A Connection Calculus for the Description Logic $\mathcal{ALC}$

Fred Freitas[1](✉) and Jens Otten[1,2]

[1] Informatics Center, Federal University of Pernambuco (CIn - UFPE),
Recife, Brazil
fred@cin.ufpe.br, jeotten@cs.uni-potsdam.de
[2] Institut für Informatik, University of Potsdam, Potsdam, Germany

Abstract. This paper presents a connection calculus for the description logic (DL) $\mathcal{ALC}$. It replaces the usage of Skolem terms and unification by additional annotation and introduces blocking, a typical feature of DL provers, by a new rule, to ensure termination in the case of cyclic ontologies. Besides the connection calculus, a simplified clausal form normalization is presented. Furthermore, termination, soundness and completeness of the calculus are proven.

Keywords: Description logic · Connection method · Inference system · Reasoning

1 Introduction

Description logics (DL) are widely used for knowledge representation and modelling ontologies, e.g., they are the standard ontology language for the semantic web [1]. One approach to reason with DL is the embedding of DL into classical first-order logic (FOL) and using existing provers for FOL. On the one hand, when using a FOL prover to answer DL queries, no alterations are required, since DL correspond to decidable fragments of FOL [6, 8]. Moreover, the main FOL reasoners were exhaustively tested, and thus, their usage is reliable. On the other hand, in a preliminary study, Tsarkov et al. reported that, even running one of the fastest FOL provers, the FOL approach displays a performance much slower than a specially crafted DL reasoner [12].

This paper proposes a connection calculus θ-CM for $\mathcal{ALC}$, an important fragment of DL. It relies on annotations instead of Skolem terms, and includes a rule for copying clauses that implements blocking, similarly to existing tableau-based DL reasoners. It is a significantly enhanced version of the connection calculus proposed earlier [3], which essentially translates a DL formula into a FOL matrix, thus including Skolem terms and unification, the latter here replaced by a similar procedure, θ-substitution.

The calculus was implemented in RACCOON (ReAsoner based on the Connection Calculus Over Ontologies) [4]. Therefore, such a calculus and its implementation can serve any Semantic Web application that deals with $\mathcal{ALC}$, an important DL language.

Section 2 presents the DL $\mathcal{ALC}$; its normalization is shown in Sect. 3. Section 4 explains the formal connection calculus for $\mathcal{ALC}$, while its termination, soundness and completeness are proven in Sect. 5. Section 6 concludes with a summary.

R. Khoury and C. Drummond (Eds.): Canadian AI 2016, LNAI 9673, pp. 243–256, 2016.
DOI: 10.1007/978-3-319-34111-8_30

2 The Description Logic $\mathcal{ALC}$

An ontology O in $\mathcal{ALC}$ is a set of axioms over a signature (N_C, N_R, N_O), where N_C is the *set of concept names* (unary predicate symbols), N_R is the *set of role or property names* (binary predicate symbols); N_O is the set of *individual names* (constants) [1]. *Concept expressions* are inductively defined as follows. N_C includes $\top$, the *universal concept* that subsumes all concepts, and $\bot$, the *bottom concept* subsumed by all concepts; all concept names belong to N_C. If $r \in N_R$ is a *role* and $C, D \in N_C$ are *concepts,* then these formulae are also *concepts*: (i) $C \sqcup D$ (ii) $C \sqcap D$, (iii) $\neg C$, (iv) $\forall r.C$; (v) $\exists r.C$.

A knowledge base in DL consists of a set of basic axioms (TBox), and a set of axioms specific to a particular situation (ABox). Two axiom types are allowed in a TBox $\mathcal{T}$: (i) $C \sqsubseteq D$; (ii) $C \equiv D$, standing for $C \sqsubseteq D$ and $D \sqsubseteq C$. An ABox $\mathcal{A}$ w.r.t. a TBox $\mathcal{T}$ is a finite set of assertions of two types: (i) a *concept assertion* is a statement of the form $C(a)$, where $a \in N_O$, $C \in N_C$ and (ii) a *role assertion* $r(a, b)$, where a, $b \in N_O$, $r \in N_R$. An $\mathcal{ALC}$ formula is either an axiom or an assertion; an ontology O is an ordered pair $(\mathcal{T}, \mathcal{A})$.

There are two major ways of defining the semantics of $\mathcal{ALC}$. The first one relies on the definitions of interpretation, model, etc., over a domain Δ [1]. Another way is by mapping $\mathcal{ALC}$ constructs to FOL (see [1], 2.2.1.3) and exploiting the semantics defined for FOL, see, e.g., [2]. This approach defines a translation ϕ that maps a concept C to a unary predicate $\phi_C(x)$ with a free variable x. If C is a concept and r a role, then, e.g., $\exists r.C$ is translated into the FOL formula $\phi_{\exists r.C}(y) = \exists x r(y,x) \wedge \phi_C(x)$ [1].

The work described in this paper uses an adaption of the translation approach, hence, taking advantage of existing concepts of connection calculi for FOL.

As for notation, in this paper, words starting with a capital letter denote concepts; roles start with small letters. Individuals are denoted by the lowercase letters a, b, c, d; variables are denoted by x, y, z, k, u, v. Terms are either variables or individuals.

3 Normal Form and Matrix for $\mathcal{ALC}$

Definition 1 (Query). A *query* α (a TBox or ABox axiom) against an ontology O is an $\mathcal{ALC}$ formula for which the logical consequence $O \vDash \alpha$ should be proven.

Definition 2 ($\mathcal{ALC}$ Disjunctive Normal Form, Clause, Matrix, Graphical Matrix). *Literals* (of $\mathcal{ALC}$) are atomic concepts or roles, possibly negated. Literals involved in an universal restriction $\forall r.C$ or in a existential restriction $\exists r.C$ are underlined. In case a restriction involves more than one clause, literals are indexed with the same new *column index number* at the top. Literals can participate at most in two universal restrictions in left-hand side (LHS) axiom's sub-formula or in two existential ones in the right-hand side (RHS); therefore, they can have at most two indices, e.g. $L^{i,j}$. An $\mathcal{ALC}$ formula in *disjunctive normal form (DNF)* is a disjunction of conjunctions (like $C_1 \vee \ldots \vee C_n$), where each C_i has the form $L_1 \wedge \ldots \wedge L_m$ and each L_i is a literal. The ($\mathcal{ALC}$) *matrix* of an $\mathcal{ALC}$ formula in DNF is its representation as a set $\{C_1, \ldots, C_n\}$, where each *clause* C_i has the form $\{L_1, \ldots, L_m\}$ with literals L_i. In the *graphical matrix* representation, clauses are represented as columns, and restrictions as lines; restrictions with indexes are horizontal, while those without are vertical (see Example 1).

Remark 1. To deduce $O \vDash \alpha$ the validity of the formula $C_1 \wedge \ldots \wedge C_n \rightarrow \alpha$ $(O \rightarrow \alpha)$, i.e. of $\neg O \vee \alpha$, must be proven. The effects for the DNF are: (i) axioms of the form $E \sqsubseteq D$ translate into $E \wedge \neg D$; (ii) ABox assertions are negated; (iii) free variables are existentially quantified; (iv) FOL Skolemization is applied to universal variables; and (v) the query α is not negated.

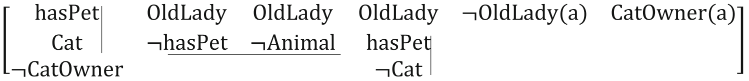
Example 1 (Query, Clause, $\mathcal{ALC}$ Matrix). The query $\{\exists hasPet.Cat \sqsubseteq CatOwner$, $OldLady \sqsubseteq \exists hasPet.Animal \sqcap \forall\ hasPet.Cat\} \vDash OldLady \sqsubseteq CatOwner$ reads in FOL as:

$$\left.\begin{array}{r}\forall x((\exists y hasPet(x,y) \wedge Cat(y)) \rightarrow CatOwner(x))\\ \forall z(OldLady(z) \rightarrow \forall k(hasPet(z,k) \rightarrow Cat(k)))\\ \forall z(OldLady(z) \rightarrow \exists v(hasPet(z,v) \wedge Animal(v)))\end{array}\right\} \vDash \forall u(OldLady(u) \rightarrow CatOwner(u))$$

and is represented by the FOL matrix (where a is a Skolem terms, f a function symbol):

$$\{\{hasPet(x,y),\ Cat(y),\ \neg CatOwner(x)\}, \{OldLady(z), hasPet(z,v), \neg Cat(v)\},\\ \{OldLady(z),\ \neg hasPet(z,f(z))\}, \{OldLady(z),\ \neg Animal(f(z))\}, \{\neg OldLady(a)\},\\ \{CatOwner(a)\}\}$$

and by the following $\mathcal{ALC}$ matrix (the column index marks the two clauses involved in he same restriction; variables are omitted as they are specified implicitly) (Fig. 1):



$$\begin{bmatrix} \mathsf{hasPet} & \mathsf{OldLady} & \mathsf{OldLady} & \mathsf{OldLady} & \neg\mathsf{OldLady(a)} & \mathsf{CatOwner(a)} \\ \mathsf{Cat} & \neg\mathsf{hasPet} & \neg\mathsf{Animal} & \mathsf{hasPet} & & \\ \neg\mathsf{CatOwner} & & & \neg\mathsf{Cat} & & \end{bmatrix}$$

Fig. 1. The query from Example 1 represented as an $\mathcal{ALC}$ matrix

$$\{\{\underline{hasPet, Cat}, \neg CatOwner\}, \{OldLady, \underline{hasPet, \neg Cat}\}, \{OldLady, \underline{\neg hasPet^1}\},\\ \{OldLady, \underline{\neg Animal^1}\}, \{\neg OldLady(a)\}, \{CatOwner(a)\}\}$$

Definition 3 (Impurity, Pure Conjunction/Disjunction). *Impurity* in an $\mathcal{ALC}$ formula is a disjunction in a conjunction, or a conjunction in a disjunction. A *pure conjunction* (PC) or *disjunction* (PD) does not contain impurities (see [3] for a formal definition).

Example 2 (Impurity, Pure Conjunction/Disjunction). (a) $\exists r.A$ and $\wedge_{i=1}^{n} A_i$ are PCs if A and each A_i is also a PC. (b) $\forall r.\big(D_0 \sqcup \ldots \sqcup D_n \sqcup (C_0 \sqcap \ldots \sqcap C_m) \sqcup \big(A_0 \sqcap \ldots \sqcap A_p\big)\big)$ is not a PD as it contains two impurities: $(C_0 \sqcap \ldots \sqcap C_m)$ and $\big(A_0 \sqcap \ldots \sqcap A_p\big)$.

Definition 4 (Two-Lined Disjunctive Normal Form). An $\mathcal{ALC}$ axiom is in *2-lined DNF* iff it is in DNF and in one of the normal forms (NFs): (i) $\hat{E} \sqsubseteq \check{D}$; (ii) $E \sqsubseteq \hat{E}$;

(iii) $\check{D} \sqsubseteq E$, where E is a concept name[1], $\hat{E}$ is a pure conjunction, and $\check{D}$ is a pure disjunction[2].

Example 3 (Two-Lined Disjunctive Normal Form). The axioms (i) $\hat{E} \sqsubseteq \check{D}$ (1NF); (ii) $E \sqsubseteq \exists r.\hat{E}$ (2NF) and (iii) $\forall r.\check{D} \sqsubseteq E$ (3NF), where $\hat{E} = \wedge_{i=1}^{n} C_i$ and $\check{D} = \vee_{j=1}^{m} D_j$ (Fig. 2).

$$i)\ 1NF\colon \begin{bmatrix} C_1 \\ \vdots \\ C_n \\ \neg D_1 \\ \vdots \\ \neg D_m \end{bmatrix} \qquad ii)\ 2NF\colon \begin{bmatrix} E & \cdots & \cdots & E \\ \underline{\neg r} & \underline{\neg C_1} & \underline{\cdots} & \underline{\neg C_n} \end{bmatrix} \qquad iii)\ 3NF\colon \begin{bmatrix} \underline{\neg r} & \underline{D_1} & \underline{\cdots} & \underline{D_m} \\ \neg E & \cdots & \cdots & \neg E \end{bmatrix}$$

Fig. 2. Examples of the three two-lined normal forms' representations in $\mathcal{ALC}$

Example 4 (Two-Lined DNF). Table 1 shows examples of quantification restrictions. Vertical lines represent existential restrict ions ($\exists r.C$), horizontal lines represent universal restrictions ($\forall r.C$) on the LHS axiom's sub-formula or the opposite on the RHS. Lines may overlap. Note also that, if written in FOL, Skolem functions should appear in the two last NFs in Table 1 (e.g., $\neg r(x,f(x))$ would replace $\exists y \ldots \neg r(x,y)$).

Remark 2. The motivation for relying on these NFs is a two-fold: it saves memory by avoiding redundancies in the matrix, and it helps proving the system's soundness, completeness and termination, by restricting the problematic cases to 2-lined columns.

Normalized, "purified" TBoxes may add new, introduced concepts; however, they are *conservative extensions* [5] of their originals, since to every model of the former there is a (sometimes distinct) model of the latter, and validity is preserved. Besides, for these NFs, in the worst case, the number of new concepts grows linearly with the number of impurities; in the average case, this is better than other normalizations (e.g., in [10], the number of new axioms grows linearly with the axioms' length).

Definition 5 (Cycle, Cyclic/Acyclic Ontologies and Matrices). If A and B are atomic concepts in an ontology O, *A directly uses B,* if B appears in the right-hand side of a subsumption axiom whose left-hand side is A. Let the relation *uses* be the transitive closure of *directly uses*. A *cyclic ontology* or *matrix* has a cycle when an atomic concept *uses* itself; otherwise it is *acyclic* [1]; e.g., $O = \{A \sqsubseteq \exists r.B, B \sqsubseteq \exists s.A\}$ is a cyclic ontology. Besides, in acyclic ontologies all subsumption axioms have a concept name in its LHS.

[1] The symbols E and $\hat{E}$ were chosen here to designate a concept name and a pure conjunction rather than the usual C and $\hat{C}$, to avoid confusion with clauses, that are also denoted by C.

[2] If $\exists r.\top \sqsubseteq \hat{E}$ or $\forall r. \sqsubseteq \check{D} \in \bot\mathcal{T}$, then the matrix must include axioms $A \sqsubseteq \top$, for all $A \in N_C$, too. Conversely, $\exists r.\top \sqsubseteq \hat{E}$ or $\forall r.\top \sqsubseteq \check{D}$ requires axioms $\bot \sqsubseteq A$, for all $A \in N_C$, in the matrix.

Table 1. Examples of quantification restrictions

Axiom	Matrix	Negated FOL mapping
$\exists r.\hat{E} \sqsubseteq \forall s.\breve{D}$ with $\hat{E}$ a pure conjunction, $\breve{D}$ a pure disjunction	$\begin{bmatrix} r \\ E_1 \\ \vdots \\ E_n \\ s \\ \neg D_1 \\ \vdots \\ \neg D_m \end{bmatrix}$	$\exists x \exists y \exists z$ $(r(x,y) \wedge$ $E_1(y) \wedge \ldots \wedge E_n(y)$ $\wedge$ $(s(x,z) \wedge$ $\neg D_1(z) \wedge \ldots \wedge \neg D_m(z))$
$A \sqsubseteq \exists r.\hat{E}$ A is a concept name, $\hat{E}$ as above	$\begin{bmatrix} A & \cdots & \cdots & A \\ \underline{\neg r} & \underline{\neg E_1} & \underline{\cdots} & \underline{\neg E_n} \end{bmatrix}$	$\exists x \forall y((A(x) \wedge \neg r(x,y))$ $\vee\ (A(x) \wedge \neg E_1(y))$ $\vee \ldots \vee\ (A(x) \wedge \neg E_n(y)))$
$\forall r.\breve{D} \sqsubseteq A$ $A, \breve{D}$ as above	$\begin{bmatrix} \underline{\neg r} & \underline{D_1} & \underline{\cdots} & \underline{D_m} \\ \neg A & \cdots & \cdots & \neg A \end{bmatrix}$	$\forall x \exists y$ $(\neg r(x,y) \wedge \neg A(x)) \vee$ $(D_1(y) \wedge \neg A(x)) \vee \ldots \vee$ $(D_m(y) \wedge \neg A(x))$

4 The $\mathcal{ALC}$ θ-Connection Calculus ($\mathcal{ALC}$ θ-CM)

The $\mathcal{ALC}$ θ-Connection Method (henceforth $\mathcal{ALC}$ θ-CM) differs from the FOL Connection Method (CM) by replacing Skolem functions and unification by θ-substitutions, and, just as typical DL systems, employs blocking to assure termination.

Definition 6 (Path, Connection, θ-Substitution, θ-Complementary Connection). A *path through a matrix M* contains exactly one literal from each clause in M. A *connection* is a pair of literals $\{E, \neg E\}$ with the same concept/role name, but different *polarities*. A *θ-substitution* assigns each (possibly omitted) variable an individual or another variable. A *θ-complementary* connection is a pair of $\mathcal{ALC}$ literals $\{E(x), \neg E(y)\}$ or $\{p(x,v), \neg p(y,u)\}$, with $\theta(x) = \theta(y), \theta(v) = \theta(u)$. The complement $\bar{L}$ of a literal L is E if $L = \neg E$, and it is $\neg E$ if $L = E$.

Remark 3 (θ-Substitution). Simple term unification without Skolem functions is used to calculate θ-substitutions. The application of a θ-substitution to a literal is an application to its variables, i.e. $\theta(E) = E(\theta(x))$ and $\theta(r) = r(\theta(x), \theta(y))$, where E is an atomic concept and r is a role. Furthermore, $x^\theta = \theta(x)$.

Definition 7 (Set of Concepts, Skolem Condition). The *set of concepts* $\tau(x)$ of a variable or individual x contains all concepts that were instantiated by x so far, or, more formally, $\tau(x) \overset{\text{def}}{=} \{E \in N_C | E(x) \in Path\}$. The *Skolem condition,* ensures that at most one concept is underlined in the graphical matrix form. This condition is formally defined as $\forall a \left| \left\{ \underline{E^i} \in N_C | \underline{E^i(a)} \in Path \right\} \right| \leq 1$, where i is a column index.

Lemma 1 (Equivalence Between θ-Substitution in $\mathcal{ALC}$ θ-CM and Unification in CM for $\mathcal{ALC}$ Formulae). θ-substitution is equivalent to unification for $\mathcal{ALC}$ formula, i.e., both procedures either return the same results or behave the same way w.r.t. their calculi, when given the same inputs.

Proof. The cases occurring in $\mathcal{ALC}$, all covered by θ-substitution, are in Table 2.

Table 2. Equivalence for $\mathcal{ALC}$ between CM unification and θ-substitution in $\mathcal{ALC}$ θ-CM

Unification		θ-substitution	
Input	Output	Input	Output
$L_1 = E(x)$ $L_2 = \neg E(a)$	$\sigma(L_1) = E(a)$	$L_1 = E \, or \, E(x)$ $L_2 = \neg E(a)$	$\theta(L_1) = E(a)$
$L_1 = E(x)$ $L_2 = \neg E(y)$	$\sigma(L_1) = E(y)$	$L_1 = E \, or \, E(x)$ $L_2 = \neg E \, or \, \neg E(y)$	$\theta(L_1) = E(y)$
$L_1 = E(b)$ $L_2 = \neg E(a)$	Not unifiable	$L_1 = E(b)$ $L_2 = \neg E(a)$	No θ-substitution: Not unifiable
$L_1 = E(x)$ $L_2 = \neg E(f(y))$	$\sigma(L_1) = E(f(y))$	$L_1 = E$ $L_2 = \underline{\neg E^i}$	$\theta(L_1) = E(y)$ $\tau(y) = \tau(y) \cup \{\underline{\neg E^i}\}$
$L_1 = E(g(x))$ $L_2 = \neg E(f(y))$	Not unifiable	$L_1 = \underline{E^k}$ $L_2 = \underline{\neg E^j}$	No θ-substitution, as Skolem condition does not hold: $\forall a \left\| \left\{ \underline{E^i} \middle\| \underline{E^i(a)} \in Path \right\} \right\| \leq 1$

Note that in all cases θ-substitution and unification yield the same substitution or no substitution; the only exception resides in the case where $L_1 = E(x)$ and $L_2 = \neg E(f(y))$ ($L_1 = E, L_2 = \underline{\neg E^i}$ in the notation without variables). However, Lemma 2 shows that they are equivalent, in the sense that unification in FOL CM prevents the same connections that $\mathcal{ALC}$ θ-CM and θ-unification prevent. ∎

Definition 8 (**$\mathcal{ALC}$ Connection Calculus).** Figure 3 shows the formal $\mathcal{ALC}$ connection calculus ($\mathcal{ALC}$ θ-CM), adapted from the FOL CM [9]. The rules of the calculus are applied in an analytic, bottom-up way. The basic structure is the tuple $< C, M, Path >$, where clause C is the open sub-goal, M the matrix corresponding to the query $O \vDash \alpha$ (O is an $\mathcal{ALC}$ ontology) and $Path$ is the *active path*, i.e. the (sub-)path currently checked. The index $\mu \in \mathbb{N}$ of a clause C^μ denotes that C^μ is the μ-th copy of clause C, increased when *Cop* is applied for that clause (the variable x in C^μ is denoted x_μ) – see example of copied clauses in Fig. 8. When *Cop* is used, it is followed by the application of *Ext* or *Red*, to avoid non-determinism in the rules' application. The *Blocking Condition* is defined as follows: the new individual x_μ^θ (if it is new, then $x_\mu^\theta \notin N_O$, as in the condition) has its set of concepts $\tau(x_\mu^\theta)$ compared to the set of concepts of the previous copied individual, i.e., $\tau(x_\mu^\theta)$ g $\tau\left(x_{\mu-1}^\theta\right)$ [11], to test if the former is a subset of the latter.

Remark 4 (**$\mathcal{ALC}$ Connection Calculus).** FOL CM already copies clauses, using the indexing function μ; in $\mathcal{ALC}$ θ-CM, *Cop* implements blocking [1], when no alternative connection is available and cyclic ontologies are dealt. It regulates the creation of new

$$\mathit{Axiom}\ (\mathit{Ax})\quad \frac{}{\{\},M,\mathit{Path}}$$

$$\mathit{Start\ Rule}\ (\mathit{St})\quad \frac{C_1,M,\{\}}{\varepsilon,M,\varepsilon}\quad \mathit{with}\ C_1\in\alpha$$

$$\mathit{Reduction\ Rule}\ (\mathit{Red})\quad \frac{C,M,\mathit{Path}\cup\{L_2\}}{C\cup\{L_1\},M,\mathit{Path}\cup\{L_2\}}$$

$$\mathit{with}\ \theta(L_1)=\theta(\overline{L_2})\ \mathit{and\ the\ Skolem\ condition\ holds}$$

$$\mathit{Extension\ Rule}\ (\mathit{Ext})\ \frac{C_1\backslash\{L_2\},M\backslash C_1,\mathit{Path}\cup\{L_1\}\qquad C,M,\mathit{Path}}{C\cup\{L_1\},M,\mathit{Path}}$$

$$\mathit{with}\ C_1\in M,\ L_2\in C_1,\theta(L_1)=\theta(\overline{L_2})\ \mathit{and\ the\ Skolem\ condition\ holds}$$

$$\mathit{Copy\ Rule}\ (\mathit{Cop})\ \frac{C\cup\{L_1\},M\cup\{C_2^{\mu}\},\mathit{Path}}{C\cup\{L_1\},M,\mathit{Path}}$$

$$\mathit{with}\ C_2^{\mu}\ \mathit{is\ a\ copy\ of}\ C_1,L_2\in C_2^{\mu},\theta(L_1)=\theta(\overline{L_2})\ \mathit{and\ the\ blocking\ condition\ holds}$$

Fig. 3. The connection calculus $\mathcal{ALC}$ θ-CM

individuals, thus preventing non-termination. The *Skolem condition* solves the FOL cases where the combination of Skolemization and unification correctly prevents connections (see Soundness Theorem below). The Skolem condition is easy to implement: only a flag denoting if each variable/individual in any path contains an underlined concept suffices Finally, in the *Ext* and *Red rules,* θ-substitutions replace variables by variables/individuals in the whole matrix. Any individual x can have in its set of concepts $\tau(x)$ at most a single concept name with a column index in the matrix (i.e., $\forall a\left|\left\{\underline{E^i}\in N_C|\underline{E^i(a)}\in \mathit{Path}\right\}\right|\leq 1$). This restriction avoids the situation in FOL matrices, where unification is tried with distinct Skolem functions (see Lemmas 1, 2).

Example 5 ($\mathcal{ALC}$ Connection Calculus). Figures 4 and 5 show the proof of the query from Example 1 using the matrix representation and the formal calculus, respectively.

5 Termination, Soundness and Completeness

Definition 9 (Functional Equivalence Between Decidable Inference Systems w.r.t. a Set of Formulae). An inference system A is *functionally equivalent* to a system B w. r.t. a set of formulae Σ when, for any formula α, $\Sigma\vdash_A\alpha\leftrightarrow\Sigma\vdash_B\alpha$ and $\Sigma\nvdash_A\alpha\leftrightarrow\Sigma\nvdash_B\alpha$.

Lemma 2 (Functional Equivalence Between CM and $\mathcal{ALC}$ θ-CM for Acyclic $\mathcal{ALC}$ Formulae). $\mathcal{ALC}$ θ-CM is functionally equivalent to CM w.r.t. acyclic $\mathcal{ALC}$ formulae.

Proof. Given that (i) *Cop* is not applied here; (ii) the two systems only differ on the replacement of unification by θ-substitution in the *Ext* and *Red* rules; and (iii) θ-substitution is equivalent to unification, Lemma 1 proves all cases but one: a connection with two Skolem functions. The proof for that case is inductive over the matrix structure.

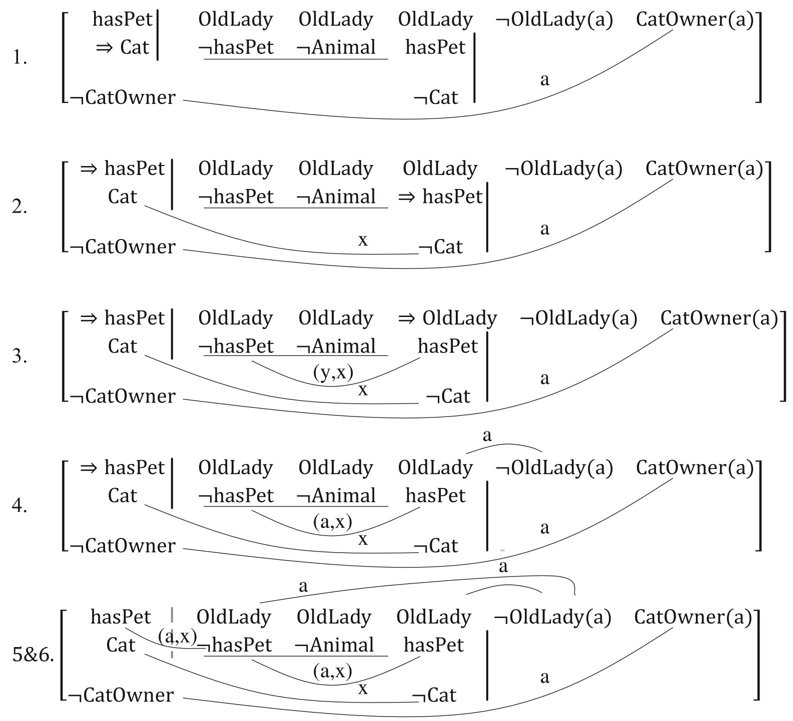


Fig. 4. The proof of the query using the graphical matrix representation. Arcs are connections whose labels are the names of the involved individual(s)/variable.

$$\dfrac{\dfrac{\dfrac{}{\{\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)},\neg O(a)\}}\,Ax}{\{O(y)\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)}\}}\,Red \qquad \dfrac{\dfrac{}{\{\},M,\{CO(a),\underline{C(x)},O(a)\}}\,Ax \qquad \dfrac{}{\{\},M,\{CO(a),\underline{C(x)}\}}\,Ax}{\{O(y)\},M,\{CO(a),\underline{C(x)}\}}\,Ext}{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\}}\,Ext$$

$$\updownarrow$$

$$\dfrac{\dfrac{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\} \qquad \dfrac{\dfrac{\dfrac{}{\{\},M,\{CO(a),\underline{\neg h(a,x)^1},\neg O(a)\}}\,Ax}{\{O(a)\},M,\{CO(a),\underline{\neg h(a,x)^1}\}}\,Red}{\{\underline{h(a,x)}\},M,\{CO(a)\}}\,Red}{\{\underline{h(a,x)},\underline{C(x)}\},M,\{CO(a)\}}\,Ext \qquad \dfrac{}{\{\},M,\{\}}\,Ax}{\dfrac{\{CO(a)\},M,\{\}}{\varepsilon,M,\varepsilon}\,St}\,Ext$$

Fig. 5. The proof of the query using the formal connection calculus, where M is an abbreviation for $M = \{\{\underline{h,C},\neg CO\},\{O,\underline{h,\neg C}\},\{O,\underline{\neg h^1}\},\{O,\underline{\neg A^1}\},\{\neg O(a)\},\{CO(a)\}\}$ (the double-ended arrow just copies the proof part to save text space).

Induction Hypothesis: $O \nvdash_{CM} \alpha$, where the only available connection is between literals with distinct Skolem functions.

Base Case: Suppose a tentative connection between literals with two distinct Skolem functions. For instance, a posed query $\exists r.A \sqsubseteq E, E \sqsubseteq \forall r.a \vDash E(a)$, represented in FOL for the CM in Fig. 6a. Unification prevents the connection (denoted by a dotted lined) in the CM. For $\mathcal{ALC}$ θ-CM, this connection is forbidden too, due to the Skolem condition. In Fig. 6b, for the new variable y, $\tau(y) = \{\underline{A^2}\}$; so it cannot contain also $\neg\underline{A^1}$, otherwise it would violate the Skolem condition $(\forall a \left|\left\{\underline{E^i} \in N_C | \underline{E^i(a)} \in Path\right\}\right| \leq 1)$.

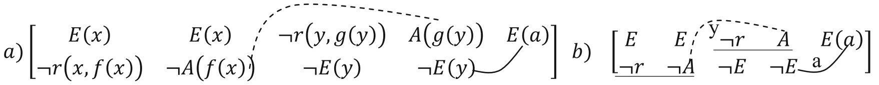


Fig. 6. Tentative connection proof for a) $\exists r.A \sqsubseteq E, E \sqsubseteq \forall r.A \nvdash_{CM} \neg E(a)$, with $\sigma = \{y/a\}$ and b) $\exists r.A \sqsubseteq E, E \sqsubseteq \forall r.A \nvdash_{ALC\theta-CM} \neg E(a)$. Dotted lines stand for forbidden connections.

Inductive Case. Suppose an individual a or a variable x that, in CM, takes part in several connections, two of two in literals with Skolem functions. Then the same behavior from the base case will occur here: CM prevents the last connection ought to unification between two different Skolem functions, while $\mathcal{ALC}$ θ-CM avoids the connection on the basis that the Skolem condition is violated. In Fig. 7, the set of concepts of y or a cannot admit two underlined concepts, i.e. $\tau(y) = \{\underline{A^1}\}$, but $\tau(y) \neq \{\underline{A^1}, \underline{\neg C^2}\}$, again due to the Skolem condition $(\left|\left\{\underline{E^i} \in N_C | \underline{E^i(y)} \in Path\right\}\right| \leq 1)$.

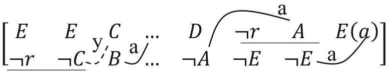


Fig. 7. Tentative $\mathcal{ALC}$ θ-CM connection proof for the inductive case

Since the two systems apply the same rules in the same order and the effects of unification in the FOL CM are the same as that of the θ-substitution in $\mathcal{ALC}$ θ-CM, the systems are equivalent for the set of acyclic $\mathcal{ALC}$ formulae. ∎

Remark 5 (Cyclic Ontologies). Lemma 2 entails that termination, soundness and completeness need only to be proven for the cyclic cases. Lemma 2.22 from [1] states that in a cyclic ontology O, for an individual name $x_i \in$ ABox $\mathcal{A}$ of concept E_i (i.e., individual $E_i(x_i)$ is an assertion), then there is a unique finite sequence of roles $r_1, \ldots, r_i (i \geq 1)$, a unique finite sequence of i role instances (the so-called *role successors*) $r_1(x_0, x_1), \ldots, r_i(x_{i-1}, x_i)$, and a unique, finite sequence of individual names $x_1, \ldots, x_{i-1}$ that creates x_i for the $\mathcal{ALCN}$ tableaux system, as defined by the authors (also holds for $\mathcal{ALC}$, since it is a subset of $\mathcal{ALCN}$). For the $\mathcal{ALC}$ θ-CM, a similar corollary is valid, without needing role successors.

Lemma 3 (Uniqueness of a Generation Sequence). Suppose E_i an $\mathcal{ALC}$ concept from a query over a cyclic ontology O, x_i an individual name of concept E_i (i.e.,$E_i(x_i)$). Then, there is a unique finite sequence of individual names $x_0, \ldots, x_{i-1}$, x_i in a path.

Proof. Since only one rule is applied at a time, the new concept instance $E_i(x_i)$ (with the new individual name x_i) is created in the end of the active path. Since the active path contains the unique sequence $E(x_0), \ldots, E(x_{i-1})$, the corollary holds. ∎

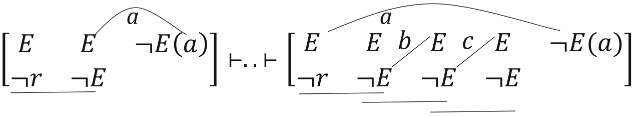


Fig. 8. The creation of the unique sequence $\{\neg E(a), \neg E(b), \neg E(c)\}$ to arrive at $\neg E(c)$ for the unfinished query $E \sqsubseteq \exists r.E \nvdash_{ALC\theta-CM} \neg E(a)$, with $\mu(\{E, \underline{\neg E}\}) = 2$

On the other hand, it is easy to see why role successors [1, Lemma 2.22] are not needed in $\mathcal{ALC}$ θ-CM (for notation $\vdash_{ALC\theta-CM}$ and $\vdash_{CM}$ stand for deductions carried out with $\mathcal{ALC}$ θ-CM, CM respectively). Observe the generation of $\neg E(c)$ for the query described as E v $\exists r.E \vdash_{ALC\theta-CM} \neg E(a)$ in Fig. 8: the active path in this case is $\{\neg E(a), \neg E(b), \neg E(c)\}$. In tableaux systems, role successors $\neg r(a, b)$ and $\neg r$ (b, c) would also need to be created to arrive at $\neg E(c)$.

Theorem 1 (Termination). Given M, the matrix representing the arbitrary query $O \vdash_{ALC\theta-CM} \alpha$, and a chosen initial clause C, any rule sequence in the $\mathcal{ALC}$ θ-CM applied over the tuple "$\varepsilon, M, \varepsilon$" terminates.

Proof. Case (1) $O \vdash_{ALC\theta-CM} \alpha$, which has itself three sub-cases:

(a) $\mathcal{ALC}$ θ-CM *does not apply the Copy rule, and M is proven valid.* This case reduces to the last, since the proof is found and has no cycle; then, $\mathcal{ALC}$ θ-CM terminates.

(b) $\mathcal{ALC}$ θ-CM *uses cyclic axioms, but the Copy rule is only applied with already existent individuals.* Again, $\mathcal{ALC}$ θ-CM is equivalent to CM, because the already existent individuals are not created by *Cop*; thus, $x_\mu^\theta \in N_O$, which never meets the blocking conditions. Consequently, the indexing function μ is incremented, the cyclic columns C_2^μ are copied, and the process repeats that of the CM. Besides, after the copy, θ-substitutions work just as unifications (Lemma 1). Since CM terminates (see proof at [2, III.6.4.]), $\mathcal{ALC}$ θ-CM terminates for this case, too.

(c) $\mathcal{ALC}$ θ-CM *uses the cyclic axioms and the Copy rule (Cop) to create new individuals.* The blocking condition from *Cop* ensures termination, given that it prevents *Cop* to be applied indefinitely, thus generating infinite repetitions of finite sequence(s) of individual names $x_0, \ldots, x_{j-1}, x_j$ in the active path p that would characterize the loop. The blocking condition identifies such repetitions, by checking if the generated concept instances are new individuals names (testing if the new instance is not in the ABox already, i.e., if $x_\mu^\theta \notin N_O$), and if their set of concepts (τ) is changing (by testing whether $\tau(x_{\mu}^{\theta})$ g $\tau\left(x_{\mu-1}^{\theta}\right)$, i.e., if the new instance`s set of concepts is a subset of the set of the previously created instance). If both conditions are met, this active path is blocked; $\mathcal{ALC}$ θ-CM runs until a proof is found (see Completeness Theorem below) and halts.

Case (2) $O \nvdash_{ALC\theta-CM} \alpha$, which has two sub-cases:

(a) The *Copy rule is not applied or is applied with existent individuals, not created by previous Copy rule applications* (i.e., $x_{\mu}^{\theta} \in N_O$), there are open subgoals which make $O \nvdash_{ALC\theta-CM} \alpha$. This case is analogous to cases (1) (a) and (b).

(b) When the *Copy rule creates new x_i individuals*, the case is similar to (1) (c). The infinite open cycles are detected by the *Copy rule* and blocked, and the proof fails due to open subgoals. Hence, $\mathcal{ALC}$ θ-CM terminates for this case, and thus, for all cases. ∎

Theorem 2 (Soundness). An $\mathcal{ALC}$ formula in the two-lined disjunctive normal form M is valid if there is a connection proof for "$\varepsilon, M, \varepsilon$" in the $\mathcal{ALC}$ θ-CM, i.e. there exists a derivation in which all leaves are axioms.

Proof. CM is a decision procedure for $\mathcal{ALC}$, since $\mathcal{ALC}$ corresponds to the decidable FOL fragment L^2 [8]. Thus, $O \vdash_{CM} \alpha$ implies in $O \vDash \alpha$. Hence, it suffices to prove that $O \vdash_{ALC\theta-CM} \alpha$ implies in $O \vdash_{CM} \alpha$. For the cyclic cases and when M originally contains (Skolem) functions, the $\mathcal{ALC}$ formulae will be converted to the 2NF and 3NF, respectively $E \sqsubseteq \exists r.\hat{E}$(or $E \sqsubseteq \hat{E}$); and $\forall r.\check{D} \sqsubseteq E$ (or $\check{D} \sqsubseteq E$), E being an atomic concept, $\hat{E}$ a pure conjunction and $\check{D}$ a pure disjunction. This case is proved by the contrapositive: $O \nvdash_{CM} \alpha$ must imply in $O \nvdash_{ALC\theta-CM} \alpha$.

The contrapositive proof is by structural induction on the structure of the finite sequence of individual names $x_1, \ldots, x_{i-1}$ that generates the next individual of the cycle x_i. The cases of each of the two normal forms are proven in a similar way, as they differ only in the polarity of the class(es) involved in the existential/universal restriction. Note that in any case, the normal forms only generate columns with two elements, which facilitates the inductive proof. The proof for second normal form comes next. The set of formulae in the first, second, third normal forms is denoted by S_{1NF}, S_{2NF} and S_{3NF}.

Induction Hypothesis: $O \nvdash_{CM} \alpha$, where $a \in O$, S_{2NF} (also works for S_{3NF}).

Base Case: $O = \{E \sqsubseteq \exists r.E\} \in S_{2DNF}, O \nvdash_{CM} \alpha$, α being an arbitrary formula, e.g. $\alpha = \{\neg E(a)\}$, as shown in Fig. 9a, b. After the connection $\{E(x), \neg E(a)\}$, with $\sigma = \{x/a\}$, due to the lack of complement for $\neg E(f(x))$), first FOL CM copies the second clause increasing this clause's μ (see Fig. 9a). Then, occurs-check blocks the new connection, and, therefore, $E \sqsubseteq \exists r.E \nvdash_{CM} \neg E(a)$.

As for the $\mathcal{ALC}$ θ-CM, the case is portrayed in Fig. 9b. The first connection is equal to that of CM (except for applying θ instead of σ). In the second clause $(\{E, \underline{\neg E^1}\}), \underline{\neg E(b)^1}$ is built as a θ-substitution. Next (not shown in the figure), *Cop* is applied, and a new clause $\{E, \underline{\neg E^2}\}$ appears in M. Then, the connection $\{\underline{\neg E(b)^1}, E(b)\}$ is settled, and instead of generating a new individual name *c*, *b* is reused in the new literal *E(b)* (an alternative test for blocking suggested by Baader et al. [1]). A new clause copy is made, and since the connection $\{\underline{\neg E(b)^2}, E(b)\}$ reappears, the process is blocked and $E \sqsubseteq \exists r.E \nvdash_{ALC\theta-CM} \neg E(a)$. The lines below the last matrix represent the fact that each new clause copy shall not represent the same individual instantiated in *E*, i.e., a new individual must instantiate $\neg E$. So, for the base case $a \in O$, S_{2NF}, $O \nvdash_{CM} \alpha$ implies in $O \nvdash_{ALC\theta-CM} \alpha$, q.e.d.

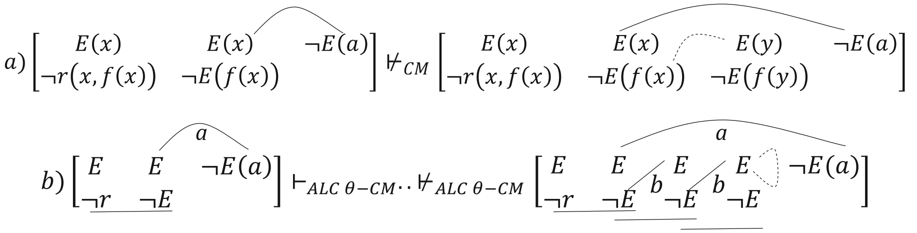


Fig. 9. Tentative connection proof, showing that a) $E \sqsubseteq \exists r.E \nvdash_{CM} \neg E(a)$, $\sigma = \{x/a\}$, with $\mu(\{E(x), \neg E(f(x))\}) = 1$, and that b) $E \sqsubseteq \exists r.E \nvdash_{ALC\theta-CM} \neg E(a)$, with $\mu(\{E, \underline{\neg E}\}) = 2$

Inductive Case: Suppose $O = \{E \sqsubseteq \exists r.B, B \sqsubseteq \hat{E}\}, \alpha = \{\neg E(y_0)\}, \alpha$, α an arbitrary formula), y_0 an individual name, $\hat{E}$ a pure conjunction that *uses E*. In that case, $\hat{E}$ is in one of the following forms: $E \sqcap \hat{A}, \exists r.(E \sqcap \hat{A})$ or $\exists r.(E \sqcap \hat{A}) \sqcap \hat{A}$, being $\hat{A}, \hat{F}$ also pure conjunctions. In either form, *M* contains the column $\{B(x), \neg E(f(x))\}$ for CM and $\{B(x), \underline{\neg E(y)^1}\}$ for $\mathcal{ALC}$ θ-CM. Therefore, even after pursuing long finite sequences of individual names $x_1, \ldots, x_{i-1}$, CM fails just as in the base case, by occurs-check or looping. Similarly, $\mathcal{ALC}$ θ-CM generates $\underline{\neg E(y_1)^1}$ after the first loop and in the next; then, the blocking condition is reached, and $\mathcal{ALC}$ θ-CM halts.

For the inductive case where $a \subseteq O, S_{2NF}, O \nvdash_{CM} \alpha$ implies in $O \nvdash_{ALC\theta-CM} \alpha$. So, the contrapositive $O \vdash_{ALC\theta-CM} \alpha$ implies in $O \vDash \alpha$ and $\mathcal{ALC}$ θ-CM is sound. ∎

Theorem 3 (Completeness). There is a connection proof for "$\varepsilon, M, \varepsilon$", i.e.. there exists a derivation in which all leaves are axioms, if the $\mathcal{ALC}$ formula *F* that corresponds to the matrix *M* is valid.

Proof. Analogously to the soundness proof, it suffices to prove that if $O \vDash \alpha$ then $O \vdash_{ALC\theta-CM} \alpha$. To show that, it is enough to demonstrate $O \vdash_{CM} \alpha$ implies in $O \vdash_{ALC\theta-CM} \alpha$, again, by the contrapositive: $O \nvdash_{ALC\theta-CM} \alpha$ must imply in $O \nvdash_{CM} \alpha$, when *M* contains cycles. $O \nvdash_{ALC\theta-CM} \alpha$ has two sub-cases:

(a) When the *Copy rule is not applied or is applied with already existent individual names, not created by previous Copy rule applications* (i.e., $x_{\mu}^{\theta} \in N_O$), there are open subgoals which make $O \nvdash_{ALC\theta-CM} \alpha$. This case is analogous to the last case: FOL CM fails with the same open subgoals, and $O \nvdash_{CM} \alpha$ for this case, too. Thus, $O \nvdash_{CM} \alpha$.

(b) The case when $O \nvdash_{ALC\theta-CM} \alpha$ *and the Copy rule creates new x_i instances*, can be shown by an inductive proof similar to the one of soundness. The idea is to show that FOL CM loops or finishes by occur-check when blocking takes place in $\mathcal{ALC}$ θ-CM for the base and inductive cases. Therefore, when $\mathcal{ALC}$ θ-CM fails after exhausting all possible connections and θ-substitutions, CM, by occur-checks and/or loops, is also unable to find a proof, i.e., $O \nvdash_{CM} \alpha$.

Indeed, the soundness theorem has shown that there is a functional equivalence between the two systems for the cyclic case too: $\mathcal{ALC}$ θ-CM blocks the cases that CM either loops or halts and vice-versa.

Hence, $O \nvdash_{ALC\theta-CM} \alpha$ implies $O \nvdash_{CM} \alpha$, and ALC θ-CM is complete. ∎

6 Conclusions, Ongoing and Future Work

In the current work, $\mathcal{ALC}$ θ-CM is introduced, a connection method for DL that presents two novelties: (i) it replaces Skolem functions and unification by θ-substitutions that emulate the process of creating instances in the model that is typical for DL tableaux systems; and (ii) it introduces a blocking scheme (with a new *Copy rule*) to deal with cyclic ontologies in order to assure termination.

For the inference process, it employs a normal form that minimizes redundancy in the representation and in the proof search. Moreover, termination, soundness and completeness were proven with the aid of these NFs, which restrict the more convoluted cases to matrix columns of only two literals. This facilitates to portrait the correspondence between FOL unification and θ-substitution/blocking for $\mathcal{ALC}$ θ-CM.

For future work, we will tackle cardinality restrictions ($\geq / \leq n\ r$ for $\mathcal{ALCN}$ and $\geq / \leq n\ r.C$ for $\mathcal{SHQ}$) by dealing with equality between instances. We also aim to create more sophisticated blocking schemes for dynamic and double blocking for DL constructs like inverse roles [7] or dealing with nominals.

Acknowledgements. The authors would like to thank Pernambuco's state sponsoring agency FACEPE for a grant to support the stay of Jens Otten at UFPE.

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook. Cambridge University Press, Cambridge (2003)
2. Bibel, W.: Automated Theorem Proving. Vieweg Verlag, Wiesbaden (1987)
3. Freitas, F.: A connection method for inferencing over the description logic ALC. In: Description Logics Workshop, Barcelona, Spain (2011)
4. Melo, D., Otten, J., Freitas, F.: RACCOON: an experimental automated reasoner for description logics using a modified version of the connection method (2016). https://github.com/dmfilho/raccoon
5. Ghilardi, S., Lutz, C., Wolter, F.: Did I damage my ontology: a case of conservative extensions of description logics. In: Proceedings of 10th International Conference of Principles of KR&R (KR). AAAI Press (2006)

6. Graedel, E., Otto, M., Rosen, E.: Two-variable logic with counting is decidable. In: Proceedings of 12th IEEE Symposium on Logic in Computer Science, LICS 1997 (1997)
7. Horrocks, I., Sattler, U.: A description logic with transitive and inverse roles and role hierarchies. J. Logic Comput. **9**(3), 385–410 (1999)
8. Mortimer, M.: On languages with two variables. Zeitschrift fur mathematische Logik und Grundlagen der Mathematik **21**, 135–140 (1975)
9. Otten, J.: Restricting backtracking in connection calculi. AI Commun. **23**(2–3), 159–182 (2010)
10. Schlicht, A., Stuckenschmidt, H.: Peer-to-peer reasoning for interlinked ontologies. Int. J. Seman. Comput. 4(1), 27–58 (2010). (Special Issue on Web Scale Reasoning)
11. Schmidt, R., Tishkovsky, D.: Analysis of blocking mechanisms for description logics. In: Proceedings of the Workshop on Automated Reasoning (2007)
12. Tsarkov, D., Riazanov, A., Bechhofer, S., Horrocks, I.: Using vampire to reason with OWL. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 471–485. Springer, Heidelberg (2004)