Generative Structure of Entailment in Finite Constraint Networks

FRANCE DACAR france.dacar@ijs.si
Jožef Stefan Institute, Jamova 39, 61000 Ljubljana, Slovenia

Abstract. We characterize complete inference systems for entailment in negative finite constraint networks. The precise statement of the result is as follows. We fix an order ideal E of tuples, where each tuple in E is interpreted as a disallowed combination of values; this interpretation determines the entailment in E. We then identify certain inferences in E, which we call *prime*, and show that they are partitioned into classes so that the following holds: a set of minimal inferences, together with all unary inferences, generates the entailment in E if and only if it contains at least one prime inference from each class. This prime decomposition theory can be developed for a general closure operator in a finite set; we actually do it first in this wider context and then specialize to negative constraint networks.

It is one thing to know that prime inferences exist, and quite another to actually determine them. As of now we are able to present all prime inferences in the set of all at most ternary tuples on up to seven two-valued variables, and all prime inferences in the set of all at most binary tuples on up to six three-valued variables. Prime inferences can be used to determine the power of a local consistency technique; we give examples.

Keywords: finite constraint satisfaction problem, negative constraint network, local consistency technique, almost complete inference system, prime inference.

1. Introduction

A negative finite constraint network is simply a set of tuples, where each tuple represents a disallowed combination of values; an assignment of values to all variables thus satisfies the constraint network if and only if it is incompatible with every tuple in the network. This notion of satisfaction determines the corresponding semantic entailment in the set of all tuples for the given variables and their domains: a semantic consequence of a set of tuples is any tuple incompatible with every assignment of values to all variables that is incompatible with every tuple in the given set. We may also observe the semantic entailment on a subset of the set of all tuples, say on the set of all tuples whose scopes belong to some given scheme. The central theme of the paper is examination of complete inference systems for the entailment in certain sets of tuples, and the main task is characterization of complete inference systems that are as small as possible.

The enquiry into the generative structure of retricted entailment has grown out of the work on the problem of determining the least level of local consistency we must enforce to derive all semantic consequences on the initially given set of tuples (the enforcing of local consistency will normally be done on a larger set of tuples, but we are interested only in the derived tuples that belong to the initial set). Our problem is a variation of the problem of determining the level of local consistency sufficient to ensure global consistency. Results in this direction relate the necessary level of local consistency to properties of the underlying scheme, or to properties of the constraints. Examples of the former are width (Freuder, 1982), and acyclicity (Beeri et al., 1983, and Dechter and Pearl, 1989); examples of the latter are binary networks of monotone relations (Montanari, 1974), tightness and looseness (van Beek and Dechter, 1994), functional and monotone constraints (Van Hentenryck et al., 1992), and 0/1/all-constraints (Cooper et al., 1994). There is one result that expresses the level of local consistency only in terms of the largest size of a domain and the maximum arity of constraints (Dechter, 1992), without relying on any really restrictive properties of either the scheme or the constraints.

Our approach departs from this line of enquiry in two respects: it does not require that constraint networks have any special properties, except that they must be within a given set of tuples; and, we are interested only in the consequences of networks that belong to the given set of tuples. The rationale of this approach is that the level of local consistency which need be enforced to derive the desired consequences may be much lower than the level needed to ensure the global consistency.

It is at this point that inference systems enter the scene. Suppose we have a set of (valid) inference rules using which we can derive all semantic inferences in the given set of tuples. We can verify whether enforcing a certain level of local consistency suffices to derive all desired consequences, by enforcing it on the sets of premises of all inference rules, to see whether this will get us to their conclusions; if we can derive all the inference rules in this manner, then the level of local consistency is sufficient, otherwise it is not. Since it is clearly desirable that there are as few inference rules as possible, the following questions naturally arise: What are minimal complete inference systems like? What properties they have? How can we find them? The paper tries to answer these questions.

The central result is the following structural characterization of minimal inference systems in the case when the set of tuples on which we observe the entailment is an order ideal of the set of all tuples. We identify certain inferences, which we call *prime*, and show that they are partitioned into classes so that the following holds: a set of minimal inferences, together with all unary inferences, is a complete inference system (in the given order ideal of tuples) if and only if it contains at least one prime inference from each class. We will in fact develop this 'prime decomposition theory' first for a general closure operator in a finite set and will then specialize it to the restricted entailment.

The paper is organized in two parts. The first part examines general closure operators and minimal inference systems that generate them. The second part specializes the results of the first part to the context of negative constraint networks. In the last two sections we list all prime inferences of small sizes for ternary two-valued and binary three-valued networks, and then use them to determine the strength of a low-level local consistency.

2. Prime inferences for closure operators

Many problems related to semantic entailment in negative finite constraint networks can be better understood and more elegantly solved when they are formulated for general closure operators in finite sets.

In this section we first review basic facts about closure operators and inference systems. This done, we introduce derivation with subsumption and prime inferences, two themes that will reappear later on in the context of constraint networks.

Derivation with subsumption generalizes ideas of propositional resolution with subsumption. We present it as a reduction relation and give some sufficient conditions for uniqueness of a reduced result.

Prime inferences are, informally speaking, irreducible units of inference; they are partitioned into classes so that within each class they are interchangeable with each other. If we choose a representative prime inference from each class and add the unary inferences, we always obtain a complete inference system. We will identify a property of closure operators which guarantees that, conversely, every inference system consisting of minimal and unary inferences contains at least one prime inference from each class.

2.1. Basics

A *closure operator* in a finite set E is a mapping $\gamma : \mathcal{P}E \to \mathcal{P}E$ which is

- *expansive*: $X \subseteq \gamma(X)$;
- *increasing*: $X \subseteq Y$ implies $\gamma(X) \subseteq \gamma(Y)$;
- *idempotent*: $\gamma(\gamma(X)) = \gamma(X)$.

For any subset X of E, the subset $\gamma(X)$ of E is called the *γ-closure*, or simply the *closure*, of the subset X. A set $X \subseteq E$ satisfying $\gamma(X) = X$ is said to be *closed under γ*, or *γ-closed*, or simply closed. The set $\mathcal{F} = \mathcal{F}(\gamma)$ of all γ-closed sets is a *closure system* in E, meaning that it is closed under intersections. The closure system $\mathcal{F}$ determines the closure operator γ, since the closure $\gamma(X)$ of a set X is the least of the sets in $\mathcal{F}$ containing X as a subset. Every closure system is associated with a unique closure operator in this way.

For two closure operators β and γ in the same set E, we define $\beta \leq \gamma$ to mean that $\beta(X) \subseteq \gamma(X)$ for every $X \subseteq E$. The relation $\beta \leq \gamma$ between closure operators is equivalent to the inclusion $\mathcal{F}(\beta) \supseteq \mathcal{F}(\gamma)$ between the corresponding closure systems.

A subset I of the set E is called *γ-independent* if $x \notin \gamma(I \setminus \{x\})$ for every $x \in I$.

Let γ be a closure operator in a finite set E, and A a subset of E. We define a closure operator $\gamma \,|\, A$ in the set A by $(\gamma \,|\, A)(X) := A \cap \gamma(X)$ for any subset X of A; the closure operator $\gamma \,|\, A$ is said to be *induced* in the subset A by the closure operator γ.

Dual to the notion of a closure operator is the notion of an *interior operator* $\alpha: \mathcal{P}E \to \mathcal{P}E$, which is

- *erosive*: $X \supseteq \alpha(X)$;
- *increasing*: $X \subseteq Y$ implies $\alpha(X) \subseteq \alpha(Y)$;
- *idempotent*: $\alpha(\alpha(X)) = \alpha(X)$.

A subset X of E such that $\alpha(X) = X$ is said to be *α-open*. The set $\mathcal{O} = \mathcal{O}(\alpha)$ of all α-open sets is an *interior system*, that is, it is closed under unions. Every interior system $\mathcal{O}$ determines an interior operator α, where $\alpha(X)$ is the largest subset of X belonging to $\mathcal{O}$. For any two interior operators α and α', $\alpha \leq \alpha'$ is equivalent to $\mathcal{O}(\alpha) \subseteq \mathcal{O}(\alpha')$. Complementation establishes a bijection between closure operators and interior operators in E: for every closure operator γ we have the companion interior operator $X \mapsto E \setminus \gamma(E \setminus X)$, and conversely.

If X and Y are two subsets of E such that $\gamma(X) \supseteq Y$, then we say that elements of the set Y are *γ-consequences* of the *set of premises* X, and write $X \vdash_\gamma Y$, or simply $X \vdash Y$ when the closure operator γ is known. A pair (X, y) such that $X \subseteq E$ and $y \in E$ and $\gamma(X) \ni y$ is called a *γ-inference* and is written $X \vdash_\gamma y$, or simply $X \vdash y$. An inference $X \vdash y$ with $|X| = m$ is said to be m-ary. The set of all γ-inferences is the *closed inference system* associated with the closure operator γ; it is closed in the sense that it satisfies the following two conditions:

- *reflexivity*: $X \vdash x$ for every $x \in X$;
- *transitivity*: if $X \vdash y$ for every $y \in Y$ (ie. $X \vdash Y$), and $Y \vdash z$, then $X \vdash z$.

A non-reflexive inference $X \vdash y$ (ie. with $y \notin X$) is said to be *proper*. Every closed inference system $\vdash$ corresponds to a unique closure operator γ, where the closure $\gamma(X)$ of $X \subseteq E$ is the set of all $y \in E$ satisfying $X \vdash y$.

More generally, an *inference system* in the set E is any set $\mathcal{R}$ of pairs (P, r) with $P \subseteq E$ and $r \in E$, called *inference rules*; an inference rule (P, r) is usually written $P \vdash r$. If X is a subset of E, let $\mathcal{R}(X)$ denote the set of all $y \in E$ for which there exists in $\mathcal{R}$ an inference rule $P \vdash y$ with $P \subseteq X$. A subset X of E is said to be *$\mathcal{R}$-closed* if $\mathcal{R}(X) \subseteq X$. All $\mathcal{R}$-closed sets form a closure system, whose corresponding closure operator γ is said to be *generated* by the inference system $\mathcal{R}$. This closure operator γ can be described as follows. Define $\mathcal{R}_0(X) := X$ and $\mathcal{R}_{n+1}(X) := X \cup \mathcal{R}(\mathcal{R}_n(X))$ for any natural number n; then $\gamma(X)$ is the union of the ascending chain of sets $X = \mathcal{R}_0(X) \subseteq \mathcal{R}_1(X) \subseteq \mathcal{R}_2(X) \subseteq \ldots$ (since we are in a finite set E, the chain eventually reaches $\gamma(X)$).

Formation of the closure $\gamma(X)$, where γ is the closure operator generated by an inference system $\mathcal{R}$, can be also described using derivations. An *$\mathcal{R}$-derivation* of a set $Y \subseteq E$ from a set $X \subseteq E$ is a finite (possibly empty) sequence of inference rules taken from $\mathcal{R}$, say $P_1 \vdash r_1$, $P_2 \vdash r_2$, ..., $P_n \vdash r_n$, so that $P_i \subseteq X \cup \{r_1, \ldots, r_{i-1}\}$ for $i = 1, 2, \ldots, n$, and $Y \subseteq X \cup \{r_1, \ldots, r_n\}$. We have $X \vdash_\gamma Y$ precisely when there exists an $\mathcal{R}$-derivation of Y from X.

If $\mathcal{R}$ is an inference system and γ is the closure operator generated by $\mathcal{R}$, then the γ-inferences $X \vdash_\gamma y$ are said to be *derivable* from $\mathcal{R}$. The reflexivity and transitivity conditions for closed inference systems can be perceived as inference rules in the set $(\mathcal{P}E) \times E$, and as such they determine the *derivation closure operator* in this set. The derivation closure of an inference system $\mathcal{R} \subseteq (\mathcal{P}E) \times E$ is the least closed inference system $\mathcal{R}^* \supseteq \mathcal{R}$; it consists of all inferences derivable from $\mathcal{R}$.

Let γ be a closure operator, $\mathcal{R}$ an inference system, and ρ the closure operator generated by $\mathcal{R}$. The inference system $\mathcal{R}$ is said to be *sound* relative to the closure operator γ if $\rho \leq \gamma$, and it is said to be *complete* relative to γ if $\rho \geq \gamma$.

The γ-closure of the empty set may be nonempty, for a general closure operator γ; elements $a \in \gamma(\emptyset)$ correspond to nullary γ-inferences $\vdash_\gamma a$. Of particular interest are the unary γ-inferences $x \vdash_\gamma y$ with $x, y \in E$; they determine a preorder in the set E. Let us call a closure operator γ *simple*, if $\gamma(\emptyset) = \emptyset$ and the relation $x \vdash_\gamma y$ is antisymmetric, (ie. it is a partial order, not just a preorder).

With any closure operator γ can be associated a simple closure operator γ', as follows. Put $O := \gamma(\emptyset)$, denote the equivalence relation "$x \vdash_\gamma y$ and $y \vdash_\gamma x$" in the set $E \setminus O$ by $x \,\omega\, y$, and let σ be the canonical projection of the set $E \setminus O$ onto the quotient set $E' := (E \setminus O)/\omega$. Define the quotient closure operator $\gamma' = \gamma/\omega$ in the set E' by $\gamma'(X') := \sigma(\gamma\sigma^{-1}(X') \setminus O)$ for $X' \subseteq E'$; then γ' is a simple closure operator. If we know the set O, the equivalence relation ω, and the simple closure operator γ', we are able to reconstruct γ, since we have $\gamma(X) = O \cup \sigma^{-1}\gamma'\sigma(X \setminus O)$ for $X \subseteq E$.

This reduction of an arbitrary closure operator to a simple one means that we can almost always assume that a closure operator under discussion is simple, since any results formulated for simple closure operators can be extended to arbitrary closure operators in an obvious way.

Now we introduce the notion of a minimal inference. We will define it only for a simple closure operator γ; it easily generalizes to an arbitrary closure operator. Let $\vdash$ be the corresponding closed inference system. When referring to the set E as an ordered set, we shall always have in mind the ordering relation $x \vdash y$.

A *minimal consequence* of a set $X \subseteq E$ is a minimal element in the set of all consequences of X. That is, an element y is a minimal consequence of a set X if $X \vdash y$ holds and $X \vdash y' \vdash y$ implies $y' = y$. A set X is said to be a *minimal set of premises* for a consequence y, if $X \vdash y$ while $X' \not\vdash y$ for any proper subset X' of X. Now a *minimal inference* is an inference $X \vdash y$, where y is a minimal consequence of X and X is a minimal set of premises for y. The only unary minimal inferences, and also the only reflexive minimal inferences, are the identities $x \vdash x$.

Note that in an inference $X \vdash y$ whose set of premises X is minimal for the consequence y, the premises are independent. Indeed, suppose that $X \setminus \{x\} \vdash x$ for some $x \in X$; then $X \setminus \{x\} \vdash X \vdash y$, contradicting minimality of X.

Let $X \vdash y$ be an inference. Choose any minimal subset X' of X such that still $X' \vdash y$, and then choose a minimal consequence y' of X' such that $y' \vdash y$. The inference $X' \vdash y'$ is minimal, since $X'' \subseteq X'$ and $X'' \vdash y' \vdash y$ imply $X'' = X'$.

Choosing X' and y' in the opposite order, first a minimal consequence y' of X such that $y' \vdash y$ and then a minimal subset X' of X such that $X' \vdash y'$, we also obtain a minimal inference $X' \vdash y'$.

If $X \vdash y$ is any γ-inference, then either $x \vdash y$ for some $x \in X$, or $x \nvdash y$ for all $x \in X$ and there exists a proper minimal inference $P \vdash r$ with $P \subseteq X$ and $r \vdash y$. It follows from this observation that the proper minimal inferences together with the proper unary inferences generate the closure operator.

2.2. Derivation with subsumption

We will describe a procedure that computes the set of all minimal consequences for an arbitrary closure operator γ in a finite set E. The procedure generalizes the propositional resolution with subsumption.

In order to facilitate the discussion we assume that the closure operator γ is simple. We will write the corresponding closed inference system as $\vdash$. The set E is partially ordered by the relation $x \vdash y$; for any subset X of E, denote by $\mathrm{Min}(X)$ the set of all minimal elements of X in this partial ordering. Let ν be the closure operator in E generated by the unary γ-inferences. Then $X \vdash_\nu y$ means that $x \vdash y$ for some $x \in X$, that is, the closure $\nu(X)$ of $X \subseteq E$ is the order filter of the set E generated by the subset X. Note that $x \vdash_\nu y$ is equivalent to $x \vdash y$.

Assume, from now on, that we have an inference system $\mathcal{R}$ such that $\mathcal{R}$ and ν together generate γ. A subset X of E is then closed under γ if and only if it is closed under both $\mathcal{R}$ and ν.

LEMMA 1 *With E, γ, ν, and $\mathcal{R}$ as in the text, the following properties of a subset X of E are equivalent to each other:*

1. $\gamma(X) = \nu(X)$;
2. $X \supseteq \mathrm{Min}(\gamma(X))$;
3. $\mathcal{R}(\nu(X)) \subseteq \nu(X)$.

Proof: Because $\gamma(X) = \gamma(\nu(X))$, condition 1 means that $\nu(X)$ is closed under γ. Since $\nu(X)$ is closed under ν, it is closed under γ if and only if it is closed under $\mathcal{R}$; this proves the equivalence of conditions 1 and 3.

If $\gamma(X) = \nu(X)$, then $\mathrm{Min}(\gamma(X)) = \mathrm{Min}(\nu(X)) = \mathrm{Min}(X) \subseteq X$. Conversely, if $X \supseteq \mathrm{Min}(\gamma(X))$, then $\gamma(X) = \nu(\mathrm{Min}(\gamma(X))) \subseteq \nu(X)$. The conditions 1 and 2 are thus equivalent. ∎

We associate with $\mathcal{R}$ and ν a reduction relation $\rightarrow_{\mathcal{R}\nu}$ in the set $\mathcal{P}E$, in the following way:

- if $X \subseteq E$ contains two different elements p and q such that $p \vdash q$, then we may remove the element q from X, denoting the removal as $X \rightarrow_\nu X \setminus \{q\}$;

- if there is an inference rule $P \vdash r$ in $\mathcal{R}$ such that $P \subseteq X$ and $r \notin \nu(X)$, then we may add r to X, denoting the addition as $X \rightarrow_{\mathcal{R}} X \cup \{r\}$;

- the reduction relation $\rightarrow_{\mathcal{R}\nu}$ is the union of the relations $\rightarrow_\nu$ and $\rightarrow_{\mathcal{R}}$.

We will call $X \rightarrow_\nu X'$ a *subsumption step*, and $X \rightarrow_{\mathcal{R}} X'$ an *$\mathcal{R}$-derivation step*, or simply a *derivation step*.

The reduction relation $\rightarrow_{\mathcal{R}\nu}$ is terminating, since a derivation step strictly increases $\nu(X)$, while a subsumption step leaves $\nu(X)$ unchanged but strictly decreases X. Every sequence of reduction steps starting with a set X will eventually stop with a reduced set X_*. The result of a reduction is not necessarily unique; to ensure the uniqueness, we have to impose on $\mathcal{R}$ and ν an additional condition.

Let us call a subset X of E *subclosed under $\mathcal{R}$ relative to ν*, if $\mathcal{R}(X) \subseteq \nu(X)$. We shall say that the inference system $\mathcal{R}$ is *subsumable relative to ν*, if whenever a subset X of E is subclosed under $\mathcal{R}$ relative to ν, the set $\nu(X)$ is closed under $\mathcal{R}$ (ie. if $\mathcal{R}(X) \subseteq \nu(X)$ implies $\mathcal{R}(\nu(X)) \subseteq \nu(X)$).

THEOREM 1 *Let E, γ, ν, and $\mathcal{R}$ be as in the text, and suppose that $\mathcal{R}$ is subsumable relative to ν. Then the reduction relation $\rightarrow_{\mathcal{R}\nu}$ is terminating and reduces every subset X of E to a unique reduced set, namely the set $\mathrm{Min}(\gamma(X))$.*

Proof: We have already shown that $\rightarrow_{\mathcal{R}\nu}$ is terminating. Suppose that a set $X \subseteq E$ reduces (in any way) to a reduced set X_*. We have $\gamma(X_*) = \gamma(X)$, because γ-closure is preserved along any reduction path. Since no subsumption applies to X_*, we have $\mathrm{Min}(X_*) = X_*$. Since no derivation step applies to X_*, we have $\mathcal{R}(X_*) \subseteq \nu(X_*)$, hence by subsumability $\mathcal{R}(\nu(X_*)) \subseteq \nu(X_*)$, implying that $\gamma(X_*) = \nu(X_*)$. Then $X_* = \mathrm{Min}(X_*) = \mathrm{Min}(\nu(X_*)) = \mathrm{Min}(\gamma(X_*)) = \mathrm{Min}(\gamma(X))$, so we have the unique reduced result as claimed. ■

In this paper we will meet only with the following special type of subsumability. Let us say that the inference system $\mathcal{R}$ is *submersible relative to ν* if the following holds: if $P \vdash r$ is any inference rule in $\mathcal{R}$, and we have for each $p \in P$ an element $p' \in E$ such that $p' \vdash p$ but $p' \not\vdash r$, then there exists in $\mathcal{R}$ an inference rule $P' \vdash r'$, whose set of premises P' is a subset of the set of all elements p' for $p \in P$, and whose conlusion r' satisfies $r' \vdash r$.

LEMMA 2 *Submersibility implies subsumability.*

Proof: Indeed, let $\mathcal{R}$ be submersible relative to ν. Let X be a subset of E such that $\mathcal{R}(X) \subseteq \nu(X)$, and suppose that there is an inference rule $P \vdash r$ in $\mathcal{R}$ with $P \subseteq \nu(X)$ and $r \notin \nu(X)$, thus violating $\mathcal{R}(\nu(X)) \subseteq \nu(X)$. For each $p \in P$ choose a $p' \in X$ such that $p' \vdash p$, and let P' be the set of all p'. By submersibility there exists in $\mathcal{R}$ an inference rule $P'' \vdash r'$ with $P'' \subseteq P' \subseteq X$ and $r' \vdash r$, whence r' belongs to $\nu(X)$, and so does r, contradiction. ■

2.3. Prime inferences

Throughout this section γ is a simple closure operator in a finite set E and $\vdash$ is the corresponding closed inference system. By a closed set we will always mean a γ-closed set, by an inference a γ-inference, and so on. Whenever we shall refer to the set E as an ordered set, we shall have in mind the ordering relation $x \vdash y$.

We denote by $\mathcal{M}$ the set of all proper minimal inferences $P \vdash r$, and by $\mathcal{U}$ the set of all proper unary inferences $p \vdash r$. All minimal and unary inferences mentioned without qualifications will be assumed proper.

We shall say that a subset $\mathcal{G}$ of $\mathcal{M}$ is an *almost complete* inference system if the inference system $\mathcal{G} \cup \mathcal{U}$ is complete (ie. it generates the closure operator γ).

The closure of a set X will be, for our purposes, a useful measure of the 'size' of the set X or an inference $X \vdash y$. This notion of the closure measuring the size is reflected by the following definition: given a subset A of E, we shall say that a subset X of E, or an inference $X \vdash y$, is A-*small*, if $\gamma(X)$ is a *proper* subset of $\gamma(A)$; we shall say that an element $x \in E$ is A-small, if $\{x\}$ is A-small.

We shall say that a subset X of E *reduces to* a subset X' of E if there exists a derivation of X' from X that uses only X-small minimal inferences and any unary inferences, and will refer to any such derivation as a *reduction* of X to X'. Likewise we shall say that an inference $X \vdash y$ *reduces to* another inference $X' \vdash y'$ if $y' = y$ and X reduces to X', and will consider any reduction of X to X' to be a reduction of $X \vdash y$ to $X' \vdash y$.

LEMMA 3 *If every element of a set $X \subseteq E$ is X-small, then any reduction of X to some other set uses only X-small unary inferences. This holds, in particular, when X is an independent set of at least two elements.*

Proof: We shall show that every element z derived along a reduction starting with X is X-small. This is true by the assumption about X when z belongs to X. If z is derived using an X-small minimal inference $P \vdash z$, then $\gamma(z) \subseteq \gamma(P) \subset \gamma(X)$. Finally, if z is derived using a unary inference $p \vdash z$, then since p is X-small and $p \vdash z$ is proper, we have $\gamma(z) \subset \gamma(p) \subset \gamma(X)$. ∎

All unary inferences, and hence all inferences, used in a reduction of a minimal inference $P \vdash r$ to some other inference, are P-small because P is independent and has at least two elements.

The relation of reducibility between minimal inferences is transitive, that is, it defines a preorder in the set $\mathcal{M}$. We shall say that two minimal inferences are *associated* if they reduce to each other; association is an equivalence relation in $\mathcal{M}$. We regard the reducibility relation as going downwards, so that when a minimal inference reduces to another minimal inference, the latter is below the former.

Minimal members of the set $\mathcal{M}$, where this set is preordered by the reducibility relation, will be called *prime inferences*.

If a prime inference $P \models r$ reduces to a minimal inference $Q \models r$, then $Q \models r$ reduces back to $P \models r$ and is therefore associated with $P \models r$. Any minimal inference associated with a prime inference is itself prime. The set of all prime inferences is partitioned into classes of associated prime inferences. Choosing one prime inference from each class we get a *set of representative prime inferences*.

We call an inference $X \models y$ *decomposable* if the consequence y can be derived from the premises X using only X-small minimal inferences and any unary inferences, and refer to any such derivation, in case it exists, as a *decomposition* of the inference $X \models y$. Clearly an inference $X \models y$ is decomposable if and only if it reduces to some X-small inference. An *indecomposable* inference is one that is not decomposable.

Lemma 4 *A prime inference is indecomposable.*

Proof: Let $X \models y$ be a decomposable minimal inference. There is some X-small minimal or unary inference $P \models y$ such that X reduces to P. Because $X \models P$, y is a minimal consequence of P, so $P \models y$ cannot be a proper unary inference. The minimal inference $X \models y$ therefore reduces to the minimal inference $P \models y$, which clearly does not reduce back, whence $X \models y$ is not prime. ■

We now consider the generative power of prime inferences.

Proposition 1 *Every set of representative prime inferences is an almost complete inference system.*

Proof: Let $\mathcal{R}$ be a set of representative prime inferences. It suffices to prove that any minimal inference $X \models y$ is derivable using the inferences taken from $\mathcal{R} \cup \mathcal{U}$. We will reason by induction on the closure $\gamma(X)$. The minimal inference $X \models y$ reduces to some prime inference and hence also to some prime inference $P \models y$ in $\mathcal{R}$. Every minimal inference used in a reduction of $X \models y$ to $P \models y$ is X-small, thus by induction hypothesis derivable using $\mathcal{R} \cup \mathcal{U}$, and it follows that the minimal inference $X \models y$ is also so derivable. ■

We now turn to the questions whether a set of representative prime inferences can have a proper subset which is still an almost complete inference system, and whether there are inclusion-minimal almost complete inference systems that are not sets of representative prime inferences. Both can happen, in general. In order to ensure that sets of representative prime inferences coincide with the minimal almost complete inference systems, we must impose some condition onto the simple closure operator γ. Here is such a condition:

We shall say that a simple closure operator γ is *skew* if for any two minimal inferences $P \models r$ and $Q \models s$, $P = Q$ implies $r = s$.

The following equivalent formulation of skewness might elucidate why we have chosen this term. The simple closure operator γ is skew if and only if the following holds: if $P \models r$ is a minimal inference and s is a minimal consequence of P different from r, then s is a consequence of a proper subset Q of the set P. Note that in the situation just described the inference $Q \vdash s$ is P-small since the closure of Q is disjoint with $P \setminus Q$, P being independent.

Given an inference $P \vdash r$ with the set P of premises minimal for the consequence r, we can always choose a minimal consequence $r' \vdash r$ of P and obtain a minimal inference $P \vdash r'$. When the closure operator γ is skew, it is clear that there is only one r' to choose.

We now show that when the closure operator is skew, the classes of associated prime inferences are independent in a very strong sense.

PROPOSITION 2 *Suppose that γ is skew, and let $P \vdash r$ be a prime inference. Then avery derivation of r from P by minimal and unary inferences uses some prime inference associated with $P \vdash r$.*

Proof: Let us have a derivation of r from P by minimal and unary inferences. We can assume that the derivation stops the first time it reaches r. Since $P \vdash r$ is indecomposable, the derivation must use some inference $Q \vdash s$ with $\gamma(Q) = \gamma(P)$; take the first such inference $Q \vdash s$ used in the derivation. We have $Q \vdash r$, and clearly P reduces to Q. We cannot have $Q = \{q\}$, since then we would have $P \vdash q \vdash r$, and the derivation would have already stopped with $q = r$; thus $Q \vdash s$ is a minimal inference. Because $P \vdash Q \vdash r$, r is a minimal consequence of Q. We must have $s = r$, for otherwise r would be a consequence of a proper subset of Q and we would have a decomposition of $P \vdash r$. The prime inference $P \vdash r$ reduces to the minimal inference $Q \vdash r$, thus the latter is a prime inference associated with the former. ∎

When the closure operator γ is skew, a removal of an entire class of associated prime inferences from the complete inference system $\mathcal{M} \cup \mathcal{U}$ yields an incomplete inference system. We have then the following characterization of the minimal almost complete inference systems:

THEOREM 2 *For a skew closure operator, the minimal almost complete inference systems are precisely the sets of representative prime inferences.*

Let us mention here one special and easily recognizable type of prime inferences. An inference $P \vdash r$ is called *primitive* if it is minimal and satisfies $\gamma(P) = P \cup \{r\}$. In a primitive inference $P \vdash r$, take any $p \in P$. The closure of the set $P \setminus \{p\}$ is a subset of $P \cup \{r\}$ and contains neither r (since $P \vdash r$ is minimal) nor p (because P is independent); that is, the set $P \setminus \{p\}$ is closed, whence it follows that every proper subset of P is closed. It is clear from this that a primitive inference can reduce only to itself, so it is certainly prime. A primitive inference $P \vdash r$ is completely independent of other inferences in the sense that when we remove it from the closed inference system, the set P becomes closed under all remaining inferences.

Let A be a subset of the set E. If $P \vdash r$, where $P \subseteq A$ and $r \in A$, is a minimal inference for the closure operator γ, then it is also a minimal inference for the induced closure operator $\gamma\,|\,A$. The converse does not always hold. However, if A is an order ideal of the ordered set E, then every minimal inference for $\gamma\,|\,A$ is also a minimal inference for γ. So we have the following:

PROPOSITION 3 *If the closure operator γ is skew and A is an order ideal of the ordered set E, then the induced closure operator $\gamma\,|\,A$ is also skew.*

We have seen that every prime inference is an indecomposable minimal inference. The converse does not hold, not even for a skew closure operator. But under an additional assumption about a minimal inference, its indecomposability does imply that it is prime. We shall say that an inference $X \vdash y$ is *nondescending* if $y \not\vdash x$ for every $x \in X$. For example, if y is a maximal element of the ordered set E, then every proper inference $X \vdash y$ is nondescending.

LEMMA 5 *Let the closure operator γ be skew. If $P \vdash r$ is a minimal inference and $Q \vdash r$ is a nondescending inference, then $P \vdash Q$ implies that P reduces to Q.*

Proof: For each $q \in Q$ choose first a minimal consequence $q' \vdash q$ of P, then a minimal set of premises $P_q \subseteq P$ for q'. Since $r \not\vdash q$, q' is a minimal consequence of P different from r, so all sets P_q are proper subsets of P. Since P is independent, every $\gamma(P_q)$ is a proper subset of $\gamma(P)$. It follows that a sequence of all the proper minimal inferences $P_q \vdash q'$, followed by a sequence of all the proper unary inferences $q' \vdash q$, is a reduction of P to Q. ■

PROPOSITION 4 *If the closure operator γ is skew, then every nondescending indecomposable minimal inference is prime.*

Proof: Let $P \vdash r$ be a nondescending indecomposable minimal inference. The minimal inference $P \vdash r$ reduces to some prime inference $Q \vdash r$. Since $P \vdash r$ is indecomposable, we have $\gamma(Q) = \gamma(P)$, hence $Q \vdash P$. Since $P \vdash r$ is nondescending, $Q \vdash r$ reduces to $P \vdash r$, and it follows that $P \vdash r$ is a prime inference (associated with the prime inference $Q \vdash r$). ■

3. Prime inferences in negative constraint networks

We will now apply the general 'prime decomposition theory', which we have developed for closure operators, to the semantic entailment in negative constraint networks.

This section has three parts. The first part introduces the necessary notions: positive and negative constraint networks, semantic entailment for negative constraint networks, local consistency of various types and degrees, and so on. We also take a look at the form the minimal semantic inferences take in negative constrain networks. In the second part we examine prime inferences for the entailment closure operator acting on negative constraint networks on the scheme consisting of

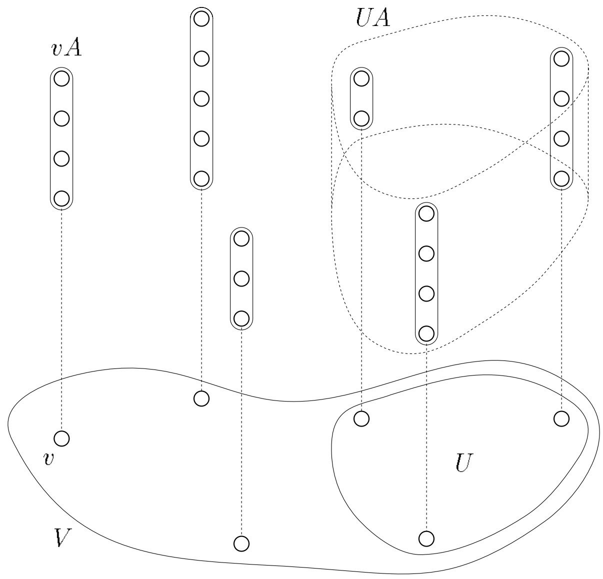

Figure 1.

all sets of at most m variables, for some natural number $m \geq 2$. In the last part we determine some prime inferences of small sizes.

3.1. Preliminaries

A constraint network consists of a set of variables which have to be assigned values from their domains so as to satisfy a set of constraints. Each constraint is a relation on some subset of variables. All constrained subsets of variables form the scheme of the constraint network; it is because of the scheme that we perceive constraints as arranged in a 'network'.

We will consider only finite constraint networks. That is, it will be always understood that there are only finitely many variables involved, and that each variable takes values in a finite domain.

In this section we specify mathematical structures that will be used to represent constituents of constraint networks and constraint networks themselves.

Let us have a set V of variables and a family $A = (\, A_v \mid v \in V\,)$ of their domains. We will refer to pairs (v, α), where $v \in V$ and $\alpha \in A_v$, as *tokens*. We say that a token $a = (v, \alpha)$ is *over* the variable v; we denote the variable v of the token a by $\operatorname{var}(a)$. For any subset U of the set of variables V we denote by UA the set of all tokens over variables $u \in U$; for a variable v we write vA instead of $\{v\}A$. We will graphically represent variables and tokens as in Figure 1.

We say that a variable v *appears* in a set of tokens $x \subseteq VA$ if it is the variable of some token in x; we define the *scope* of x, denoted $\operatorname{scope}(x)$, as the set of all variables appearing in x. For any set of sets of tokens $X \subseteq \mathcal{P}(VA)$ we call the set of all scopes of sets belonging to X the *underlying scheme* of X, and define the scope of X as the union of scopes of its members.

Let $U \subseteq V$. For any set of tokens x we call the set of tokens $x[U] := x \cap UA$ the *restriction* of x to U (we do not require U to be a subset of scope(x)). For any set of sets of tokens we call $X[U] := \{ x[U] \mid x \in X \}$ the *projection* of X to U.

We define a *tuple* as a set of tokens $x \subseteq VA$ that contains precisely one token $x(u)$ over each variable u in its scope. We denote the set of all tuples by $A^{\#}$. The *empty tuple* $\square$ is just the empty set considered a tuple. An *m-ary tuple* is a tuple consisting of m tokens. A set X of tuples such that $\bigcup X$ is a tuple is said to be *compatible*; it is easy to see that X is compatible if and only if any two tuples in X are compatible.

If x is a tuple and scope$(x) = U$, we say that x is a tuple *over* U. The set of all tuples over a set of variables $U \subseteq V$ is the direct product of the domains A_u for $u \in U$; we denote it by A^U. (This 'overloading' of exponential notation will never cause any confusion.) Sets A^U, for all $U \subseteq V$, form a partition of $A^{\#}$.

A *relation on a set of variables* $U \subseteq V$ is any subset of A^U. An *m-ary relation* is a relation on a set of m variables.

We shall refer to any set of subsets of V as a *scheme* and call its members *scopes*. We shall say that a scheme is *descending* if it is an order ideal of the powerset $\mathcal{P}V$ ordered by inclusion. Let C be any scheme. We denote the union of sets A^U for $U \in C$ by $A(C)$. Suppose we have a family $(R_U \mid U \in C)$ of relations $R_U \subseteq A^U$. If R is the union of all relations R_U for $U \in C$, then each relation can be recovered as $R_U = R \cap A^U$. This means that families of relations over scopes of the scheme C can be represented by subsets of $A(C)$.

For any natural number m we denote by $V^{(m)}$ the scheme in V of all m-element subsets of V, and by $V^{(*m)}$ the scheme of all at most m-element subsets of V. We shall also write $A^{(m)} := A(V^{(m)})$ and $A^{(*m)} := A(V^{(*m)})$.

Now, a *constraint network* is given by a set V of variables, a family A of domains for variables in V, a scheme C in V, and a family R of relations on scopes in C; we shall usually refer to it as the constraint network R on the scheme C, supposing that variables and domains are known. We will always assume that each domain contains at least two values. A constraint network is said to be *k-valued* if every domain consists of at most k values, and is said to be *m-ary* if every scope consists of at most m variables.

Given a constraint network, we must still specify how precisely the relations constrain the variables. This depends on whether we interpret the relations positively or negatively.

In the *positive* interpretation, each relation R_U is a set of admissible tuples over the scope U. Formally, a tuple s over V satisfies a positive constraint network R on a scheme C if for each scope U in the scheme C the restriction $s[U]$ belongs to R_U.

The *negative* interpretation is complementary, on the side of relations, to the positive one. A tuple s over V is taken to satisfy a negative constraint network Q on a scheme C if it satisfies the positive network $A(C) \setminus Q$ on C, that is, if $s[U] \notin Q_U$

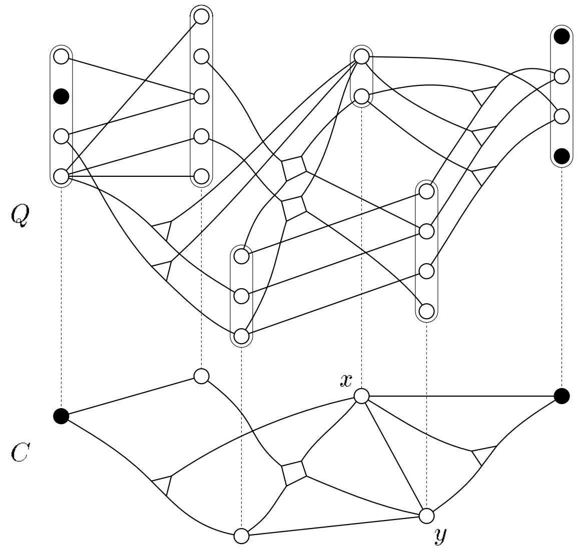

Figure 2.

for each scope U in the scheme C. Equivalently, a tuple $s \in A^V$ satisfies a negative network Q on C if $q \nsubseteq s$ for every tuple $q \in Q$.

A tuple that satisfies a constraint network (be it positive or negative) is also called a *solution* of the constraint network.

We will graphically represent a constraint network as shown in Figure 2. Unary scopes and tuples are the black nodes. For a binary scope or tuples we draw the line connecting the two nodes. Scopes and tuples of arity three or more are represented as 'neurons'; this notation is far more readable than the usual one, where sets of nodes are indicated by drawing fences around them. Note that the relation $Q_{\{x,y\}}$ is empty. Were the constraint network Q positive, it would be obviously unsatisfiable. As it happens, the network Q is negative, and so the emptiness of $Q_{\{x,y\}}$ means that the given constraint on the scope $\{x, y\}$ does not really constrain anything. Even so it makes sense to have the scope $\{x, y\}$ in the scheme, since an algorithm run on the network may install a proper constraint on it.

Sometimes we want to change the scheme of a constraint network to some smaller or larger scheme. When we narrow down the scheme by removing some scopes, we lose the constraints on the removed scopes; when we enlarge it by additional scopes, we install on each added scope an always satisfied constraint. Formally, let $D \subseteq C$ be two schemes. We define the *restriction* of a network $R \subseteq A(C)$ on the scheme C to the scheme D as the network $R|D := R \cap A(D)$ on the scheme D; we let the restricted network have the same polarity, positive or negative, as the network R. If a network S on D is the restriction of a network R on C, we also say that the network R is an *extension* of the network S. The *default extension* of a network S on the scheme D to the scheme C depends on polarity of S: if S is positive, the default extension is the positive network $S \cup A(D \setminus C)$; if S is negative, the default extension is S itself, regarded as a negative network on the scheme D.

We may also restrict a scheme, or a constraint network, to a subset V' of the set of variables. The restriction of a scheme C to V' is the scheme $C\,|\,V'$, consisting of all scopes $U \in C$ that are subsets of V'. For a network R on a scheme C, we define the restriction $R\,|\,V'$ as the restriction of R to the restricted scheme $C\,|\,V'$.

We shall use multiplicative notation for unions of tuples that have disjoint scopes and, occasionally, for unions of disjoint sets of variables. If we have, say, tuples x and y and a token a, then we write the union $x \cup y \cup \{a\}$ as xya, provided the scopes $\mathrm{scope}(x)$, $\mathrm{scope}(y)$, and $\mathrm{scope}(\{a\})$ are disjoint. We extend multiplicative notation to sets of tuples X and Y that have $\mathrm{scope}(X)$ and $\mathrm{scope}(Y)$ disjoint, writing XY for the set of all tuples xy with $x \in X$ and $y \in Y$. When $X \subseteq A^U$ and $Y \subseteq A^W$ are relations on disjoint sets U and W, the relation $XY \subseteq A^{U \cup W}$ is the direct product of relations X and Y.

3.2. Entailment, local consistency

Here we define the semantic entailment for negative constraint networks, and then introduce several types of local consistency of positive and negative constraint networks. All networks considered are assumed to be given on the set V of variables with domains A_v, $v \in V$.

Let us look, for a start, at negative constraint networks on the largest possible scheme $\mathcal{P}V$. In this case a negative constraint network is just any set of tuples $X \subseteq A^{\#}$; recall that a tuple $s \in A^V$ satisfies X if and only if $x \not\subseteq s$ for every tuple $x \in X$. This satisfaction relation determines the semantic entailment in the set $A^{\#}$. If X is a set of tuples, y is a tuple, and every $s \in A^V$ that satisfies X also satisfies y, then we write $X \models y$, saying that X *entails* y, and call y a *semantic consequence* of X. Told in the opposite direction, $X \models y$ means that every tuple $s \in A^V$ which includes the tuple y also includes some tuple $x \in X$. The entailment $\models$ is a closed inference system; we denote the corresponding closure operator by Conseq. It is easily shown that a tuple x entails a tuple y if and only if x is a subtuple of y; since also $\mathrm{Conseq}(\emptyset) = \emptyset$, the closure operator Conseq is simple.

Now let C be any scheme in V. We denote by Conseq_C the restriction of the closure operator Conseq to the subset $A(C)$ of $A^{\#}$; in the special case when $C = V^{(*m)}$ we write the restricted closure operator as Conseq_m. We will say that a negative network Q on C is *closed under entailment* if it is closed under Conseq_C.

Consider a positive network R on C. Let $S \subseteq A^V$ be the set of all solutions of R, and define the positive network R° on C by $R^\circ_U := S\,[U]$ for $U \in C$. Since R° has the same solutions as R, we have $R^{\circ\circ} = R^\circ$, whence $R \mapsto R^\circ$ is an interior operator in $A(C)$. It takes but a moment to see that this interior operator is just the companion to the closure operator Conseq_C, that is, $R^\circ = A(C) \setminus \mathrm{Conseq}_C(A(C) \setminus R)$.

The closure operator Conseq_C and its companion $R \mapsto R^\circ$ are complete in the sense that the former finds all semantic consequences belonging to $A(C)$ while the latter removes all tuples that cannot be extended to solutions. Since it is NP-hard to do what these two do, they are computationally expensive. Luckily they have many distant relatives, collectively known as 'local consistency techniques', which are computationally more affordable. A local consistency technique is a sound, but generally incomplete, operator (closure operator for negative, interior operator for positive networks), which enforces certain local 'compatibilities' or 'consistencies', that are present in every network produced by the complete operator.

We shall introduce three types of local consistency: internal consistency, weak relative consistency, and strong relative consistency.

From now on we consider only constraint networks on descending schemes; so let C be a descending scheme in V.

A positive network R on the scheme C is said to be *internally consistent* if $R_U[W] = R_W$ for any two scopes $W \subseteq U$ in C; it clearly suffices if this condition is satisfied for pairs of scopes $W \subset U$ where U differs from W by a single variable. Every union of internally consistent networks is an internally consistent network, so there exists in each positive network R on C a largest internally consistent subnetwork $R' \subseteq R$; we say that R' is obtained from R by enforcing the internal consistency. The mapping $R \mapsto R'$ is an interior operator in $A(C)$. Since the network R° is internally consistent, we have always $R^\circ \subseteq R'$.

Let us transfer the notion of internal consistency from positive to negative networks, via complementation in $A(C)$. A negative network Q on C is then internally consistent if and only if for any pair of scopes W and Wu in C, where u is a variable not in W, it satisfies the conditions

- for any tuple $x \in Q_W$ and any token $a \in uA$, the tuple xa belongs to Q_{Wu},
- if $x \in A^W$ is such that $xa \in Q_{Wu}$ for every token $a \in uA$, then $x \in Q_W$,

which can be read as inference rules in $A(C)$, namely:

1. $x \vdash xa\,,$ for any tuple $x \in A^W$ and any token $a \in A^u$;
2. $A^u x \vdash x\,,$ for any tuple $x \in A^W$.

These inference rules generate a closure operator in $A(C)$, the companion to the interior operator $R \mapsto R'$ defined in the preceding paragraph; we will call the corresponding closed inference system, written $\vdash$, the *internal inference* on the scheme C.

The internal inferences of type 1 obviously generate the unary inferences in $A(C)$. For any $Q \subseteq A(C)$, let $\mathrm{Ext}_C(Q)$ denote the closure of Q under the unary internal inferences in $A(C)$, ie. $\mathrm{Ext}_C(Q)$ is the set of all tuples in $A(C)$ that include some tuple belonging to Q. The set of all internal inferences of type 2, together with the unary inferences, generates the internal inference; this set of inferences is not subsumable, but can be expanded to a submersible set of internal inferences, which we now proceed to do. We will say that a set of tuples $P \subseteq A^\#$ is a *resolution fan with a hub* $v \in V$, if it satisfies the following conditions:

- every tuple in P contains a token over the variable v;
- every token over the variable v belongs to precisely one tuple in P;
- $r = (\bigcup P) \setminus (vA)$ is a tuple.

The tuple r is called the *resolvent* of the resolution fan P; the resolvent r is a semantic consequence of P, since every tuple $s \in A^V$ which includes r contains some token a over v, hence includes the tuple $p \in P$ which contains the token a; the inference $P \models r$ is called a *resolution inference*. A resolution fan P such that $\mathrm{scope}(P) \in C$, and the corresponding resolution inference, are said to be C-bounded.

Internal inferences of type 2 are special C-bounded resolution inferences. Conversely, if P is a C-bounded resolution fan with a hub v and a resolvent r, then the corresponding resolution inference is an internal inference, since it can be derived using the internal unary inferences $p \vdash p(v)r$ for $p \in P$, and the internal inference $A^v r \vdash r$. It follows that the C-bounded resolution inferences, together with the unary inferences in $A(C)$, generate internal inference in $A(C)$. It is clear that the set of all C-bounded resolution inferences is submersible. The derivation with subsumption which uses C-bounded resolution inferences for derivation steps therefore 'works'; we will call it the *C-bounded resolution* (*with subsumption*). We have the following result:

PROPOSITION 5 *For any negative constraint network Q on a descending scheme C, the negative constraint network $\mathrm{Ext}_C(Q)$ on C is internally closed if and only if Q is subclosed under C-bounded resolution inferences. The C-bounded resolution, when applied to an arbitrary negative constraint network Q on C, produces the set of all minimal internal consequences of Q.*

The condition in the first assertion of the proposition means, when spelled out, that for every C-bounded resolution inference $P \vdash r$ with $P \subseteq Q$, the resolvent r includes some tuple in Q.

When the scheme is $C = V^{(*m)}$, $m \geq 2$, the C-bounded resolution inferences are those resolution inferences whose resolvents are at most $(m-1)$-ary. We will name the $V^{(*m)}$-bounded resolution inferences according to the maximum allowed size of their resolvents, calling them $(m-1)$*-resolution inferences*, and will refer to the corresponding $V^{(*m)}$-bounded resolution as the $(m-1)$*-resolution*.

We can apply the $(m-1)$-resolution to any negative constraint network $Q \subseteq A^{\#}$, which may contain tuples of arity greater than m. There is a well-known notion of local consistency, the *strong m-consistency*, introduced by E. C. Freuder (Freuder, 1978, 1982). It can be characterized, for negative constraint networks, by means of the $(m-1)$-resolution inferences:

PROPOSITION 6 *A negative constraint network $Q \subseteq A^{\#}$ is strongly m-consistent if and only if it is subclosed under $(m-1)$-resolution inferences. The $(m-1)$-resolution enforces strong m-consistency.*

We will now define the notions of weak and strong relative consistency, this time only for negative constraint networks. Let $C \subseteq D$ be two descending schemes.

We will say that a negative constraint network Q on the scheme C is *weakly consistent relative to the scheme* D, if for every scope $W \in D$ the restricted negative network $Q|W$ is closed under the entailment closure operator $\mathrm{Conseq}_{C|W}$. It is clear that the set of all $Q \subseteq A(C)$ that are weakly consistent relative to D is a closure system in $A(C)$, and that the corresponding closure operator is generated by all semantic inferences $P \models r$ such that $\mathrm{scope}(P \cup \{r\}) \in D$.

We will say that a negative constraint network Q on the scheme C is (*strongly*) *consistent relative to the scheme* D, if it can be extended to an internally consistent negative network on D. Once more we have a closure system. The corresponding closure operator can be described as follows: given a negative network $Q \subseteq A(C)$, we regard it as a negative network on D, then apply to it the internal closure operator in $A(D)$, and finally restrict the result back to C. The strong consistency on C relative to D can thus be enforced by the D-bounded resolution, where we retain only those derived consequences that lie in $A(C)$.

When $C = V^{(*m)}$ and $D = V^{(*\ell)}$, $\ell \geq m \geq 2$, we will refer to the weak (strong) consistency on C relative to D as the *weak* (*strong*) *relative ℓ-consistency* of m-ary negative networks.

Let $m \geq 2$ and $k \geq 2$ be natural numbers, and suppose that we have for each family $A = (\, A_v \mid v \in V \,)$ of k-valued domains a closure operator γ^A in the set $A^{(*m)}$. We will call the family $\left(\gamma^A\right)$ a *consistency technique* if the closure operators γ^A satisfy the following conditions:

- Each closure operator γ^A is sound relative to the semantic entailment in $A^{(*m)}$, that is, $\gamma^A(Q) \subseteq \mathrm{Conseq}_m^A(Q)$ for every negative constraint network $Q \subseteq A^{(*m)}$.

- If $B = A|V'$ is a subfamily of a family A of k-valued domains, and Q is a subset of $B^{(*m)}$ (hence also a subset of $A^{(*m)}$), then $\gamma^B(Q) \subseteq \gamma^A(Q)$.

(Another natural thing to require of closure operators γ^A would be that they should act isomorphically in isomorphic situations. We do not need this for the simple point we want to make.) Closure operators γ^A will in general not be complete, but they may be complete up to a certain number of variables.

We define *the reach* of a consistency technique $\left(\gamma^A\right)$ as the largest natural number n such that $\gamma^A = \mathrm{Conseq}_m^A$ whenever the number of variables does not exceed n. The reach of a consistency technique $\left(\gamma^A\right)$ can be also characterized as the largest level of weak relative consistency enforced by every closure operator γ^A; this follows easily from the second requirement above.

Enforcing strong relative ℓ-consistency of m-ary k-valued constraint networks, where $\ell \geq m$, is certainly a consistency technique by our definition; we will denote its reach by $\mathrm{reach}(m, k; \ell)$. Obviously $\mathrm{reach}(m, k; \ell) \geq \ell$. It would be worthwhile to know how fast $\mathrm{reach}(m, k; \ell)$ increases as a function of ℓ, for fixed m and k (where either $m \geq 3$ of $k \geq 3$, to exclude the trivial case $m = k = 2$). A simple estimate shows that any rate of growth greater than $\ell \ln \ell$ would make the enforcement of strong relative ℓ-consistency, with ℓ depending on the number of variables, into a subexponential complete consistency technique, provided we had a good, efficiently computable lower bound for the reach. Since this is almost surely not true, it is a good guess that $\text{reach}(m, k; \ell) = O(\ell \ln \ell)$ (where the factor implicit in O depends on m and k). If the reach grows approximately as $\ell \ln \ell$, then the enforcement of strong relative ℓ-consistency is roughly equivalent, with regard to time complexity, to the direct enumeration.

We are still very far from any reliably close estimate of the reach. All we know at this moment are some rather unimpressive lower bounds for the reach, and values of the $\text{reach}(3, 2; \ell)$ and $\text{reach}(2, 3; \ell)$ for small values of ℓ. For example, it can be shown that

$$\text{reach}(2, 3; \ell) \geq 2\ell - 3$$

(we will not prove this here). Towards the end of the paper we will find that $7 \leq \text{reach}(3, 2; 5) \leq 8$ and that $\text{reach}(2, 3; 4) = 7$ (the latter is larger than the estimate $2 \cdot 4 - 3 = 5$).

3.3. Minimal inferences in negative constraint networks

In this section we examine the structure of minimal semantic inferences in negative constraint networks. We start off with the simple but important principle of localization, then introduce minimal cotautologies, which are closely related to minimal inferences, and finally turn to the minimal inferences themselves. A set V of variables and a family A of their domains will be fixed throughout.

We introduce the principle of localization in the form which shows how a semantic inference can be localized to an arbitrary set of tokens.

Lemma 6 (Localization) *If we have $\{p_1, \ldots, p_n\} \models r$ in $A^{\#}$, and $t \subseteq VA$ is any set of tokens, then $\{p_1 \cap t, \ldots, p_n \cap t\} \models r \cap t$ holds.*

Proof: Suppose that a tuple $s \in A^V$ includes $r \cap t$. Put $W := V \setminus \text{scope}(r \setminus t)$, and construct the tuple $s' := s[W]\,(r \setminus t) \in A^V$. Since s' includes r, it includes p_i for some $i = 1, \ldots, n$, whence $s \supseteq s[W] = s'[W] \supseteq s' \cap t \supseteq p_i \cap t$. ■

Taking $t = \bigcup P$, we get the following consequence of the localization lemma:

Corollary 1 *If $P \models r$, then $P \models r \cap \bigcup P$. In particular, if the inference $P \models r$ is minimal, then $r \subseteq \bigcup P$.*

Each of the statements in the corollary is in fact equivalent to the principle of localization as stated in the localization lemma.

Next on our agenda are unsatisfiable negative constraint networks; we are going to call them *cotautologies*, for the following reason. A negative constraint network

$Q \subseteq A^{\#}$ is unsatisfiable precisely when each tuple over scope(Q) includes at least one tuple from Q. We can reformulate this in terms of truth-valued functions. Let us associate with a negative constraint network Q the function $h_Q\colon A^V \to \{0,1\}$, defined by

$$h_Q(s) := \bigvee_{q \in Q} \bigwedge_{a \in q} (a \in s)\,, \quad \text{for any tuple } s \in A^V$$

(we assume here that logical formulas, like $(a \in s)$, evaluate to truth values $0 = \mathit{false}$ and $1 = \mathit{true}$.) The function h_Q is the negated interpretation of the negative constraint network Q, since $h_Q(s) = 0$ precisely when s satisfies Q. The negative constraint network Q is therefore unsatisfiable if and only if h_Q is the constant function $s \mapsto 1$. This is why we will refer to unsatisfiable negative constraint networks as cotautologies (and also because it is shorter).

A cotautology is *minimal* if none of its proper subsets is a cotautology.

For any set of tuples $Q \subseteq A^{\#}$ and any set of variables $U \subseteq V$ denote by $Q\langle U\rangle$ the set of tuples $\{\, q[U] \mid q \in Q \text{ and } q[U] \neq \square \,\}$.

PROPOSITION 7 *If Q is a minimal cotautology, then $Q\langle U\rangle$ is a cotautology for every nonempty set of variables $U \subseteq$ scope(Q).*

Proof: Write $W :=$ scope(Q). Suppose that for some nonempty subset U of W, $Q\langle U\rangle$ is not a cotautology. That is, there exists a tuple $p \in A^U$ so that $q[U] \not\subseteq p$ for every tuple $q \in Q$ with a nonempty restriction $q[U]$. For every tuple $r \in A^{W \setminus U}$ the tuple $pr \in A^W$ includes some tuple $q \in Q$; since then $q[U] \subseteq p$, the restriction $q[U]$ must be empty and thus $q \subseteq r$. But this means that $Q' = Q \mid (W \setminus U)$ is a cotautology. Because each variable $u \in U \neq \emptyset$ appears in some tuple belonging to Q, the cotautology Q' is a proper subset of the cotautology Q, contrary to the minimality of Q. ∎

COROLLARY 2 *If Q is a minimal cotautology with* scope$(Q) = W$*, then* $\bigcup Q = WA$.

Proof: For every variable $w \in W$, $Q\langle w\rangle$ is a cotautology, thus $Q\langle w\rangle = A^w$. This means that every token $a \in WA$ belongs to some tuple $q \in Q$. ∎

COROLLARY 3 *The underlying scheme of a cotautology Q is connected.*

Proof: Suppose that the underlying scheme of Q is not connected. Then there exists a partition of the set of variables $W :=$ scope(Q) into two nonempty subsets U and U' such that for every tuple $q \in Q$ either scope$(q) \subseteq U$ or scope$(q) \subseteq U'$. But then the cotautology $Q\langle U\rangle = Q|U$ is a proper subset of the minimal cotautology Q, contradiction. ∎

We will now relate minimal inferences to cotautologies. For any set of tuples $P \subseteq A^{\#}$ and any tuple $r \in A^{\#}$, define

$$P - r := \{\, p \setminus r \mid p \in P \text{ is compatible with } r \,\}$$

Writing $U := \mathrm{scope}(r)$, we have $p \setminus r = p\,[V \setminus U]$ for every $p \in P$ compatible with r.

LEMMA 7 *Let $P \subseteq A^{\#}$, $r \in A^{\#}$. Then $P \models r$ if and only if $P - r$ is a cotautology.*

Proof: Put $U := \mathrm{scope}(r)$.

Suppose that $P \models r$. Since the scopes of tuples in $P - r$ are subsets of the set $V \setminus U$, we will prove $P - r$ to be a cotautology if we show that every tuple over $V \setminus U$ includes sume tuple from $P - r$. Let then $s \in A^{V \setminus U}$. Since the tuple $sr \in A^V$ includes the consequence r, it includes some premise $p \in P$, which is compatible with r, and s includes $p\,[V \setminus U] \in P - r$.

Conversely, suppose that $P - r$ is a cotautology. Take any tuple $s \in A^V$ that includes r. The tuple s includes $p \setminus r$ for some tuple $p \in P$ compatible with r, and then also includes $(p \setminus r)\, r \supseteq p$. ■

PROPOSITION 8 *Let $P \subseteq A^{\#}$ and $r \in A^{\#}$. Then $P \models r$ holds, with the set of premises P minimal for the consequence r, if and only if the following conditions are satisfied:*

- *every tuple in P is compatible with the tuple r;*
- *the mapping $P \to (P - r) : p \mapsto (p \setminus r)$ is a bijection;*
- *$P - r$ is a minimal cotautology.*

Proof: Suppose that $P \models r$ with P minimal for r. Let P' be the set of those tuples in P that are compatible with r. Since $P' - r = P - r$ is a cotautology, $P' \models r$ holds, so we must have $P' = P$ because of the minimality of P. The mapping $\tau : P \to (P - r) : p \mapsto (p \setminus r)$ is surjective by the definition of $P - r$. For every tuple q in $P - r$ choose a tuple q' in P for which $q' \setminus r = q$, and let P' be the set of all chosen tuples q'. Then again $P' - r = P - r$ and hence $P' = P$, so τ is a bijection. Finally, choose a minimal cotautology $Q \subseteq P - r$ and put $P' := \tau^{-1}(Q)$; since $P' - r = \tau(P') = Q$ is a cotautology, we have $P' = P$, thus also $Q = P - r$.

Conversely, suppose that P and r satisfy the conditions. Because $P - r$ is a cotautology, we have $P \models r$. If P' is a proper subset of P, then $P' - r$ is clearly a proper subset of the minimal cotautology $P - r$, whence $P' \not\models r$. ■

If the set of premises P in $P \models r$ is minimal for the consequence r, and $P - r$ contains the empty tuple $\square$, then $P - r = \{\square\}$ and $P = \{p\}$ with $p \subseteq r$; otherwise $P - r$ is a set of nonempty tuples, thus in every tuple $p \in P$ there appear variables not appearing in r.

COROLLARY 4 *Let $P \models r$, with P minimal for r. Writing $W := \mathrm{scope}(P - r) = \mathrm{scope}(P) \setminus \mathrm{scope}(r)$, we have $WA \subseteq \bigcup P \subseteq WA \cup r$. If also r is minimal for P, then $\bigcup P = WA \cup r$.*

Proof: Suppose P is minimal for r in $P \models r$. Since $P - r$ is a minimal cotautology with $\mathrm{scope}(P - r) = W$, we have $WA = \bigcup (P - r) \subseteq \bigcup P$. On the other hand, every tuple $p \in P$ satisfises $p \subseteq (p \setminus r)\, r \subseteq WA \cup r$, and thus $\bigcup P \subseteq WA \cup r$.

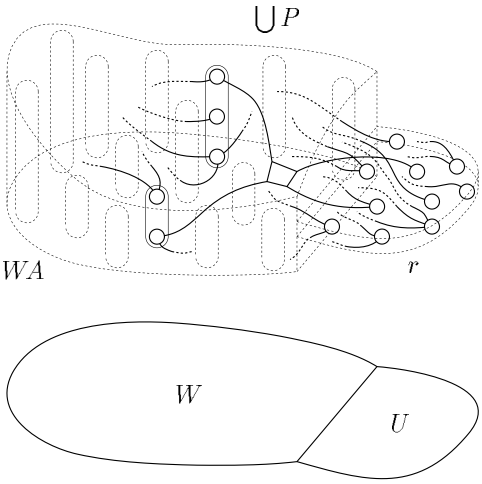

Figure 3.

Now let also r be a minimal consequence of P. Localizing $P \models r$ to $P \models r'$ with $r' := r \cap \bigcup P$, we find that $r = r' \subseteq \bigcup P$. ■

Corollary 5 *If P is minimal for r in $P \models r$, then $r \cap \bigcup P$ is the only minimal consequence of P among the subtuples of the tuple r. No variable appearing in $r \setminus \bigcup P$ appears in P.*

Proof: Write $W := \mathrm{scope}(P - r)$ and $r' := r \cap \bigcup P$, and let $r'' \subseteq r$ be a minimal consequence of P. Because $WA \subseteq \bigcup P \subseteq WA \cup r$, we have $\bigcup P = WA \cup r'$. Because P is still minimal for r'', we have also $\bigcup P = WA \cup r''$. It follows that $r'' = \bigcup P \setminus WA = r'$. A variable appearing in $r \setminus \bigcup P = r \setminus r'$ does not appear in WA or in r', hence does not appear in $\bigcup P = WA \cup r'$. ■

If the inference $P \models r$ is minimal, then the consequence r is the 'flat part' of the set of tokens $\bigcup P$ (see Figure 3 for an impression of the shape of a minimal inference). In particular, in a minimal inference the set of premises uniquely determines the conclusion. Formally:

Proposition 9 *If $P \models r$ and $P' \models r'$ are minimal, then $P = P'$ implies $r = r'$.*

In short, the entailment closure operator is skew.

3.4. Prime inferences

Let A be a family of domains for variables in V. Since the closure operator Conseq in $A^{\#}$ is skew, the results we have obtained for general skew closure operators can be applied to it and also to any closure operator induced by it in an order ideal of $A^{\#}$. It would be worthwhile to know more about prime inferences in an arbitrary order ideal of $A^{\#}$; here we shall restrict the discussion to the closure operator Conseq_m, induced in the order ideal $A^{(*m)}$ of $A^{\#}$ by the entailment closure operator Conseq, for some natural number $m \geq 2$.

Let us take a quick look at the prime inferences $P \models r$ in $A^{(*m)}$ with $|r| < m$; we will not be much concerned with them. It is not hard to figure out what the classes of associated prime inferences are in this case.

PROPOSITION 10 *If r is a tuple with $m-1$ elements, then for each variable v not in* scope(r) *there is in $A^{(*m)}$ one class of associated prime inferences, consisting of the prime inference $A^v r \models r$ alone. For each tuple r with at most $m-2$ elements there is in $A^{(*m)}$ one class of associated prime inferences that have the conclusion r, which consists of all minimal inferences $P \models r$ in $A^{(*m)}$ such that $r \subseteq p$ for every $p \in P$. There are no other prime inferences with at most $(m-1)$-ary conclusions.*

We omit the easy proof. It is also easy to show that an indecomposable minimal inference in $A^{(*m)}$ with an at most $(m-1)$-ary conclusion is always prime, and hence every indecomposable minimal inference in $A^{(*m)}$ is prime.

From now on let r be an m-ary tuple. All sets of tuples mentioned will be assumed to be subsets of $A^{(*m)}$. Since r is maximal in the set $A^{(*m)}$, a minimal inference $P \models r$ reduces to a minimal inference $Q \models r$ if and only if $P \models Q$.

We will need localized versions of reduction and primeness. We say that an inference $P \models r$ *locally reduces* to an inference $Q \models s$, if $s = r$ and $P \models Q$ and $\bigcup Q \subseteq \bigcup P$. Local reducibility is transitive. We call an inference *locally prime* if it is minimal, and has the property that whenever it locally reduces to another minimal inference, this other inference reduces back to it. (Note that we do not require of the other inference to reduce back *locally*.) If a locally prime inference $P \models r$ locally reduces to a minimal inference $Q \models r$, then $Q \models r$ is a locally prime inference associated with $P \models r$. It is obvious that every prime inference is locally prime. The converse also holds. We first prove two lemmas.

LEMMA 8 *Every premise of a locally prime inference is m-ary.*

Proof: Let $P \models r$ be a minimal inference that has a premise $p \in P$ with $|p| < m$. Choose a variable u appearing in r but not in p, and let a be the token $r(u)$. Put $P' := (P \setminus \{p\}) \cup \{pa\}$. Clearly $P \models P' \models r$, where $P' \models r$ is minimal and $\bigcup P' = \bigcup P$. We will now show that $P' \not\models p$. Because the inference $P \models r$ is minimal, there exists a tuple s over scope(P) which includes r and p and excludes $P \setminus \{p\}$. In the tuple s, change the token $s(u) = a$ to some different token over u, and denote the resulting tuple by s'. The tuple s' still excludes $P \setminus \{p\}$, it also excludes the tuple pa, hence excludes the whole P', while it includes the tuple p. This proves that $P \models r$ is not prime. ■

Lemma 9 *If a locally prime inference $P \models r$ reduces to a prime inference $Q \models r$, then $\bigcup Q \subseteq \bigcup P$.*

Proof: For each $q \in Q$ choose a minimal consequence $q' \subseteq q$ of P, and let Q' be a minimal subset of the set of all q' such that $Q' \models r$; then $P \models Q' \models r$ and $\bigcup Q' \subseteq \bigcup P$. Here $Q' \models r$ is a proper minimal inference: if $Q' = \{q'\}$, then $P \models q' \models r$ implies $r = q' = q \in Q$, which is not possible because $Q \models r$ is proper. Since $P \models r$ is locally prime, $Q' \models r$ is also locally prime. But then every tuple in Q' is m-ary and therefore belongs to Q. Since we have $Q' \models r$ with $Q' \subseteq Q$, and $Q \models r$ is minimal, it follows that $Q' = Q$. ∎

Now we have:

Proposition 11 *Every locally prime inference is prime.*

Proof: Let $P \models r$ be a locally prime inference; it reduces to some prime inference $Q \models r$. Since we have $\bigcup Q \subseteq \bigcup P$, and $P \models r$ is locally prime, the prime inference $Q \models r$ reduces to $P \models r$, which is therefore also prime. ∎

Corollary 6 *If a prime inference $P \models r$ reduces to a minimal inference $Q \models r$, then $\bigcup P = \bigcup Q$.*

Proof: Since $Q \models r$ is also prime and reduces back to $P \models r$, and because prime inferences coincide with locally prime inferences, we have both inclusions between $\bigcup P$ and $\bigcup Q$. ∎

For any natural number h let γ_h be the closure operator in $A^{(*m)}$ which enforces the weak relative $(m+h)$-consistency of negative networks on the scheme $V^{(*m)}$, and let $\vdash_h$ be the corresponding closed inference system. The closure operator γ_0 enforces the internal consistency on $V^{(*m)}$. Closure operators γ_h increase with increasing h, up to the entailment closure operator Conseq_m, where $\gamma_h = \mathrm{Conseq}_m$ for every $h \geq |V| - m$.

We know that any representative set of prime inferences for Conseq_m is an almost complete inference system for this closure operator; we will show that it also provides almost complete inference systems for each of the closure operators γ_h.

We say that an inference system $\mathcal{Q}$ in $A^{(*m)}$ *essentially generates* a closure operator β in $A^{(*m)}$, if $\mathcal{Q}$ and $\vdash_0$ together generate β. We will say that a prime inference $P \models r$ in $A^{(*m)}$ with $|r| = m$ and $|\mathrm{scope}(P - r)| = h$ *is of height h*. Note that a prime inference $P \models r$ of height h or less satisfies $P \vdash_h r$.

If $\mathcal{R}$ is any set of prime inferences in $A^{(*m)}$, then for each natural number $h > 0$ we denote by $\mathcal{R}_h$ the set of all prime inferences from $\mathcal{R}$ that are of height h, and put $\mathcal{R}_{*h} := \mathcal{R}_1 \cup \cdots \cup \mathcal{R}_h$.

Proposition 12 *If $\mathcal{R}$ is any representative set of prime inferences for* Conseq_m*, then the closure operator γ_h is essentially generated by $\mathcal{R}_{*h}$, for each natural number $h > 0$.*

Proof: The closure operator γ_h is generated by all minimal inferences $P \models r$ in $A^{(*m)}$ such that $|\,\mathrm{scope}(P)\,| \leq m+h$, together with all unary inferences in $A^{(*m)}$. Since for every unary inference $p \models q$ in $A^{(*m)}$ we have $p \vdash_0 q$, it suffices to prove that any minimal inference $P \models r$ in $A^{(*m)}$ with $m' := |\,\mathrm{scope}(P)\,| \leq m+h$ is derivable using $\mathcal{R}_h$ and $\vdash_0$. The proof will be by induction on the set $\mathrm{Conseq}_m(P)$. For $m' \leq m$ we have $P \vdash_0 r$, so suppose that $m' > m$.

If $|r| < m$, choose a set U of m variables such that $\mathrm{scope}(r) \subset U \subseteq \mathrm{scope}(P)$, and let Q be the set of all tuples over U that extend the tuple r. For each $q \in Q$ choose a minimal set of premises $P_q \subseteq P$ for the consequence q, and then choose a minimal consequence $q' \subseteq q$ of P_q; let Q' be the set of all tuples q'. Since all tuples in P_q are compatible with q, while there are tuples in P incompatible with q, each P_q is a proper subset of P, hence P-small because P is independent. Since $\mathrm{scope}(P_q) \subseteq \mathrm{scope}(P)$, we have $|\,\mathrm{scope}(P_q)\,| \leq m+h$. By induction hypothesis, all minimal inferences $P_q \models q'$ are derivable using $\mathcal{R}_h$ and $\vdash_0$, and moreover, we have $Q' \vdash_0 Q \vdash_0 r$.

Suppose that $|r| = m$. The minimal inference $P \models r$ locally reduces, in $A^{(*m)}$, to a locally prime inference $P' \models r$, which is in fact prime and therefore reduces to a prime inference $Q \models r$ in $\mathcal{R}$. Since $\bigcup Q = \bigcup P' \subseteq \bigcup P$, the height of the prime inference $Q \models r$ is at most h. For each $q \in Q$ choose P_q and q' as in the previous case; also the rest of the proof is just as in the previous case, except that now we have the prime inference $Q \models r$ instead of $Q \vdash_0 r$. ■

3.5. Prime inferences of heights 1 and 2

In this section we describe all prime inferences of heights 1 and 2 for the restricted entailment closure operator Conseq_m in $A^{(*m)}$, where $A = (\,A_v \mid v \in V\,)$ is an arbitrary family of domains, and $m \geq 2$.

PROPOSITION 13 *Prime inferences of height* 1 *for the closure operator* Conseq_m *are precisely those resolution inferences in* $A^{(*m)}$ *in which all premises and the resolvent are* m*-ary tuples. All prime inferences of height* 1 *are primitive.*

Proof: Clearly any prime inference of height 1 is a resolution inference in which all premises and the resolvent are m-ary tuples.

Conversely, let $P \models r$ be a resolution inference with $P \subseteq A^{(m)}$ and $r \in A^{(m)}$. The set of premises P is minimal for r. The resolution procedure, applied to P, produces the resolvent r and stops. Since all tuples in $P \cup \{r\}$ are maximal in $A^{(*m)}$, we have $\mathrm{Conseq}_m(P) = P \cup \{r\}$, ie. the inference $P \models r$ is primitive. ■

Characterization of prime inferences of height 2 is slightly more complicated. Let w_1 and w_2 be two different variables. The general form of a minimal cotautology Q with $\mathrm{scope}(Q) = \{w_1, w_2\}$ is

$$Q = X_1 \cup X_2 \cup Y_1 Y_2\,,$$

where

$$X_i \subseteq A^{w_i}, \quad Y_i = A^{w_i} \setminus X_i \neq \emptyset, \quad \text{for } i = 1, 2\,.$$

Let r be an m-ary tuple such that $U := \mathrm{scope}(r)$ is disjoint with a set of variables W. If $P \models r$ is an inference with $\mathrm{scope}(P - r) = W$ in which the set P of premises is minimal for the conclusion r, then $Q := P - r$ is a minimal cotautology on W, and each premise $p \in P$ is of the form $p = q\tau(q)$ with $\tau(q) \subseteq r$ for a unique $q \in Q$. Conversely, given a minimal cotautology Q on W, if we choose for each $q \in Q$ an arbitrary subtuple $\tau(q)$ of the tuple r and let P be the set of all tuples $q\tau(q)$ for $q \in Q$, then we obtain an inference $P \models r$ whose set P of premises is minimal for r. With the inference $P \models r$ represented in this way, we write $\tau(Q') := \bigcup_{q' \in Q'} \tau(q')$ for any subset Q' of Q.

Proposition 14 *Using the notation introduced in the text, the following holds: a minimal inference $P \models r$ with* $\mathrm{scope}(P - r) = W = \{w_1, w_2\}$*, whose premises and the conclusion are m-ary tuples, is prime if and only if*

$$\begin{aligned} \tau(X_1 \cup Y_1 y_2) &= r \quad \textit{for every } y_2 \in Y_2\,, \quad \textit{and} \\ \tau(X_2 \cup y_1 Y_2) &= r \quad \textit{for every } y_1 \in Y_1\,. \end{aligned}$$

All prime inferences of height 2 are primitive.

Proof: First we prove that the conditions are sufficient by proving that they imply primitivity of the inference $P \models r$. We will show that $P \cup \{r\}$ is the set of all minimal consequences of P for the entailment closure operator Conseq in $A^{\#}$. For each $y_2 \in Y_2$ we have the resolution fan $\{\, x_1\tau(x1) \mid x_1 \in X_1 \,\} \cup \{\, y_1 y_2 \tau(y_1 y_2) \mid y_1 \in Y_1 \,\}$ with the resolvent $\tau(X_1) \cup y_2\tau(Y_1 y_2) = y_2\tau(X_1 \cup Y_1 y_2) = y_2 r$. Symmetrically we get a resolvent $y_1 r$ for each $y_1 \in Y_1$. Since this exhausts all resolution fans in the set $P \cup \{r\}$, this set is subclosed under resolution in $A^{\#}$ and is therefore the set of all minimal consequences of the set P. Since all the tuples in $P \cup \{r\}$ are maximal in $A^{(*m)}$, it follows that $\mathrm{Conseq}_m(P) = P \cup \{r\}$, proving primitivity of the inference $P \models r$.

Now assume that the conditions of the proposition are not satisfied. There is, say, an $y_2 \in Y_2$ such that $\tau(X_1 \cup Y_1 y_2) =: r'$ is a proper subtuple of the tuple r. As in the first part of the proof there is the resolvent $y_2\tau(X_1 \cup Y_1 y_2) = y_2 r'$, but now we have $|y_2 r'| \leq m$. Choose an $(m-1)$-ary tuple r'' such that $r' \subseteq r'' \subseteq r$, and put $p' := y_2 r''$. Also put $p := y_1 y_2 \tau(y_1 y_2) \in P$ for some $y_1 \in Y_1$, and construct $P' := (P \setminus \{p\}) \cup \{p'\}$. Clearly $P \models P' \models r$; we shall show that $P' \not\models p$. Since $|p \cap r| = |\tau(y_1 y_2)| = m-2 \leq m-1 = |r''| = |p' \cap r|$, the tuple p' is not a subtuple of the tuple p, so there exists a tuple s over $W \cup U$ such that $p \subseteq s$ and $p' \not\subseteq s$. Clearly the tuple s excludes $P \setminus \{p\}$ because it includes $y_1 y_2$. The tuple s excludes P' and includes p, showing that $P' \not\models p$. Let $P'' \subseteq P'$ be a minimal set of premises for r. The minimal inference $P \models r$ is not prime, because it reduces to the minimal inference $P'' \models r$ which does not reduce back. ∎

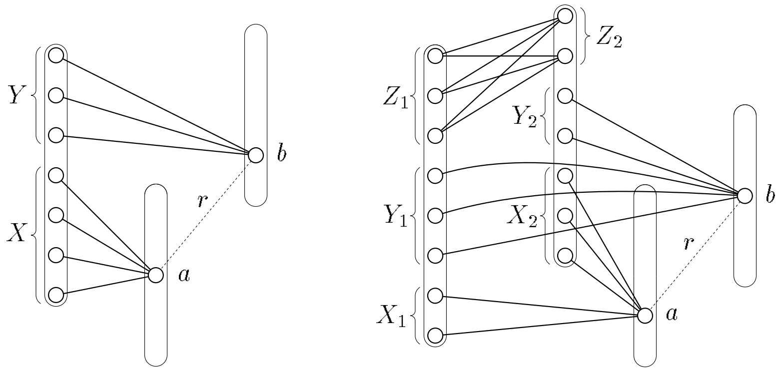

Figure 4.

Let us look at the prime inferences $P \models r$ of heights 1 and 2 for binary networks. Denote by W the set $\mathrm{scope}(P - r)$, and write the binary tuple r as $r = ab$, where a and b are tokens over two different variables not in W. Specializing the propositions 13 and 14 we arrive at the following characterization:

PROPOSITION 15 *Let W, P, and $r = ab$ be as in the text.*

Prime inferences of height 1 over a complete binary scheme are precisely the resolution inferences

$$Xa \cup Yb \models ab\,,$$

where X and Y are nonempty complementary subsets of the set A^w $(W = \{w\})$.

Prime inferences of height 2 are precisely the inferences

$$X_1 a \cup Y_1 b \cup X_2 a \cup Y_2 b \cup Z_1 Z_2 \models ab\,,$$

where X_i, Y_i, Z_i are nonempty sets forming a partition of the set A^{w_i} for $i = 1, 2$ $(W = \{w_1, w_2\})$.

Figure 4 shows binary prime inferences of heights 1 and 2 as described by the proposition 15. A moment's thought shows that prime inferences of height 1 enforce path consistency. Prime inferences of height 2 thus embody the simplest type of (weak relative) consistency beyond the path consistency.

3.6. Weak primitivity of prime inferences

Prime inferences for the closure operator Conseq_m have the following property, which will be referred to as *weak primitivity*:

LEMMA 10 *Let $P \models r$ be a prime inference for Conseq_m; write $W := \mathrm{scope}(P - r)$. If $q \notin P \cup \{r\}$ is an at most m-ary tuple such that $q[W] \subseteq p[W]$ for some tuple $p \in P$, then $P \not\models q$.*

Proof: Taking $q' := q \cap \bigcup P$, we have $P \models q$ if and only if $P \models q'$. Since q' is at most m-ary, $q' \notin P \cup \{r\}$, and $q'[W] \subseteq p[W]$, we can assume that already $q \subseteq \bigcup P$, and in particular, that q is compatible with r. When $q[W] = \square$, the tuple q is a proper subtuple of the minimal consequence r of the premises P, whence $P \not\models q$; from now on we assume that the tuple $q[W]$ is not empty.

In the case $|q| = m$ we cannot have $q \cap r \subseteq p \cap r$, since then we would have $q = p$. Suppose that $|q| < m$. We can extend the tuple q to an m-ary subtuple q' of the tuple $q \cup r$ in such a way that $q' \nsubseteq p$; by the choice of q' we have $q'[W] = q[W]$. The tuple q' is certainly different from r. Moreover, q' does not belong to P: if $q'[W]$ is a proper subtuple of $p[W]$, then q' does not belong to P because $P - r$ is a minimal cotautology; if $q'[W] = p[W]$, then $q' \neq p$ does not belong to P because the mapping $P \to (P - r) : x \mapsto x[W]$ is a bijection. If we show that $P \not\models q'$, it will follow that $P \not\models q$. In short, we can assume that q is an m-ary tuple.

We must show that $P \not\models q$; suppose that, on the contrary, $P \models q$ holds. Construct the set $P' := (P \setminus \{p\}) \cup \{q\}$. Then $P \models P'$, and since $q[W] \subseteq p[W]$, $P' - r$ is a cotautology, so we have $P' \models r$. Choose a minimal set of premises $P'' \subseteq P'$ for the consequence r. There exists a tuple s over scope(P) which includes r and p but excludes $P \setminus \{p\}$. In the tuple s, replace some token $a \in (q \setminus p) \cap r \neq \square$ by a different token over the same variable. The resulting tuple s' excludes P'' but includes p, whence $P'' \not\models p$, contradicting primeness of $P \models r$. ■

For any minimal inference $P \models r$, every token over a variable in scope$(P - r)$ belongs to some premise $p \in P$, so we have the following corollary of the lemma 10:

Corollary 7 *If $P \models r$ is a prime inference and $q \notin P \cup \{r\}$ is an at most m-ary tuple of the form $q = cr'$, where r' is a subtuple of the tuple r and c is a token over a variable not appearing in r, then $P \not\models q$.*

The following corollary of lemma 10 is also worth mentioning:

Corollary 8 *Every premise of a prime inference $P \models r$ is a minimal consequence of the set of premises P.*

Proof: If p' is a proper subtuple of some premise $p \in P$, then $p' \notin P \cup \{r\}$ and $p'[W] \subseteq p[W]$, hence $P \not\models p'$ by weak primitivity of $P \models r$. ■

3.7. All prime (3,2)-inferences up to height 4

This subsection presents all prime inferences of heights 1, 2, 3, and 4 for the closure operator Conseq_3 on two-valued variables; we will call them *prime* $(3,2)$*-inferences.* We will give proofs for the prime inferences of heights 1, 2, and 3, but will only list the prime inferences of height 4 without a proof that the list is complete (primeness of the listed inferences can be easili checked).

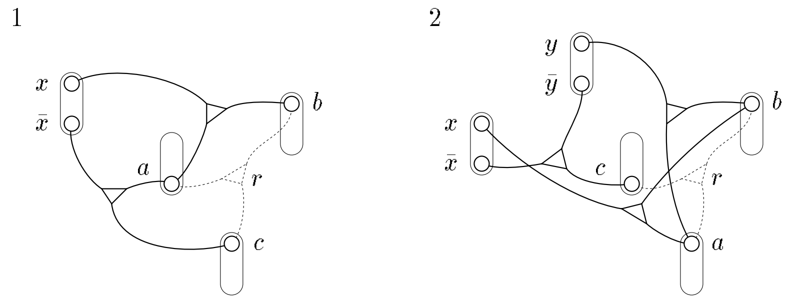

Figure 5.

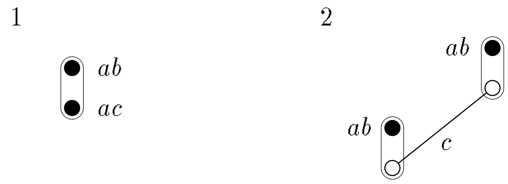

Figure 6.

Let us have a tuple $r = abc$, where a, b, and c are tokens over three different variables, and let us alos have a set W of at most four variables disjoint with scope(r). We will determine all prime inferences $P \models r$ with scope$(P - r) = W$. The two tokens in wA, for $w \in W$, will be written x and $\bar{x}$, or y and $\bar{y}$, and so on.

We start by extracting the prime (3,2)-inferences of heights 1 and 2 from the general propositions 13 and 14.

PROPOSITION 16 *All prime* $(3,2)$*-inferences of height* 1 *are of the form*

$$\{xabc,\ \bar{x}ac\} \models abc\,,$$

while all prime $(3,2)$*-inferences of height* 2 *are of the form*

$$\{xab,\ yab,\ \bar{x}\bar{y}c\} \models abc\,.$$

All prime $(3,2)$*-inferences of heights* 1 *and* 2 *are primitive.*

Proof: The case of height 1 is trivial.
Height 2. In the notation of proposition 14, we cannot have $X_1 = \emptyset$, since in this case the tuple $\tau(Y_1 y_2)$ would consist of at most two tokens, and similarly we cannot have $X_2 = \emptyset$. Every prime (3,2)-inference of height 2 is thus of the form $\{x\tau(x),\ y\tau(y),\ \bar{x}\bar{y}\tau(\bar{x}\bar{y})\} \models abc$, where $\tau(x)$ and $\tau(y)$ are binary subtuples and $\tau(\bar{x}\bar{y})$ is a unary subtuple of the tuple abc, such that $\tau(x) \cup \tau(\bar{x}\bar{y}) = abc = \tau(y) \cup \tau(\bar{x}\bar{y})$. From this we see that, up to a permutation of the tokens a, b, and c, we have $\tau(x) = ab = \tau(y)$ and $\tau(\bar{x}\bar{y}) = c$. ∎

Figure 5 shows the prime (3,2)-inferences of heights 1 and 2. We will prefer to draw them (and other prime inferences) in the abbreviated form shown in Figure 6;

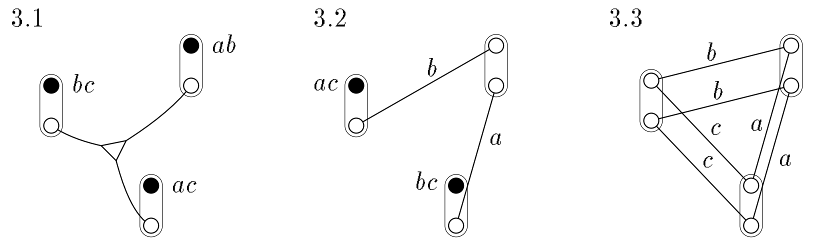

Figure 7.

in this form we represent a prime inference $P \models r$ by the cotautology $P - r$, where for each premise $p \in P$ we mark the tuple $p \setminus r \in P - r$ by the tuple $p \cap r$, whenever this tuple is nonempty.

Next lemma gives a handful of conditions satisfied by any prime (3,2)-inference $P \models r$. Some conditions constrain the cotautology $P - r$, and some constrain how subtuples of r are attached to tuples in $P - r$ to form the tuples in P.

Lemma 11 *Let $P \models r$ be a prime $(3,2)$-inference. Then the following holds (where x, y, and z are tokens over different variables in $W = \mathrm{scope}(P - r)$, while α, β, and γ are subtuples of the tuple $r = abc$):*

(i) $P - r$ does not contain both a tuple xyz and a tuple $xy\bar{z}$;

(ii) $P - r$ does not contain both a tuple xy and a tuple $x\bar{y}z$;

(iii) $P - r$ does not contain both a tuple xy and a tuple $x\bar{y}$;

(iv) $P - r$ does not contain two different inclusion-comparable tuples;

(v) when $|W| > 1$, $P - r$ does not contain a pair of tuples x, $\bar{x}$;

(vi) when $|W| > 2$, $P - r$ does not contain a triple of tuples x, y, $\bar{x}\bar{y}$;

(vii) if P contains tuples $xy\alpha$ and $\bar{x}\beta$, then $\alpha \cup \beta = abc$;

(viii) if P contains tuples $x\alpha$, $y\beta$, and $\bar{x}\bar{y}\bar{z}$, then $\alpha \cup \beta = abc$;

(ix) if P contains tuples $xy\alpha$, $\bar{x}z\beta$, and $\bar{y}z\gamma$, then $\alpha \cup \beta \cup \gamma = abc$.

Proof: Conditions (iv), (v), and (vi) are satisfied because $P - r$ is a minimal cotautology. All other properties listed in the lemma follow easily from the weak primitivity of prime inferences (lemma 10). For example, in (ix) we have $\{xy\alpha, \bar{x}z\beta, \bar{y}z\gamma\} \models z(\alpha \cup \beta \cup \gamma)$, hence by corollary 7 the tuple $z(\alpha \cup \beta \cup \gamma)$ must consist of at least four tokens. ∎

It is now a fairly simple task to determine all prime (3,2)-inferences of height 3.

Proposition 17 *Every prime* $(3,2)$*-inference has one of the following three forms (shown in Figure 7 in abbreviated graphic notation):*

$$\begin{aligned} xbc,\ yac,\ zab,\ \bar{x}\bar{y}\bar{z} &\models abc\,; \\ xac,\ ybc,\ \bar{x}zb,\ \bar{y}\bar{z}a &\models abc\,; \\ xyc,\ xzb,\ yza,\ \bar{x}\bar{y}c,\ \bar{x}\bar{z}b,\ \bar{y}\bar{z}a &\models abc\,. \end{aligned}$$

All prime $(3,2)$*-inferences are primitive.*

Proof: We will refer to the properties of prime (3,2)-inferences in lemma 11 without mentioning the lemma. The proof is divided into cases according to the number of unary tuples in the cotautology $Q := P - abc$. This cotautology can contain at most three unary tuples, since due to (v) each of the unary tuples must be over a different variable.

(1) The cotautology Q contains three unary tuples x, y, and z. The tuple $\bar{x}\bar{y}\bar{z}$ contains a tuple in Q, which cannot be binary, because of (vi), hence must be ternary, ie. the tuple $\bar{x}\bar{y}\bar{z}$ itself belongs to Q. The cotautology Q thus consists of the tuples x, y, z, and $\bar{x}\bar{y}\bar{z}$. To each of the unary tuples in Q is attached a binary subtuple of the tuple abc, to make the corresponding tuple in P. It follows from (viii) that xbc, yac, zab is the only possibility, up to a permutation of the tokens a, b, c. Running the resolution on the set $P = \{xbc,\ yac,\ zab,\ \bar{x}\bar{y}\bar{z}\}$, we find that the only minimal consequences consisting of at most three tokens are the tuples in P and the tuple abc, which means that the inference $P \models abc$ is primitive.

(2) The cotautology Q contains precisely two unary tuples x and y. Each of the tuples $\bar{x}\bar{y}z$ and $\bar{x}\bar{y}\bar{z}$ includes a tuple in Q. We cannot have the two ternary tuples themselves in Q, since this would violate (i), hence the cotautology Q must contain at least one binary subtuple of the one or the other ternary tuple. The tuple $\bar{x}\bar{y}$ does not belong to Q, because of (vi). We may assume that Q contains the subtuple $\bar{x}z$ of the tuple $\bar{x}\bar{y}z$ (otherwise we swap the variables of the tokens x and y, or exchange tokens z and $\bar{z}$, or do both). The tuple $\bar{x}\bar{y}\bar{z}$ does not belong to Q, since otherwise Q would contain tuples $\bar{x}\bar{y}\bar{z}$ and $\bar{x}z$, which it does not, because of (vi). Thus one of the binary subtuples of the tuple $\bar{x}\bar{y}\bar{z}$ belongs to Q. We have already seen that this binary subtuple cannot be $\bar{x}\bar{y}$; it also cannot be $\bar{x}\bar{z}$, since this would violate (iii). The only possibility is the tuple $\bar{y}\bar{z}$, which in fact gives us a minimal cotautology Q consisting of the tuples x, y, $\bar{x}z$, and $\bar{y}\bar{z}$. Let the corresponding tuples of P be $x\alpha$, $y\beta$, $\bar{x}z\gamma$, and $\bar{y}\bar{z}\delta$. We see from (vii) that α and γ complement each other in abc, and that the same holds for β and δ. Can we have $\gamma = \delta$, say $\gamma = c = \delta$? Then we would have $\alpha = ab = \beta$, $P = \{xab,\ yab,\ \bar{x}zc,\ \bar{y}\bar{z}c\}$, and $P \models abc$ would reduce to the inference $\{xab,\ yab,\ \bar{x}\bar{y}c\} \models abc$ of height 2, hence would not be prime. So we must have $\gamma \neq \delta$, say $\gamma = b$ and $\delta = a$, and therefore $P = \{xac,\ ybc,\ \bar{x}zb,\ \bar{y}\bar{z}a\}$. We verify that the inference $P \models abc$ is primitive using the method of the previous case.

(3) There is only one unary tuple x in the cotautology Q. The token $\bar{x}$ belongs to some binary or some ternary tuple in Q; we will discuss these two possibilities in two subcases.

(3a) The cotautology Q contains the unary tuple x and a binary tuple $\bar{x}y$. Neither of the tuples $\bar{x}\bar{y}z$, $\bar{x}\bar{y}\bar{z}$ belongs to Q, because this would violate (ii). The tuple $\bar{x}\bar{y}z$ therefore includes a binary tuple in Q, which cannot be $\bar{x}\bar{y}$ because of (iii), so it is one of the tuples $\bar{x}z$, $\bar{y}z$. For the same reason one of the binary subtuples $\bar{x}\bar{z}$, $\bar{y}\bar{z}$ of the tuple $\bar{x}\bar{y}\bar{z}$ belongs to Q. If the tuple $\bar{x}\bar{y}z$ includes, say, the tuple $\bar{x}z \in Q$, then because of $\{x, \bar{x}z\} \models z \models \bar{y}z$ the tuple $\bar{y}z$ does not belong to the minimal cotautology Q. The tuples $\bar{x}z$ and $\bar{y}z$ cannot be both in Q, and likewise $\bar{x}\bar{z}$ and $\bar{y}\bar{z}$ cannot be both in Q. Since according to (iii), Q does not contain both $\bar{x}z$ and $\bar{x}\bar{z}$, and also does not contain both $\bar{y}z$ and $\bar{y}\bar{z}$, there remains only one possibility, up to a symmetry, namely that Q contains the tuples $\bar{x}z$ and $\bar{y}\bar{z}$. This indeed yields a minimal cotautology $Q = \{x, \bar{x}y, \bar{x}z, \bar{y}\bar{z}\}$. But this Q does not correspond to any prime inference $P \models abc$. If we have, say, $xbc \in P$, then by (vii) we have $\bar{x}ya \in P$ and $\bar{x}za \in P$, contradicting (ix), according to which the three tuples $\bar{x}y$, $\bar{x}z$, and $\bar{y}\bar{z}$ from Q should be attached via P to three different tokens in abc.

(3b) The cotautology Q contains the unary tuple x and a ternary tuple $\bar{x}yz$. We may assyme that there is in Q no binary tuple containing the token $\bar{x}$, as this possibility has been already discussed in the subcase (3a). Then the tuple $\bar{x}\bar{y}z$ should include the binary tuple $\bar{y}z$ in Q, or it should itself belong to Q; both is impossible, the former in view of (ii) and the latter in view of (i).

We have shown that there is no prime (3,2)-inference of height 3 whose corresponding cotautology would contain only one unary tuple.

(4) The cotautology Q contains no unary tuples. First we show that Q does not contain any ternary tuple. Assume that, on the contrary, there is in Q a ternary tuple xyz. Since each of the three tokens in abc belongs to some tuple in P, the cotautology Q must contain at least three binary tuples. Because of (ii) and (iv) only the three binary tuples $\bar{x}\bar{y}$, $\bar{x}\bar{z}$, and $\bar{y}\bar{z}$ can belong to Q, so the cotautology Q contains all three of them, and contains no other binary tuple. But then the tuple $xy\bar{z}$, which does not include any of the tuples in Q, should belong to Q, in contradiction to (i).

The cotautology Q thus consists entirely of binary tuples. Over any pair of variables in W we have in Q either no tuple, or one tuple $\xi\eta$, or two tuples $\xi\eta$ and $\bar{\xi}\bar{\eta}$, where ξ is a token over one and η a token over the other of the two variables; this holds because of (iii). The cotautology Q is therefore, up to a symmetry, a subset of the set $\{xy, \bar{x}\bar{y}, xz, \bar{x}\bar{z}, y\bar{z}, \bar{y}z\}$, or of the set $\{xy, \bar{x}\bar{y}, xz, \bar{x}\bar{z}, yz, \bar{y}\bar{z}\}$. The first set is not a cotautology, so the cotautology Q cannot be its subset. The second set is a minimal cotautology, hence coincides with Q. By several applications of (ix) we find that tokens a, b, and c can be attached in only one way, up to a permutation, to tuples in Q, namely as in $P = \{xyc, xzb, yza, \bar{x}\bar{y}c, \bar{x}\bar{z}b, \bar{y}\bar{z}a\}$. We verify primitivity of $P \models abc$ just as we have verified it in the cases (1) and (2). ■

What have we done here? We have searched for and found all sets P of ternary tuples $p \subseteq WA \cup abc$ such that $P \models abc$ is a prime inference. The presence or absence of a particular tuple in a set P can be regarded as a propositional variable. Our search was guided by constraints on these variables stemming from minimality

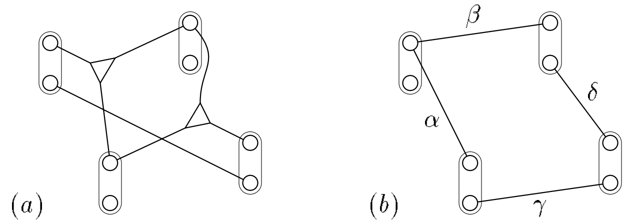

Figure 8.

and weak primitivity of prime inferences. The constraints were not strong enough to completely characterize the prime inferences, as it became apparent in the case (2) of the proof. In short, we formulated a 'relaxed' propositional constraint network, whose solutions were candidates for prime inferences; we then generated all solutions, cutting down on symmetries as we went along, and checked each solution for primeness.

The same method should work for all heights, at least in principle. In practice it soon grows into an onerous and error-prone task to carry out by hand. Determining all prime (3,2)-inferences of height four was already a considerable project, even though the search was further narrowed down by additional constraints. Two of constraints satisfied by prime (3,2)-inferences of height four (or more) can be seen in Figure 8: (a) is a combination of tuples that never appears in the cotautology $P - abc$, while in the situation (b) always $\alpha \cup \beta \cup \gamma \cup \delta = abc$. All types of prime (3,2)-inferences of height four are listed in Figure 9. Most of them are primitive, with the exception of the inferences 4.10, 4.11, and 4.13. For example, the prime inference 4.10 has a minimal consequence yub, which is different from abc and is not one of the premises. Each of the three exceptional prime inferences, while not primitive, is *isolated* in the sense that it is associated only with itself.

All prime (3,2)-inferences of heights 3 and 4 can be derived by 4-resolution; see Figure 10 for one such derivation of the prime (3,2)-inference 4.18. On the other hand, 4-resolution is unable to derive the prime inference of height 6 in Figure 11. So we have

PROPOSITION 18 $\quad 7 \leq \mathrm{reach}(3,2;5) \leq 8.$

The remaining gap at height 5 ought not to take long to fill in.

3.8. All prime (2,3)-inferences up to height 4

In this subsection we give all prime inferences of heights 1, 2, 3, and 4 for the closure operator Conseq_2 on three-valued variables; we call them *prime* $(2,3)$-*inferences.*

We have a binary tuple $r = ab$ and a set W of at most four variables (x, y, z, and u) disjoint with $\mathrm{scope}(r)$. The three tokens over the variable x in W will be written x_0, x_1, and x_2, and so on.

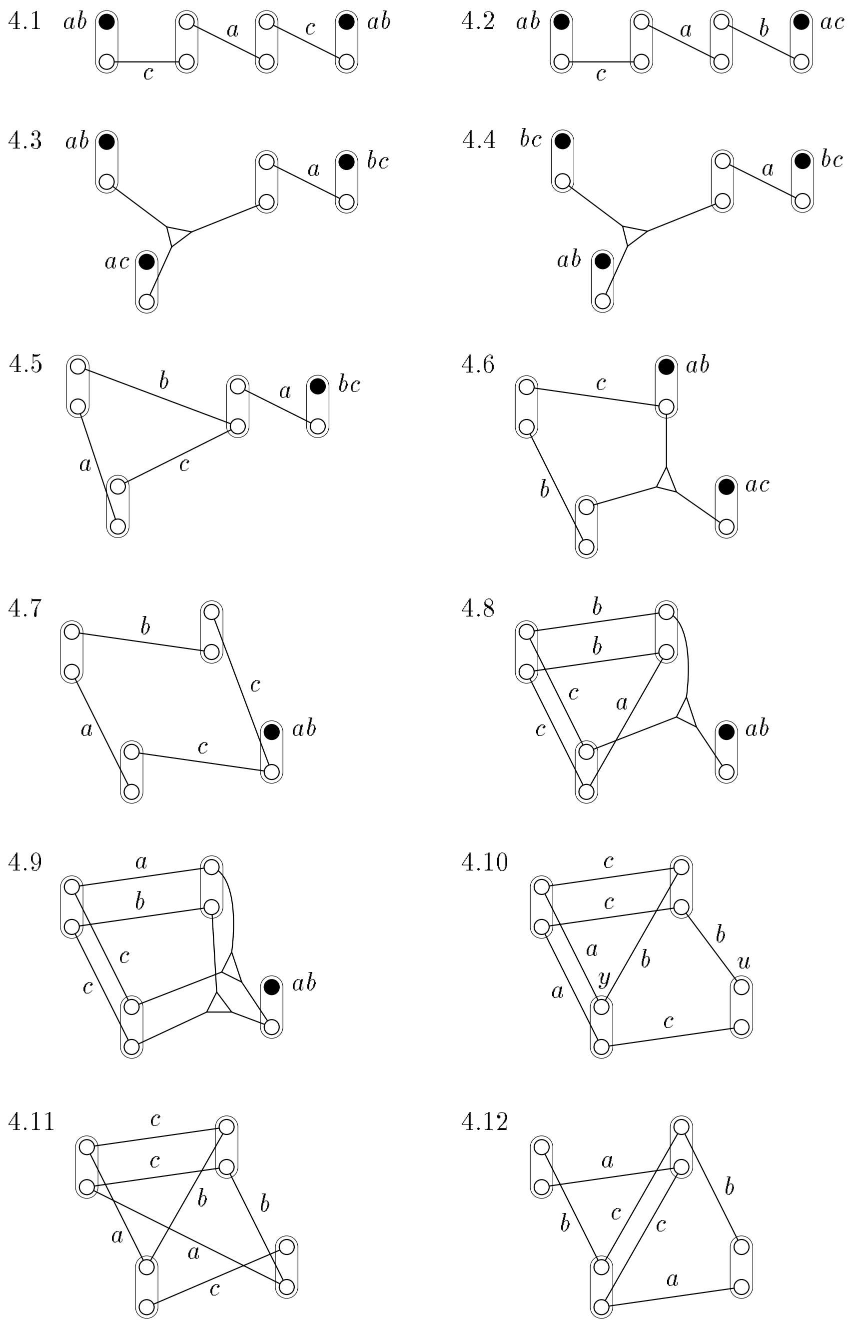
4.1
ab
ab
4.2
ab
ac
4.3
ab
bc
ac
4.4
bc
bc
ab
4.5
bc
4.6
ab
ac
4.7
ab
4.8
ab
4.9
ab
4.10
4.11
4.12

Figure 9.

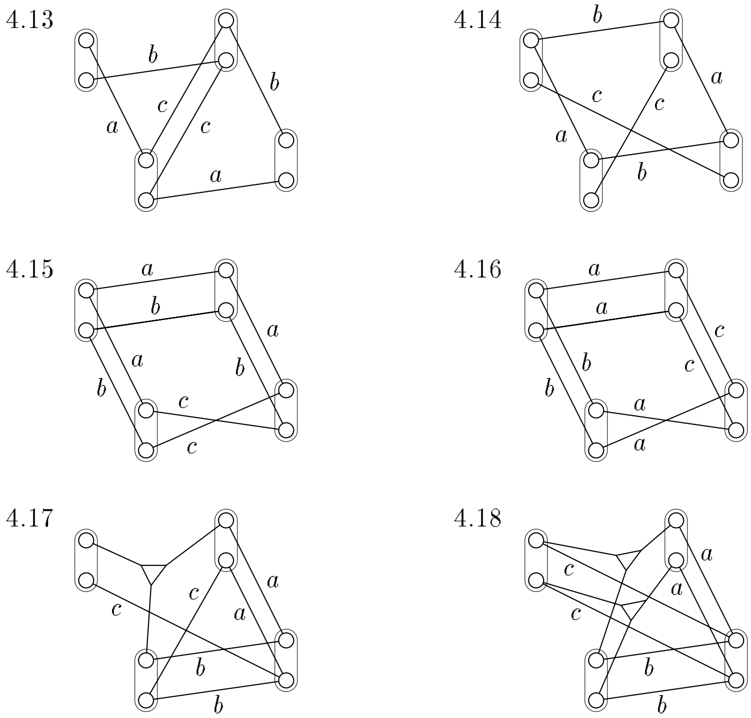

Figure 9. (*Continued*)

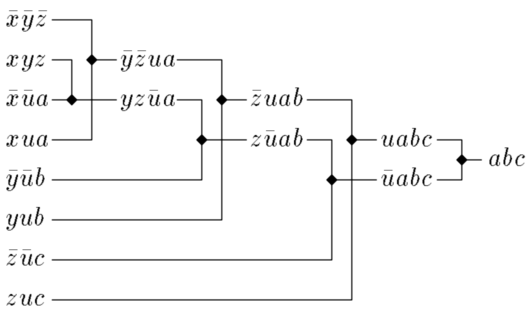

Figure 10.

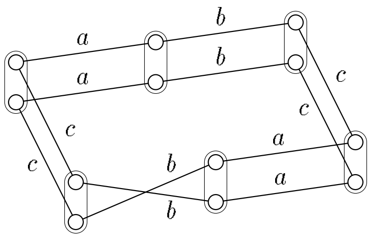

Figure 11.

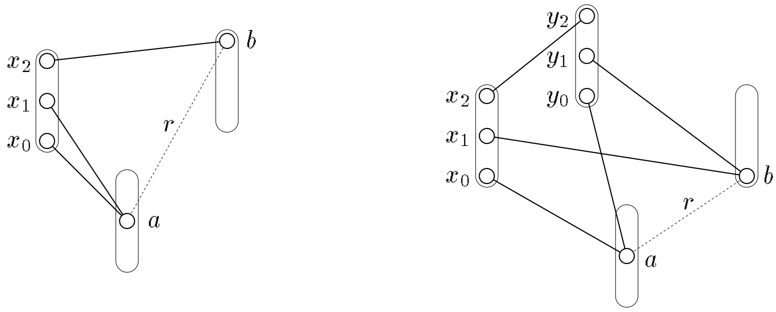

Figure 12.

Figure 13.

PROPOSITION 19 *All prime* $(2,3)$*-inferences of height* 1 *are of the form*

$$\{x_0a,\ x_1a,\ x_2b\} \models ab\,,$$

while all prime $(2,3)$*-inferences of height* 2 *are of the form*

$$\{x_0a,\ x_1b,\ y_0a,\ y_1b,\ x_2y_2\} \models ab\,.$$

Prime $(2,3)$*-inferences of heights* 1 *and* 2 *are primitive.*

Prime $(2,3)$-inferences of heights 1 and 2 are shown in Figure 12 in normal notation, and in Figure 13 in abbreviated notation.

Figure 14 lists all prime inferences of heights 3 and 4; all are primitive. Also, all are derivable by a 3-resolution on three-valued variables; a sample derivation of the prime (2,3)-inference 4.14 is shown in Figure 15.

There is a prime (2,3)-inference of height 6 (Figure 16) which is not derivable by 3-resolution. But this time it can be shown (we omit the proof), without explicitly determining all prime (2,3)-inferences of height 5, that each of these inferences can be derived by a judicious elimination of variables in W which never produces a more than ternary resolvent. In this case we therefore know the precise reach:

PROPOSITION 20 $\mathrm{reach}(2,3;4) = 7.$

4. Conclusion

We have shown the existence of prime inferences, we have characterized minimal (almost) complete inference systems for entailment in negative constrain networks

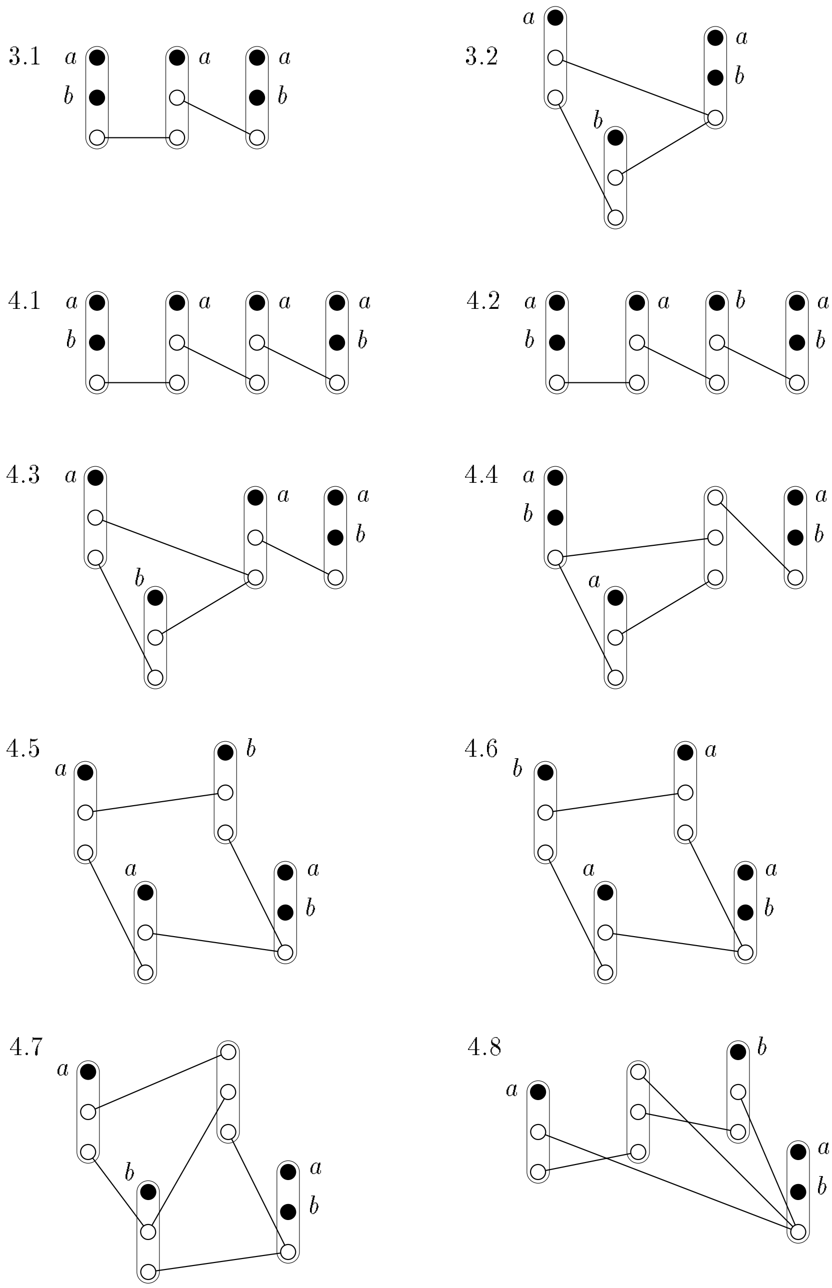

Figure 14.

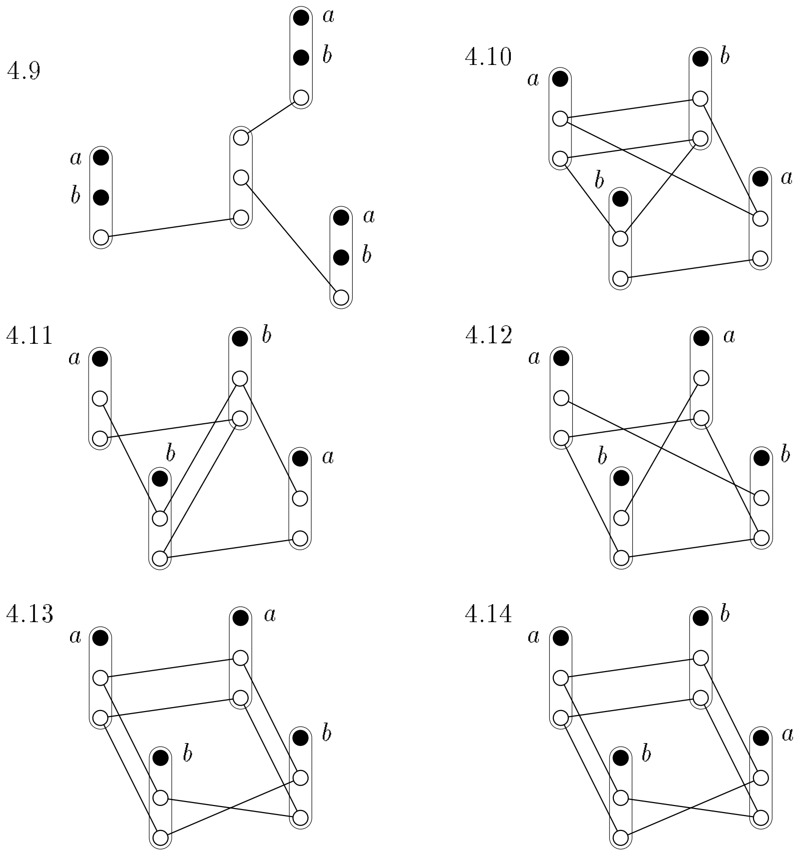

Figure 14. (*Continued*)

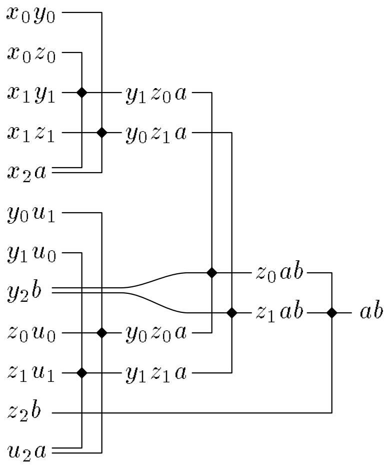

Figure 15.

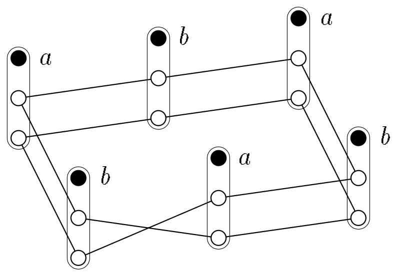

Figure 16.

as the representative sets of prime inferences, and, using complete lists of prime inferences of small heights, we have determined the reach of a low-level strong relative consistency for two special types of negative constraint networks. Practical results at this point are meager, to say the least. We are still lackig deep enough insight into properties of prime inferences, which would enable us to produce a provably close lower bound for the reach of the relative consistency.

But the prime inferences are worth further study. It is intriguing enough that they provide us with so neatly structured complete inference systems (even though the reason why this is so, namely the skewness of the entailment, is quite simple). By unearthing prime inferences we have in a sense revealed the first lower layer of complexity inherent in the finite constraint satisfaction problem. If anything, we must dig deeper, trying to find regularities in the structure of prime inferences, to figure out how they relate to each other, to discover ways to construct them, and so on. This will be the direction of further research.

References

Beeri, C., Fagin, R., Maier, D., and Yannakakis, M., "On the desirability of acyclic database schemes," *J. ACM*, **30**, 1983, pp. 479–513.

Cooper, M. C., Cohen, D. A., and Jeavons, P. G., "Characterizing tractable constraints," *Artif. Intell.*, **65**, 1994, pp. 347–361.

Dechter, R., and Pearl, J., "Tree clustering for constraint networks," *Artif. Intell.*, **38**, 1989, pp. 353–366.

Dechter, R., "From local to global consistency," *Artiff. Intell.*, **55**, 1992, pp. 87–107.

Freuder, E. C., "Synthesizing constraint expressions," *Comm. ACM*, **21**, 1978, pp. 958–966.

Freuder, E. C., "A sufficient condition for backtrack-free search," *J. ACM*, **29**, 1982, pp. 24–32.

Montanari, U., "Networks of constraints: Fundamental properties and applications to picture processing," *Inform. Sci.*, **7**, 1974, pp. 95–132.

van Beek, P., and Dechter, R., "Constraint tightness versus global consistency," in *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning*, Bonn, Germany, 1994, pp. 572–582.

Van Hentenryck, P., Deville, Y., and Teng, C.-M., "A generic arc consistency algorithm and its specializations," *Artif. Intell.*, **57**, 1992, pp. 291–321.