(1) an *until* if it has the form $?C \circ \mathsf{until}(A_1, B_1) \circ \ldots \circ \mathsf{until}(A_n, B_n)$, for some $0 < n \in \omega$;
(2) a *test* if it has the form $?A$;
(3) a *since* if has the form $?C \circ \mathsf{since}(A_1, B_1) \circ \ldots \circ \mathsf{since}(A_n, B_n)$, for some $0 < n \in \omega$.
(4) a *directed component* if it is either an until, or a test, or a since.
(5) in *directed disjunctive normal form* (DDNF) if it is a union whose components are directed.

Proposition D.3.1 *Let R be a relation term of* LTL2 *and $\mathfrak{M} = \langle M, <^{\mathfrak{M}}, r_i^{\mathfrak{M}} \rangle_{i \in \omega}$ be a flow of time. Then the following hold:*
(a) $[\![R]\!]_{\mathfrak{M}} \subseteq <^{\mathfrak{M}}$, *when R is an until;*
(b) $[\![R]\!]_{\mathfrak{M}} \subseteq I_M$, *when R is a test;*
(c) $[\![R]\!]_{\mathfrak{M}} \subseteq >^{\mathfrak{M}}$, *when R is a since.*

PROOF. Items (a) and (c) follow immediately by induction on n, using the fact that $<^{\mathfrak{M}}$ is transitive. Item (b) is immediate. ∎

Proposition D.3.1, implies that, when an until term R, is evaluated true at a pair of points a, b, in a flow of time, point b is in the *future* of point a. Analogously, b is in the *present* of a when R is a test and b is in the *past* of a when R is a since. Hence, terms in DDNF can be —if we do not take into account the possible references to *past, present, and future* made by the set terms occurring in it— considered as "separated" in a sense close to that given to this word in [42]. The use of this "separated" normal form will reduce the problem of defining intersection to that of defining intersection of the meaning of "separated" terms. As we will show at the end of the section, this latter problem has a very simple positive solution on the class of linear flows of time, implying that, when restricted to this semantics, intersection is already present in LTL2.

Besides, the nomenclature introduced in Definition D.3.4, we shall also make use of the following concepts:

Definition D.3.5 Let R be a relation term. We say that R is:
(1) an *atom* if it is either $\mathsf{until}(A, B)$, or $?A$, or $\mathsf{since}(A, B)$.
(2) a *basic composition* if it is a composition of atoms.

Theorem D.3.2 (DDNF Theorem) *Every relation term is equivalent to a relation term in* DDNF.

Proof. The proof is in two rounds. First, we prove in Lemma D.3.4 that every relation term can be rewritten as a disjunction of basic compositions. Second, we prove in Lemma D.3.5 that every basic composition is equivalent to a relation term in DDNF. From these we get the result. ∎

Lemma D.3.4 *Every relation term is equivalent to a relation term having the form*

$$\bigsqcup_{1 \leq i \leq n} R_i,$$

where $n \in \omega$ and each R_i, $1 \leq i \leq n$, is a basic composition.

Proof. By distributing $\sqcup$ over $\circ$, we get the result. ∎

To prove Lemma D.3.5 we will make use of the following identities that convert compositions of two atoms to DDNF.

Proposition D.3.2 *For any set terms A, B, C and D, the following hold, being equivalences* (a) *to* (c) *over the class of all models:*
(a) $?A \circ ?B \approx ?(A \wedge B)$;
(b) $\mathsf{until}(A, B) \circ ?C \approx \mathsf{until}(A \wedge C, B)$;
(c) $\mathsf{since}(A, B) \circ ?C \approx \mathsf{since}(A \wedge C, B)$;
(d) *Over the class of linear flows of time, we have:*

$$\begin{aligned} \mathsf{until}(A, B) \circ \mathsf{since}(C, D) \approx\ & \mathsf{until}(B \wedge C \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D)), B) \\ & \sqcup \\ & ?(C \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D))) \\ & \sqcup \\ & ?(D \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D))) \circ \mathsf{since}(C, D); \end{aligned}$$

(e) *Over the class of linear flows of time, we have:*

$$\begin{aligned}\mathsf{since}(A,B)\circ\mathsf{until}(C,D) \;\approx\;& \mathsf{since}(B\wedge C\wedge \mathsf{dom}(\mathsf{since}(A,B\wedge D)),B)\\ & \sqcup\\ & ?(C\wedge \mathsf{dom}(\mathsf{since}(A,B\wedge D)))\\ & \sqcup\\ & ?(D\wedge \mathsf{dom}(\mathsf{since}(A,B\wedge D)))\circ \mathsf{until}(C,D).\end{aligned}$$

Proof. Let $\mathfrak{M}$ be a model and a, b points in M.

To prove D.3.2(a):

$$\begin{aligned} a,b \Vdash\, ?A\circ ?B \quad &\text{iff}\quad \exists c\in M : a,c\Vdash\, ?A \text{ and } c,b\Vdash\, ?B\\ &\text{iff}\quad \exists c\in M : a=c=b, a\Vdash A, \text{ and } a\Vdash B\\ &\text{iff}\quad a=b \text{ and } a\Vdash A\wedge B\\ &\text{iff}\quad a,b\Vdash\, ?(A\wedge B).\end{aligned}$$

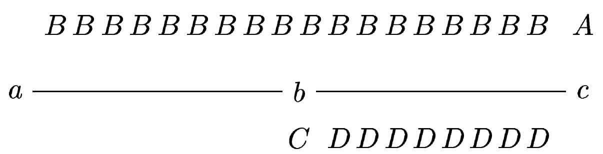
To prove D.3.2(b):

$$\begin{aligned} a,b\Vdash \mathsf{until}(A,B)\circ ?C \quad &\text{iff}\quad \exists c\in M : a,c\Vdash \mathsf{until}(A,B) \text{ and } c,b\Vdash\, ?C\\ &\text{iff}\quad \exists c\in M : a<c, c\Vdash A, (a,c)\Vdash B, c=b, \text{ and } c\Vdash C\\ &\text{iff}\quad a<b, b\Vdash A\wedge C, \text{ and } (a,b)\Vdash B\\ &\text{iff}\quad a,b\Vdash \mathsf{until}(A\wedge C,B).\end{aligned}$$

The proof of D.3.2(c) is entirely analogous to the proof of D.3.2(b). Now, to prove items (d) and (e), suppose $\mathfrak{M}$ is a flow of time.

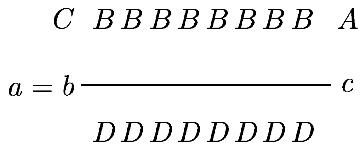
To prove D.3.2(d): To establish the inclusion from left to right, suppose $a, b \Vdash \mathsf{until}(A,B)\circ \mathsf{since}(C,D)$. So, by definition, $\exists c\in M$ for which $a,c\Vdash \mathsf{until}(A,B)$ and $c,b\Vdash \mathsf{since}(C,D)$. From this, we see that $a<c$, $c\Vdash A$, $(a,c)\Vdash B$, $b<c$, $b\Vdash C$, and $(b,c)\Vdash D$. As $a,b<c$ and $\mathfrak{M}$ is linear, there are exactly three possibilities a can find a place in relation to b, at the total ordering of M.

If $a<b$, then $\mathfrak{M}$ looks like:



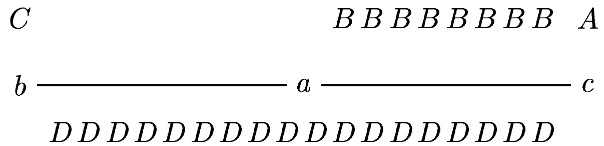
In this case, $a, b \Vdash \mathsf{until}(B \wedge C \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D)), B)$: by hypothesis, $a < b$; $b \Vdash B$ comes from $(a, c) \Vdash B$ and $a < b < c$; $b \Vdash C$ is given; $b \Vdash \mathsf{dom}(\mathsf{until}(A, B \wedge D))$ because $b < c$, $c \Vdash A$, and $(b, c) \Vdash D$ are given, moreover $(b, c) \Vdash B$ since $(a, c) \Vdash B$ and $a < b < c$; finally, $(a, b) \Vdash B$ because $(a, c) \Vdash B$ and $a < b < c$.

If $a = b$, then $\mathfrak{M}$ looks like:



In this case, we have $a, b \Vdash ?(C \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D)))$: by hypothesis, $a = b$; $a \Vdash C$ comes from $b \Vdash C$ and $a = b$; $a \Vdash \mathsf{dom}(\mathsf{until}(A, B \wedge D))$ because $a < c$, $c \Vdash A$, and $(a, c) \Vdash B$ are given, and $(a, c) \Vdash D$ since $(b, c) \Vdash D$ and $a = b$.

In the last case, when $b < a$, the linear flow of time $\mathfrak{M}$ looks like:



In this case, we have $a, b \Vdash ?(D \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D))) \circ \mathsf{since}(C, D)$: $a \Vdash D$ comes from $(b, c) \Vdash D$ and $b < a < c$; $a \Vdash \mathsf{dom}(\mathsf{until}(A, B \wedge D))$ because $a < c$, $c \Vdash A$, and $(a, c) \Vdash B$ are given, moreover $(a, c) \Vdash D$ because $(b, c) \Vdash D$ and $(a, c) \subset (b, c)$; $a, b \Vdash \mathsf{since}(C, D)$ because $b < a$ by hypothesis, $b \Vdash C$ is given, and $(b, a) \Vdash D$ comes from $(b, c) \Vdash D$ and $(b, a) \subset (b, c)$.

To prove the converse, we consider the three cases. First, suppose $a, b \Vdash \mathsf{until}(B \wedge C \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D)), B)$. Hence, $a < b$, $b \Vdash B$, $b \Vdash C$, $b \Vdash \mathsf{dom}(\mathsf{until}(A, B \wedge D))$, and $(a, b) \Vdash B$. Hence, $\exists c \in M$ such that $b, c \Vdash \mathsf{until}(A, B \wedge D))$. From this, we have $b < c$, $c \Vdash A$, $(b, c) \Vdash B$, and $(b, c) \Vdash D$. To prove $a, c \Vdash \mathsf{until}(A, B)$: $a < c$ comes from $a < b < c$ and $<$ is transitive; $c \Vdash A$ is given; $(a, c) \Vdash B$ comes from $(a, b) \Vdash B$, $b \Vdash B$, $(b, c) \Vdash B$ and $(a, c) = (a, b) \cup \{b\} \cup (b, c)$. To prove $c, b \Vdash \mathsf{since}(C, D)$: $b < c$, $b \Vdash C$, and $(b, c) \Vdash D$ are given.

Next, assume $a, b \Vdash ?(C \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D)))$. Hence, $a = b$, $a \Vdash C$, and $a \Vdash \mathsf{dom}(\mathsf{until}(A, B \wedge D))$. So, $\exists c \in M$ such that $a, c \Vdash \mathsf{until}(A, B \wedge D))$. From this, we have $a < c$, $c \Vdash A$, $(a, c) \Vdash B$, and $(a, c) \Vdash D$. To prove $a, c \Vdash \mathsf{until}(A, B)$: $a < c$, $c \Vdash A$, and $(a, c) \Vdash B$ are given. To prove $c, b \Vdash \mathsf{since}(C, D)$: $b < c$, $b \Vdash C$, and $(b, c) \Vdash D$ follows directly from $a = b$, $a < c$, $a \Vdash C$, and $(a, c) \Vdash D$.

Finally, suppose $a, b \Vdash ?(D \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D))) \circ \mathsf{since}(C, D)$. Hence, $\exists c \in M$ for which $a, c \Vdash ?(D \wedge \mathsf{dom}(\mathsf{until}(A, B \wedge D)))$ and $c, b \Vdash \mathsf{since}(C, D)$ hold. From this, we have: $a = c$, $a \Vdash D$, $a \Vdash \mathsf{dom}(\mathsf{until}(A, B \wedge D))$, $b < c$, $b \Vdash C$, and $(b, c) \Vdash D$. Now, take $d \in M$ such that $a, d \Vdash \mathsf{until}(A, B \wedge D))$. From this, we have $a < d$, $d \Vdash A$, $(a, d) \Vdash B$, and $(a, d) \Vdash D$. To prove $a, d \Vdash \mathsf{until}(A, B)$: $a < d$, $d \Vdash A$, and $(a, d) \Vdash B$ are given. To prove $d, b \Vdash \mathsf{since}(C, D)$: $b < d$ because $b < c = a < d$ and $<$ is transitive; $b \Vdash C$ is given; $(b, d) \Vdash D$ because $(b, d) \Vdash D$, $a \Vdash D$, $(a, d) \Vdash D$, $a = c$, and $(b, d) = (b, a) \cup \{a\} \cup (a, d)$.
The proof of D.3.2(e) is entirely analogous to the proof of D.3.2(d). ∎

Lemma D.3.5 *On linear flows of time, every basic composition is equivalent to a relation term in* DDNF.

PROOF. Let R be a basic composition. We prove the result by induction on the number of $\circ$'s occurring in R. If R is an atom, then the result is true because atoms are already in DDNF. If R is a composition of two atoms, it has one of the following nine forms:

$$\begin{array}{ccc} ?A\circ ?B & \underline{?C \circ \mathsf{until}(A, B)} & \underline{?C \circ \mathsf{since}(A, B)} \\ \mathsf{until}(A, B)\circ ?C & \underline{\mathsf{until}(A, B) \circ \mathsf{until}(C, D)} & \mathsf{until}(A, B) \circ \mathsf{since}(C, D) \\ \mathsf{since}(A, B)\circ ?C & \mathsf{since}(A, B) \circ \mathsf{until}(C, D) & \underline{\mathsf{since}(A, B) \circ \mathsf{since}(C, D)} \end{array}$$

The underlined ones are already in DDNF. The others ones, by Proposition D.3.2, are equivalent to relation terms in DDNF.

Now, assume the result holds for any basic composition having n occurrences of $\circ$ and let $R = \mathsf{atom} \circ S$ be a basic composition having $n + 1$ occurrences of $\circ$. By IH, $S \approx \bigsqcup_{1 \leq i \leq n} S_i$, where each S_i, $1 \leq i \leq n$, is a directed component. Hence $R \approx \mathsf{atom} \circ (\bigsqcup_{1 \leq i \leq n} S_i) \approx \bigsqcup_{1 \leq i \leq n} (\mathsf{atom} \circ S_i)$ that is in DDNF. ∎

Theorem D.3.3 *Intersection is definable in* LTL2*, over the class of linear flows of time.*

PROOF. According to Definition D.3.2, we have to proof that for all relation terms R, S, there exists a relation term T such that: $[\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}} = [\![T]\!]_{\mathfrak{M}}$, for any flow of time $\mathfrak{M}$. Let R, S be relation terms and $\mathfrak{M}$ be a linear flow of time. By Theorem DDNF, we may assume $R \approx \bigsqcup_{1 \leq i \leq m} R_i$ and $S \approx \bigsqcup_{1 \leq j \leq n} S_j$, where $m, n \in \omega$ and each member of R_i, S_j is either a since, or a test, or an until relation term. Hence, we have:

$$\begin{aligned} [\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}} &= \bigcup_i [\![D_i]\!]_{\mathfrak{M}} \cap \bigcup_j [\![E_j]\!]_{\mathfrak{M}} \\ &= \bigcup_{i,j} ([\![D_i]\!]_{\mathfrak{M}} \cap [\![E_j]\!]_{\mathfrak{M}}). \end{aligned}$$

Hence, if we prove that the intersection of directed relation terms is definable, we are done. Let R, S be two directed relation terms. Since intersection is commutative, we just need to consider four cases.

First, if $R = ?A$ and $S = ?B$ are tests, then, for any $a, b \in M$, we have:

$$\begin{aligned} a, b \Vdash ?A \text{ and } a, b \Vdash ?B \quad &\text{iff} \quad a = b, a \Vdash A, \text{ and } a \Vdash B \\ &\text{iff} \quad a = b, a \Vdash A \wedge B \\ &\text{iff} \quad a, b \Vdash ?A \wedge B. \end{aligned}$$

So, the intersection of $?A$ and $?B$ is definable by $?(A \wedge B)$. Second, if one is a test and the other not, then by Proposition D.3.1, their intersection is empty, whence definable by $?\bot$. Similarly, when one is an until and the other a since relation term, again by Proposition D.3.1, their intersection is empty and definable by $?\bot$. So, the only interesting cases are when both are basic compositions in the same direction. We treat these two cases into details, now. First, assume

$$R = ?E \circ \mathsf{until}(A_1, B_1) \circ \cdots \circ \mathsf{until}(A_m, B_m)$$

is an until relation term. We prove by induction on the number of $\circ$'s in R that for any until relation term

$$S = ?F \circ \mathsf{until}(C_1, D_1) \circ \cdots \circ \mathsf{until}(C_n, D_n),$$

the intersection $[\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}}$ is definable.

The base case is when $m = 1$. In this case, $R = ?E \circ \mathsf{until}(A_1, B_1)$ and, over flows of time, we have the equivalence:

$$a, b \Vdash ?E \circ \mathsf{until}(A_1, B_1) \quad \text{and} \quad a, b \Vdash ?F \circ \mathsf{until}(C_1, D_1) \circ \cdots \circ \mathsf{until}(C_n, D_n)$$

if and only if

$$a, b \Vdash ?(E \wedge F) \circ \mathsf{until}(B_1 \wedge C_1, B_1 \wedge D_1) \circ \mathsf{until}(B_1 \wedge C_2, B_1 \wedge D_2) \circ \ldots \circ \mathsf{until}(B_1 \wedge C_{n-1}, B_1 \wedge D_{n-1}) \circ \mathsf{until}(A_1 \wedge C_n, B_1 \wedge D_n).$$

This can be proved as follows. Let $\mathfrak{M}$ be a flow of time and a, b time points in M. Then, we have:

$a, b \Vdash ?E \circ \mathsf{until}(A_1, B_1)$
and
$a, b \Vdash ?F \circ \mathsf{until}(C_1, D_1) \circ \ldots \circ \mathsf{until}(C_n, D_n)$

if and only if

$a \Vdash E$ and $a \Vdash F$, and $a, b \Vdash \mathsf{until}(A_1, B_1)$,
and
there are time points $c_1, \ldots, c_{n-1} \in M$ for which
$a, c_1 \Vdash \mathsf{until}(C_1, D_1)$ and $\ldots$, and $c_{n-1}, b \Vdash \mathsf{until}(C_n, D_n)$.

if and only if

there are time points $c_1, c_2, \ldots, c_{n-1} \in M$ for which
$a \Vdash E \wedge F$, and
$a < b$, and $b \Vdash A_1$, and $(a, b) \Vdash B_1$, and
$a < c_1$, and $c_1 \Vdash C_1$, and $(a, c_1) \Vdash D_1$, and
$c_1 < c_2$, and $c_2 \Vdash C_2$, and $(c_1, c_2) \Vdash D_2$, and $\ldots$,
$c_{n-2} < c_{n-1}$, and $c_{n-1} \Vdash C_{n-1}$, and $(c_{n-2}, c_{n-1}) \Vdash D_{n-1}$, and
$c_{n-1} < b$, and $b \Vdash C_n$, and $(c_{n-1}, b) \Vdash D_n$.

Since $\mathfrak{M}$ is a flow of time, in the presence of $a < c_1 < \ldots < c_{n-1} < b$ we have that $a < b$, and in the presence of these together with $(a, b) \Vdash B_1$ we obtain that all the time points $c_1, c_2 \ldots, c_{n-1}$ as well as all the intervals $(a, c_1), (c_1, c_2), \ldots, (c_{n-1}, b)$ force the set term B_1 be true. Hence, the last statement above is equivalent to:

there are time points $c_1, c_2, \ldots, c_{n-1} \in M$ for which
$a \Vdash E \wedge F$, and
$a < c_1$, and $c_1 \Vdash B_1 \wedge C_1$, and $(a, c_1) \Vdash B_1 \wedge D_1$, and
$c_1 < c_2$, and $c_2 \Vdash B_1 \wedge C_2$, and $(c_1, c_2) \Vdash B_1 \wedge D_2$, and $\ldots$,
$c_{n-2} < c_{n-1}$, and $c_{n-1} \Vdash B_1 \wedge C_{n-1}$, and $(c_{n-2}, c_{n-1}) \Vdash B_1 \wedge D_{n-1}$, and
$c_{n-1} < b$, and $b \Vdash A_1 \wedge C_n$, and $(c_{n-1}, b) \Vdash B_1 \wedge D_n)$

if and only if

$\exists c_1, c_2, \ldots, c_{n-1} \in M$ for which
$a \Vdash E \wedge F$, and
$a, c_1 \Vdash \mathsf{until}(B_1 \wedge C_1, B_1 \wedge D_1)$, and
$c_1, c_2 \Vdash \mathsf{until}(B_1 \wedge C_2, B_1 \wedge D_2)$, and $\ldots$,
$c_{n-2}, c_{n-1} \Vdash \mathsf{until}(B_1 \wedge C_{n-1}, B_1 \wedge D_{n-1})$, and
$c_{n-1}, b \Vdash \mathsf{until}(A_1 \wedge C_n, B_1 \wedge D_n)$

if and only if

$$a, b \Vdash ?(E \wedge F) \circ \mathsf{until}(B_1 \wedge C_1, B_1 \wedge D_1) \circ \mathsf{until}(B_1 \wedge C_2, B_1 \wedge D_2) \circ \ldots \circ \mathsf{until}(B_1 \wedge C_{n-1}, B_1 \wedge D_{n-1}) \circ \mathsf{until}(A_1 \wedge C_n, B_1 \wedge D_n).$$

This concludes the base case.

Now, let $R = ?E \circ \mathsf{until}(A_1, B_1) \circ \cdots \circ \mathsf{until}(A_m, B_m)$, with $1 < m$, and suppose that the intersection of any until relation term, having less occurrences of $\circ$ than R, with any other until relation term is definable on flows of time.

Take $S = ?F \circ \mathsf{until}(C_1, D_1) \circ \mathsf{until}(C_2, D_2) \circ \cdots \circ \mathsf{until}(C_n, D_n)$. Given a flow of time $\mathfrak{M}$ and time points $a, b \in M$, we have that the ordered pair (a, b) belongs to the intersection $[\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}}$ iff there are time points $c, d_1, d_2, \ldots, d_{n-1}$ for which both conditions hold:

$$a \Vdash E \text{ and } a, c \Vdash \mathsf{until}(A_1, B_1), \text{ and } c, b \Vdash \mathsf{until}(A_2, B_2) \circ \cdots \circ \mathsf{until}(A_m, B_m)$$

and

$$a \Vdash F, \text{ and } a, d_1 \Vdash \mathsf{until}(C_1, D_1), \text{ and } \ldots, \text{ and } d_{n-1}, b \Vdash \mathsf{until}(C_n, D_n).$$

Since $\mathfrak{M}$ is a flow of time and for any set relations A, B the set relation $\mathsf{until}(A, B)$ is a subrelation of $<^{\mathfrak{M}}$, the time points $a, b, c, d_1, \ldots, d_{n-1}$ should be displayed in $\mathfrak{M}$ as $a < c < b$ and $a < d_1 < \cdots < d_{n-1} < b$. Therefore, we have three cases, according to the place c finds at the linear ordering $a < d_1 < \cdots < d_{n-1} < b$.

First, $a < c < d_1$; second $d_{n-1} < c < b$; and third there exists i, $1 \leq i \leq n-1$, such that $d_i < c < d_{i+1}$. We treat just the last case, the others are similar. So, we suppose i, $1 \leq i \leq n-1$, such that $d_i < c < d_{i+1}$. For any such i, we obtain:

- $a, c \Vdash ?F \circ \mathsf{until}(C_1, D_1) \circ \cdots \circ \mathsf{until}(C_i, D_i) \circ \mathsf{until}(D_{i+1}, D_{i+1})$: since $a, d_i \Vdash ?F \circ \mathsf{until}(C_1, D_1) \circ \cdots \circ \mathsf{until}(C_i, D_i)$ is given and $d_i < c < d_{i+1}$ and $d_i, d_{i+1} \Vdash \mathsf{until}(C_{i+1}, D_{i+1})$ gives, directly, $d_i c \Vdash \mathsf{until}(D_{i+1}, D_{i+1})$.

- $c, b \Vdash \mathsf{until}(C_{i+1}, D_{i+1}) \circ \cdots \circ \mathsf{until}(C_n, D_n)$: this time, $d_i < c < d_{i+1}$ and $d_i, d_{i+1} \Vdash \mathsf{until}(C_{i+1}, D_{i+1})$ gives, directly, $c, d_{i+1} \Vdash \mathsf{until}(D_{i+1}, D_{i+1})$; moreover, $d_{i+1}, b \Vdash \mathsf{until}(C_{i+2}, D_{i+2}) \circ \cdots \circ \mathsf{until}(C_n, D_n)$ is given.

Hence, in this case, to say that the ordered pair (a, b) is in both $[\![R]\!]_{\mathfrak{M}}$ and $[\![S]\!]_{\mathfrak{M}}$ is equivalent to say that there is a time point $c \in M$ for which:

$$\begin{array}{l} a, c \Vdash ?E \circ \mathsf{until}(A_1, B_1) \\ \text{and} \\ a, c \Vdash ?F \circ \mathsf{until}(C_1, D_1) \circ \cdots \circ \mathsf{until}(D_{i+1}, D_{i+1}) \end{array}$$

and

$$\begin{array}{l} c, b \Vdash \mathsf{until}(A_2, B_2) \circ \cdots \circ \mathsf{until}(A_m, B_m) \\ \text{and} \\ c, b \Vdash \mathsf{until}(C_{i+1}, D_{i+1}) \circ \cdots \circ \mathsf{until}(C_n, D_n). \end{array}$$

By the IH, we can deal with both intersections. Hence, in this case, a, b forces both R and S iff it forces the union:

$$\bigsqcup_{i=1}^{n-1} \begin{array}{c} ?E \circ \mathsf{until}(A_1, B_1) \ \sqcap \ ?F \circ \mathsf{until}(C_1, D_1) \circ \cdots \circ \mathsf{until}(D_{i+1}, D_{i+1}) \\ \circ \\ \mathsf{until}(A_2, B_2) \circ \cdots \circ \mathsf{until}(A_m, B_m) \ \sqcap \ \mathsf{until}(C_{i+1}, D_{i+1}) \circ \cdots \circ \mathsf{until}(C_n, D_n). \end{array}$$

Again, since this is a union of relation terms we conclude that $[\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}}$ is definable.

Finally, suppose $d_n < c < b$. Under this assumption, we obtain:

- $a, c \Vdash ?F \circ \mathsf{until}(C_1, D_1) \circ \ldots \circ \mathsf{until}(C_{n-1}, D_{n-1}) \circ \mathsf{until}(D_n, D_n)$: by a reasoning similar to the previous;

- $c, b \Vdash \mathsf{until}(C_n, D_n)$: since $c < b$ and $b \Vdash C_n$ are given; moreover, $(d_n, b) \Vdash D_n$ together with $(c, b) \subset (d_n, b)$ implies $(c, b) \Vdash D_n$.

Hence, in this case, to say that the ordered pair (a, b) belongs to both $[\![R]\!]_{\mathfrak{M}}$ and $[\![S]\!]_{\mathfrak{M}}$ is equivalent to say that there exists a time point $c \in M$ such that:

$$a, c \Vdash ?E \circ \mathsf{until}(A_1, B_1) \text{ and } a, c \Vdash ?F \circ \mathsf{until}(C_1, D_1) \circ \ldots \mathsf{until}(D_n, D_n)$$

and

$$c, b \Vdash \mathsf{until}(A_2, B_2) \circ \ldots \circ \mathsf{until}(A_n, B_n) \text{ and } c, b \Vdash \mathsf{until}(C_n, D_n).$$

By the IH, we can deal with both intersections, whence $[\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}}$ is definable by the composition of

$$?E \circ \mathsf{until}(A_1, B_1) \ \sqcap \ ?F \circ \mathsf{until}(D_1, D_1) \circ \mathsf{until}(C_1, D_1) \circ \ldots \mathsf{until}(D_n, D_n)$$

with

$$\mathsf{until}(A_2, B_2) \circ \ldots \circ \mathsf{until}(A_n, B_n) \ \sqcap \ \mathsf{until}(C_n, D_n).$$

This completes the proof that intersection is definable. ■

LTL2 is closed under complementation

Finally, we shall prove that complementation of relation terms is definable in LTL2, on the class of discrete flows of time. Our approach is based on equational reasoning, as follows. First, we shall expand LTL2 with intersection and complementation, obtaining an auxiliary relational calculus $\overline{\mathsf{LTL2}}$. Second, we shall present a series of equalities of $\overline{\mathsf{LTL2}}$ that are valid on the class of all discrete flows of time. Using these identities, we shall show how the complemented relation terms of LTL2 may be rewritten without occurrences of negation but, possibly, with occurrences of intersection. This means that $\overline{\mathsf{LTL2}}$ and LTL2 expanded with intersection are equally expressive. Since, by Theorem D.3.3, the expansion of LTL2 by way of intersection does not increase its expressive power, we get the result.

Definition D.3.6 (1) Starting with the syntax of LTL2, we add the following to constitute the language of $\overline{\mathsf{LTL2}}$:

(1.1) the relational operators $\sqcap$ and $^-$;

(1.2) to the generating grammar the "rule":

$$R ::= R \sqcap R \mid \overline{R}.$$

(2) $\overline{\mathsf{LTL2}}$ has the same class of models as LTL2.

(3) Starting with the semantics of LTL2, we add to the definition of meaning the clauses below, to constitute the language of $\overline{\mathsf{LTL2}}$:

$$[\![R \sqcap S]\!]_{\mathfrak{M}} ::= [\![R]\!]_{\mathfrak{M}} \cap [\![S]\!]_{\mathfrak{M}},$$

$$[\![\overline{R}]\!]_{\mathfrak{M}} ::= [\![R]\!]^c_{\mathfrak{M}}.$$

Clauses above can also be rewritten by using a modal logic-like notation, respectively, as:

$$a, b \Vdash R \sqcap S \quad \text{iff} \quad a, b \Vdash R \text{ and } a, b \Vdash S,$$

$$a, b \Vdash \overline{R} \quad \text{iff} \quad a, b \nVdash R.$$

The elementary axioms of a total linear ordering can be written in the language of $\mathsf{LTL2}$ as in Table D.5.

Irr)	$\mathsf{I} \sqcap < \approx \mathsf{O}$
Tra)	$< \circ < \sqsubseteq <$
Con)	$> \sqcup \mathsf{I} \sqcup < \approx \mathsf{E}$

Table D.5: Axioms of order.

Irr, Tra, and Con are valid on linear flows of time, and permit us to obtain some general consequences about their structure. For instance, the following equalities follows:

$$> \sqcap \mathsf{I} \approx \mathsf{O}$$

$$> \sqcap < \approx \mathsf{O}$$

Axioms Irr and Con together with equalities above warrant that, given any linear flow of time $\mathfrak{M}$, the relations $>^{\mathfrak{M}}$, I_M, and $<^{\mathfrak{M}}$ form a partition of the universe M. But to a proper algebraic treatment of the expressive power of $\mathsf{LTL2}$, we shall need a lot more. First, we prove some simple validities on tests and intersection.

Proposition D.3.3 *Let A, B be set terms and R, S be relation terms. Then the following equivalences are valid.*

(a) $?A \circ (R \sqcap S) \approx (?A \circ R) \sqcap S$.

(b) $?A \circ R \sqcap ?\neg A \circ S \approx \mathsf{O}$.

(c) $?A \circ (R \sqcap S) \approx ?A \circ R \sqcap ?A \circ S$.

PROOF. To prove D.3.3(a):

$$a, b \Vdash ?A \circ (R \sqcap S) \quad \text{iff} \quad \exists c \in M : a = c \text{ and } a \Vdash A, \text{ and } c, b \Vdash R \text{ and } a, b \Vdash S$$

$$\text{iff} \quad \exists c \in M : a, c \Vdash ?A; \text{ and } c, b \Vdash R; \text{ and } a, b \models S$$

$$\text{iff} \quad a, b \Vdash (?A \circ R) \sqcap S.$$

The other are analogous. ■

Now, we prove some useful identities on complementation. The proposition below shows that, in the presence of complementation, until and since are definable.

Proposition D.3.4 *Let A, B be set terms. Then the following equalities are valid:*
(a) $\mathsf{until}(A, B) \approx < \circ\, ?A \sqcap \overline{< \circ\, ?\neg B \circ <}$.
(b) $\mathsf{since}(A, B) \approx > \circ\, ?A \sqcap \overline{> \circ\, ?\neg B \circ >}$.

PROOF. Immediate from definitions. ■

Since we have complementation, we can use Proposition D.3.4 to obtain useful inclusions.

Corollary D.3.1 *Let A, B be set terms and R be a relation term. Then, we have:*
(a) $\mathsf{until}(\top, \bot) \approx < \sqcap \overline{< \circ <}$.
(b) *If* $R \sqsubseteq <$, *then* $\mathsf{until}(\top, \bot) \sqsubseteq \overline{R \circ <}$.
(c) $\mathsf{until}(\top, \neg A) \sqsubseteq \overline{\mathsf{until}(A, B) \circ <}$.

PROOF. To prove D.3.1(a):

$$\begin{aligned} \mathsf{until}(\top, \bot) &\approx < \circ\, ?\top \sqcap \overline{< \circ\, ?\neg\bot \circ <} && (D.3.4(a)) \\ &\approx < \sqcap \overline{<\circ ?\top \circ <} \\ &\approx < \sqcap \overline{< \circ <}. \end{aligned}$$

To prove D.3.1(b): Let $R \sqsubseteq <$. By RA, we have $R \circ < \sqsubseteq < \circ <$. Hence, by BA, $\overline{< \circ <} \sqsubseteq \overline{R \circ <}$. From this, we obtain the result as follows:

$$\begin{aligned} \mathsf{until}(\top, \bot) &\approx < \sqcap \overline{< \circ <} && (D.3.1(a)) \\ &\sqsubseteq \overline{< \circ <} && (\mathsf{BA}) \\ &\sqsubseteq \overline{R \circ <}. \end{aligned}$$

To prove D.3.1(c), first, note that:

$$\begin{aligned}
\mathsf{until}(A,B)\circ < \;&\approx\; (<\circ ?A \sqcap \overline{<\circ ?\neg B\circ <})\circ < && (D.3.4(a))\\
&\sqsubseteq\; <\circ ?A\circ < \sqcap \overline{<\circ ?\neg B\circ <}\circ < && (\mathsf{RA})\\
&\sqsubseteq\; <\circ ?A\circ < && (\mathsf{BA})\\
&\sqsubseteq\; \overline{<\circ ?\top} \sqcup <\circ ?A\circ < && (\mathsf{BA})\\
&\approx\; \overline{<\circ ?\top \sqcap \overline{<\circ ?A\circ <}} && (\mathsf{BA})\\
&\approx\; \overline{\mathsf{until}(\top,\neg A)} && (D.3.4(a)).
\end{aligned}$$

Hence, by BA, $\mathsf{until}(\top,\neg A)\sqsubseteq \overline{\mathsf{until}(A,B)\circ <}$. ■

Proposition D.3.5 *Let A, B be set terms and R, S be relation terms. Then, the following equivalences are valid on linear flows of time.*

(a) $\overline{R\circ ?A}\sqcap R\approx R\circ ?\neg A$ *and* $\overline{?A\circ R}\sqcap R\approx ?\neg A\circ R$.

(b) $?C\circ (R\sqcap \overline{S})\approx ?C\circ R\sqcap \overline{?C\circ S}$.

(c) $?C\circ <\sqcap \overline{<\circ ?A}\approx ?C\circ <\circ ?\neg A$.

(d) $?C\circ <\sqcap \overline{?C\circ \mathsf{until}(A,B)\circ <}\approx ?C\circ <\sqcap \overline{\mathsf{until}(A,B)\circ <}$.

Proof. We establish just D.3.5(a) and (d), by semantic reasoning.

To prove D.3.5(a): For any $b\in M$, we have $b\Vdash A$ iff $\exists c\in M : c,b\Vdash ?A$. Hence:

$$\begin{aligned}
& a,b\Vdash \overline{R\circ ?A}\sqcap R\\
&\quad \text{iff}\;\; \forall c\in M : \text{if } a,c\Vdash R, \text{ then } c,b\nVdash ?A;\; \text{and } a,b\Vdash R\\
&\quad \text{iff}\;\; a,b\Vdash R \text{ and } b\nVdash A\\
&\quad \text{iff}\;\; a,b\Vdash R \text{ and } b\Vdash \neg A\\
&\quad \text{iff}\;\; a,b\Vdash R\circ ?\neg A.
\end{aligned}$$

The other equivalence is analogous.

To prove D.3.5(d): Let $\mathfrak{M}$ be a flow of time and $a,b\in M$. We prove the double implication $a,b\Vdash ?C\circ <\sqcap \overline{?C\circ \mathsf{until}(A,B)\circ <}$ iff $a,b\Vdash ?C\circ (<\sqcap \overline{\mathsf{until}(A,B)\circ <})$.

To prove the implication from left to right, suppose both $a,b\Vdash ?C\circ <$ and $a,b\Vdash \overline{?C\circ \mathsf{until}(A,B)\circ <}$. Let $c\in M$ be such that $a,c\Vdash ?C$ and $c,b\Vdash <$, and suppose, for a contradiction, that $c, b \Vdash \mathsf{until}(A, B) \circ <$. By choosing $d \in M$ for which $c, d \Vdash \mathsf{until}(A, B)$ and $d < b$, we obtain $a, c \Vdash ?C$, $c, d \Vdash \mathsf{until}(A, B)$, and $d < b$. But this implies $a, b \Vdash ?C \circ \mathsf{until}(A, B) \circ <$, a contradiction.

To prove the converse, suppose $a, b \Vdash ?C(\circ < \sqcap \overline{\mathsf{until}(A, B) \circ <})$ and let $c \in M$ be such that $a, c \Vdash ?C$, $c, b \Vdash <$, and $c, b \Vdash \overline{\mathsf{until}(A, B) \circ <}$. Besides, suppose, for a contradiction, that $a, b \Vdash ?C \circ \mathsf{until}(A, B) \circ <$, and let $d, e \in M$ such that $a, d \Vdash ?C$, $d, e \Vdash \mathsf{until}(A, B)$, and $e, b \Vdash <$. From $a, c \Vdash ?C$ and $a, d \Vdash ?C$ we obtain $a = c = d$. Hence, we have $c, e \Vdash \mathsf{until}(A, B)$ and $e < b$, implying $c, b \Vdash \mathsf{until}(A, B) \circ <$, a contradiction. ■

Proposition D.3.6 *Let A be a set term and R be a relation term. Then the following equivalences are valid on linear flows of time.*
(a) *If $R \sqsubseteq <$, then $\overline{R} \approx \geq \sqcup (\overline{R} \sqcap <)$.*
(b) $< \circ < \approx (< \circ < \sqcap \mathsf{until}(\top, \neg A)) \sqcup \mathsf{until}(A, \top) \circ <$.

Proof. To prove D.3.6(a): Let $R \sqsubseteq <$. Then:

$$
\begin{aligned}
\overline{R} &\approx \overline{R} \sqcap \mathsf{E} && \text{(BA)}\\
&\approx \overline{R} \sqcap (> \sqcup \mathsf{I} \sqcup <) && \text{(Con)}\\
&\approx (\overline{R} \sqcap >) \sqcup (\overline{R} \sqcap \mathsf{I}) \sqcup (\overline{R} \sqcap <) && \text{(BA)}\\
&\approx > \sqcup \mathsf{I} \sqcup (\overline{R} \sqcap <)\\
&\approx \geq \sqcup (\overline{R} \sqcap <).
\end{aligned}
$$

To prove D.3.6(b): First, note that:

$$
\begin{aligned}
&< \circ <\\
&\approx\ < \circ < \sqcap \mathsf{E} && \text{(BA)}\\
&\approx\ < \circ < \sqcap (\overline{< \circ ?A \circ <} \sqcup < \circ ?A \circ <) && \text{(BA)}\\
&\approx\ (< \circ < \sqcap \overline{< \circ ?A \circ <}) \sqcup (< \circ < \sqcap < \circ ?A \circ <) && \text{(BA)}\\
&\approx\ (< \circ < \sqcap < \sqcap \overline{< \circ ?A \circ <}) \sqcup (< \circ < \sqcap < \circ ?A \circ <) && \text{(Tra)}\\
&\approx\ (< \circ < \sqcap < \circ ?\top \sqcap \overline{< \circ ?A \circ <}) \sqcup (< \circ < \sqcap < \circ ?A \circ <)\\
&\approx\ (< \circ < \sqcap < \circ ?\top \sqcap \overline{< \circ ?A \circ <}) \sqcup (< \circ ?A \circ <) && \text{(BA).}
\end{aligned}
$$

Now, $\overline{< \circ ?\neg\top \circ <} \approx \mathsf{E}$. Then:

$$
\begin{aligned}
&< \circ < \\
&\approx \; (< \circ < \sqcap < \circ ?\top \sqcap \overline{< \circ ?A \circ <}) \sqcup (< \circ ?A \sqcap \overline{< \circ ?\neg\top \circ <}) \circ < \qquad (\mathsf{BA}, \mathsf{RA}).
\end{aligned}
$$

Finally, by Proposition D.3.4(a),

$$\mathsf{until}(\top, \neg A) \approx < \circ ?\top \sqcap \overline{< \circ ?\neg\neg A \circ <}$$

and

$$\mathsf{until}(A, \top) \approx < \circ ?A \sqcap \overline{< \circ ?\neg\top \circ <}.$$

Then, we have the equivalence:

$$< \circ < \approx (< \circ < \sqcap \mathsf{until}(\top, \neg A)) \sqcup \mathsf{until}(A, \top) \circ <.$$

This completes the proof. ∎

Finally, we shall need one more equivalence, valid on the class of discrete flows of time.

Proposition D.3.7 *For any set terms A, B,*

$$\mathsf{until}(A, \top) \circ < \sqcap \overline{\mathsf{until}(A, B) \circ <} \approx (\mathsf{until}(A, \neg A) \sqcap < \circ ?\neg B \circ <) \circ <$$

is valid over the class of discrete flows of time.

Proof. Let $\mathfrak{M}$ be a flow of time and $a, b \in M$. We prove that $a, b \Vdash \mathsf{until}(A, \top) \circ <$ and $a, b \Vdash \overline{\mathsf{until}(A, B) \circ <}$ iff $a, b \Vdash (\mathsf{until}(A, \neg A) \sqcap < \circ ?\neg B \circ <) \circ <$.

To prove the implication from left to right, suppose $a, b \Vdash \mathsf{until}(A, \top) \circ <$ and $a, b \Vdash \overline{\mathsf{until}(A, B) \circ <}$. Hence, $\exists c \in M : s, u \Vdash \mathsf{until}(A, \top)$ and $u < t$ and $\forall d \in M$: if $a, d \Vdash \mathsf{until}(A, B)$, then $d \not< b$. Now, let the set $X ::= \{x \in M : a, x \Vdash \mathsf{until}(A, \top)$ and $x < b\}$. Since a is a lower bound of X, $X \neq \emptyset$, and $<$ is discrete, X has a first element e. We prove $a, e \Vdash \mathsf{until}(A, \neg A) \sqcap < \circ ?\neg B \circ <$. Since $e < b$, this gives us the result.

- To prove $a, e \Vdash \mathsf{until}(A, \neg A)$: we have $a < e$ and $e \Vdash A$ because $a, e \Vdash \mathsf{until}(A, \top)$; we have $(a, e) \Vdash \neg A$ because, otherwise, $\exists e' \in M : a < e' < e$ and $e' \Vdash A$). From this we could get $a, e' \Vdash \mathsf{until}(A, \top)$ and $e' < t$, contradicting the choice of e.

- To prove $a, e \Vdash < \circ ?\neg B \circ <$: suppose, for the sake of a contradiction, that $a, e \nVdash < \circ ?\neg B \circ <$. Since $a, e \Vdash \mathsf{until}(A, \top)$ gives us $a < e$ and $e \Vdash A$, we have $a, e \Vdash < \circ ?A$. Hence, $a, e \Vdash < \circ ?A \sqcap \overline{< \circ ?\neg B \circ <}$. Hence, by D.3.4(a), $a, e \Vdash \mathsf{until}(A, B)$. From this and $a, b \Vdash \overline{\mathsf{until}(A, B) \circ <}$, we have $e \not< b$, contradicting $e \in X$.

To prove the converse, suppose $a, b \Vdash (\mathsf{until}(A, \neg A) \sqcap < \circ ?\neg B \circ <) \circ <$. Hence, $\exists c \in M : a, c \Vdash \mathsf{until}(A, \neg A) \sqcap < \circ ?\neg B \circ <$ and $c < b$. Define $X ::= \{x \in M : a, x \Vdash \mathsf{until}(A, \neg A) \sqcap < \circ ?\neg B \circ <$ and $x < b\}$. Since X is lower bounded by a, $X \neq \emptyset$, and $<$ is discrete, X has a first element d.

- To prove $a, b \Vdash \mathsf{until}(A, \top) \circ <$: we have $a, e \Vdash \mathsf{until}(A, \top)$ because $s < w$ and $e \Vdash A$ come from $a, e \Vdash \mathsf{until}(A, \neg A)$, and $(a, e) \Vdash T$ is obviously true; we have $e < b$ because $e \in X$.

- To prove $a, b \Vdash \overline{\mathsf{until}(A, B) \circ <}$: suppose, for a contradiction, that $a, b \Vdash \mathsf{until}(A, B) \circ <$. Let $d \in T$ be such that $a < d < b$ and $d \Vdash A$, and $(a, d) \models B)$. Since e is the first element of X, we have $a < e < b$ and $e \Vdash A$, and $(a, e) \Vdash \neg A)$. Hence, $e \leq d$. But $a, e \Vdash < \circ ?\neg B \circ <$ gives us $\exists c \in M : a < c < e$ and $c \nVdash B)$, contradicting $(a, d) \Vdash B$.

This finishes the proof. ■

We are ready to present a detailed proof that complementation is definable. According to Definition D.3.3, we have to show that for every relation term R of LTL2, there exists a relation term S of LTL2, such that $\overline{[\![R]\!]_{\mathfrak{M}}} = [\![S]\!]_{\mathfrak{M}}$, for any discrete flow of time $\mathfrak{M}$. Since in some algebraic proofs we will be working in an expansion of LTL2 with complementation, we sometimes will think as proving that, in fact, complementation is *eliminable* from $\overline{\text{LTL2}}$, over discrete flows of time.

Theorem D.3.4 *Complementation is definable in* LTL2*, over the class of discrete flows of time.*

PROOF. By Theorem D.3.2, $R \approx \bigsqcup_{1 \leq i \leq n} R_i$, where $n \in \omega$ and each R_i, $1 \leq i \leq n$, is either a since, or a test, or an until relation term. Hence, given any flow of time

$\mathfrak{M}$, we have:

$$\overline{[\![R]\!]_{\mathfrak{M}}} \approx \overline{[\![\bigsqcup_{1\leq i\leq n} R_i]\!]_{\mathfrak{M}}}$$

$$\approx \bigsqcap_{1\leq i\leq n} \overline{[\![R_i]\!]_{\mathfrak{M}}}.$$

By Theorem D.3.3, we know that intersection is definable in LTL2 on the class of flows of time. So, it is enough to prove the complement of test, since, and until relation terms is definable over discrete flows of time. As we shall see below, complementation of a test is trivially definable. The case of until relation terms will be treated by induction. To define the complement of a since relation term, we will apply Lemma D.3.2 and the already established result that since relation terms are definable.

First, we show how to define the negation of a test, without any restrictions over the class of flows of time. Let $\mathfrak{M}$ be a model and a, b time points in M. Then, we have:

$$\begin{aligned}
a, b \Vdash \overline{?A} \quad & \text{iff} \quad a \neq b \text{ or } a \nVdash A \\
& \text{iff} \quad a \neq b; \text{ or } a \neq b \text{ or } a = b, \text{ and } a \nVdash A \\
& \text{iff} \quad a \neq b, \text{ or } a \neq b \text{ and } a \nVdash A, \text{ or } a = b \text{ and } a \nVdash A \\
& \text{iff} \quad a \neq b, \text{ or } a = b \text{ and } a \nVdash A \\
& \text{iff} \quad a < b \text{ or } a > b, \text{ or } a = b \text{ and } a \Vdash \neg A \\
& \text{iff} \quad a > b \text{ or } a, b \Vdash ?\neg A, \text{ or } a < b \\
& \text{iff} \quad a, b \Vdash \; > \sqcup \; ?\neg A \; \sqcup \; <.
\end{aligned}$$

Second, to eliminate complementation of until relation terms R, we apply induction on the number of $\circ$'s occurring in R. The base case is when $R = ?C \circ \mathsf{until}(A, B)$. We treat this case, based on the validities given previously. First, we have:

$$\begin{aligned}
?C \circ \mathsf{until}(A, B) \; & \approx \; ?C \circ (< \circ \, ?A \sqcap \overline{< \circ \, ?\neg B \circ <}) \qquad D.3.4(a) \\
& \approx \; (?C \circ < \circ \, ?A) \sqcap \overline{< \circ \, ?\neg B \circ <} \qquad D.3.5(b).
\end{aligned}$$

Hence, by BA, $\overline{?C \circ \mathsf{until}(A,B)} \approx \overline{?C \circ < \circ ?A} \sqcup < \circ ?\neg B \circ <$. Now, since $?C \circ < \circ ?A$ is a subrelation of $<$, we have:

$$\begin{aligned}
&\overline{?C \circ \mathsf{until}(A,B)} \\
&\approx \; \geq \sqcup (\overline{?C \circ < \circ ?A} \sqcap <) \sqcup < \circ ?\neg B \circ < && (D.3.6(a)) \\
&\approx \; \geq \sqcup (\overline{?C \circ < \circ ?A} \sqcap (?\neg C \circ < \sqcup ?C \circ <)) \sqcup < \circ ?\neg B \circ < \\
&\approx \; \geq \sqcup (\overline{?C \circ < \circ ?A} \sqcap ?\neg C \circ <) \\
&\quad \sqcup (\overline{?C \circ < \circ ?A} \sqcap ?C \circ <) \sqcup < \circ ?\neg B \circ < && (\mathsf{BA}) \\
&\approx \; \geq \; \sqcup \; ?\neg C \circ < \; \sqcup \; ?C \circ < \circ ?\neg A \; \sqcup \; < \circ ?\neg B \circ < && (\mathsf{BA}, D.3.3(b), \\
& && D.3.5(c)).
\end{aligned}$$

This completes the base case.

Now, suppose $\overline{S}$ is definable for the until relation term S, and take a relation term $R = S \circ \mathsf{until}(A,B)$. Then, since $R \sqsubseteq <$, by Proposition D.3.6(a), $\overline{R} \approx \; \geq \sqcup (< \sqcap \overline{R})$. The first disjunct is definable, by Lemma D.3.1. To the second, since $S \circ \mathsf{until}(A,B)$ is a subrelation of $S \circ <$ that is, in its turn, a subrelation of $<$, we have:

$$\begin{aligned}
< \sqcap \overline{R} &\approx (< \sqcap \overline{R} \sqcap \overline{S \circ <}) \sqcup (< \sqcap \overline{R} \sqcap S \circ <) && (\mathsf{BA}) \\
&\approx (< \sqcap \overline{S \circ <}) \sqcup (S \circ < \sqcap \overline{R}) && (\mathsf{BA}).
\end{aligned}$$

Using closure under intersection again, we see it is enough to show how to define $\overline{S \circ <}$ and $S \circ < \sqcap \overline{S \circ \mathsf{until}(A,B)}$. These are postponed to Lemmas D.3.6 and D.3.7.

Finally, let $R := ?C \circ \mathsf{since}(A_1,B_1) \ldots \circ \mathsf{since}(A_n,B_n)$ be a since relation term. Recall that $\mathsf{since}(A,B)^{\mathsf{T}} \approx ?A \circ \mathsf{until}(\top, B)$ and $?C^{\mathsf{T}} \approx ?C$, and $\mathsf{until}(A,B) \circ ?C \approx \mathsf{until}(A \wedge C, B)$. Hence, by RA, we have:

$$\begin{aligned}
R &\approx ?C^{\mathsf{T}\mathsf{T}} \circ \mathsf{since}(A_1,B_1)^{\mathsf{T}\mathsf{T}} \circ \ldots \circ \mathsf{since}(A_n,B_n)^{\mathsf{T}\mathsf{T}} \\
&\approx (\mathsf{since}(A_n,B_n)^{\mathsf{T}} \circ \ldots \circ \mathsf{since}(A_1,B_1)^{\mathsf{T}} \circ ?C^{\mathsf{T}})^{\mathsf{T}} \\
&\approx (?A_n \circ \mathsf{until}(A_{n-1},B_n) \circ \mathsf{until}(A_{n-2},B_{n-1}) \circ \ldots \circ \mathsf{until}(C,B_1))^{\mathsf{T}}.
\end{aligned}$$

Hence, by RA again, we obtain:

$$\overline{R} \approx \overline{?A_n \circ \mathsf{until}(A_{n-1},B_n) \ldots \circ \mathsf{until}(C,B_1)}^{\mathsf{T}},$$

that, in its turn, by Lemma D.3.2 and the fact that since relation terms are definable, is equivalent to a relation term without occurrences of neither reversion nor complementation. ∎

Lemma D.3.6 *Let R be an until relation term. Then $\overline{R \circ <}$ is definable.*

Proof. Given an until relation term R, we show, by induction on the number of $\circ$'s in R, how to construct a relation term equivalent to $\overline{R \circ <}$ on the class of all discrete flows of time.

The base case is $R = ?C \circ \mathsf{until}(A, B)$. First, we reduce the definability of $\overline{?C \circ \mathsf{until}(A, B) \circ <}$, to that of $< \sqcap \overline{\mathsf{until}(A, B) \circ <}$. Since $?C \circ \mathsf{until}(A, B) \circ <$ is a subrelation of $<$, we have:

$$
\begin{aligned}
&\overline{?C \circ \mathsf{until}(A, B) \circ <} \\
&\quad \approx \; \geq \sqcup (\overline{?C \circ \mathsf{until}(A, B) \circ <} \sqcap <) && (D.3.6(a)) \\
&\quad \approx \; \geq \sqcup (\overline{?C \circ \mathsf{until}(A, B) \circ <} \sqcap (?\neg C \circ < \sqcup\, ?C \circ <)) \\
&\quad \approx \; \geq \sqcup (\overline{?C \circ \mathsf{until}(A, B) \circ <} \sqcap ?\neg C \circ <) \\
&\qquad\; \sqcup (\overline{?C \circ \mathsf{until}(A, B) \circ <} \sqcap ?C \circ <) && (\mathsf{BA}) \\
&\quad \approx \; \geq \sqcup\, ?\neg C \circ < \sqcup (\overline{?C \circ \mathsf{until}(A, B) \circ <} \sqcap ?C \circ <) && (D.3.3(b), \mathsf{BA}) \\
&\quad \approx \; \geq \sqcup\, ?\neg C \circ < \sqcup\, ?C \circ (< \sqcap \overline{\mathsf{until}(A, B) \circ <}) && (D.3.5(d)).
\end{aligned}
$$

Now, we rewrite $< \sqcap \overline{\mathsf{until}(A, B) \circ <}$ as an union of until relation terms.

$$
\begin{aligned}
&< \sqcap \overline{\mathsf{until}(A, B) \circ <} \\
&\quad \approx \; (< \sqcap \mathsf{E}) \sqcap \overline{\mathsf{until}(A, B) \circ <} && (\mathsf{BA}) \\
&\quad \approx \; (< \sqcap (\overline{< \circ <} \sqcup < \circ <)) \sqcap \overline{\mathsf{until}(A, B) \circ <} && (\mathsf{BA}) \\
&\quad \approx \; (< \sqcap \overline{< \circ <} \sqcap \overline{\mathsf{until}(A, B) \circ <}) \\
&\qquad\; \sqcup (< \sqcap < \circ < \sqcap \overline{\mathsf{until}(A, B) \circ <}) && (\mathsf{BA}) \\
&\quad \approx \; (\mathsf{until}(\top, \bot) \;\sqcap\; \overline{\mathsf{until}(A, B) \circ <}) \\
&\qquad\; \sqcup (< \circ < \sqcap \overline{\mathsf{until}(A, B) \circ <}) && (D.3.1(a), \mathsf{Tra})
\end{aligned}
$$

$$\approx \quad \mathsf{until}(\top,\bot) \sqcup (< \circ < \sqcap \overline{\mathsf{until}(A,B) \circ <}) \qquad (\mathsf{BA})$$

$$\begin{aligned}\approx \quad & \mathsf{until}(\top,\bot) \\ & \sqcup ((< \circ < \sqcap \mathsf{until}(\top,\neg A)) \sqcup \mathsf{until}(A,\top) \circ <) \\ & \quad \sqcap \overline{\mathsf{until}(A,B) \circ <}) \qquad (D.3.1(b),\ D.3.6(b))\end{aligned}$$

$$\begin{aligned}\approx \quad & \mathsf{until}(\top,\bot) \\ & \sqcup (< \circ < \sqcap \mathsf{until}(\top,\neg A) \sqcap \overline{\mathsf{until}(A,B) \circ <}) \\ & \sqcup (\mathsf{until}(A,\top) \circ < \sqcap \overline{\mathsf{until}(A,B) \circ <}) \qquad (\mathsf{BA})\end{aligned}$$

$$\begin{aligned}\approx \quad & \mathsf{until}(\top,\bot) \sqcup (< \circ < \sqcap \mathsf{until}(\top,\neg A)) \\ & \sqcup (\mathsf{until}(A,\neg A) \sqcap < \circ ?\neg B \circ <) \circ < \qquad (D.3.1(c),\ D.3.7).\end{aligned}$$

Finally, we simplify the first disjunct until obtain the form desired. The third line below is obtained by rewriting $<$ and $\overline{< \circ <}$, respectively, as $< \circ ?\top$ and $\overline{< \circ <} \sqcap \overline{< \circ ?\neg\neg A \circ <}$.

$$\mathsf{until}(\top,\bot) \sqcup (< \circ < \sqcap \mathsf{until}(\top,\neg A))$$

$$\approx \quad (\overline{< \circ <} \sqcap <) \sqcup (< \circ < \sqcap \mathsf{until}(\top,\neg A)) \qquad (D.3.4(a))$$

$$\begin{aligned}\approx \quad & (\overline{< \circ <} \sqcap < \sqcap < \circ ?\top \sqcap \overline{< \circ ?\neg\neg A \circ <}) \\ & \sqcup (< \circ < \sqcap \mathsf{until}(\top,\neg A)) \qquad (D.3.4(a), \mathsf{BA})\end{aligned}$$

$$\approx \quad (\overline{< \circ <} \sqcap \mathsf{until}(\top,\neg A)) \sqcup (< \circ < \sqcap \mathsf{until}(\top,\neg A)) \qquad (D.3.4(a))$$

$$\approx \quad (\overline{< \circ <} \sqcup < \circ <) \sqcap \mathsf{until}(\top,\neg A) \qquad (\mathsf{BA})$$

$$\approx \quad \mathsf{E} \sqcap \mathsf{until}(\top,\neg A) \qquad (\mathsf{BA})$$

$$\approx \quad \mathsf{until}(\top,\neg A) \qquad (\mathsf{BA}).$$

Hence, we have:

$$< \sqcap \overline{\mathsf{until}(A,B) \circ <} \approx \mathsf{until}(\top,\neg A) \sqcup (\mathsf{until}(A,\neg A) \sqcap < \circ ?\neg B \circ <) \circ < \qquad (\mathrm{D.2})$$

on the class of all discrete linear orders. Discreteness was used when Proposition D.3.7 was applied. This concludes the base case.

Now, suppose $\overline{S \circ <}$ is equivalent to a $\mathsf{LTL2}$ relation term and let $R = S \circ \mathsf{until}(A, B)$. Since $\mathsf{until}(A, B)$ and $< \circ <$ are subrelations of $<$, we have:

$$\begin{aligned} R \circ < &= S \circ \mathsf{until}(A, B) \circ < \\ &\sqsubseteq S \circ < \circ < && (\mathsf{RA}) \\ &\sqsubseteq S \circ < && (\mathsf{RA}). \end{aligned}$$

Then, $\overline{S \circ <} \sqsubseteq \overline{R \circ <}$, and from this, we have:

$$\begin{aligned} \overline{R \circ <} &\approx (\overline{R \circ <}) \sqcap (\overline{S \circ <} \sqcup S \circ <) && (\mathsf{BA}) \\ &\approx (\overline{S \circ <} \sqcap \overline{R \circ <}) \sqcup (S \circ < \sqcap \overline{R \circ <}) && (\mathsf{BA}) \\ &\approx \overline{S \circ <} \sqcup (S \circ < \sqcap \overline{R \circ <}) && (\mathsf{BA}). \end{aligned}$$

By the IH, the first disjunct is definable. Now, we show how to find a $\mathsf{LTL2}$ relation term equivalent to $S \circ < \sqcap \overline{R \circ <}$, on the class of discrete flows of time.

Besides range, the following relation will be useful. For any until relation term R and model $\mathfrak{M}$, we define $\mathsf{max}(R)$ as the ternary relation on T given by:

$$\mathsf{max}(R) := \{(a, b, u) : \mathfrak{M} \models x < z < y \wedge xRz \wedge \neg\exists w(z < w < y \wedge xRw)\ [a, b, u]\}.$$

Note $\mathsf{max}(R)$ is the relation that holds between three time points a, b, c exactly when $a < b$ and c is an R of a in the interval (a, b), and it is the last such time point before b. Given an until relation term R, we use the expressions '$\mathsf{ran}(R)$' and '$\mathsf{max}(R)$' in three ways. To refer to the relations defined above; to refer to the respective first-order formulas $\exists x(xRy)$ and $x < z < y \wedge xRz \wedge \neg\exists w(z < w < y \wedge xRw)$, defining them; and, according to Lemmas D.3.3 and D.3.9, to the respective definitions of these relations in $\mathsf{LTL2}$.

By Theorem D.3.5 and Lemma D.3.9 from the next section, the IH, and (D.2), to define $S \circ < \sqcap \overline{R \circ <}$, it is sufficient to prove that:

$$\begin{aligned} \mathfrak{M}, a, b &\Vdash S \circ < \sqcap \overline{S \circ \mathsf{until}(A, B) \circ <} \\ &\text{iff} \\ \mathfrak{M} &\models \exists z \exists z'(x < z \leq z' < y \\ &\qquad \wedge\ xSz \wedge x\ \overline{S \circ <}\ z \\ &\qquad \wedge\ \mathsf{max}(S)(x, z', y) \\ &\qquad \wedge\ z'\ \overline{\mathsf{until}(A, B) \circ <}\ y \\ &\qquad \wedge\ (z = z' \vee z\ \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))\ z')\ [a, b], \end{aligned}$$

for any discrete model $\mathfrak{M}$ and time points $a, b \in M$.

Let $\mathfrak{M}$ be a model and $a, b \in M$. To prove the implication from left to right, suppose that $a, b \Vdash S \circ < \sqcap \overline{S \circ \mathsf{until}(A,B) \circ <}$. Hence, $\exists c \in M : a, c \Vdash S$ and $c < b$), and $a, b \nVdash S \circ \mathsf{until}(A,B) \circ <$. Let $X ::= \{x \in M : a, x \Vdash S \text{ and } x < b\}$. Since $X \neq \emptyset$, X is bounded below and above by a and b, respectively, and $<$ is discrete, X has a first and a last element. Let us call them d and d', respectively. First, we prove that:

$$\begin{aligned}\mathfrak{M} \models\ & x < z \leq z' < y \\ & \wedge x S z \wedge x\ \overline{S \circ <}\ z \\ & \wedge \mathsf{max}(S)(x, z', y) \\ & \wedge (z'\ \overline{\mathsf{until}(A,B) \circ <}\ y\ \ [a, b, v, v'].\end{aligned}$$

- $a < d \leq d' < b$ and $a, d \Vdash S$, and $a, d \nVdash S \circ <$, and $\mathsf{max}(S)(a, b, d')$ follow from the choice of d and d'.
- $d', b \nVdash \mathsf{until}(A,B) \circ <$ because, otherwise, $\exists c \in M : d', c \Vdash \mathsf{until}(A,B)$ and $c < b$). Since $d' \in X$, we have $a, d' \Vdash S$. Hence, $a, d' \Vdash S$ and $d', c \Vdash \mathsf{until}(A,B)$, and $c < b$. But this would imply $a, b \Vdash S \circ \mathsf{until}(A,B) \circ <$, contradicting the hypothesis.

Now, we prove, using Lemma D.3.10, that:

$$\mathfrak{M} \models z = z' \vee z\ \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))\ z'\ [d, d'].$$

Suppose $d \neq d'$. We prove that $d, d' \Vdash \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))$.

- $d < d'$ because $d \leq d'$ and $d \neq d'$.
- Now, assume, for a contradiction, that $\exists d'' \in M(d < d'' \leq d'$ and $d'' \Vdash \mathsf{ran}(R))$. Then there are time points $c, c', c'' \in M$ such that $c < d < c''$ and $c' < b$, and $d, d' \Vdash S$, and $d', d'' \Vdash \mathsf{until}(A,B)$. We have two cases. First, if $d' \leq c$, then $(c, d'') \Vdash \mathsf{until}(A,B)$ because $c'' \Vdash A$ and $(d, c'') \subseteq (c', c'') \Vdash B$. Since $a, d \Vdash S$, we have $a, b \Vdash S \circ \mathsf{until}(A,B) \circ <$, contradicting the hypothesis. Second, if $d < c' < c''$, we have $a < d < c' < d'$ and $a, d \Vdash S$, and $a, d' \Vdash S$, and $c' \Vdash \mathsf{ran}(S)$. Hence, by Lemma D.3.10, we conclude $a, c' \Vdash S$. Therefore, we have $a, c' \Vdash S$ and $c', c'' \Vdash \mathsf{until}(A,B)$, and $c'' < b$, implying

$a, b \Vdash S \circ \mathsf{until}(A, B) \circ <$, contradicting the hypothesis. Since both cases are contradictory, we have $d, d' \Vdash \neg\mathsf{ran}(S \circ \mathsf{until}(A, B))$.

Now, to prove the implication from right to left, suppose:

$$\begin{array}{rl} \mathfrak{M} \models \exists z z' (& x < z \leq z' < y \\ & \wedge x S z \wedge x\ \overline{S \circ <}\ z \\ & \wedge \mathsf{max}(S)(x, z', y) \\ & \wedge z'\ \overline{\mathsf{until}(A, B) \circ <}\ y \\ & \wedge (z = z' \vee z\ \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))\ z') \quad) \ [a, b]. \end{array}$$

Let $d, d' \in M$ be such that $a < d \leq d' < b$ and $a, d \Vdash S$, and $a, d \Vdash \overline{S \circ <}$, and $\mathsf{max}(S)(a, b, d')$, and $d', b \Vdash \overline{\mathsf{until}(A, B) \circ <}$, and $d, d' \Vdash \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))$ if $d \neq d'$. We have:

- $a, b \Vdash S \circ <$ because $a, d \Vdash S$ and $d < b$.

- $a, b \Vdash \overline{S \circ \mathsf{until}(A, B) \circ <}$ because, otherwise, we get a contradiction reasoning as follows. First, let $c, c' \in M$ be time points for which $a, c \Vdash S$ and $c, c' \Vdash \mathsf{until}(A, B)$, and $c' < t$). From this, we obtain $d \leq c < c' \leq d'$. In fact, $c < d$ contradicts $a, d \not\Vdash S \circ <$; $c' \leq c$ contradicts $c, c' \Vdash \mathsf{until}(A, B)$; and if we suppose $d' < c'$, since $c \leq d'$ (because $d' < c$ contradicts $\mathsf{max}(S)(a, b, d')$), we would have $c' \Vdash A$ and $(d', c') \Vdash B$ (from $(d', c') \subseteq (c, c') \Vdash B$), implying that $(d', c') \Vdash \mathsf{until}(A, B)$, a contradiction with $d', b \not\Vdash \mathsf{until}(A, B) \circ <$. Now, we have $d < d'$ implying $d, d' \Vdash \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))$. But, $a, c' \Vdash S \circ \mathsf{until}(A, B)$ implies $c' \Vdash \mathsf{ran}(R)$ and since we proved $d < c' \leq d'$, we get a contradiction.

This completes the proof. ∎

Lemma D.3.7 *Let R be an until relation term. Then $R \circ < \sqcap \overline{R \circ \mathsf{until}(A, B)}$ is definable.*

Proof. Given R, we prove that there exists an until relation term S such that:

$$[\![R \circ <]\!]_{\mathfrak{M}} \cap \overline{[\![R \circ \mathsf{until}(A, B)]\!]_{\mathfrak{M}}} = [\![S]\!]_{\mathfrak{M}},$$

for any discrete flow of time $\mathfrak{M}$. We evoke Theorem D.3.5, by showing that:

$$a, b \Vdash R \circ < \sqcap \overline{R \circ \mathsf{until}(A, B)}$$
$$\text{iff}$$
$$\mathfrak{M} \models \exists z (x < z < y \wedge \mathsf{max}(R)(x, y, z) \wedge z \overline{\mathsf{until}(A, B)} y)\ [a, b].$$

for any discrete flow of time $\mathfrak{M}$ and time points $a, b \in M$. Since $\overline{\mathsf{until}(A, B)}$ is definable, by Lemma D.3.9, we get the result.

To prove the implication from left to right. Suppose $\exists c \in M : a, c \Vdash R$ and $c < b)$ and $\forall d \in M$: if $d, b \Vdash \mathsf{until}(A, B)$, then $a, d \nVdash R)$. Let $X = \{x \in M : a, x \Vdash R \text{ and } x < b\}$. Since $X \neq \emptyset$, b is an upper bound of X, and $<$ is discrete, X has a last element e. We prove:

$$\mathfrak{M} \models (x < z < y \wedge \mathsf{max}(R)(x, y, z) \wedge z\overline{\mathsf{until}(A, B)}y)\ [a, b, e].$$

- To prove $a < e < b$: we have $a < e$ because $a, e \Vdash R$ and R is an until relation term; we have $e < b$ because $e \in X$.

- To prove $\mathsf{max}(R)(a, b, e)$, we just need to prove $(e, b) \nVdash R$. Suppose, for a contradiction, that $\exists e' \in M : e < e' < b$ and $a, e' \Vdash R)$, hence $e' \in X$ and $e < e'$, contradicting e is the last element of X.

- To prove $e, b \nVdash \mathsf{until}(A, B)$: suppose, for a contradiction, $e, b \Vdash \mathsf{until}(A, B)$. Hence, by hypothesis, we obtain $a, e \nVdash R$, contradicting $e \in X$.

To prove the implication from right to left, suppose there is a time point e such that $a < e < b$ and $\mathsf{max}(R)(a, b, e)$, and $e, b \nVdash \mathsf{until}(A, B)$.

- To prove $a, b \Vdash R \circ <$: we have $a, e \Vdash R$ because $\mathsf{max}(R)(a, b, e)$; $e < b$ is given.

- To prove $a, b \nVdash R \circ \mathsf{until}(A, B)$: suppose, for a contradiction, that $\exists c \in M$: $a, c \Vdash R$ and $c, b \Vdash \mathsf{until}(A, B)$.

  From $\mathsf{max}(R)(a, b, e)$ we have $c \leq e$. From $c, b \Vdash \mathsf{until}(A, B)$, we have $b \Vdash A$ and $(c, b) \Vdash B$. As $(e, b) \subseteq (c, b)$ we conclude $(e, b) \Vdash B$, implying $e, b \Vdash \mathsf{until}(A, B)$, a contradiction with the hypothesis.

The proof is complete. ■

D.3.3 Translating from $+\exists$FOL(LTL2) to LTL2 via graphs

Now, we prove the two facts used in the proof that complementation is definable. First, we show how the special first-order formulas used there can be rewritten as terms of LTL2. We shall take a playful approach using special relational graphs.

Definition D.3.7 (1) Let $\textsc{IVar} = \{x_i : i \in \omega\}$ be a set of individual variables, typically denoted by x, y, z. The *formulas* of the *positive existential first-order language over relation terms*, $+\exists\mathsf{FOL}(\mathsf{LTL2})$, typically denoted by φ, ψ, are generated by the following grammar:

$$\varphi \quad ::= \quad xRy \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x\varphi,$$

where R is a relation term.
(2) Formulas are interpreted on models $\mathfrak{M} = (M, <^{\mathfrak{M}}, P_i^{\mathfrak{M}})_{i\in\omega}$ as in the usual first-order logic with the proviso that given an assignment β to the individual variables in M:

$$\mathfrak{M}, \beta \models xRy \text{ iff } \mathfrak{M}, \beta x, \beta y \Vdash R.$$

From Theorems C.5.3 and C.5.4, considering $\textsc{RVar}$ as the set of relational terms of $\mathsf{LTL2}$, the formulas of $+\exists\mathsf{FOL}(\mathsf{LTL2})$ can be seeing as graphs whose arcs are labeled with relational terms. In this way, most of the syntactical machinery developed in Chapter C can be applied directly to formulas viewed as graphs. For instance, by applying the SNF Theorem, we have:

Corollary D.3.2 *Every graph of* $+\exists\mathsf{FOL}(\mathsf{LTL2})$ *is equivalent to a graph whose arcs are labeled with terms of the form* $\mathsf{until}(A, B)$, $?A$, *or* $\mathsf{since}(A, B)$.

In what follows, we denote set terms of $\mathsf{LTL2}$ and sets of arcs of graphs ambiguously by A, B. The graphs of $+\mathsf{RG}$ and that of $+\exists\mathsf{FOL}(\mathsf{LTL2})$ have some important semantical differences. Maybe, the most important one is that, due to the presence of negation inside the set terms occurring at the labels of graphs, there are graphs of $+\exists\mathsf{FOL}(\mathsf{LTL2})$ that are *inconsistent.*

Definition D.3.8 (1) Let G be a graph of $+\exists\mathsf{FOL}(\mathsf{LTL2})$. We say that G is *empty* if $[\![G]\!]_{\mathfrak{M}} = \emptyset$, for every model $\mathfrak{M}$.
(2) We, ambiguously, denote all the empty graphs by $\emptyset$.

For instance, the graphs $G_1 = (\{x\}, x?\neg\top x, x, x)$ and $G_2 = (\{x, y\}, x?\neg\top y, x, y)$, whose arcs are labeled with a test, are empty.

Given a graph G whose components are $C_i, 1 \leq i \leq m$, we have that the meaning of G on a model $\mathfrak{M}$ is $[\![G]\!]_{\mathfrak{M}} = \bigcup_{1\leq i\leq m} [\![C_i]\!]_{\mathfrak{M}}$. So, empty components do not change the meaning of a graph and may be deleted or included when necessary. In other words, we are saying that the following rule is obviously sound in both directions, for any graph G:

$$\mathsf{Empt)}\ \frac{G+\emptyset}{G}$$

Now, we go via series of refinements of Corollary D.3.2, to prove that, over the class of flows of time, every graph of $+\exists\mathsf{FOL}(\mathsf{LTL2})$ is equivalent to a relation term of LTL2, in DDNF.

Lemma D.3.8 *Let G be a graph of $+\exists\mathsf{FOL}(\mathsf{LTL2})$. Then, over linear flows of time, G is equivalent to a graph having just components of the following forms:*

- *$x?Ay$, where x, y are the same node;*
- *$x?Cx + xR_1z_1 + z_1R_2z_2 + \cdots + z_nR_ny + y?Dy$, where $1 \leq n$, $x, y, z_1, \ldots, z_n$ are distinct nodes, and each R_i, $1 \leq i \leq n$, is of the form $\mathsf{since}(A, B)$ or each R_i, $1 \leq i \leq n$, is of the form $\mathsf{until}(A, B)$.*

PROOF. Let $G = (N_i, A_i, x_i, y_i)_{i\in\{1,\ldots,m\}}$ be a graph of $+\exists\mathsf{FOL}(\mathsf{LTL2})$. We shall show how to construct a new equivalent graph H, in a series of steps as follows. For each component C_i of G, we denote by:

– $N_i = \{u^i_1, u^i_2, \ldots, u^i_{n_i}\}$ the set of n_i nodes of C_i;

– $\Lambda_i = \{\lambda_1, \ldots, \lambda_{n_i!}\}$ the set of all $n_i!$ linear orderings of N_i;

– W_i the set of all 2^{n_i-1} words of n_i-1 letters over the alphabet $\{?\top, \mathsf{until}(\top, \top)\}$. In what follows, we denote $?\top$ by I and $\mathsf{until}(\top, \top)$ by $<$.

To each component $C_i = (N_i, A_i, x_i, y_i)$ of G, each linear order λ_j in Λ_i, and word $w_k = R_1R_2 \ldots R_{n_i-1}$ in W_i, we assign a new component C_{ijk} as follows:

- The set of nodes of C_{ijk} is $N_{ijk} = N_i$;

- The set of arcs of C_{ijk} is $A_{ijk} = A_i \cup B$, where B is a set of *new arcs* obtained from λ_j and w_k as follows: for each pair u, v of nodes in N_i, we put the arc uR_pv in B iff u the node that occupies position p in the order λ_j and v is the immediate successor of u.

- The distinguished nodes of C_{ijk} are $x_{ijk} = x_i$ and $y_{ijk} = y_i$.

Hence, C_{ijk} is the component obtained from C_i, λ_j, and w_k by considering N_i ordered according to λ_j, constructing a path labeled with the elements of w_k in the order in which they are given in w_k, from the first element of N_i to its last element, and, finally, to add to this path the arcs in A_i. Note that each component obtained by this process is connected, in the sense of graph theory.

Now, we take:

$$G_1 = \bigsqcup_{ijk} G_{ijk}.$$

First, we show that $[\![G]\!]_{\mathfrak{M}} = [\![G_1]\!]_{\mathfrak{M}}$, for every linear flow of time $\mathfrak{M}$. So, let $\mathfrak{M}$ be a flow of time and $a, b \in M$. To prove the inclusion from left to right, suppose $(a, b) \in [\![G]\!]_{\mathfrak{M}}$ is witnessed by $C_i = (N_i, A_i, x_i, y_i)$ and $g : N_i \to M$. Let $\sim$ be the equivalence relation on N_i induced by g, in the usual way: for all $u, v \in N_i$, we have $u \sim v$ iff $gu = gv$. Denote by $[N_i] = \{[u] : u \in N_i\}$ the quotient set of N_i by $\sim$ and, based on $[N_i]$, define a total ordering λ_j of N_i by putting, for all $u, v \in N_i$:

- $u\lambda_j v$ when $[u] = [v]$ and u precedes v in the usual ordering of nodes;

- $u\lambda_j v$ when $[u] \neq [v]$ and $gu <^{\mathfrak{M}} gv$.

Now, let w_k be the word $R_1R_2 \ldots R_{n_i-1}$, over the alphabet $\{\mathsf{I}, <\}$, defined from λ_j by taking, for each pair u, v of nodes in N_i, that are consecutive in λ_j:

$$R_i ::= \begin{cases} \mathsf{I} & \text{if } [u] = [v] \\ < & \text{if } [u] \neq [v] \text{ and } u\lambda_j v. \end{cases}$$

We are ready to prove that $(a, b) \in [\![G_1]\!]_{\mathfrak{M}}$ is witnessed by C_{ijk} and g. First, we have $gx_{ijk} = gx_i = a$ and $gy_{ijk} = gy_i = b$. Now, let $uRv \in A_{ijk}$. We have two cases. If $uRv \in A_i$, then $gu, gv \Vdash R$, by hypothesis. If $uRv \notin A_i$, we consider two sub-cases. When R is I, we have $[u] = [v]$ and $u \sim v$ implies $gu = gv$. Hence, we have $gu, gv \Vdash \mathsf{I}$.

Otherwise, R is $<$ and, in this case, $[u] \neq [v]$ and $u\lambda_j v$. So, $gu <^{\mathfrak{M}} gv$, and we have $gu, gv \Vdash <$.

To prove the converse inclusion, suppose $(a, b) \in [\![G_1]\!]_{\mathfrak{M}}$ is witnessed by C_{ijk} and $g : N_{ijk} \to M$, for some i, j, k as above. Since $gx_i = gx_{ijk} = a$, $gy_i = gy_{ijk} = b$, and $A_i \subseteq A_{ijk}$, the result is immediate.

Now, we simplify G_1 to obtain a more suitable graph H. During the process, we shall make strong use of the fact that the graphs of $+\exists\mathsf{FOL}(\mathsf{LTL2})$ are being interpreted over linear flows of time.

Let vRu be an arc of N_i. We say that vRu is *contrary* when $u\lambda_j v$. Every component C_{ijk} of G_1 is equivalent to a component without contrary arcs. To prove this we apply rules in Table D.6. Note that these rules are sound in both directions and that their application does not change any new arc in B.

$$\mathsf{ConU}\quad \frac{N, A + v\mathsf{until}(A, B)u, x, y}{N, A + u?Au + u\mathsf{since}(\top, B)v, x, y}$$

$$\mathsf{ConT}\quad \frac{N, A + v?Au, x, y}{N, A + u?Av, x, y}$$

$$\mathsf{ConS}\quad \frac{N, A + v\mathsf{since}(A, B)u, x, y}{N, A + u?Au + u\mathsf{until}(\top, B)v, x, y}$$

Table D.6: Rules for eliminating contrary arcs.

By applying the rules in Table D.6 to each component of G_1 we obtain a graph G_2 without occurrences of contrary arcs.

Let uRv be an arc of G_2. We say that uRv is a *short cut* when there is a node w such that $u\lambda_j w$ and $w\lambda_j v$. Every component C of G_2 is equivalent to a component without short cuts. To prove this we apply rules in Table D.7 in the following way. First, assume $C = (N, A, x, y)$ has just one short cut uRv. Since C does not have contrary arcs, we have $u\lambda_j v$. Let $w_1, \ldots, w_p$ be all the arcs of N in between u and v, according to λ_j and let uR_0w_1, $w_1R_1w_2$, …, w_pR_pv, associated to the subordering $u\lambda_j w_1 \lambda_j w_2 \ldots w_{p-1}\lambda_j w_p \lambda_j v$, be the new arcs introduced when constructing G_1.

Now, assume that R is $?A$. We have two sub-cases. If R_q is I for every q, $1 \leq q \leq p$, then apply rule TTes, of Table D.10, $p-1$ times to replace the occurrences of $u, w_1, w_2, \ldots, w_p$ by occurrences of v eliminating the short cut $u?Av$. If there is a q, $1 \leq q \leq p$, for which R_q is $<$, apply the rule SCutL replacing C by $\emptyset$ and eliminating the short cut $u?Av$. Now, assume R is $\mathsf{until}(A, B)$. We also have two cases. If R_q is $<$ for every q, $1 \leq q \leq p$, then apply rule SCutU to eliminate the short cut $u\mathsf{until}(A, B)v$. If there is a q, $1 \leq q \leq p$, for which R_q is I, apply rule TTes, of Table D.10, as many times as necessary, to eliminate all occurrences of I, labeling consecutive arcs whose nodes are in $v, w_1, \ldots, w_p, v$ and reduces to the previous case or end up with an arc $u?Au$, eliminating the short cut. Finally, assume R is $\mathsf{since}(A, B)$. In this case, we apply rule SCutS to eliminate the short cut $u\mathsf{since}(A, B)v$. Note that these rules are sound in both directions and that their application does not introduce neither contrary nor new short cut arcs.

$$\mathsf{SCutU} \quad \frac{N, A + u < w_1 + w_1 < w_2 + \cdots + w_n < v + u\mathsf{until}(A, B)v, x, y}{N, A + u\mathsf{until}(B, B)w_1 + w_1\mathsf{until}(B, B)w_2 + \cdots + w_n\mathsf{until}(A, B)v, x, y}$$

$$\mathsf{SCutL} \quad \frac{H + (N, A + u?Av + u \leq w_1 \cdots \leq w_{i-1} < w_i \leq w_{i+1} \cdots \leq w_p \leq v, x, y)}{H + \emptyset}$$

$$\mathsf{SCutS} \quad \frac{H + (N, A + u < v + u\mathsf{since}(A, B)v, x, y)}{H + \emptyset}$$

Table D.7: Rules for eliminating short cuts.

Now we proceed by induction on the number of short cuts in G_2, applying procedure above to each component of G_2 until obtain a graph G_3 without occurrences neither of contrary nor short cut arcs.

Let uRv and $u'Sv'$ be arcs of G_3. We say that uRv and $u'Sv'$ are *parallel* when $u = u'$ and $v = v'$. Every component C of G_3 is equivalent to a component without parallel arcs. To prove this, we apply rules in Table D.8. Observe that these rules are sound in both directions and that their application does not introduce neither contrary nor short cut nor parallel arcs.

ParUU	$\dfrac{N, A + u\mathsf{until}(A, B)v + u\mathsf{until}(C, D)v, x, y}{N, A + u\mathsf{until}(A \wedge C, B \wedge D)v, x, y}$
ParUT	$\dfrac{H + (N, A + u\mathsf{until}(A, B)v + u?Cv, x, y)}{H + \emptyset}$
ParUS	$\dfrac{H + (N, A + u\mathsf{until}(A, B)v + u\mathsf{since}(C, D)v, x, y)}{H + \emptyset}$
ParTT	$\dfrac{N, A + u?Av + u?Bv, x, y}{N, A + u?(A \wedge B)v, x, y}$
ParTS	$\dfrac{H + (N, A + u?Av + u\mathsf{since}(A, B)v, x, y)}{H + \emptyset}$
ParSS	$\dfrac{N, A + u\ \mathsf{since}(A, B)\ v + u\ \mathsf{since}(C, D)\ v, x, y}{N, A + u\ \mathsf{since}(A \wedge C, B \wedge D)\ v, x, y}$

Table D.8: Rules for eliminating parallel arcs.

By applying the rules in Table D.8 to each component of G_3 we obtain a graph G_4 without any occurrences neither of contrary nor short cut nor parallel arcs.

Let C be a component of G_4 and $u?Au$ be a test of C. We say that $u?Au$ is *out of place* when u is not the first element of N_i, according to the ordering of λ_j of N_i. Every component of G_4 is equivalent to a component without tests out of place. To prove this we apply the rules in Table D.9. Observe that these rules are sound in both directions and that their application does not introduce neither contrary nor short cut, nor parallel arcs, nor tests out of place.

By applying the rules in Table D.9 to each component of G_4 we obtain a graph G_5 without occurrences either of contrary or short cut or parallel arcs or tests out of place. Since the application of each one of the rules in Tables D.6, D.7, D.8, and

$$\text{Out1} \quad \frac{N, A + u\mathsf{until}(A, B)v + v?Cv, x, y}{N, A + u\mathsf{until}(A \wedge C, B)v, x, y}$$

$$\text{Out2} \quad \frac{N, A + u\mathsf{since}(A, B)v + v?Av, x, y)}{N, A + u\mathsf{since}(A \wedge C, B)v, x, y}$$

Table D.9: Rules for eliminating tests out of place.

D.9 preserves connectedness, each component of a such graph is empty or a path, where a component C is a *path* if it has the following form:

$$C = (\{u_0, \ldots, u_n\}, \{u_0?Au_0, u_0R_1u_1, u_1R_2u_2, \ldots, u_{n-2}R_{n-1}u_{n-1}, u_{n-1}R_nu_n\}, x, y),$$

where $u_0, u_1, \ldots, u_n$ are distinct arcs and $R_1, R_2, \ldots, R_n$ are all until atoms. Every component of G_5 is empty or a path. To prove this, we apply rules in Table D.10.

$$\text{TUnt} \quad \frac{H + (N, A + u\ \mathsf{until}(A, B)\ u, x, y)}{H + \emptyset}$$

$$\text{TTes} \quad \frac{N, A + u\ ?A\ v, x, y}{\mathtt{ren}_u^v N, \mathtt{ren}_u^v A + \mathtt{ren}_u^v u\ ?A\ \mathtt{ren}_u^v v, \mathtt{ren}_u^v x, \mathtt{ren}_u^v y}$$

$$\text{TSin} \quad \frac{H + (N, A + u\ \mathsf{since}(A, B)\ u, x, y)}{H + \emptyset}$$

$$\text{LSin} \quad \frac{H + (N, A + u\ \mathsf{since}(A, B)\ v, x, y)}{H + \emptyset}$$

Table D.10: Rules for eliminating useless untils, tests, and sinces.

Suppose C is a component of G_5 such that $C \neq \emptyset$. Let uRv be an arc of C. Note that $u\lambda_jv$, because G_5 has no contrary arcs. We have two cases. If $u = v$, then u is the first element of N_i, according to λ_j, and R is $?A$, because, otherwise, either G_5 contains arcs $u\mathsf{until}(A,B)u$ or arcs $u\mathsf{since}(A,B)u$ and, by rules TUnt and TSin it should be empty, or G_5 should contains arcs out of place. If $u \neq v$, then v is the immediate successor of u, according to λ_j, because, otherwise, G_5 should have a short cut, contradicting its definition. Hence, G_5 has the required format. Now, let us examine which possibilities for R can exist when u is not the first element of N_i. Since G_5 does not have a test out of place, we just have two possibilities. If R were of the form $\mathsf{since}(A,B)$, we would have a contradiction with $u\lambda_j v$. So, we conclude R is of the form $\mathsf{until}(A,B)$ and C is a path. Now, we come up with a graph G_6 having just paths as components.

Now, let $P = (N, A, x, y)$ be a path of G_6. We have dom and ran in the language. We also have the rules ConU —which replaces until in since— and Out2, which eliminates occurrences of tests out of place associated to since. So, an exhaustive analysis of cases shows that every path is equivalent to a component having one of the following forms:

- $(\{x, y, u_1, \ldots, u_{n-1}\}, \{x?Ax, xR_1u_1, \ldots, u_{n-1}R_ny\}, x, y)$, where every R_i, $1 \leq i \leq n$, is of the form $\mathsf{until}(A,B)$.
- $(\{x\}, \{x?Ax\}, x, x)$.
- $(\{x, y, u_1, \ldots, u_{n-1}\}, \{x?Ax, xR_1u_1, \ldots, u_{n-1}R_ny\}, x, y)$, where every R_i, $1 \leq i \leq n$, is of the form $\mathsf{since}(A,B)$.

In fact, note that paths are just of two types: those for which $x = y$ and those having distinct distinguished nodes. In the first case we conclude that the path is equivalent to a test. In the second case, we have that it is equivalent to an until or to a since *directed path*, as required. ∎

Now, we can prove the main result of this section:

Theorem D.3.5 *Let φ be a formula of* $+\exists\mathsf{FOL}(\mathsf{LTL2})$ *having at most two free variables. Then there is a relational term R of* LTL2 *such that:*

$$[\![\varphi]\!]_{\mathfrak{M}} = [\![R]\!]_{\mathfrak{M}},$$

for every flow of time $\mathfrak{M}$.

PROOF. The proof of Lemma above can be converted in a procedure for rewriting a formula of $+\exists\mathsf{FOL}(\mathsf{LTL2})$ into a relation term in directed disjunctive normal form. ■

D.3.4 max is definable in $+\exists\mathsf{FOL}(\mathsf{LTL2})$

Finally, we use Theorem D.3.5 to prove that max is definable and finish the proof that complementation is definable.

Lemma D.3.9 *Let R be an until relation term. Then $\mathsf{max}(R)$ is definable as a positive existential formula.*

PROOF. Let R be an until relation term. We show, by induction on the number of $\circ$'s in R, how to construct a positive existential first-order formula $\alpha(x, y, z)$ over relation terms, having the free variables x, y, z, such that:

$$\mathsf{max}(R)(a, b, c) \text{ iff } \mathfrak{M} \models \alpha(x, y, z)\ [a, b, c],$$

for any discrete flow of time $\mathfrak{M}$ and time points $a, b, c \in M$.

The base case is $R = ?C \circ \mathsf{until}(A, B)$. We prove, without using discreteness, that $\mathsf{max}(R)(a, b, c)$ iff

$$\mathfrak{M} \models x < z < y \wedge x\ ?C \circ \mathsf{until}(A, B)\ z \wedge (z\ ?\neg B\ z \vee z\ \overline{\mathsf{until}(A, B) \circ <}\ y)\ [a, b, c]. \tag{D.3}$$

To prove the implication from left to right, suppose $\mathsf{max}(?C \circ \mathsf{until}(A, B))(a, b, c)$. Hence, $a < c < b$ and $a, c \Vdash ?C \circ \mathsf{until}(A, B)$, and $\forall d \in M$: if $c < d < b$, then $a, d \nVdash ?C \circ \mathsf{until}(A, B))$. To prove (D.3), we just need to prove that $c \nVdash \neg B$ implies $u, t \Vdash \overline{\mathsf{until}(A, B) \circ <}$. So, suppose $c \Vdash B$ and, for a contradiction, that $\exists d \in M$: $c, d \Vdash \mathsf{until}(A, B)$ and $d < b$). Since $s < v$ and $v \Vdash A$, and $(a, d) \Vdash B$ (because $(a, d) = (a, c) \cup \{u\} \cup (c, d)$ and $(a, c), u, (c, d) \Vdash B$), we have $a, d \Vdash \mathsf{until}(A, B)$. But this, together with $s \Vdash C$, gives $a, d \Vdash ?C \circ \mathsf{until}(A, B)$, a contradiction.

To prove that (D.3) implies $\mathsf{max}(?C \circ \mathsf{until}(A, B))(a, b, c)$, suppose $a < c < b$ and $s \Vdash C$, and $a, c \Vdash \mathsf{until}(A, B)$, and $u, t \Vdash \overline{\mathsf{until}(A, B) \circ <}$ whenever $u \Vdash B$. To prove $\mathsf{max}(?C \circ \mathsf{until}(A, B))(a, b, c)$ holds, we just need to prove $a, d \nVdash ?C \circ \mathsf{until}(A, B)$ for all $d \in (c, b)$. Suppose, for a contradiction, that $\exists d \in M$: $c < d < b$ and $a, d \Vdash$ $?C \circ \mathsf{until}(A, B))$. We have, $(c, d) \Vdash B$ because $(a, d) \Vdash B$ and $(c, d) \subset (a, d)$. This, together with $u < v$ and $v \Vdash A$, gives $c, d \Vdash \mathsf{until}(A, B)$, that, in its turn, together with $d < b$ gives $u, t \Vdash \mathsf{until}(A, B) \circ <$, a contradiction.

This concludes the base case.

Let S be an until relation term for which there is a positive existential first-order formula $\beta(x, y, z)$ over relation terms s.t. $\mathsf{max}(S)(a, b, c)$ iff $\mathfrak{M} \models \beta(x, y, z)\ [a, b, c]$, for any discrete flow of time $\mathfrak{M}$ and time points $a, b, c \in M$.

As before, we use $\mathsf{max}(T)(x, y, z)$ to denote the formula defining the relation $\mathsf{max}(T)$, for any until relation term T.

Let $R = S \circ \mathsf{until}(A, B)$. We prove that, for any discrete flow of time $\mathfrak{M}$ and $a, b, c \in M$, $\mathsf{max}(R)(a, b, c)$ iff

$$\begin{aligned}\mathfrak{M} \models\ & \exists z'(x < z' < z < y \wedge \mathsf{max}(S)(x, y, z') \wedge \mathsf{max}(\mathsf{until}(A, B))(z', y, z)) \\ & \vee \\ & \exists z'(x < z \leq z' < y \wedge \mathsf{max}(S)(x, y, z') \wedge z'\ \overline{\mathsf{until}(A, B) \circ <}\ y \\ & \wedge\, xRz \wedge (z \approx z' \vee z\ \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))z'))\ [a, b, c]. \qquad \text{(D.4)}\end{aligned}$$

By the IH, $\mathsf{max}(S)$ is definable; by the base case $\mathsf{max}(\mathsf{until}(A, B))$ is definable; we already know that $< \cap\, \overline{\mathsf{until}(A, B) \circ <}$ is definable; and, by Lemma D.3.3, $\mathsf{ran}(R)$ is definable.

To prove the implication from left to right, suppose $\mathsf{max}(S \circ \mathsf{until}(A, B))(a, b, c)$. Hence $a < c < b$ and $a, c \Vdash S \circ \mathsf{until}(A, B)$, and $\forall d \in M$: if $u < d < b$, then $a, d \nVdash S \circ \mathsf{until}(A, B)$. Now, let $X = \{x \in M : s, x \Vdash S$ and $x, c \Vdash \mathsf{until}(A, B)\}$ and $Y = \{y \in T : a, y \Vdash S$ and $y < b\}$. Since, by hypothesis, $X, Y \neq \emptyset$ and b is an upper bound for X and Y, and $<$ is discrete, both X and Y has a last element. Let us call them c' and e, respectively.

Observe that, since $c' < c < b$ and $a, c' \Vdash S$, we have $c' \leq e$. Hence, we have two cases. When, $c' = e$, we have:

- $a < e < c < b$ is given;
- $\mathsf{max}(S)(a, b, e)$ because e is the last element of Y and $c' = e$;
- $\mathsf{max}(\mathsf{until}(A, B))(e, b, c)$ because:

– $e < c < b$ is given;

– $e, c \Vdash \mathsf{until}(A, B)$ because $c' \in X$;

– Admitting, for a contradiction, that $\exists d \in M : c < d < b$ and $e, d \Vdash \mathsf{until}(A, B)$, we would have $a, d \Vdash S \circ \mathsf{until}(A, B)$ and $c' < d$, contradicting the fact that c' is the last element of X.

Hence, we have:

$$\mathfrak{M} \models \exists z'(x < z' < z < y \wedge \mathsf{max}(S)(x, y, z') \wedge \mathsf{max}(\mathsf{until}(A, B))(z', y, z)) \; [a, b, c]$$

and (D.4) follows at once.

When $c' < e$, first, observe that $c \leq e$ because, otherwise, $e < c$ and $c \Vdash A$, and $(e, c) \Vdash B$ (because $(c', c) \Vdash B$ and $(e, c) \subset (c', c)$), gives $e, c \Vdash \mathsf{until}(A, B)$. But this, together with $a, e \Vdash S$ gives $a, c \Vdash S \circ \mathsf{until}(A, B)$, contradicting the fact that c' is the last element of X. Now, we have:

- $a < c \leq e < b$ because $a < c < b$, $c \leq e$, and $e < b$ since $e \in Y$.
- $\mathsf{max}(S)(a, b, e)$ because e is the last element of Y.
- $e, b \not\Vdash \mathsf{until}(A, B) \circ <$ because, otherwise, $\exists d \in M : e, d \Vdash \mathsf{until}(A, B)$ and $d < b$ and hence, $a, d \Vdash S \circ \mathsf{until}(A, B)$, and $c < d < b$, contradicting $\mathsf{max}(S \circ \mathsf{until}(A, B))(a, b, c)$.
- $a, c \Vdash R$ is given.
- If $c \neq e$, we prove $c, e \Vdash \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))$ as follows:

  – $c < e$ because $c \leq e$ and $c \neq e$.

  – Assume, for a contradiction, that there exists $d \in (c, e)$ for which $d \Vdash \mathsf{ran}(R)$. Let $d', d'' \in M$ be such that $d'', d' \Vdash S$ and $d', d \Vdash \mathsf{until}(A, B)$. Observe that $c < d \leq w$ and $d'' < d' < d$. Hence, we have two cases. If $d' < c$, then we have $c', d \Vdash \mathsf{until}(A, B)$ because $c' < d$ is given; $d \Vdash A$ comes from $d', d \Vdash \mathsf{until}(A, B)$; and $(c', d) \Vdash B$ comes from $(c', c) \Vdash B$ and $(d', d) \Vdash B$, and $d' < c$. Hence, $a, d \Vdash S \circ \mathsf{until}(A, B)$ and $c < d$, contradicting $\mathsf{max}(R)(a, b, c)$. If $c \leq d'$, then we have: $a, c' \Vdash S$; $a, e \Vdash S$; $a < c' < d' < e$; and $d' \Vdash \mathsf{ran}(S)$.

Hence, by Lemma D.3.10, $a, d' \Vdash S$ implying $a, d \Vdash S \circ \mathsf{until}(A, B)$ and $c < d$, contradicting $\mathsf{max}(R)(a, b, c)$ again. As both possibilities are contradictory, we have $\forall d \in M$: if $u < v \leq w$, then $d \Vdash \neg\mathsf{ran}(R)$.

Hence, we have:

$$\begin{aligned} \mathfrak{M} \models\; & \exists z'(x < z \leq z' < y \wedge \mathsf{max}(S)(x, z', y) \wedge z'\ \overline{\mathsf{until}(A, B) \circ <}\ y \wedge xRz \wedge \\ & (z \approx z' \vee z\ \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))z')\ [a, b, c] \end{aligned}$$

and (D.4) is proved.

To prove (D.4) implies $\mathsf{max}(R)(a, b, c)$, we just have to examine two cases, both leading to $\mathsf{max}(S \circ \mathsf{until}(A, B))(a, b, c)$.

First, assume there is a time point $c' \in M$ such that $a < c' < c < b$ and $\mathsf{max}(S)(a, b, c')$, and $\mathsf{max}(\mathsf{until}(A, B))(c', b, c)$. To prove $\mathsf{max}(S \circ \mathsf{until}(A, B))(a, b, c)$, since $a < c < b$ and $a, c \Vdash S \circ \mathsf{until}(A, B)$ are given, we just have to show $\forall d \in M$: if $u < d < b$ then $a, d \nVdash S \circ \mathsf{until}(A, B)$. So, suppose, for a contradiction, the opposite and take $d \in M$ such that $c < d < b$ and $a, d \Vdash S \circ \mathsf{until}(A, B)$. Hence, $\exists d' \in T(a, d' \Vdash S$ and $d', d \Vdash \mathsf{until}(A, B))$. Since $\mathsf{max}(S)(a, b, c')$ and $a < d' < d < b$, and $a, d' \Vdash S$ we have $d' \leq c'$. But, from $(c', d) \subseteq (d', d)$ and $d', d \Vdash \mathsf{until}(A, B)$, we obtain $c', d \Vdash \mathsf{until}(A, B)$ which, together with $c < d < b$, contradicts $\mathsf{max}(\mathsf{until}(A, B))(c', b, c)$.

Now, assume there is a time point $e \in M$ such that $a < c \leq e < b$ and $\mathsf{max}(S)(a, b, e)$, $e, b \nVdash \mathsf{until}(A, B) \circ <$, $a, c \Vdash R$, and $c, e \Vdash \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))$ when $c \neq e$. To prove $\mathsf{max}(R)(a, b, c)$, it is enough to prove $\forall d \in M$: if $u < d < b$, then $a, d \nVdash S \circ \mathsf{until}(A, B)$. Suppose, for a contradiction, that d, d' are time points in M for which $c < d < b$ and $a < d' < d$, and $a, d' \Vdash S$, and $d', d \Vdash \mathsf{until}(A, B))$. We have two cases. If $c = e$, then $d' \leq c$ contradicts $e, b \nVdash \mathsf{until}(A, B) \circ <$; and $c < d'$ contradicts $\mathsf{max}(S)(a, b, e)$. If $c < e$, then $c < d \leq e$ contradicts $c, e \Vdash \mathsf{until}(\neg\mathsf{ran}(R), \neg\mathsf{ran}(R))$; and $d' \leq e < d$ contradicts $e, b \nVdash \mathsf{until}(A, B) \circ <$; and $e < d' < d$ contradicts $\mathsf{max}(S)(a, b, e)$. As both cases are contradictory, we have $\forall d \in M$: if $c < d < b$, then $a, d \nVdash S \circ \mathsf{until}(A, B)$.

This concludes the proof. ∎

Lemma D.3.10 *Let R be an until relation term and $\mathfrak{M}$ be a flow of time. For all time points $a, b, c, d \in M$ such that $a < c \leq d < b$, if $a, c \Vdash R$ and $a, b \Vdash R$, and $d \Vdash \mathsf{ran}(R)$, then also $a, d \Vdash R$.*

PROOF. Let R be an until relation term. We prove the result by induction on the number of $\circ$'s in R.

The base case is $R = ?C \circ \mathsf{until}(A, B)$. Hence, suppose a, b, c, d are time points such that $a < c \leq d < b$ and $a \Vdash C$, and $a, c \Vdash \mathsf{until}(A, B)$, and $a, b \Vdash \mathsf{until}(A, B)$, and $\exists d' \in M : d', d \Vdash ?C \circ \mathsf{until}(A, B)$. We have $a, d \Vdash ?C \circ \mathsf{until}(A, B)$ because:

- $a < d$ an $a \Vdash C$ are given.
- $d \Vdash A$ comes from $d', d \Vdash \mathsf{until}(A, B)$.
- $(a, d) \Vdash B$ comes from $(a, b) \Vdash B$ and $(a, d) \subset (a, b)$.

Let S be an until relation term such that for all time points $a, b, c, d \in M$, if $a < c \leq d < b$, when $a, c \Vdash S$ and $a, b \Vdash S$, and $d \Vdash \mathsf{ran}(S)$, then we also have $a, d \Vdash S$.

Now, let $R = S \circ \mathsf{until}(A, B)$ and suppose a, b, c, d are time points such that $a < c \leq d < b$, and $a, c \Vdash R$, and $a, b \Vdash R$, and $d \Vdash \mathsf{ran}(R)$. Hence, there are time points c', d', e, e' for which:

$$a, c' \Vdash S \text{ and } c', c \Vdash \mathsf{until}(A, B),$$

$$a, d' \Vdash S \text{ and } d', b \Vdash \mathsf{until}(A, B),$$

and

$$e, e' \Vdash S \text{ and } e', d \Vdash \mathsf{until}(A, B).$$

Since $e' < d$, we have two cases:

If $e' < c$, then we have $c', d \Vdash \mathsf{until}(A, B)$ as follows: $c' < d$ because $c' < c \leq d$; $d \Vdash A$ because $e', d \Vdash \mathsf{until}(A, B)$; $(c', d) \Vdash B$ because $(c', d) = (c', c) \cup \{c\} \cup (c, d)$ and $(c', c) \Vdash B$, and $(e', d) \Vdash B$, and $(c, d) \subset (e', d)$. Hence, $a, d \Vdash S \circ \mathsf{until}(A, B)$ because $a, c' \Vdash S$.

If $c \leq e' < d$, then we have two sub-cases. First, when $d' \leq e'$, we have $d', d \Vdash \mathsf{until}(A, B)$ as follows: $d' < d$ because $d' \leq e' < d$; $d \Vdash A$ because $e', d \Vdash$

$\mathsf{until}(A, B)$; $(d', d) \Vdash B$ because $(d', d) \subset (d', b)$ and $(d', b) \Vdash B$. Hence, in this case, since $a, d' \Vdash S$, we have $a, d \Vdash S \circ \mathsf{until}(A, B)$. Second, if $e' < d'$, then we have $a < c' < e' < d'$ and $a, c' \Vdash S$, and $a, d' \Vdash S$ and $e' \Vdash \mathsf{ran}(S)$. Then, by the IH, $a, e' \Vdash S$. As $e', d \Vdash \mathsf{until}(A, B)$, in this case, we also obtain $a, d \Vdash S \circ \mathsf{until}(A, B)$, as required.

This completes the proof. ∎

Referências Bibliográficas

[1] ANDRÉKA, H., "Representations of distributive lattice-ordered semigroups with binary relations", *Algebra Univers.*, v. 28, pp. 12–25, 1991.

[2] ANDRÉKA, H., NÉMETI, I., SAIN, I., "Algebraic Logic", tech. rep., Alfréd Rényi Institute of Mathematics, Hungarian Academy of Sciences, 2003. http://www.math-inst.hu/pub/algebraic-logic/Contents.html.

[3] ARECES, C., *Logic Engineering: the case of description and hybrid logics.* PhD thesis, Institute for Logic, Language and Computation (ILLC), University of Amsterdam, Amsterdam, 2000.

[4] ARECES, C., BLACKBURN, P., "Bringing them all together", *J. Logic Comput.*, v. 11, pp. 657–669, 2001.

[5] ARECES, C., BLACKBURN, P., MARX, M., "Hybrids logics: Characterization, interpolation and complexity", *J. Symbolic Logic*, v. 66, pp. 997–1010, 2001.

[6] BENEDIKT, M., FAN, W., KUPER, G., "Structural properties of XPath fragments", *Theoret. Comput. Sci.*, v. 336, pp. 3–31, 2005.

[7] BERGHAMMER, R., ZIERER, H., "Relational algebraic semantics of deterministic and nondeterministic programs", *Theoret. Comput. Sci.*, v. 43, pp. 123–147, 1986.

[8] BLACKBURN, P., "Nominal tense logic", *Notre Dame J. Formal Logic*, v. 14, pp. 56–83, 1993.

[9] BLACKBURN, P., "Internalizing labelled deduction", *J. Logic Comput.*, v. 10, pp. 137–168, 2000.

[10] BLACKBURN, P., "Representation, Reasoning, and relational Structures: a Hybrid Logic Manifesto", *Log. J. IGPL*, v. 8, pp. 339–365, 2000.

[11] BLACKBURN, P., SELIGMAN, J., "What are hybrid languages?", In: *Advances in Modal Logic*, v. 1 of *CSLI Publications*, pp. 41–62, Stanford, Stanford University, 1998.

[12] BLACKBURN, P., TZAKOVA, M., "Hybrid completeness", *Log. J. IGPL*, v. 6, pp. 625–650, 1998.

[13] BLACKBURN, P., TZAKOVA, M., "Hybrid languages and temporal logics", *Log. J. IGPL*, v. 7, pp. 27–54, 1999.

[14] BRINK, C., KAHL, W., (EDS.), G. S., *Relational Methods in Computer Science.* Vienna, Springer, 1997.

[15] BROWN, C., HUTTON, G., "Categories, allegories and circuit design", In: *Proceedings of the Ninth Annual IEEE Symposium on Logic in Computer Science, 4-7 July 1994*, (Paris, France), pp. 372–381, IEEE, 1994.

[16] BURGESS, J., "Axioms for tense logic I. "Since" and "Until"", *Notre Dame J. Formal Logic*, v. 23, pp. 367–374, 1982.

[17] BURGESS, J., GUREVICH, Y., "The decision problem for linear temporal logic", *Notre Dame J. Formal Logic*, v. 26, pp. 115–128, 1985.

[18] CANTONE, D., FORMISANO, A., OMODEO, E., *et al.*, "Compiling dyadic specifications into map algebra", *Theoret. Comput. Sci.*, v. 293, pp. 447–475, 2003.

[19] CHIN, L. H., TARSKI, A., "Distributive and modular laws in the arithmetics of relation algebras", *University of California Publications in Mathematics (N.S.)*, v. 1, pp. 341–384, 1951.

[20] CHIPMAN, J., "The foundations of utility", *Econometrica*, v. 28, pp. 193–224, 1960.

[21] COPILOWISH, I., "Matrix development of the calculus of relations", *J. Symbolic Logic*, v. 13, pp. 193–203, 1948.

[22] COUSOT, P., "Methods and logics for proving programs", In: *Handbook of Theoretical Computer Science* (VAN LEEUWEN, J., ed.), v. B: *Formal Models and Semantics*, pp. 841–993, Amsterdam, Elsevier, 1990.

[23] CURTIS, S., LOWE, G., "Proofs with graphs", *Sci. Comput. Programming*, v. 26, pp. 197–216, 1996.

[24] DE BAKKER, J., DE ROEVER, W.-P., "A calculus for recursive program schemes", In: *Proc. 1st International Coll. on Automata, Languages, and Programmimg, Proc. Symp. (IRIA), 3–7 july, 1972, Rocquencourt* (NIVAT, M., ed.), pp. 167–196, Amsterdam, North-Holland, 1973.

[25] DE FREITAS, R., VIANA, J., "A completeness result for relation algebra with binders", In: *IX Workshop on Logic, Language, Information and Computation (WoLLIC'02), Rio de Janeiro, 30 de julho a 02 de agosto de 2002, Eletronic Notes in Theoretical Computer Science (ENTCS), 67*, (Amsterdam), pp. 1–14, Elsevier, 2002.

[26] DE FREITAS, R., VIANA, J., BENEVIDES, M., *et al.*, "Squares in Fork Arrow Logic", *J. Philos. Logic*, v. 32, pp. 343–355, 2003.

[27] DE FREITAS, R., VIANA, J., VELOSO, P., *et al.*, "On hybrid arrow logic", In: *Proceedings of HyLo@LICS—4th Workshop on Hybrid Logics, LICS 2002 Affiliated Wokshop, Copenhagen, Denmark, July 2002*, (Copenhagen), pp. 53–67, 2002.

[28] DE RIJKE, M., "The modal logic of inequality", *J. Symbolic Logic*, v. 57, pp. 566–584, 1992.

[29] DE RIJKE, M., "The Logic of Peirce Algebras", *J. Logic Lang. Inform.*, v. 4, pp. 227–250, 1995.

[30] DÓSEN, K., "Functions redefined", *Amer. Math. Monthly*, v. 105, pp. 631–635, 1998.

[31] DOUGHERTY, D., GUTIÉRREZ, C., "Normal forms and reduction theories of binary relations", In: *Rewriting Techniques and Applications, 11th International Conference, RTA2000, Norwich, UK, July, 2000, Proc. LNCS 1833*, pp. 95–109, Springer, 2000.

[32] EBBINGHAUS, H.-D., FLUM, J., THOMAS, W., *Mathematical Logic.* second ed. New York, Springer, 1994.

[33] FORMISANO, A., OMODEO, E., SIMEONI, M., "A graphical approach to relational reasoning", In: *RelMiS 2001, Relational Methods in Software (a Satellite Event of ETAPS 2001) Genova, Italy, 7–8 April 2001, Eletronic Notes in Theoretical Computer Science (ENTCS), 44*, pp. 1–22, Elsevier, 2003.

[34] FREYD, P., SCEDROV, A., *Categories, Allegories*, v. 39 of *North-Holland Mathematical Library.* Amsterdam, North-Holland, 1990.

[35] FRIAS, M., "Independence of the axiomatization of fork", *Algebra Univers.*, v. 39, pp. 211–215, 1998.

[36] FRIAS, M., *Fork Algebras in Algebra, Logic and Computer Science*, v. 2 of *Advances in Logic.* World Scientific, 2002.

[37] FRIAS, M., BAUM, G., HAEBERER, A., "Fork algebras in algebra, logic and computer science", *Fund. Inform.*, v. 32, pp. 1–25, 1997.

[38] FRIAS, M., BAUM, G., HAEBERER, A., "Representability and program construction within fork algebras", *Log. J. IGPL*, v. 6, pp. 227–257, 1997.

[39] FRIAS, M., HAEBERER, A., VELOSO, P., "A finite axiomatization for fork algebras", *Log. J. IGPL*, v. 5, pp. 311–319, 1997.

[40] FRIAS, M., VELOSO, P., BAUM, G., "Fork algebras: past, present and future", *Jounal on Relational Methods in Computer Scince*, v. 1, pp. 181–216, 2004.

[41] FRICK, M., GROHE, M., "The complexity of first-order and monadic second-order logic revisited", *Ann. Pure Appl. Logic*, v. 130, pp. 3–31, 2004.

[42] GABBAY, D., HODKINSON, I., REYNOLDS, M., *Temporal Logic: Mathematical Foundations and Computational Aspects.* Oxford, Oxford University Press, 1994.

[43] GABBAY, D., PNUELI, A., SHELAH, S., *et al.*, "On the temporal analysis of fairness", In: *7th ACM Symposium on Principles of Programming Languages*, (Las Vegas), pp. 163–173, ACM, 1980.

[44] GARGOV, G., GORANKO, V., "Modal logic with names", *J. Philos. Logic*, v. 22, pp. 607–636, 1993.

[45] GIVANT, S., "Tarski's development of logic and mathematics based on the calculus of relations", In: *Algebraic Logic* (ANDRÉKA, H., MONK, J. D., NÉMETI, I., eds.), v. 54 of *Colloquia Mathematica Societatis János Bolyai*, pp. 189–215, Amsterdam, North-Holland, 1991.

[46] GORANKO, V., PASSY, S., "Using the universal modality: gains and questions", *J. Logic Comput.*, v. 2, pp. 5–30, 1992.

[47] GOTTLOB, G., KOCH, C., PICHLER, R., "The complexity of XPath evaluation", In: *ACM SIGMOD/PODS 2003, San Diego, California, June 9-12, 2003*, (San Diego), pp. 179–190, ACM, 2003.

[48] GOTTLOB, G., KOCH, C., PICHLER, R., "Efficient algorithms for processing XPath queries", In: *Proceedings of the 28th VLDB Conference, Hong Kong, China 2002*, (Hong Kong), pp. 95–106, 2003.

[49] GRIES, D., SCHNEIDER, F., "Teaching Math more effectively, through calculational proofs", *Amer. Math. Monthly*, v. 102, pp. 691–697, 1995.

[50] GRIES, D., SCHNEIDER, F., "Adding the everywhere operator to propositional logic", *J. Logic Computat.*, v. 8, pp. 119–129, 1998.

[51] GUTIÉRREZ, C., "Normal forms for connectedness in categories", *Ann. Pure Appl. Logic*, v. 108, pp. 237–247, 2001.

[52] GYURIS, V., "A short proof of representability of fork algebras", *Theoret. Comput. Sci.*, v. 188, pp. 211–220, 1997.

[53] HAEBERER, A., FRIAS, M., BAUM, G., *et al.*, "Fork algebras", In: *Relational Methods in Computer Science* (BRINK, C., KAHL, W., SCHIMIDT, G., eds.), Advances in Computing Science, pp. 54–69, Vienna, Springer, 1997.

[54] HENKIN, L., "Internal semantics and algebraic logic", In: *Truth, syntax and modality*, v. 68 of *Studies in Logic and the Foundations of Math*, pp. 226–246, Amsterdam, North-Holland, 1973.

[55] HODKINSON, I., "Expressive completeness of Until and Since over dedekind complete linear time", In: *Modal Logic and Process Algebra*, v. 53 of *CSLI Publications*, pp. 171–185, Stanford, Stanford University, 1995.

[56] HODKINSON, I., "Atom structures of cylindric algebras and relation algebras", *Ann. Pure Appl. Logic*, v. 89, pp. 117–148, 1997.

[57] HUTTON, G., "A relational derivation of a functional program", In: *Lecture Notes of the STOP Summer School on Constructive Algorithms, Ameland, The Netherlands*, 1992.

[58] IMMERMAN, N., KOZEN, D., "Definability with bounded number of bound variables", *Inf. Comput.*, v. 83, pp. 121–139, 1989.

[59] JÓNSSON, B., "Representation of modular lattices and of relation algebras", *Trans. Amer. Math. Soc.*, v. 92, pp. 449–464, 1959.

[60] KAMP, J., *Tense Logic and the Theory of Order.* PhD thesis, University of California, Los Angeles, 1968.

[61] KOZEN, D., TIURYN, J., "Logics of programs", In: *Handbook of Theoretical Computer Science*, pp. 789–840, New York, Elsevier, 1990.

[62] KURTONINA, N., DE RIJKE, M., "Bisimulations for temporal logic", *J. Logic Lang. Inform.*, v. 6, pp. 403–425, 1997.

[63] LICHTESTEIN, O., PNUELI, A., "Propositional temporal logics: decidability and completeness", *Log. J. IGPL*, v. 8, pp. 55–85, 2000.

[64] LYNDON, R., "The Representation of Relational Algebras", *Annals Math.*, v. 51, pp. 707–729, 1950.

[65] LYNDON, R., "The Representation of Relational Algebras, II", *Annals Math.*, v. 63, pp. 294–307, 1956.

[66] MADDUX, R., "Finitary algebraic logic", *Z. Math. Logik Grundlag. Math.*, v. 35, pp. 321–332, 1989.

[67] MADDUX, R., "Finitary algebraic logic II", *Math. Log. Quart.*, v. 39, pp. 566–569, 1993.

[68] MADDUX, R., "On the derivation of identities involving projection functions", In: *Logic Colloquium'92* (CSIRMAZ, GABBAY, D., DE RIJKE, M., eds.), CLSI, (Stanford), pp. 145–163, 1995.

[69] MARX, M., *Algebraic Relativization and Arrow Logic*. PhD thesis, Institute for Logic, Language and Computation (ILLC), University of Amsterdam, Amsterdam, 1995.

[70] MARX, M., "Relation Algebras with Binders", *J. Logic Comp.*, v. 11, pp. 691–700, 2001.

[71] MARX, M., "Conditional XPath, the first order complete XPath dialect", In: *PODS* (DEUTSCH, A., ed.), pp. 13–22, ACM, 2004.

[72] MARX, M., "XPath with conditional axis relations", In: *Advances in Database Technology - EDBT 2004: 9th International Conference on Extending Database Technology, Heraklion, Crete, Greece, March 14-18, 2004* (ET AL., E. B., ed.), pp. 477–494, Springer, 2004.

[73] MARX, M., "First order paths in ordered trees", In: *ICDT* (EITER, T., LIBKIN, L., eds.), v. 3363 of *Lecture Notes in Computer Science*, pp. 114–128, Springer, 2005.

[74] MARX, M., VENEMA, Y., *Multi Dimensional Modal Logic*. Kluwer Academic Publishers, 1997.

[75] MORGAN, A. D., "On the syllogism: IV, and on the logic of relations, (read April 23, 1860)", *Cambridge Phil. Soc. Trans.*, v. 10, pp. 331–358, 1864.

[76] NÉMETI, I., "Algebraization of quantifier logics, an introductory overview", *Studia Logica*, v. 50, pp. 485–569, 1991.

[77] ONO, H., NAKAMURA, A., "On the size of refutation Kripke models for some linear modal and tense logics", *Stud. Log.*, v. 39, pp. 325–333, 1980.

[78] PEIRCE, C., "Description of a notation for the logic of relatives, resulting from an amplification of the conceptions of Boole's calculus of logic", *Memoirs of the American Academy of Sciences*, v. 9, pp. 317–378, 1870.

[79] PEIRCE, C., "On the algebra of logic", *Amer. J. Math.*, v. 3, pp. 15–57, 1880.

[80] PEIRCE, C., “Note B: the logic of relatives”, In: *Studies in Logic by Members of the Johns Hopkins University* (PEIRCE, C., ed.), pp. 187–203, Boston, Little, Brown, and Co., 1883.

[81] PEIRCE, C., “On the algebra of logic: A contribution to the philosophy of notation”, *Amer. J. Math.*, v. 7, pp. 180–196, 1885.

[82] PNUELI, A., “The temporal logic of programs”, In: *Proceedings of the Eighteenth IEEE Symposium on Foundations of Computer Science*, (Providence, RI), pp. 46–57, IEEE, 1977.

[83] REYNOLDS, M., “The complexity of the temporal logic with “until” over general linear time”, *JCSS*, v. 66, pp. 393–426, 2003.

[84] REYNOLDS, M., “The complexity of temporal logic over the reals”. Submitted, 2004.

[85] SCHEIN, B., “Relation algebras and function semigroups”, *Semigroup Forum*, v. 1, pp. 1–62, 1970.

[86] SCHEIN, B., “Representation of subreducts of Tarski relation algebras”, In: *Algebraic Logic* (ANDRÉKA, H., MONK, J. D., NÉMETI, I., eds.), v. 54 of *Colloquia Mathematica Societatis János Bolyai*, pp. 621–635, Amsterdam, North-Holland, 1991.

[87] SCHMIDT, G., STRÖHLEIN, T., *Relation and Graphs: Discrete Mathematics for Computer Scientists.* EACTS Monographs on Theoretical Computer Science. Berlin, Springer-Verlag, 1993.

[88] SCHRÖDER, E., *Vorlesungen über die Algebra der Logik (Exakte Logik). Dirtter Band: Algebra und Logik der Relative.* Leipzig, B.G. Teubner, 1895.

[89] SHAPIRO, S., “Second-order languages and mathematical pratice”, *J. Symbolic Logic*, v. 50, pp. 714–742, 1985.

[90] SISTLA, A., CLARKE, E., “The complexity of propositional linear temporal logics”, *JACM*, v. 32, pp. 733–749, 1985.

[91] SUPPES, P., “Elimination of quantifiers in the semantics of natural language by use of extended relation algebras”, *Rev. Internat. Philos.*, v. 117–118, pp. 243–259, 1976.

[92] TARSKI, A., "On the calculus of relations", *J. Symbolic Logic*, v. 6, pp. 73–89, 1941.

[93] TARSKI, A., "Some metalogical results concerning the calculus of relations", *J. Symbolic Logic*, v. 18, pp. 188–189, 1953.

[94] TARSKI, A., "A simplified formalization of predicate logic with identity", *Arch. Math. Logik Grundlagenforsch.*, v. 7, pp. 61–79, 1965.

[95] TARSKI, A., GIVANT, S., *A Formalization of Set Theory without Variables.* Colloquium Publications. Providence, Rhode Island, American Mathematical Society, 1987.

[96] TEN CATE, B., *Model Theory for Extended Modal Languages.* PhD thesis, Institute for Logic, Language and Computation (ILLC), University of Amsterdam, Amsterdam, 2004.

[97] TEN CATE, B., "Interpolation for extended modal languages", *J. Symbolic Logic*, v. 70, pp. 223–234, 2005.

[98] VAN DE VEL, M., "Interpreting first-order theories into a logic of records", *Stud. Log.*, v. 72, pp. 411–432, 2002.

[99] VAN DE VEL, M., "Relation algebras with feature symbols". *Vrije Universiteit Amsterdam*, preprint, 2002.

[100] VELOSO, P., DE FREITAS, R., VIANA, P., *et al.*, "On fork arrow logic and its expressive power", *J. Philos. Logic*, 2006. to appear.

[101] VELOSO, P., "On the independence of the axioms for fork algebras", *Bull. Sect. Logic Univ. Łódź*, v. 26, pp. 197–209, 1997.

[102] VELOSO, P., "On eight independent equational axiomatisations for fork algebras", *Bull. Sect. Logic Univ. Łódź*, v. 27, pp. 117–129, 1998.

[103] VELOSO, P., "On fork relations and programming", In: *PRATICA: Proofs, Types and Categories* (HAEUSLER, E., PEREIRA, L., eds.), pp. 5–53, Rio de Janeiro, PUC-Rio, 1999.

[104] VENEMA, Y., *Many-Dimensional Modal Logic.* PhD thesis, Institute for Logic, Language and Computation (ILLC), University of Amsterdam, Amsterdam, 1992.

[105] VENEMA, Y., "Completeness via completeness", In: *Diamons and Defaults: Studies in Pure and Applied Intensional Logic*, v. 229 of *Synthese Library*, pp. 349–358, Amsterdam, Kluwer, 1993.

[106] VENEMA, Y., "Completeness through flatness in two-dimensional temporal logic", In: *Temporal Logic, First International Conference, ICTL'94, LNCS 827*, (Berlin), pp. 149–164, Springer, 1995.

[107] XU, M., "On some US-tense logics", *J. Philos. Logic*, v. 17, pp. 181–202, 1988.