Circumscription: A Reduction Algorithm

Patrick Doherty, Witold Łukaszewicz, Andrzej Szałas, Computing Circumscription Revisited: A Reduction Algorithm, Journal of Automated Reasoning, Dec. 1995

Definitions

Definition 1 (Second-Order Circumscription) Let $\overline{P}$ be a tuple of distinct predicate constants, $\overline{S}$ be a tuple of distinct function and/or predicate constants disjoint from $\overline{P}$, and let $T(\overline{P}, \overline{S})$ be a sentence. The *second-order circumscription* of $\overline{P}$ in $T(\overline{P}, \overline{S})$ with variable $\overline{S}$, written $Circ_{SO}(T; \overline{P}; \overline{S})$, is the sentence

$$T(\overline{P}, \overline{S}) \wedge \forall \overline{\Phi}\ \overline{\Psi}. \neg\ [T(\overline{\Phi}, \overline{\Psi}) \wedge \overline{\Phi} < \overline{P}] \tag{1}$$

where $\overline{\Phi}$ and $\overline{\Psi}$ are tuples of variables similar to $\overline{P}$ and $\overline{S}$, respectively. ∎

(1) can be rewritten as

$$T(\overline{P}, \overline{S}) \wedge \forall \overline{\Phi}\ \overline{\Psi}.[T(\overline{\Phi}, \overline{\Psi}) \wedge [\overline{\Phi} \leq \overline{P}]] \supset [\overline{P} \leq \overline{\Phi}]].$$

We consider only the second-order part of the above formula, i.e.

$$\forall \overline{\Phi}\ \overline{\Psi}.[T(\overline{\Phi}, \overline{\Psi}) \wedge [\overline{\Phi} \leq \overline{P}]] \supset [\overline{P} \leq \overline{\Phi}]].$$

Definition 2 A predicate variable Φ occurs *positively* (resp. *negatively*) in a formula A if conjunctive normal form of A contains a subformula of the form $\Phi(\overline{t})$ (resp. $\neg\Phi(\overline{t})$). A formula A is said to be *positive* (resp. *negative*) w.r.t. Φ iff all occurrences of Φ in A are positive (resp. negative). ∎

Definition 3 A first-order sentence T is said to be *existential* (*universal*) iff it is of the form $\exists\overline{x}.\, T_1$ (or $\forall\overline{x}.\, T_1$, respectively), where T_1 is quantifier free. ∎

Definition 4 Let ϕ be either a predicate constant or a predicate variable and $\overline{\phi}$ be a tuple of predicate constants or a tuple of predicate variables. Then

- a formula $T(\phi)$ is said to be *separated* w.r.t. ϕ iff it is of the form $T_1(\phi) \wedge T_2(\phi)$ where $T_1(\phi)$ is positive w.r.t. ϕ and T_2 is negative w.r.t. ϕ

- a sentence T is said to be *separable w.r.t.* $\overline{\phi}$ iff T is equivalent to a formula of the form

$$\bigvee_{i=1}^{m} [B_i(\overline{\phi}) \wedge (\overline{U}_i \leq \overline{\phi})] \tag{2}$$

where $B_i(\overline{\phi})$ is a formula containing no positive occurrences of predicate constants (variables) from $\overline{\phi}$ and each $\overline{U}_i$ is an n-tuple of predicate expressions not containing predicate constants (variables) of $\overline{\phi}$.■

Useful tautologies

Proposition 1 The following pairs of formulas are equivalent.

Here Q stands for any quantifier
A, B, C are formulas such that C does not contain free occurrence of a variable x.
In clauses (15) and (16), $\overline{t}, \overline{t}_1, \ldots, \overline{t}_n$ are n-tuples of terms and it is assumed that neither C nor any term from $\overline{t}, \overline{t}_1, \ldots, \overline{t}_n$ contains variables from $\overline{x}$.
In clause (17), f is a function variable which does not occur in A.

(1)	$\neg\neg A$	and	A
(2)	$\neg(A\wedge B)$	and	$\neg A\vee\neg B$
(3)	$\neg(A\vee B)$	and	$\neg A\wedge\neg B$
(4)	$\neg\forall xA(x)$	and	$\exists x\neg A(x)$
(5)	$\neg\exists xA(x)$	and	$\forall x\neg A(x)$
(6)	$\exists x(A(x)\vee B(x))$	and	$\exists xA(x)\vee\exists xB(x)$
(7)	$\forall x(A(x)\wedge B(x))$	and	$\forall xA(x)\wedge\forall xB(x)$
(8)	$Qx(A(x))\wedge C$	and	$Qx(A(x)\wedge C)$
(9)	$C\wedge Qx(A(x))$	and	$Qx(C\wedge A(x))$
(10)	$Qx(A(x))\vee C$	and	$Qx(A(x)\vee C)$
(11)	$C\vee Qx(A(x))$	and	$Qx(C\vee A(x))$
(12)	$QxQyA$	and	$QyQxA$
(13)	$A\wedge(B\vee C)$	and	$(A\wedge B)\vee(A\wedge C)$
(14)	$(A\vee B)\wedge C$	and	$(A\wedge C)\vee(B\wedge C)$
(15)	$A(\bar{t})$	and	$\forall\bar{x}(A(\bar{t}\leftarrow\bar{x})\vee\bar{x}\neq\bar{t})$
(16)	$A(\bar{t}_1)\vee\cdots\vee A(\bar{t}_n)$	and	$\exists\bar{x}(\bar{x}=\bar{t}_1\vee\cdots\vee\bar{x}=\bar{t}_n)\wedge A(\bar{t}_1\leftarrow\bar{x}))$
(17)	$\forall\bar{x}\exists yA(\bar{x}\ldots)$	and	$\exists f\forall\bar{x}A(\bar{x},y\leftarrow f(\bar{x}),\ldots)$
(18)	$A(\bar{t}_1)\wedge\cdots\wedge A(\bar{t}_n)$	and	$\forall\bar{x}.(\bar{x}\neq\bar{t}_1\wedge\cdots\wedge\bar{x}\neq\bar{t}_n)\vee A(\bar{t}_1\leftarrow\bar{x}))$. ∎

(15) – used by Ackermann.

(16),(18) – generalization of Ackermann's technique; makes the technique work in the case of clauses containing more than one positive (or negative) occurrence of the eliminated predicate.

(17) – allows to perform Skolemization (i.e. elimination of existential quantifiers) and unskolemization (i.e. elimination of Skolem functions) in such a way that equivalence is preserved. We call this equivalence *second-order skolemization*.

The elimination algorithm

Based on the following lemma, proved by Ackermann in 1934.

Lemma 1 (Ackermann) Let Φ be a predicate variable and $A(\bar{x})$, $B(\Phi)$ be formulae without second-order quantification. Let $B(\Phi)$ be positive w.r.t. Φ and let A contain no occurrences of Φ at all. Then the following equivalences hold:

$$\exists\Phi\forall\bar{x}[\Phi(\bar{x})\vee A(\bar{x},\bar{z})]\wedge B(\Phi\leftarrow\neg\Phi) \;\equiv\; B(\Phi\leftarrow A(\bar{x},\bar{z})) \quad (3)$$

$$\exists\Phi\forall\bar{x}[\neg\Phi(\bar{x})\vee A(\bar{x},\bar{z})]\wedge B(\Phi) \;\equiv\; B(\Phi\leftarrow A(\bar{x},\bar{z})) \quad (4)$$

where in the righthand formulae the arguments $\bar{x}$ of A are each time substituted by the respective actual arguments of Φ (renaming the bound variables whenever necessary). ∎

Proposition 2

Let A be a formula of the form $pref(A_1 \wedge \cdots \wedge A_q)$, where $pref$ is a prefix of first-order quantifiers and $A_1, \ldots, A_q$ are disjunctions of literals. In addition, let Φ be a predicate variable occuring in A and $Conj(A)$ be those conjuncts in A where ϕ occurs. Assume that for any conjunct in $Conj(A)$, Φ occurs either positively, or both positively and negatively (or analogously, negatively, or both negatively and positively). Then

$$\exists \Phi A \equiv pref(A_{i_1} \wedge \cdots \wedge A_{i_r}) \tag{5}$$

where $i_1, \ldots, i_r \in \{1, \ldots, q\}$ and $A_{i_1}, \ldots, A_{i_r}$ are all the conjuncts that do not contain occurrences of Φ (the empty conjunction is regarded to be equivalent to T). ∎

The algorithm

Input: a formula of the form $\exists\Phi.A$, where A is a first-order formula
Output: its first-order equivalent or failure*.

The algorithm can also be used for formulae of the form $\forall\Phi.A$, since it is equivalent to $\neg\exists\Phi.\neg A$. Thus, by repeating the algorithm one can deal with formulae containing arbitrarily many second-order quantifiers.

*The failure of the algorithm does not mean that the second-order formula at hand cannot be reduced to its first-order equivalent. The problem we are dealing with is not even partially decidable.

The algorithm consists of three basic phases:

1. preprocessing;
2. preparation for Ackermann's lemma;
3. application of Ackermann's lemma.

It is always assumed that (1) whenever the goal specific for a current phase is reached, then the remaining steps of the phase are skipped, (2) every time the equivalence (5) of Proposition 2 is applicable, it should be applied.

Preprocessing

The purpose of this phase is to transform the formula $\exists\Phi.A$ into a form that separates positive and negative occurrences of the quantified predicate variable Φ. The form we want to obtain is*

$$\exists\bar{x}\exists\Phi[(A_1(\Phi)\wedge B_1(\Phi))\vee\cdots\vee(A_n(\Phi)\wedge B_n(\Phi))], \tag{6}$$

where, for each $1 \leq i \leq n$, $A_i(\Phi)$ is positive w.r.t. Φ and $B_i(\Phi)$ is negative w.r.t. Φ.†

To achieve the goal of this phase, apply the steps below in the following order.

*It should be emphasized that not every formula is reducible into this form.

†To increase the strength of the algorithm, it is essential to move as many existentially quantified variables as possible into the prefix of (6).

1. Eliminate the connectives $\supset$ and $\equiv$ using the usual definitions. Remove redundant quantifiers. Rename individual variables until all quantified variables are different and no variable is both bound and free. Using the usual equivalences, move the negation connective to the right until all its occurrences immediately proceed atomic formulae.

2. Move universal quantifiers to the right and existential quantifiers to the left applying as long as possible the following equivalences (below $Q \in \{\forall, \exists\}$, $\circ \in \{\vee, \wedge\}$ and B contains no occurrences of variables $\bar{x}$):

   - $Q\bar{x}(A(\bar{x}) \circ B) \equiv (Q\bar{x}A(\bar{x})) \circ B$
   - $Q\bar{x}(B \circ A(\bar{x})) \equiv B \circ Q\bar{x}A(\bar{x})$.

3. Move to the right the existential quantifiers that are in the scope of univeral quantifiers using the equivalences of step 2.

4. Repeat (2) and (3) as long as no new existentially quantified variable can be moved into the prefix.

5. In the matrix of the formula obtained so far, distribute all top-level conjunctions over the disjunctions that occur among their conjuncts. For this purpose, apply the following equivalences:

   - $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$
   - $(A \vee B) \wedge C \equiv (A \wedge C) \vee (B \wedge C)$

   If the resulting formula is not in the form (6), then report the failure of the algorithm. Otherwise replace (6) by its

equivalent given by

$$\exists \bar{x}(\exists \Phi(A_1(\Phi) \wedge B_1(\Phi)) \vee \cdots \vee \exists \Phi.(A_n(\Phi) \wedge B_n(\Phi))). \quad (7)$$

For each disjunct $\exists \Phi(A_i(\Phi) \wedge B_i(\Phi))$ of (7) try to find its first-order equivalent by performing the next phases to the formula $\exists \Phi(A_i(\Phi) \wedge B_i(\Phi))$. If all the equivalents are obtained, return their disjunction, preceded by the prefix $\exists \bar{x}$, as the output of the algorithm.

The following example illustrates the described phase.

Example 1 Consider the formula

$$\exists\Phi[\forall x\exists y(P(y)\vee\exists t(\Phi(t)\vee P(x)\vee R(x,t)))\wedge\exists z\Phi(z)\wedge\exists u\neg\Phi(u)].$$

The following lines show the subsequent transformations.

$$\begin{aligned}
&\exists\Phi[\forall x\exists y(P(y)\vee\exists t(\Phi(t)\vee P(x)\vee R(x,t)))\wedge\exists z\Phi(z)\wedge\exists u\neg\Phi(u)] && \equiv \quad \text{(by 2)}\\
&\exists zu\exists\Phi[\forall x\exists y(P(y)\vee\exists t(\Phi(t)\vee P(x)\vee R(x,t)))\wedge\Phi(z)\wedge\neg\Phi(u)] && \equiv \quad \text{(by 3)}\\
&\exists zu\exists\Phi[\forall x(\exists yP(y)\vee\exists t(\Phi(t)\vee P(x)\vee R(x,t)))\wedge\Phi(z)\wedge\neg\Phi(u)] && \equiv \quad \text{(by 2)}\\
&\exists zu\exists\Phi[(\exists yP(y)\vee\forall x\exists t(\Phi(t)\vee P(x)\vee R(x,t)))\wedge\Phi(z)\wedge\neg\Phi(u)] && \equiv \quad \text{(by 2)}\\
&\exists zuy\exists\Phi[(P(y)\vee\forall x\exists t(\Phi(t)\vee P(x)\vee R(x,t)))\wedge\Phi(z)\wedge\neg\Phi(u)] && \equiv \quad \text{(by 5)}\\
&\exists zuy\exists\Phi[(P(y)\wedge\Phi(z)\wedge\neg\Phi(u))\vee(\forall x\exists t(\Phi(t)\vee P(x)\vee R(x,t))\wedge\Phi(z)\wedge\neg\Phi(u))]
\end{aligned}$$

■

Preparation for Ackermann's lemma

The goal of this phase is to transform a formula of the form $\exists\Phi(A(\Phi)\wedge B(\Phi))$, where $A(\Phi)$ (resp. $B(\Phi)$) is positive (resp. negative) w.r.t. Φ, into one of the forms (3) or (4) given in Lemma 1. Both forms can always be obtained and both transformations should be performed because none, one or both forms may require Skolemization. Unskolemization, which occurs in the next phase, could fail in one form, but not in the other. In addition, one form may be substantially smaller than the other. The steps of this phase are based on equivalences (13) – (17) from Proposition 1. Due to the symmetry of Ackermann's lemma, the steps stated below describe only one of those transformations, namely that leading to the form (3).

1. Transform $A(\Phi)$ into the form

$$pref[(\Phi(\bar{t}_{11})\vee\cdots\vee\Phi(\bar{t}_{1n_1})\vee C_1)\wedge\cdots\wedge(\Phi(\bar{t}_{k1})\vee\cdots\vee\Phi(\bar{t}_{kn_k})\vee C_k)\wedge D],$$

   where $pref$ is a prefix of first-order quantifiers and Φ does not occur in $C_1,\ldots,C_k,D$. This step is always possible by applying the usual technique of obtaining CNF.

2. Transform each conjunct in Step 1. of the form $(\Phi(\bar{t}_{i1})\vee\cdots\vee\Phi(\bar{t}_{in_i})\vee C_i)$, where $n_i > 1$, into its equivalent

$$\exists\bar{x}_i(\forall\bar{y}(\Phi(\bar{y})\vee\bar{x}_i\neq\bar{y}\vee C_i)\wedge(\bar{x}_i=\bar{t}_{i1}\vee\cdots\vee\bar{x}_i=\bar{t}_{in_i}\vee C_i)$$

   and move all existential quantifiers into the prefix $pref$.* Move each of the second conjuncts, $(\bar{x}_i=\bar{t}_{i1}\vee\cdots\vee\bar{x}_i=\bar{t}_{in_i}\vee C_i)$, into D and rename it D'.

*This step is justified by equivalence (16) and (15) of Proposition 1.

3. Transform each conjunct in Step 1 of the form $(\Phi(\bar{t}_{i1} \vee C_i)$ (case $n_i = 1$) into its equivalent $\forall \bar{y}(\Phi(\bar{y}) \vee \bar{y} \neq \bar{t}_{i1} \vee C_i)$

4. Remove all existential quantifiers from the prefix $pref$ using the equivalence of Skolem given by

$$\forall \bar{x}.\exists y.A(\bar{x}, y, \ldots) \equiv \exists f.\forall \bar{x}.A(\bar{x}, y \leftarrow f(\bar{x}), \ldots), \tag{8}$$

where f is a new function variable. After this transformation the input formula takes the form

$$\exists \bar{f} \exists \Phi pref'[\forall \bar{y}(\Phi(\bar{y}) \vee \bar{x}_1 \neq \bar{y} \vee C_1) \wedge \cdots \wedge \forall \bar{y}(\Phi(\bar{y}) \vee \bar{x}_k \neq \bar{y} \vee C_k) \wedge E], \tag{9}$$

where $\bar{f}$ is the tuple of the introduced Skolem functions and $pref'$ contains only universal quantifiers and E is $D' \wedge B(\Phi)$.

5. Transform (9) into its equivalent given by

$$\exists \bar{f} \exists \Phi \forall \bar{y} [\Phi(\bar{y}) \vee pref'((\bar{x}_1 \neq \bar{y} \vee C_1) \wedge \cdots \wedge (\bar{x}_k \neq \bar{y} \vee C_k)) \wedge pref' E. \quad (10)$$

Example 1 [continued] There are two formulae to be considered in this phase, namely

$\exists\Phi(P(y)\wedge\Phi(z)\wedge\neg\Phi(u))$ and

$\exists\Phi\forall x\exists t(\Phi(t)\vee P(x)\vee R(x,t))\wedge\Phi(z)\wedge\neg\Phi(u)$.

We apply phase 2 to the former of the above formulae.

$$\begin{array}{ll} \exists\Phi(P(y)\wedge\Phi(z)\wedge\neg\Phi(u)) & \equiv \text{ (by 3)} \\ \exists\Phi(P(y)\wedge\forall r(\Phi(r)\vee z\neq r)\wedge\neg\Phi(u)) & \end{array}$$

Applying phase 2 to the second formula proceeds as follows.

$$\begin{array}{ll} \exists\Phi\forall x\exists t(\Phi(t)\vee P(x)\vee R(x,t))\wedge\Phi(z)\wedge\neg\Phi(u) & \equiv \text{(by 3)} \\ \exists\Phi\forall x\exists t\forall r(\Phi(r)\vee r\neq t\vee P(x)\vee R(x,t))\wedge\forall r(\Phi(r)\vee z\neq r)\wedge\neg\Phi(u) & \equiv \text{(by 4)} \\ \exists f\exists\Phi\forall x\forall r(\Phi(r)\vee r\neq f(x)\vee P(x)\vee R(x,f(x)))\wedge\forall r(\Phi(r)\vee z\neq r)\wedge\neg\Phi(u) & \\ & \equiv \text{(by 5)} \\ \exists f\exists\Phi\forall r[\Phi(r)\vee(\forall x(r\neq f(x)\vee P(x)\vee R(x,f(x)))\wedge z\neq r)]\wedge\neg\Phi(u) & \blacksquare \end{array}$$

Application of Ackermann's lemma

The goal of this phase is to eliminate the second-order quantification over Φ, applying Ackermann's lemma, and then to unskolemize the introduced function variables (by (17) from Proposition 1). The phase consists of the following two steps.

1. Apply Ackermann's lemma to the formula (10). The resulting formula is of the form

$$\exists \bar{f}[prefE(\neg\Phi \leftarrow pref((\bar{x}_1 \neq \bar{y} \vee C_1) \wedge \cdots \wedge (\bar{x}_k \neq \bar{y} \vee C_k)))]$$

2. Try to remove all existential quantifiers over function variables using the equivalence (8). If both transformations mentioned in the beginning of phase 2 failed, report failure of the algorithm; otherwise try the remaining possibility.

Simplification.

The formula obtained after phase 3 can often be simplified using Proposition 1 (15) or its generalization (18) in the left to right direction.

Example 1 [continued] We apply phase 3 for the pair of formulae obtained as the result of phase 2.

$$\exists\Phi(\forall r(\Phi(r)\vee z\neq r)\wedge P(y)\wedge\neg\Phi(u)) \;\equiv\; \text{(by 1)}$$
$$P(y)\wedge z\neq u.$$

$$\exists f\exists\Phi\forall r[\Phi(r)\vee(\forall x(r\neq f(x)\vee P(x)\vee R(x,f(x)))\wedge z\neq r)]\wedge\neg\Phi(u)$$
$$\equiv \text{(by 1)}$$
$$\exists f\forall x(u\neq f(x)\vee P(x)\vee R(x,f(x)))\wedge z\neq u \;\equiv\; \text{(by 2)}$$
$$\forall x\exists t(u\neq t\vee P(x)\vee R(x,t))\wedge z\neq u.$$

Collecting the above results, we get the first-order equivalent of the input formula, namely
$\exists zuy(P(y)\wedge z\neq u\vee\forall x\exists t(u\neq t\vee P(x)\vee R(x,t))\wedge z\neq u)$. ∎

Discussion of the algorithm

Assume that a second-order formula A is of the form

$$\exists \Phi.[(pref B) \wedge (pref' C)], \tag{11}$$

where,

- $pref$ and $pref'$ are sequences of first-order quantifiers,
- B and C are quantifier-free formulae in conjunctive normal forms,
- B is positive w.r.t. Φ, and
- C is negative w.r.t. Φ.

The following proposition holds.

Proposition 3 Let A be an input formula of the form (11). Then, as a result, the algorithm returns a first-order formula provided that unskolemization (if necessary) succeeds. ∎

Skolem functions are introduced in the second step of the algorithm whenever existential quantifiers are to be eliminated. These can appear in the input formula or may be introduced via application of the equivalence (16) of Proposition 1.

In the following proposition, we formulate conditions under which no Skolem functions are introduced and the algorithm terminates successfully.

Proposition 4 If one of the following conditions holds

- B is universal and each conjunct of B contains at most one occurrence of Φ, or

- C is universal and each conjunct of C contains at most one occurrence of $\neg\Phi$,

then the algorithm always returns a first-order formula as output. ∎

If the input formula cannot be transformed into the form (11) then the algorithm fails.

Examples

Example 2 (Block example) Let $\Gamma(Ab, On)$ be the theory

$$[b1 \neq b2 \wedge B(b1) \wedge B(b2) \wedge \neg On(b1)] \\ \wedge [\forall x.B(x) \wedge \neg Ab(x) \supset On(x)], \tag{12}$$

where B and On are abbreviations for $Block$ and $Ontable$, respectively. The circumscription of $\Gamma(Ab, On)$ with Ab minimized and On varied is

$$Circ_{SO}(\Gamma(Ab, On); Ab; On) \equiv \\ \Gamma(Ab, On) \wedge \forall \Phi \forall \Psi [[\Gamma(\Phi, \Psi) \wedge \Phi \leq Ab] \supset [Ab \leq \Phi]], \tag{13}$$

where

$$\begin{aligned}
\Gamma(\Phi,\Psi) \equiv\ & [b1 \neq b2 \wedge B(b1) \wedge B(b2) \wedge \neg\Psi(b1)] \\
& \wedge [\forall x.B(x) \wedge \neg\Phi(x) \supset \Psi(x)] && (14)\\
\Phi \leq Ab \equiv\ & \forall x.\Phi(x) \supset Ab(x) && (15)\\
Ab \leq \Phi \equiv\ & \forall x.Ab(x) \supset \Phi(x). && (16)
\end{aligned}$$

In the following, we will reduce

$$\forall\Phi\forall\Psi[\Gamma(\Phi,\Psi) \wedge [\Phi \leq Ab] \supset [Ab \leq \Phi]] \qquad (17)$$

in (13). Negating (17), we get

$$\exists\Phi\exists\Psi[\Gamma(\Phi,\Psi) \wedge [\Phi \leq Ab] \wedge \neg[Ab \leq \Phi]]. \qquad (18)$$

Since we will try to remove Φ first, we replace (18) by the equivalent

$$\exists\Psi\exists\Phi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\wedge\neg[Ab\leq\Phi]]. \tag{19}$$

Preprocessing. Replacing $\Gamma(\Phi,\Psi)$, $\Phi\leq Ab$ and $Ab\leq\Phi$ by their equivalent forms (14)–(16), eliminating $\supset$ and renaming individual variables, we obtain

$$\begin{aligned}\exists\Psi\exists\Phi[&b1\neq b2\wedge B(b1)\wedge B(b2)\wedge\neg\Psi(b1)\\ &\wedge\forall x(\neg B(x)\vee\Phi(x)\vee\Psi(x))\\ &\wedge\forall y(\neg\Phi(y)\vee Ab(y))\wedge\exists z(Ab(z)\wedge\neg\Phi(z))].\end{aligned} \tag{20}$$

We next move $\exists z$ to the left, obtaining

$$\begin{aligned}\exists z\exists\Psi\exists\Phi[&b1\neq b2\wedge B(b1)\wedge B(b2)\wedge\neg\Psi(b1)\\ &\wedge\forall x(\neg B(x)\vee\Phi(x)\vee\Psi(x))\\ &\wedge\forall y(\neg\Phi(y)\vee Ab(y))\wedge(Ab(z)\wedge\neg\Phi(z))].\end{aligned} \tag{21}$$

Preparation for Ackermann's lemma. (21) is in the form suitable for application of Ackermann's lemma. To make this more explicit, we rewrite (21) as

$$\begin{aligned}&\exists z\exists\Psi\exists\Phi\forall x[(\Phi(x)\vee\neg B(x)\vee\Psi(x))\\&\quad\wedge\forall y(\neg\Phi(y)\vee Ab(y))\wedge\neg\Phi(z)\wedge Ab(z)\\&\quad\wedge\neg\Psi(b1)\wedge b1\neq b2\wedge B(b1)\wedge B(b2)].\end{aligned}\tag{22}$$

Application of Ackermann's lemma. Applying Ackermann's lemma to (22), we obtain

$$\begin{aligned}&\exists z\exists\Psi[\forall y(\neg B(y)\vee\Psi(y)\vee Ab(y))\\&\qquad\wedge(\neg B(z)\vee\Psi(z))\wedge Ab(z)\\&\quad\wedge\neg\Psi(b1)\wedge b1\neq b2\wedge B(b1)\wedge B(b2)].\end{aligned}\tag{23}$$

We next try to remove Ψ in (23).

Preprocessing. The formula (23) is already in the form which is the goal of this phase. To see this, we rewrite it as

$$\exists z \exists \Psi [\neg \Psi(b1) \wedge \forall y (\Psi(y) \vee \neg B(y) \vee Ab(y)) \\ \wedge (\Psi(z) \vee \neg B(z)) \wedge Ab(z) \wedge b1 \neq b2 \wedge B(b1) \wedge B(b2)]. \quad (24)$$

Preparation for Ackermann's lemma. Applying Proposition 1 (15) to $\neg \Psi(b1)$ in (24), we obtain

$$\exists z \exists \Psi \forall s [(\neg \Psi(s) \vee s \neq b1) \wedge \forall y (\Psi(y) \vee \neg B(y) \vee Ab(y)) \\ \wedge (\Psi(z) \vee \neg B(z)) \wedge Ab(z) \wedge b1 \neq b2 \wedge B(b1) \wedge B(b2)]. \quad (25)$$

Application of Ackermann's lemma. We apply Ackermann's lemma to (25), obtaining

$$\exists z \forall y [(y \neq b1 \vee \neg B(y) \vee Ab(y)) \\ \wedge (z \neq b1 \vee \neg B(z)) \wedge Ab(z) \wedge b1 \neq b2 \wedge B(b1) \wedge B(b2)]. \quad (26)$$

Let us simplify (26) before negating by using Proposition 1(15). (26) is equivalent to

$$\exists z[(\neg B(b1) \vee Ab(b1)) \\ \wedge (z \neq b1 \vee \neg B(z)) \wedge Ab(z) \wedge b1 \neq b2 \wedge B(b1) \wedge B(b2)]. \quad (27)$$

Negating and simplifying (27) results in

$$\forall z[(B(b1) \wedge \neg Ab(b1)) \\ \vee (z = b1 \wedge B(z) \vee \neg Ab(z) \vee b1 = b2 \vee \neg B(b1) \vee \neg B(b2)]. \quad (28)$$

The first-order formula (28) is logically equivalent to the second-order formula (17). Consequently,

$$Circ_{SO}(\Gamma(Ab, On); Ab; On) \equiv \Gamma(Ab, On) \wedge \\ \forall z[(B(b1) \wedge \neg Ab(b1)) \\ \vee (z = b1 \wedge B(z) \vee \neg Ab(z) \vee b1 = b2 \vee \neg B(b1) \vee \neg B(b2)]. \quad (29)$$

At this stage, the algorithm terminates but we can continue simplifying relative to the original theory. (28) together with (12) implies

$$Ab(b1) \rightarrow \forall z(\neg Ab(z) \vee (z = b1 \wedge B(z))) \tag{30}$$

and thus implies

$$\forall z(Ab(z) \supset (z = b1 \wedge B(z))), \tag{31}$$

which states that for any object z, either it is normal ($\neg Ab(z)$) or it is a block and $b1$. In other words, the only abnormal object is the block $b1$.

Example 3 (Birthday Example) This example contains both existentially quantified and universal formulae. In addition, it contains both unary and binary predicates.

Let $\Gamma(Ab, G)$ be the theory

$$[\exists x \exists y(B(y) \wedge F(x,y) \wedge \neg G(x,y))] \\ \wedge [\forall x \forall y(B(y) \wedge F(x,y) \wedge \neg Ab(x,y) \supset G(x,y))], \qquad (32)$$

where B, F and G are abbreviations for $Birthday$, $Friend$ and $Gives - Gift$, respectively. Here $Ab(x,y)$ has the following intuitive interpretation: "x behaves abnormally w.r.t. y in the situation when y has a birthday and x is a friend of y". The circumscription of $\Gamma(Ab, G)$ with Ab minimized and G varied is

$$Circ_{SO}(\Gamma(Ab, G); Ab; G) \equiv \\ \Gamma(Ab, G) \wedge \forall \Phi \forall \Psi[\Gamma(\Phi, \Psi) \wedge [\Phi \leq Ab] \supset [Ab \leq \Phi]], \qquad (33)$$

where

$$\begin{aligned}
\Gamma(\Phi,\Psi) &\equiv [\exists x\exists y(B(y)\wedge F(x,y)\wedge\neg\Psi(x,y))]\wedge \\
&\quad\ [\forall x\forall y(B(y)\wedge F(x,y)\wedge\neg\Phi(x,y)\supset\Psi(x,y))] && (34)\\
\Phi\leq Ab &\equiv \forall x\forall y[\Phi(x,y)\supset Ab(x,y)] && (35)\\
Ab\leq\Phi &\equiv \forall x\forall y[Ab(x,y)\supset\Phi(x,y)]. && (36)
\end{aligned}$$

In the following, we will reduce

$$\forall\Phi\forall\Psi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\supset[Ab\leq\Phi]] \qquad (37)$$

in (33). Negating (37), we obtain

$$\exists\Phi\exists\Psi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\wedge\neg[Ab\leq\Phi]]. \qquad (38)$$

We remove Ψ first.

Preprocessing. Replacing $\Gamma(\Phi, \Psi)$, $\Phi \leq Ab$ and $Ab \leq \Phi$ by their equivalents given by (34)–(36), eliminating $\supset$, renaming individual variables and moving existential quantifiers over individual variables to the left, we obtain

$$\begin{aligned}\exists x \exists y \exists q \exists r \exists \Phi \exists \Psi [B(y) \wedge F(x,y) \wedge \neg \Psi(x,y) \\ \wedge \forall u \forall z (\neg B(z) \vee \neg F(u,z) \vee \Phi(u,z) \vee \Psi(u,z)) \\ \wedge \forall s \forall t (\neg \Phi(s,t) \vee Ab(s,t)) \wedge Ab(q,r) \wedge \neg \Phi(q,r)]. \end{aligned} \tag{39}$$

Preparation for Ackermann's lemma. (39) is in the form suitable for application of Ackermann's lemma. To see this, we rewrite it as

$$\begin{aligned}\exists x \exists y \exists q \exists r \exists \Phi \exists \Psi \forall u \forall z [(\Psi(u,z) \vee \neg B(z) \vee \neg F(u,z) \vee \Phi(u,z)) \\ \wedge \neg \Psi(x,y) \wedge B(y) \wedge F(x,y) \\ \wedge \forall s \forall t (\neg \Phi(s,t) \vee Ab(s,t)) \wedge Ab(q,r) \wedge \neg \Phi(q,r)]. \end{aligned} \tag{40}$$

Application of Ackermann's lemma. Applying Ackermann's lemma to (40), we obtain

$$\exists x \exists y \exists q \exists r \exists \Phi [(\neg B(y) \vee \neg F(x,y) \vee \Phi(x,y)) \wedge B(y) \wedge F(x,y) \\ \wedge \forall s \forall t (\neg \Phi(s,t) \vee Ab(s,t)) \wedge Ab(q,r) \wedge \neg \Phi(q,r)]. \qquad (41)$$

We now remove Φ in (41).

Preprocessing. (41) is in the form which is the goal of this phase.

Preparation for Ackermann's lemma. Using Proposition 1 (15), we replace (41) by

$$\exists x \exists y \exists q \exists r \exists \Phi \forall v \forall w [(\Phi(v,w) \vee v \neq x \vee w \neq y \vee \neg B(y) \vee \neg F(x,y)) \\ \wedge \forall s \forall t (\neg \Phi(s,t) \vee Ab(s,t)) \wedge \neg \Phi(q,r) \wedge B(y) \wedge F(x,y) \wedge Ab(q,r)]. \qquad (42)$$

Application of Ackermann's lemma. Applying Ackermann's lemma to (42), we obtain

$$\exists x\exists y\exists q\exists r\forall s\forall t[(s \neq x \vee t \neq y \vee \neg B(y) \vee \neg F(x,y) \vee Ab(s,t)) \wedge (q \neq x \vee r \neq y \vee \neg B(y) \vee \neg F(x,y)) \wedge B(y) \wedge F(x,y) \wedge Ab(q,r)]. \quad (43)$$

Before negating, we simplify (43) which results in

$$\exists x\exists y\exists q\exists r[(\neg B(y) \vee \neg F(x,y) \vee Ab(x,y)) \wedge (q \neq x \vee r \neq y \vee \neg B(y) \vee \neg F(x,y)) \wedge B(y) \wedge F(x,y) \wedge Ab(q,r)]. \quad (44)$$

Negating (44), we obtain

$$\forall x\forall y\forall q\forall r[(B(y) \wedge F(x,y) \wedge \neg Ab(x,y)) \vee (q = x \wedge r = y \wedge B(y) \wedge F(x,y)) \vee \neg B(y) \vee \neg F(x,y) \vee \neg Ab(q,r)]. \quad (45)$$

(45) is logically equivalent to

$$\forall x \forall y \forall q \forall r [\neg (B(y) \wedge F(x,y)) \vee ((B(y) \wedge F(x,y)) \\ \wedge (\neg Ab(x,y) \vee (q = x \wedge r = y))) \vee \neg Ab(q,r)], \tag{46}$$

which is equivalent to

$$\forall x \forall y \forall q \forall r [\neg (B(y) \wedge F(x,y)) \vee \neg Ab(x,y) \\ \vee (q = x \wedge r = y) \vee \neg Ab(q,r)]. \tag{47}$$

The first-order formula (47) is logically equivalent to the second-order formula (37). Consequently,

$$Circ_{SO}(\Gamma(Ab,G); Ab; G) \equiv \\ \Gamma(Ab,G) \wedge \forall x \forall y \forall q \forall r [\neg (B(y) \wedge F(x,y)) \vee \neg Ab(x,y) \\ \vee (q = x \wedge r = y) \vee \neg Ab(q,r)]. \tag{48}$$

A more informative sentence, equivalent to (47), is

$$\forall x \forall y \forall q \forall r [Ab(x,y) \wedge Ab(q,r) \wedge B(y) \wedge F(x,y) \supset (q = x \wedge r = y)] \quad (49)$$

(49), together with the theory $\Gamma(Ab, G)$, states that there is exactly one pair of individuals, x and y, such that y has a birthday, x is a friend of y and x does not give a gift to y.

Example 4 (Hospital Example) Let Γ be the theory

$$[\forall x \exists y(Ab(x,y) \supset H(x,y))] \\ \wedge[\forall x \exists y(\neg Ab(x,y) \supset H(x,y))]. \tag{50}$$

Here $H(x,y)$ and $Ab(x,y)$ are to be intuitively interpreted as "x is in a hospital in a situation y" and "x behaves abnormally in a situation y", respectively. The circumscumscription of Γ, with Ab minimized and H varied is

$$Circ_{SO}(\Gamma; Ab; H) \equiv \\ \Gamma \wedge \forall \Phi \forall \Psi[\Gamma(\Phi,\Psi) \wedge [\Phi \leq Ab] \supset [Ab \leq \Phi]], \tag{51}$$

where

$$\Gamma(\Phi,\Psi) \equiv \quad \forall x \exists y[\Phi(x,y) \supset \Psi(x,y)] \\ \wedge \forall x \exists y[\neg\Phi(x,y) \supset \Psi(x,y)] \tag{52}$$

$$\Phi \leq Ab \equiv \quad \forall x \forall y[\Phi(x,y) \supset Ab(x,y)] \tag{53}$$

$$Ab \leq \Phi \equiv \quad \forall x \forall y[Ab(x,y) \supset \Phi(x,y)]. \tag{54}$$

In the following, we will reduce

$$\forall\Phi\forall\Psi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\supset[Ab\leq\Phi]] \tag{55}$$

in (51). Negating (55), we obtain

$$\exists\Phi\exists\Psi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\wedge\neg[Ab\leq\Phi]]. \tag{56}$$

Since we will try to remove Φ first, we replace (56) by its equivalent given by

$$\exists\Psi\exists\Phi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\wedge\neg[Ab\leq\Phi]]. \tag{57}$$

Preprocessing. Replacing $\Gamma(\Phi,\Psi)$, $\Phi\leq Ab$ and $Ab\leq\Phi$ by their equivalents given by (52)–(54), eliminating $\supset$ and renaming individual variables, we obtain

$$\begin{aligned}\exists\Psi\exists\Phi[&\forall x\exists y(\neg\Phi(x,y)\vee\Psi(x,y))\wedge\forall q\exists r(\Phi(q,r)\vee\Psi(q,r))\\&\wedge\forall u\forall v(\neg\Phi(u,v)\vee Ab(u,v))\wedge\exists s\exists t(Ab(s,t)\wedge\neg\Phi(s,t))].\end{aligned} \tag{58}$$

Moving $\exists s \exists t$ to the left and rearranging the resulting formula, we obtain

$$\exists s \exists t \exists \Psi \exists \Phi [\forall q \exists r (\Phi(q,r) \vee \Psi(q,r)) \wedge \forall x \exists y (\neg \Phi(x,y) \vee \Psi(x,y)) \\ \wedge \forall u \forall v (\neg \Phi(u,v) \vee Ab(u,v)) \wedge Ab(s,t) \wedge \neg \Phi(s,t)]. \qquad (59)$$

Preparation for Ackermann's lemma. Using Proposition 1 (15), we replace $\Phi(q,r)$ in (59) by $\forall z \forall w (\Phi(z,w) \vee z \neq q \vee w \neq r)$. This results in

$$\exists s \exists t \exists \Psi \exists \Phi [\forall q \exists r (\forall z \forall w (\Phi(z,w) \vee z \neq q \vee w \neq r) \\ \vee \Psi(q,r)) \wedge \forall x \exists y (\neg \Phi(x,y) \vee \Psi(x,y)) \\ \wedge \forall u \forall v (\neg \Phi(u,v) \vee Ab(u,v)) \wedge Ab(s,t) \wedge \neg \Phi(s,t)] \qquad (60)$$

which is equivalent to

$$\exists s \exists t \exists \Psi \exists \Phi [\forall q \exists r \forall z \forall w (\Phi(z,w) \vee z \neq q \vee w \neq r \vee \Psi(q,r)) \\ \wedge \forall x \exists y (\neg \Phi(x,y) \vee \Psi(x,y)) \\ \wedge \forall u \forall v (\neg \Phi(u,v) \vee Ab(u,v)) \wedge Ab(s,t) \wedge \neg \Phi(s,t)]. \qquad (61)$$

We next eliminate $\exists r$ by Skolemization and move $\forall q$ to the right and $\forall z \forall w$ to the left. The resulting formula is (below f is the introduced function variable)

$$\begin{aligned}\exists s \exists t \exists \Psi \exists f \exists \Phi \forall z \forall w [(\Phi(z,w) \vee \forall q (z \neq q \vee w \neq f(q) \\ \vee \Psi(q, f(q)))) \wedge \forall x \exists y (\neg \Phi(x,y) \vee \Psi(x,y)) \\ \wedge \forall u \forall v (\neg \Phi(u,v) \vee Ab(u,v)) \wedge Ab(s,t) \wedge \neg \Phi(s,t)]. \end{aligned} \tag{62}$$

Application of Ackermann's lemma. Applying Ackermann's lemma to (62), we obtain

$$\begin{aligned}&\exists s \exists t \exists f \exists \Psi [\forall x \exists y (\forall q (x \neq q \vee y \neq f(q) \vee \Psi(q, f(q))) \vee \Psi(x,y)) \\ &\wedge \forall u \forall v (\forall q (u \neq q \vee v \neq f(q) \vee \Psi(q, f(q))) \vee Ab(u,v)) \\ &\wedge Ab(s,t) \wedge \forall q (s \neq q \vee t \neq f(q) \vee \Psi(q, f(q)))] \end{aligned} \tag{63}$$

which is equivalent to

$$\begin{aligned}&\exists s\exists t\exists f\exists\Psi\forall x\exists y\forall q\forall u\forall v[(x\neq q\vee y\neq f(q)\vee\Psi(q,f(q))\vee\Psi(x,y))\\&\quad\wedge(u\neq q\vee v\neq f(q)\vee\Psi(q,f(q))\vee Ab(u,v))\\&\quad\wedge Ab(s,t)\wedge(s\neq q\vee t\neq f(q)\vee\Psi(q,f(q)))].\end{aligned}\tag{64}$$

Since all occurrences of Ψ in each conjunct in (64) are positive, all the conjuncts including Ψ, together with $\exists\Psi$, can be removed by Proposition 2. This yields

$$\exists s\exists t\exists f\exists\Psi\forall x\exists y\forall q\forall u\forall v.Ab(s,t)\tag{65}$$

which reduces to

$$\exists s\exists t.Ab(s,t).\tag{66}$$

Since we negated the original formula before applying the algorithm, we now negate the result, obtaining

$$\forall s \forall t. \neg Ab(s,t). \tag{67}$$

The first-order formula (67) is logically equivalent to the second-order formula (55). Consequently,

$$Circ_{SO}(\Gamma; Ab; H) \equiv \Gamma \wedge \forall s \forall t. \neg Ab(s,t). \tag{68}$$

which implies

$$\forall x \exists y H(x,y). \tag{69}$$

This is a variant of an example from Reiter. Rather than using the function $city$ as Reiter does, we will use a relation $C(x, y)$ with suitable axioms.

Example 5 (Vancouver Example) We begin by defining the binary relation C with the intention that $C(x, y)$ holds iff the city of x is y. In our axiomatization, Reiter's axiom,

$$\forall x.\neg Ab(x) \supset city(x) = city(wife(x)) \tag{70}$$

is replaced with

$$\forall x \forall y \forall z.\neg Ab(x) \wedge C(x, y) \wedge C(wife(x), z) \supset y = z. \tag{71}$$

In addition, we add the following axiom guaranteeing that C represents a function:

$$\forall x \forall y \forall z.C(x, y) \wedge C(x, z) \supset y = z. \tag{72}$$

We do not require that all people live in cities, i.e. we reject the axiom

$$\forall x \exists y . C(x, y). \tag{73}$$

So, the distinction is that our representation of the $city$ function is partial, whereas Reiter's is total. Intuitively, our choice seems to make more sense.

Let $\Gamma(Ab, C)$ be the theory

$$[\forall x \forall y \forall z . \neg Ab(x) \wedge C(x, y) \wedge C(wife(x), z) \supset y = z] \tag{74}$$

The circumscription of $\Gamma(Ab, City)$ with Ab minimized and C varied is

$$\begin{aligned} &Circ_{SO}(\Gamma(Ab, C); Ab; C) \equiv \\ &\quad \Gamma(Ab, C) \wedge \forall \Phi \forall \Psi [\Gamma(\Phi, \Psi) \wedge [\Phi \leq Ab] \supset [Ab \leq \Phi], \end{aligned} \tag{75}$$

where

$$\begin{aligned}
\Gamma(\Phi,\Psi) &\equiv [\forall x\forall y\forall z.\neg\Phi(x)\wedge\Psi(x,y)\wedge\Psi(wife(x),z)\supset y=z] \\
&\quad \wedge[\forall x\forall y\forall z.\Psi(x,y)\wedge\Psi(x,z)\supset y=z]. && (76)\\
\Phi\leq Ab &\equiv \forall x.\Phi(x)\supset Ab(x) && (77)\\
Ab\leq\Phi &\equiv \forall x.Ab(x)\supset\Phi(x). && (78)
\end{aligned}$$

In the following, we will reduce

$$\forall\Phi\forall\Psi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\supset[Ab\leq\Phi]] \qquad (79)$$

in (75). Negating (79), we obtain

$$\exists\Phi\exists\Psi[\Gamma(\Phi,\Psi)\wedge[\Phi\leq Ab]\wedge\neg[Ab\leq\Phi]]. \qquad (80)$$

We start by removing Ψ.

Preprocessing. Replacing $\Gamma(\Phi, \Psi)$, $\Phi \leq Ab$ and $Ab \leq \Phi$ by their equivalents given by (76)–(78), eliminating $\supset$ and renaming individual variables, we obtain

$$\begin{aligned}&\exists\Phi\exists\Psi[\forall x\forall y\forall z(\Phi(x)\vee\neg\Psi(x,y)\vee\neg\Psi(wife(x),z)\vee y=z)\wedge\\&\quad\forall u\forall v\forall w(\neg\Psi(u,v)\vee\neg\Psi(u,w)\vee v=w))\wedge\\&\quad\forall s(\neg\Phi(s)\vee Ab(s))\wedge\exists t(Ab(t)\wedge\neg\Phi(t))].\end{aligned}\tag{81}$$

Note that Ψ can be removed directly using Proposition 2, since all instances of Ψ occurring in (81) are negative. To prepare the latter formula into the form suitable for the application of Proposition 2, we move all quantifiers ranging over individual variables to the left. This results in

$$\begin{aligned}&\exists\Phi\exists\Psi\forall x\forall y\forall z\forall u\forall v\forall w\forall s\exists t[(\Phi(x)\vee\neg\Psi(x,y)\\&\qquad\vee\neg\Psi(wife(x),z)\vee y=z)\\&\quad\wedge(\neg\Psi(u,v)\vee\neg\Psi(u,w)\vee v=w)\\&\quad\wedge(\neg\Phi(s)\vee Ab(s))\wedge Ab(t)\wedge\neg\Phi(t)].\end{aligned}\tag{82}$$

Applying Proposition 2 to (82), we obtain

$$\exists\Phi\forall x\forall y\forall z\forall u\forall v\forall w\forall s\exists t[(\neg\Phi(s)\vee Ab(s))\wedge Ab(t)\wedge\neg\Phi(t)] \quad (83)$$

which reduces to

$$\exists\Phi\forall s\exists t[(\neg\Phi(s)\vee Ab(s))\wedge Ab(t)\wedge\neg\Phi(t). \quad (84)$$

We next try to remove Φ. Again, this can be done directly using Proposition 2, since all instances of Φ in (84) are negative. This results in

$$\forall s\exists t.Ab(t) \quad (85)$$

which is equivalent to

$$\exists t.Ab(t). \quad (86)$$

Taking the negation of (86) results in

$$\forall t.\neg Ab(t). \tag{87}$$

The first-order formula (87) is logically equivalent to the second-order formula (79). Consequently,

$$Circ_{SO}(\Gamma; Ab; C) \equiv \Gamma \wedge \forall t \neg Ab(t). \tag{88}$$

Remarks on the algorithm

- The algorithm always terminates and is sound, i.e. the output first-order formula, if obtained, is equivalent to the input second-order formula.
- There are two reasons causing the failure of the algorithm: (1) impossibility of separating positive and negative occurrences of the predicate variable Φ during the preprocessing phase; (2) the impossibility of unskolemization during the third phase.
- Although the algorithm may seem a bit complex, the calculations it describes may be performed without any computer support.

- For the sake of clarity the algorithm is not presented in its most efficient form. The possible directions of its optimization follow from the examples presented throught the paper.

- Observe that one usually deals with the elimination problem over a first-order definable class of models. In such a case it is sometimes possible to simplify considerably the input formula before running the algorithm. Such a possibility can be considered as an additional heuristics in the preprocessing phase.

Algorithm has been implemented at Linköping University and may be used on line.

http://www.ida.liu.se/labs/kplab/projects/dls/