Lemmas 4.1, and 4.6. Therefore we may iteratively repeat the above step generating $\mathcal{F}_6$, ..., etc., until for some j the constructed $S(\mathcal{F}_j)$-path contains all vertices of G. This shows that G contains a Hamiltonian path which concludes the proof of Theorem 2.

□

References

[1] C.Delobel and R.G.Casey. Decomposition of a Data Base and the Theory of Boolean Switching Functions. *IBM J.Res.Develop.*, **17**(1973), 374–386.

[2] W.F. Dowling and J.H. Gallier. Linear time algorithms for testing the satisfiability of propositional Horn formulae. *Journal of Logic Programming*, **3**(1984), 267–284.

[3] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, San Francisco, 1979.

[4] M.R. Garey, D.S. Johnson and R.E. Tarjan. The planar Hamiltonian circuit problem is NP-complete. *SIAM J.Comput.*, **5**(1976), 704–714.

[5] P.L. Hammer and A. Kogan. Horn functions and their DNFs. *Information Processing Letters*, **44**(1992), 23–29.

[6] P.L. Hammer and A. Kogan. Horn Function Minimization and Knowledge Compression in Production Rule Bases. *RUTCOR Research Report RRR 8-92*, Rutgers University, New Brunswick, NJ, March 1992.

[7] P.L. Hammer and A. Kogan. Optimal Compression of Propositional Horn Knowledge Bases: Complexity and Approximation. *Artificial Intelligence*, **64**(1993), 131–145.

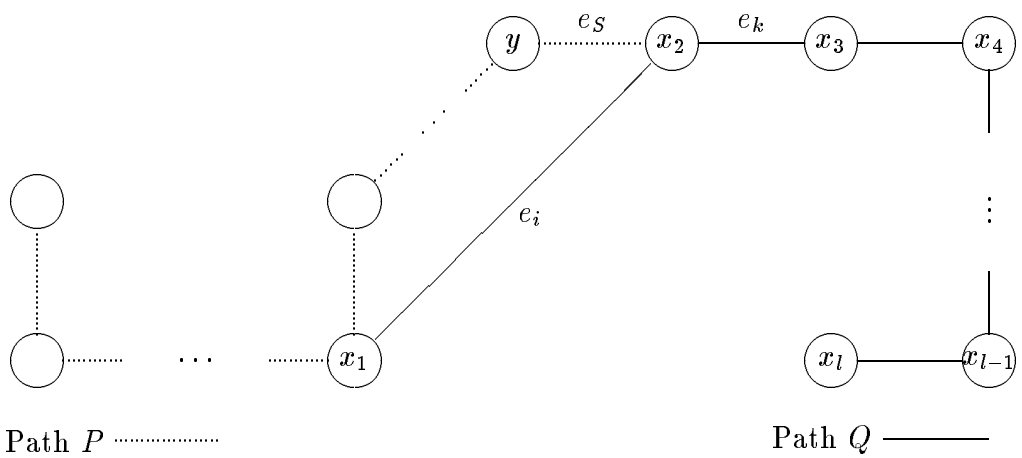

Figure 2: case $x_2 = head(P)$

- remove terms $v\overline{x}_2, v\overline{y}$ (where $\{y\} = S(\mathcal{F}_4)\setminus\{x_2\}$), $x_j x_{j+1}\overline{x}_{j+2}$ for $j = 1, 2, \ldots, \ell-2$ and $x_{\ell-1}x_\ell\overline{v}$;
- add terms $v\overline{x}_\ell, v\overline{x}_{\ell-1}, x_j x_{j-1}\overline{x}_{j-2}$ for $j = \ell, \ell-1, \ldots, 4$, $x_3x_2\overline{y}$ and $x_2x_1\overline{v}$.

Informally speaking, this modification relocates $S(\mathcal{F}_4) = \{x_2, y\}$ to the end of the path Q (making $S(\mathcal{F}_5) = \{x_\ell, x_{\ell-1}\}$), reverses the direction of Q, cuts off the edge e_i from the end of the reversed path Q, and hooks P onto the "new" end of of Q (behind the edge e_k). To prove a feasibility of this modification we again have to show how to derive all the removed terms from the modified form $\mathcal{F}_5$.

- We need to show that $\{x_2, y\} \subseteq R_{\mathcal{F}_4}(\{v\})$. First x_ℓ and $x_{\ell-1}$ are put into $R_{\mathcal{F}_4}(\{v\})$ and then by following the reversed path Q all $x \in V(Q)\setminus\{x_1\}$ (i.e. including x_2) are put in there as well. Using the term $x_3x_2\overline{y}$ finishes this part.
- Let $1 \le j \le \ell-1$ be arbitrary. By following the reversed path Q (starting at $\{x_j, x_{j+1}\}$) subsequently all $x_{j-1}, x_{j-2}, \ldots, x_2$ are put into $R_{\mathcal{F}_4}(\{x_j, x_{j+1}\})$. Using $x_3x_2\overline{y}$ first y and then by following P all $x \in V(P)$ are put into $R_{\mathcal{F}_4}(\{x_j, x_{j+1}\})$. The last edge in P includes also v and so in the case $j = \ell-1$ the term $x_{\ell-1}x_\ell\overline{v}$ is derived. If $j < \ell-1$ then x_ℓ and $x_{\ell-1}$ are put in, and by following the reversed Q the variables $x_{\ell-3}, x_{\ell-4}, \ldots, x_{j+2}$ are subsequently put in deriving the term $x_j x_{j+1}\overline{x}_{j+2}$.

Similarly as in the case $x_2 = tail(P)$ the $S(\mathcal{F}_5)$-path consisting of the reversed path Q (without the edge e_i) concatenated with the path P is longer by at least one edge than the original $S(\mathcal{F}_4)$-path P.

It is easy to check that the modified form $\mathcal{F}_5$ is again a minimal prime DNF of f_G, and that it has all the necessary properties of $\mathcal{F}_4$, in particular it satisfies the properties in

then immediately "leaves P". Lemma 4.1b guarantees that Q does not "return back into P, i.e. it guarantees that $V(P) \cap V(Q) = \{x_1, x_2\}$. Now we shall show how to use Q to make P "longer" by at least one edge.

- Case $x_2 = tail(P)$.
  We can modify $\mathcal{F}_4$ as follows :

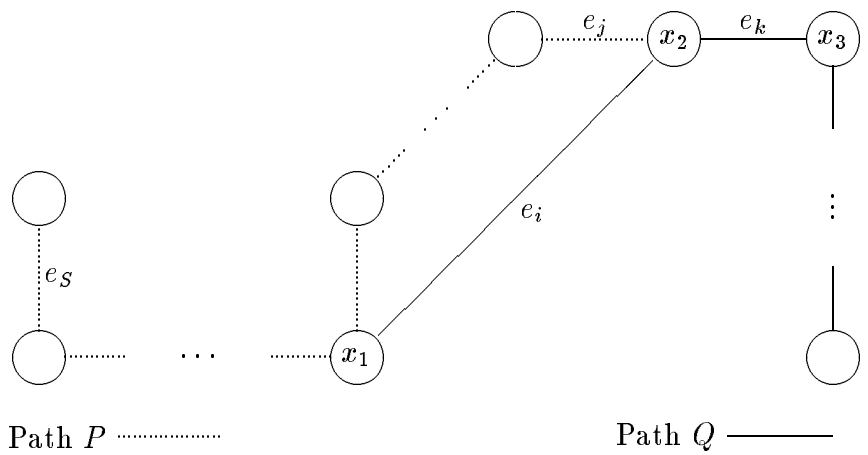

Figure 1: case $x_2 = tail(P)$

  – remove terms $e_i\overline{x}_3$ and $e_j\overline{v}$, where e_j is the last edge in P, and
  – add terms $e_j\overline{x}_3$ and $e_i\overline{v}$.

  To prove that this is a feasible modification we have to show that the terms $e_i\overline{x}_3$ and $e_j\overline{v}$ can be derived from the modified form (let us call it $\mathcal{F}_5$).

  – We need to show that $x_3 \in R_{\mathcal{F}_4}(e_i)$. The term $e_i\overline{v}$ puts v into $R_{\mathcal{F}_4}(e_i)$. In the next step $S(\mathcal{F}_4) = S(\mathcal{F}_5)$ is included into $R_{\mathcal{F}_4}(e_i)$. By following P all $x \in V(P)$ are then subsequentely put into $R_{\mathcal{F}_4}(e_i)$, including the last two, i.e. the edge e_j. Finally, the term $e_j\overline{x}_3$ gives us what we need.
  – We also have to show that $v \in R_{\mathcal{F}_4}(e_j)$. The term $e_j\overline{x}_3$ puts x_3 into $R_{\mathcal{F}_4}(e_j)$. Now we have $e_k \subseteq R_{\mathcal{F}_4}(e_j)$ and by following Q (starting at e_k) all $x \in V(Q) \setminus \{x_1\}$ are put into $R_{\mathcal{F}_4}(e_j)$. Then the last edge of Q adds v.

  The path obtained by concatenation of P and $Q \setminus \{e_i\}$ is a $S(\mathcal{F}_5)$-path which is longer than P by at least one edge (at least e_k is in $Q \setminus \{e_i\}$).

- Case $x_2 = head(P)$.
  This case is more complicated. Let us assume that $V(Q) = \{x_1, \ldots, x_\ell\}$ (in this order on the path). We then modify $\mathcal{F}_4$ as follows (call the resulting form $\mathcal{F}_5$) :

5 Minimum DNFs of f_G and Hamiltonian Paths in G

We are now almost ready to prove Theorem 2. However, before doing so it will be useful to define the notion of an e-path. Let $e \in I$ be an arbitrary edge and let $\{e = e_1, e_2, \ldots, e_l\} \subseteq E$ be a set of edges, $e_j = (y_{j-1}, y_j)$, such that $e_j \in N_{e_{j-1}}$, $j = 2, \ldots, l$ (i.e. $e_{j-1}\overline{y}_j$ is a term in $\mathcal{F}_4, j = 2, \ldots, l$) and such that $e_l \in T$ (i.e. $\mathcal{F}_4$ contains the term $e_l\overline{v}$). Then $\{e = e_1, e_2, \ldots, e_l\}$ will be called an e-path in the graph G. Note that an e-path is not uniquely defined for a given edge e. Let P be an e-path in G. Then we denote :

- $E(P) = \{e_1, e_2, \ldots, e_l\}$ the set of all edges in P;
- $V(P) = \{y_0, y_1, \ldots, y_l\}$ the set of all vertices in P;
- $head(P) = y_0$ the first vertex in P; and
- $tail(P) = y_l$ the last vertex in P.

Note that by Lemma 4.1 every e-path is a simple path, i.e. no repetition of vertices can occur. Let us now prove Theorem 2.

Proof of Theorem 2.

First let us assume that G has a Hamiltonian path $P = \{e_1, e_2, \ldots, e_{n-1}\}$, $e_i = \{x_i, x_{i+1}\}$ for $1 \leq i \leq n-1$. Then we construct a DNF $\mathcal{F}'$ of the function f_G containing exactly $m+2$ terms as follows:

$$\mathcal{F}' = (v\overline{x}_1 \vee v\overline{x}_2) \vee (\bigvee_{i=1}^{n-2} e_i\overline{x}_{i+2}) \vee (e_{n-1}\overline{v}) \vee (\bigvee_{e \in E \setminus P} e\overline{v})$$

It is not hard to check that by following the path P one can easily derive all implicants $v\overline{x}_i$ for $3 \leq i \leq n$ and by following P backwards all implicants $e_i\overline{v}$ for $n-2 \geq i \geq 1$. Therefore $\mathcal{F}'$ indeed represents f_G, which finishes one direction of the proof.

Let us assume now for the converse direction that f_G has a DNF representation $\mathcal{F}_0$ containing at most $m+2$ terms. Using Modifications 0,1,2, and 3 we first construct $\mathcal{F}_4$. By Lemmas 4.5, 4.7, and 4.6a) (in this order) we consequently get that $\mathcal{F}_4$ has exactly $m+2$ terms, $|I| = n-2$, and $S = e_S$ for some $e_S \in I$. Let P be a e_S-path in G. If $V(P) = V$ then P is Hamiltonian and we are done. So let us assume that there exists $x \notin V(P)$. It is easy to see that $V(P) \subseteq R_{\mathcal{F}_4}(\{v\})$ since the variables in $V(P)$ are subsequently put into $R_{\mathcal{F}_4}(\{v\})$ by "going along" the path P. However, $v\overline{x}$ is also an implicant of $\mathcal{F}_4$ and thus $x \in R_{\mathcal{F}_4}(\{v\})$ must hold as well. This can be achieved only if $\mathcal{F}_4$ contains a term $x_1x_2\overline{x}_3$ such that $e_i = \{x_1, x_2\} \subseteq V(P)$ and $x_3 \notin V(P)$. Since for all $e_j \in E(P)$ we have $H_{e_j} \subseteq V(P)$ (remember that $|H_{e_j}| = 1$ by Lemma 4.6), $e_i \notin E(P)$ must hold. Moreover, by Lemma 4.2, $N_{e_i} \neq \emptyset$ holds and so we may assume w.l.o.g. that $\{x_2, x_3\} = e_k \in E$. Now since G is cubic either $x_2 = tail(P)$ or $x_2 = head(P)$ must hold, since otherwise x_2 has degree at least 4 in G. Let Q be an e_i-path in G such that $e_k \in E(Q)$, i.e. the path Q "starts inside of P" and

Proof. The first inequality follows directly by Lemma 4.4, the second by property b) of Lemma 4.1, and $|D| \leq n-2$ follows by property a) of Lemma 4.1 using the inequality $|S| \geq 2$. To count the number of terms in $\mathcal{F}_4$ let us recall its form:

$$\mathcal{F}_4 = (\bigvee_{x \in S} v\overline{x}) \vee (\bigvee_{e \in I} \bigvee_{x \in H_e} e\overline{x}) \vee (\bigvee_{e \in T} e\overline{v}).$$

Thus for the number of terms we get

$$\begin{aligned} \#\text{ terms in } \mathcal{F}_4 &= |S| + \textstyle\sum_{e \in I} |H_e| + |T| = \\ &= |S| + |D| + |T| \geq \\ &\geq |S| + |I| + |T| = |S| + m \geq \\ &\geq 2 + m. \end{aligned} \tag{5}$$

The first equality here follows from property b) of Lemma 4.1. The subsequent inequalities are direct applications of the ones shown above in this lemma.

□

Now we shall state a simple yet useful lemma for the special case when $|I|$ attains its maximum possible value, i.e. when $|I| = n-2$.

Lemma 4.6 *If $|I| = n-2$ then*

a) *$S = e$ for some $e \in E$;*

b) *$|H_e| = 1$ for all $e \in I$;*

c) *for all $e \in I$ and $e' \in N_e$ one has $|e \cap e'| = 1$, i.e. the edges e' and e constitute a path of length 2 in G; and*

d) *$\mathcal{F}_4$ contains exactly $m+2$ terms.*

Proof.
a). By Lemma 4.1a) we have $|S| + |D| = n$ and by Lemma 4.5 we have $n-2 \geq |D| \geq |I| = n-2$ which gives $|D| = n-2$ and thus $|S| = 2$. On the other hand, by Lemma 4.4, S must contain an edge of G.
b). By part a) $|D| = |I|$ holds and hence using Lemma 4.1b) $|H_e| = 1$ must hold for all $e \in I$.
c). This is a trivial consequence of part b) and the fact that G is a simple graph.
d). By part b) we have $\sum_{e \in I} |H_e| = |I|$ and hence all inequalities in (5) become equalities.

□

The last lemma of this section shows that the converse to part d) of lemma 4.6 is also true.

Lemma 4.7 *If $\mathcal{F}_4$ contains exactly $m+2$ terms then $|I| = n-2$.*

Proof. All inequalities in (5) become in this case equalities, which means that $|S| = 2$ and $|I| = |D|$. However $|S| + |D| = n$ by lemma 4.1a) and hence $|I| = n-2$.

□

Lemma 4.2 *For every edge $e \in I$ we have $N_e \neq \emptyset$.*

Proof. Let us do forward chaining on $\mathcal{F}_4$ starting with an edge $e \in I$. After the first round we have the set $e \cup H_e$ included in $R(e)$. Now, if N_e were empty, we would stop at this point without including v in $R(e)$. This, however would contradict the fact that $e\overline{v}$ is a prime implicant of f_G, and thus by Lemma 3.2, v must belong to $R(e)$.

□

Lemma 4.3 *If the simple graph G has at least two edges (i.e. if $m > 1$), then $I \neq \emptyset$.*

Proof. Assume that the contrary is true, i.e. that $I = \emptyset$, and let the two edges in the graph be e and e', $e \neq e'$. Since $I = \emptyset$ we have $e, e' \in T$, i.e. $\mathcal{F}_4$ contains terms $e\overline{v}$ and $e'\overline{v}$. Also, $x \in S$ must hold for all $x \in V$, so in particular $\mathcal{F}_4$ contains terms $v\overline{x}$ for all $x \in e' \setminus e$. Now the following modification yields a DNF equivalent to $\mathcal{F}_4$, and decreases the number of terms by one.

- Delete the terms $e\overline{v}$, and $v\overline{x}$ for $x \in e' \setminus e$.
- Add the terms $e\overline{x}$ for $x \in e' \setminus e$.

It is easy to see, that if we call the result of the above modification $\mathcal{F}_5$ then $\{v\} \in R_{\mathcal{F}_5}(e)$ and $\{x\} \in R_{\mathcal{F}_5}(v)$ for all $x \in e' \setminus e$. Therefore both DNFs are indeed equivalent which is a contradiction to the minimality of $\mathcal{F}_4$.

□

Lemma 4.4 *If $m > 1$, then there exists an edge $e \in I$ for which $e \subseteq S$.*

Proof. By the previous lemma, if the simple graph G contains at least two edges, then $I \neq \emptyset$, and thus $D \neq \emptyset$, by the definition of these sets. Let $x \in D$ be arbitrary and let us investigate $R(\{v\})$. Since $v\overline{x}$ is an implicant of $f_G = \mathcal{F}_4$, $x \in R(\{v\})$ must hold. Let $e\overline{y}$ be the first intermediate term used in the above forward chaining (such a term necessarily exists since the literal $\overline{x}$ appears only once in $\mathcal{F}_4$ and that is in an intermediate term). This means that the endpoints of e were put into $R(\{v\})$ using only starting terms and therefore $e \subseteq S$.

□

Lemma 4.5 *If $m > 1$, then $|S| \geq 2$, $|I| \leq |D| \leq n - 2$, and $\mathcal{F}_4$ contains at least $m + 2$ terms.*

This is very similar to Modification 1. However, to prove the feasibility we have to show that the term $e'\overline{x}$ can be derived from the modified form of $\mathcal{F}_2$, since this time the term $v\overline{x}$ is not explicitely present as in the previous case. But it is easy to see that by the choice of e' and e the term $v\overline{x}$ can be derived from $\mathcal{F}_2$ without using the term $e'\overline{x}$, since otherwise x would have been put into $R_{\mathcal{F}_2}(\{v\})$ by $e'\overline{x}$ rather than $e\overline{x}$. Therefore $v\overline{x}$ can be derived from the modified form of $\mathcal{F}_2$ and thus so does $e'\overline{x}$ (using $v\overline{x}$ and $e'\overline{v}$). Since this modification preserves the number of terms it is feasible. If the resulting DNF (say $\mathcal{F}'$) has $S(\mathcal{F}') \cap D(\mathcal{F}')$ nonempty then use Modification 1 to get $S(\mathcal{F}') \cap D(\mathcal{F}') = \emptyset$.

Repeat the above step as long as $|A(x)| > 1$ for some x (i.e. $H_e(\mathcal{F}_2) \cap H_{e'}(\mathcal{F}_2) = \{x\}$ for some $e \neq e'$) and call the resulting form $\mathcal{F}_3$. Note that the above process cannot cycle since every step (both in Modification 1 and Modification 2) reduces the number of intermediate terms by at least one.

These prove that $\mathcal{F}_3$ is a minimum prime DNF of f_G satisfying both properties (a) and (b).

Modification 3 : If there exists an edge $e \in I(\mathcal{F}_3) \cap T(\mathcal{F}_3)$, then modify $\mathcal{F}_3$ as follows:

- remove terms $e\overline{y}$ for all $y \in H_e(\mathcal{F}_3)$, and
- add terms $v\overline{y}$ for all $y \in H_e(\mathcal{F}_3)$.

Since $e\overline{v}$ is a term in $\mathcal{F}_3$, the above is obviously a feasible modification (each $e\overline{y}$ can be derived using $v\overline{y}$ and $e\overline{v}$).

Repeat the above step as long as $I(\mathcal{F}_3) \cap T(\mathcal{F}_3) \neq \emptyset$, and call the resulting form $\mathcal{F}_4$. Clearly, these last modifications preserved properties (a) and (b). We claim that $\mathcal{F}_4$ now satisfies property (c), as well. We have, on one hand, $I(\mathcal{F}_3) \cap T(\mathcal{F}_3) = \emptyset$ by Modification 3. On the other hand, if there were edges $e \in E \setminus (I(\mathcal{F}_3) \cup T(\mathcal{F}_3))$, then $R_{\mathcal{F}_4}(e) = e$ would follow, contradicting the fact that $e\overline{v}$ is a prime implicant of f_G, and thus v must belong to $R_{\mathcal{F}_4}(e)$ by Lemma 3.2.

These prove that $\mathcal{F}_4$ is a minimum prime DNF of f_G satisfying properties (a), (b), and (c).

The above procedure necessarily terminates after a finite number of steps, since every modification step in the above procedure decreased strictly the number of intermediate terms. Since the total number of possible intermediate terms of f_G is bounded by nm, all the above modifications can be carried out in polynomial time, which concludes our proof.

□

We can show now a few further properties of $\mathcal{F}_4$. Let us introduce the notation $N_e(\mathcal{F}_4) = \{e' \in E \mid e \neq e' \subseteq e \cup H_e(\mathcal{F}_4)\}$; the set $N_e(\mathcal{F}_4)$ refers to potential terms which can be activated "next" in the forward chaining procedure after the activation of the terms involving the edge e.

In the remaining part of this section all statements will involve $\mathcal{F}_4$, hence, to simplify the notation, we drop the reference to it from the formulae, e.g. we write simply I instead of $I(\mathcal{F}_4)$, etc.

Properties (a) and (b) ensure that $\mathcal{F}'$ and $\mathcal{F}''$ are both defining the same function f, and (c) guarantees that if $\mathcal{F}'$ is a minimum DNF of the function f then so is $\mathcal{F}''$.

Modification 0: For each term of $\mathcal{F}_0$ we construct (in quadratic time, see [5]) a prime implicant of f_G absorbing it. Let $\mathcal{F}_1$ be the disjunction of these prime implicants. We claim that $\mathcal{F}_1$ is equivalent to $\mathcal{F}_0$. By the construction of $\mathcal{F}_1$, the inequality $\mathcal{F}_0 \leq \mathcal{F}_1$ holds (informally speaking $\mathcal{F}_1$ contains "shorter" terms than $\mathcal{F}_0$). On the other hand since $\mathcal{F}_1$ consists only of prime implicants of f_G we have $\mathcal{F}_1 \leq f_G = \mathcal{F}_0$. Hence $\mathcal{F}_1$ is equivalent to $\mathcal{F}_0$. Moreover since $\mathcal{F}_1$ consists only of prime implicants of f_G, it is a prime DNF of the function f_G.

Thus $\mathcal{F}_1$ must have the form (4) and it can be further modified as follows :

Modification 1: If there exists a variable $x \in V$ such that both $v\overline{x}$ and $e\overline{x}$ (for some $e \in I(\mathcal{F}_1)$) are terms of $\mathcal{F}_1$, i.e. $x \in S(\mathcal{F}_1) \cap D(\mathcal{F}_1)$, then modify $\mathcal{F}_1$ as follows :

- remove terms $e\overline{y}$ for all $y \in H_e(\mathcal{F}_1)$, i.e. including $e\overline{x}$ and
- add terms $v\overline{y}$, $y \in H_e(\mathcal{F}_1) \setminus \{x\}$ and the term $e\overline{v}$.

Clearly, the term $e\overline{y}$ is derivable from the "new" DNF, using the added terms $e\overline{v}$ and $v\overline{y}$, for each $y \in H_e(\mathcal{F}_1)$. Moreover the number of removed terms equals to the number of added terms (in both cases $|H_e(\mathcal{F}_1)|$). Hence this is a feasible modification. Note that the term $e\overline{x}$ does not have to be added since it is already present in $\mathcal{F}_1$ by the choice of x.

Repeat the above step as long as $S(\mathcal{F}_1) \cap D(\mathcal{F}_1)$ is nonempty and then call the result $\mathcal{F}_2$. We claim that after these modifications there is no variable $x \notin S(\mathcal{F}_2) \cup D(\mathcal{F}_2)$. For if there were, then the literal $\overline{x}$ would not appear at all in $\mathcal{F}_2$, and therefore the term $v\overline{x}$ could not be derived from $\mathcal{F}_2$ contradicting the fact that $v\overline{x}$ is a prime implicant of f_G. These prove that $\mathcal{F}_2$ is a minimum prime DNF of f_G obeying property (a).

Let us introduce the notation $A(x) = \{e \in E \mid x \in H_e(\mathcal{F}_2)\}$. Obviously, $A(x) \neq \emptyset$ only if $x \in D(\mathcal{F}_2)$.

Modification 2: If there exists a variable $x \in D(\mathcal{F}_2)$ such that $|A(x)| > 1$ i.e. if $\mathcal{F}_2$ contains more than one intermediate terms containing $\overline{x}$, then proceed as follows. Since $v\overline{x}$ is an implicant of $\mathcal{F}_2$, $x \in R_{\mathcal{F}_2}(\{v\})$ must hold, but by property (a) we know that $v\overline{x}$ is not in $\mathcal{F}_2$. Therefore there must be an edge $e \in A(x)$ for which the term $e\overline{x}$ is the one which places x into $R_{\mathcal{F}_2}(\{v\})$ in the forward chaining procedure. Since $|A(x)| > 1$, there exists another edge $e' \in A(x)$, $e \neq e'$. Let us now modify $\mathcal{F}_2$ as follows:

- remove terms $e'\overline{y}$ for all $y \in H_{e'}(\mathcal{F}_2)$, i.e. including $e'\overline{x}$ and
- add terms $v\overline{y}$, $y \in H_{e'}(\mathcal{F}_2) \setminus \{x\}$ and the term $e'\overline{v}$.

- $S(\mathcal{F}) = \{x \in V \mid v\overline{x} \text{ is a term of } \mathcal{F}\}$;
- $H_e(\mathcal{F}) = \{x \in V \mid e\overline{x} \text{ is a term of } \mathcal{F}\}$, $e \in E$;
- $I(\mathcal{F}) = \{e \in E \mid H_e(\mathcal{F}) \neq \emptyset\}$;
- $T(\mathcal{F}) = \{e \in E \mid e\overline{v} \text{ is a term of } \mathcal{F}\}$; and finally
- $D(\mathcal{F}) = \bigcup_{e \in I(\mathcal{F})} H_e(\mathcal{F})$.

Using the above notation any prime DNF of f_G can be written as

$$\mathcal{F} = (\bigvee_{x \in S(\mathcal{F})} v\overline{x}) \vee (\bigvee_{e \in I(\mathcal{F})} \bigvee_{x \in H_e(\mathcal{F})} e\overline{x}) \vee (\bigvee_{e \in T(\mathcal{F})} e\overline{v}) \tag{4}$$

For example, for the defining DNF $\mathcal{F}_G$ of (2) we have $S(\mathcal{F}) = V$, $H_e(\mathcal{F}) = \emptyset$ for all edges $e \in E$, thus $I(\mathcal{F}) = \emptyset$ and $D(\mathcal{F}) = \emptyset$, and finally $T(\mathcal{F}) = E$. Thinking of a forward chaining starting from v, $S(\mathcal{F})$ denotes the "starting" set of variables, encountered first by the procedure, and $I(\mathcal{F})$ and $T(\mathcal{F})$ correspond to edges appearing in "intermediate" and "terminal" steps of the chaining procedure respectively. Following this informal terminology, we shall call the terms appearing in Lemma 3.3 as category a) the starting terms and as categories b) and c) intermediate and terminal terms respectively. The set $H_e(\mathcal{F})$ is the set of "heads" of intermediate terms corresponding to the edge $e \in E$, and $D(\mathcal{F})$ denotes the "domain of heads" in intermediate terms.

In the remainder of this section we shall show that among the minimum DNFs of f_G there is always one which obeys some nice properties in terms of the above notation, moreover that one can get to such a form in polynomial time from any minimum DNF.

Lemma 4.1 *Given a minimum DNF $\mathcal{F}_0$ of f_G, one can derive from it another minimum DNF $\mathcal{F}_4$, which is prime and satisfies the following properties:*

a) *$S(\mathcal{F}_4)$ and $D(\mathcal{F}_4)$ make up a partition of V;*

b) *$H_e(\mathcal{F}_4) \cap H_{e'}(\mathcal{F}_4) = \emptyset$ for all $e \neq e'$;*

c) *$I(\mathcal{F}_4)$ and $T(\mathcal{F}_4)$ make up a partition of E;*

Proof. We shall perform a series of modifications on $\mathcal{F}_0$ to arrive to $\mathcal{F}_4$. For this we need the following notions.

We shall say that a pure Horn term $Q\overline{y}$ is *derivable* from a DNF $\mathcal{F}$ iff $y \in R_{\mathcal{F}}(Q)$, or in other words if it is an implicant of the function defined by $\mathcal{F}$.

Let $\mathcal{F}'$ be an arbitrary pure Horn DNF of a pure Horn function f. A pure Horn DNF $\mathcal{F}''$ is called a *feasible modification* of $\mathcal{F}'$ if on one hand (a) all terms of $\mathcal{F}''$ which do not appear in $\mathcal{F}'$ are prime implicants of f and, on the other hand, (b) all terms of $\mathcal{F}'$ which are not in $\mathcal{F}''$ are derivable from $\mathcal{F}''$, and finally if (c) $\mathcal{F}'$ and $\mathcal{F}''$ contain the same number of terms.

Lemma 3.2 *A pure Horn term $Q\overline{y}$ is an implicant of a pure Horn function f given by a pure Horn DNF $\mathcal{F}$ if and only if $y \in R_{\mathcal{F}}(Q)$.*

Let us consider again the pure Horn DNF $\mathcal{F}_G$ given by (2) associated to the graph $G = (V, E)$. It is easy to see that this DNF is prime, i.e. that dropping a literal from any of the terms of $\mathcal{F}_G$ would produce a term which is not an implicant of f_G. The following lemma characterizes all prime implicants of f_G.

Lemma 3.3 *Let $G = (V, E)$ be a graph and let us consider the associated function f_G defined by the DNF (2). An elementary conjunction T is a prime implicant of f_G iff one of the following is true :*

a) *$T = v\overline{x}_k$ for some $x_k \in V$; or*

b) *$T = x_i x_j \overline{x}_k$ for some $(x_i, x_j) \in E$ such that $k \notin \{i, j\}$; or*

c) *$T = x_i x_j \overline{v}$ for some $(x_i, x_j) \in E$.*

Proof. Let us recall that a subset $S \subseteq V$ of the vertices of a graph G is called *stable* if no edge of G has both of its endpoints in S. One can observe now that

$$\begin{array}{lll} R_{\mathcal{F}_G}(S) & = S & \text{if } S \subseteq V \text{ is a stable set of } G, \\ R_{\mathcal{F}_G}(\{x_i, x_j\}) & = V \cup \{v\} & \text{if } (x_i, x_j) \in E, \text{ and} \\ R_{\mathcal{F}_G}(\{v\}) & = V \cup \{v\}. & \end{array} \tag{3}$$

This implies, according to Lemma 3.2 that any term of category a), b) or c) is an implicant of f_G, moreover that no truncation of such an implicant yields an implicant, thus those terms are indeed prime implicants of f_G.

On the other hand, if T is a prime implicant of f_G, then it must be pure Horn, since all prime implicants of (the pure Horn function) f_G must be pure Horn (see e.g. [5]). Thus we may assume that $T = Q\overline{y}$ for some subset Q of the variables, and $y \notin Q$. Then $R_{\mathcal{F}_G}(Q) \neq Q$ (since it must contain y), and thus by (3) either v or an edge $(x_i, x_j) \in E$ must be contained in Q. The primeness of T thus implies in the first case that $T = v\overline{y}$, i.e. that T is of category a), and in the second case that $T = x_i x_j \overline{y}$, and so T is of category b) or c) depending on y.

□

4 Minimum Forms of f_G

In this section we characterize, in some sense, the minimum DNFs of a pure Horn function f_G associated to a graph $G = (V, E)$.

For this we introduce the following shorthand. If $e = (x_i, x_j) \in E$ is an edge of the graph G, then the terms $x_i x_j \overline{v}$ and $x_i x_j \overline{x}_k$ will also be written as $e\overline{v}$ and $e\overline{x}_k$, respectively. Now, if $\mathcal{F}$ is an arbitrary prime DNF of f_G it must consist (by Lemma 3.3) only of terms of categories a),b), and c). Let us introduce the following notation

Let us show first that problem HM is indeed in NP. For this we shall follow the lines of the proof of an analogous statement from [7]. We include here the short proof for completeness.

Lemma 3.1 *The Horn Minimization problem is in NP.*

Proof. If the answer to a given instance of HM is *yes* and we are given a (not necessarily Horn) DNF $\mathcal{F}'$ as the "certificate", we first count the number of terms in $\mathcal{F}'$ to check that this number is not larger than q. If yes, we construct for each term of $\mathcal{F}'$ (in quadratic time, see [5]) a prime implicant of $\mathcal{F}$ absorbing it. Let $\mathcal{F}''$ be the disjunction of these prime implicants. Clearly, $\mathcal{F}''$ is Horn and consists of at most q terms.

Next we check whether $\mathcal{F}$ and $\mathcal{F}''$ are equivalent. For this we check first the inequality $\mathcal{F}'' \leq \mathcal{F}$ by verifying (in quadratic time, see [2]) that every term of $\mathcal{F}''$ is an implicant of $\mathcal{F}$. If yes, then we finish by checking the reverse inequality $\mathcal{F} \leq \mathcal{F}''$ in a similar way.

If $\mathcal{F}''$ passed the above test, it itself verifies that the answer to the given instance of HM is indeed yes. On the other hand if it failed the test, i.e. if $\mathcal{F}''$ is not equivalent to $\mathcal{F}$, then neither is $\mathcal{F}'$ and we can reject it as a certificate.

□

Since our focus here is to decide the complexity of finding a minimum DNF of a given pure Horn function (recall that for every graph G the function f_G is pure Horn), and since a prime DNF can be obtained from any pure Horn DNF representation of a pure Horn function in polynomial time (see e.g. [5]) without increasing the number of terms, we may restrict our attention solely to prime implicants and prime DNFs of pure Horn functions without any loss of generality.

In verifying that a given term is an implicant of a given pure Horn function, a very useful and simple procedure is the following. Given a pure Horn DNF $\mathcal{F}$ of a pure Horn function f, we shall define a *forward chaining* procedure which associates to any subset Q of the variables of f a set R in the following way. The procedure takes as input the subset Q of variables, initializes the set $R = Q$, and at each step it looks for a pure Horn term $T\overline{y}$ in $\mathcal{F}$ such that $T \subseteq R$, and $y \notin R$. If such a term is found, the variable y is included into R, and the search is repeated as many times as possible.

FORWARD CHAINING

Input:	A pure Horn DNF $\mathcal{F} = \bigvee_{i=1}^{m} S_i \overline{x}_{j_i}$ and a subset Q of the variables.
Initialization:	Set $R = Q$.
Main Step:	**while** $\exists\ i$ s.t. $S_i \subset R$ and $x_{j_i} \notin R$ **do** $R = R \cup \{x_{j_i}\}$.
Stop:	Output $R_{\mathcal{F}}(Q) = R$.

In a relational database terminology the variables in $R_{\mathcal{F}}(Q)$ are said to be "chained" to the subset Q (see e.g. [1]). In an expert system terminology the usage of the term $T\overline{y}$ is called "firing the rule" $\bigwedge_{x \in T} x \rightarrow y$ (see e.g. [6]).

The following lemma (proved in [7]) will be frequently used in the rest of this paper.

where v denotes an extra Boolean variable. Let us observe that the DNF (2) consists of $m + n$ terms.

A graph $G = (V, E)$ is said to be *cubic*, if every vertex of G has *degree* 3, i.e. if every vertex of G is incident with exactly 3 edges.

Theorem 2 *Let $G = (V, E)$ be a cubic graph. The associated pure Horn function f_G has a DNF representation of at most $m + 2$ terms if and only if G has a Hamiltonian path, i.e. a simple path that visits every vertex exactly once.*

The proof of Theorem 2 (which is deferred until Section 5) will show that any DNF representation of f_G for a cubic graph G must contain at least $m + 2$ terms. Therefore any DNF of f_G containing at most $m+2$ terms must be minimal and must consist of exactly $m+2$ terms. It will be shown that a DNF consisting of exactly $m + 2$ terms exists iff the graph G contains a Hamiltonian path. Furthermore, the proof will be constructive in the sense that if one is given a minimum DNF of f_G consisting of $m + 2$ terms, then a Hamiltonian path of G can be constructed from it in time polynomial in n. Conversely, the proof will show that if a Hamiltonian path is given in G, then a minimum DNF representation of f_G consisting of $m + 2$ terms can be constructed, again in polynomial time. This constitutes the needed polynomial time reduction.

Let us remark that since for every graph G the DNF $\mathcal{F}_G$ is pure Horn and cubic (a formula is cubic if each of its terms consists of at most 3 literals), our proof yields the following somewhat stronger statement.

Corollary 2.1 *Problem HM remains NP-complete even if the input is restricted to the class of cubic pure Horn formulae.*

3 Prime Implicants of f_G

To present the proof of the main result, we need to introduce some further definitions.

For a Boolean function f we write $f \leq g$ if $f(x_1, \ldots, x_n) = 1$ implies $g(x_1, \ldots, x_n) = 1$ for every 0-1 vector. We say that a term T_1 *absorbes* another term T_2 if $T_2 \leq T_1$ (e.g. the term $x\overline{z}$ absorbes the term $xy\overline{z}$). A term T is called an *implicant* of the function f if $T \leq f$. An implicant T of a function is called *prime* iff there is no distinct implicant T' absorbing T, or in other words, an implicant of a function is prime iff dropping any literal from it produces a term which is not an implicant of the function.

A DNF $\mathcal{F}$ representing a function f is called *prime* iff each term of $\mathcal{F}$ is a prime implicant of the function. For the purposes of this paper a DNF of a function is called *minimum* iff there is no DNF representing the same function and containing fewer terms. It is not hard to show, that among all minimum DNFs there is always at least one which is prime.

For the case of Horn or pure Horn functions, it is well-known (see e.g. [5]) that each prime implicant of a (pure) Horn function is (pure) Horn. Thus, in particular, any prime DNF representing a pure Horn function is pure Horn.

literals respectively. An elementary conjunction

$$T = \bigwedge_{i \in I} x_i \bigwedge_{j \in J} \overline{x}_j \tag{1}$$

of literals is called a *term*, if every variable appears in it at most once, i.e. if $I \cap J = \emptyset$. The term T is called *Horn* if it contains at most one negative literal, and it is called *pure Horn* (or in some literature *definite Horn*) if it contains exactly one negative literal. To simplify notation we shall write conjunctions in the sequel as products, e.g. xy instead of $x \wedge y$.

It is well known that every Boolean function f can be represented as a disjunction of elementary conjunctions. Such an expression is called a *disjunctive normal form* (or DNF) of the Boolean function f. A DNF is called *Horn* (*pure Horn*) iff it contains only Horn (resp. pure Horn) terms. Finally, a Boolean function is called *Horn* (or *pure Horn*) iff it can be represented by at least one Horn (resp. pure Horn) DNF. To further simplify notation, let us write in the sequel an elementary pure Horn term $\bigwedge_{x \in S} x \wedge \overline{y}$ simply as $S\overline{y}$.

The problem which concerns us in this paper stems from the fact that DNF representations of Boolean functions are not unique. To find a DNF representation having some specific properties (e.g. the minimum number of terms, etc.) of a given Boolean function is usually a hard problem, not well solved even in the special cases of Horn or pure Horn functions. The problem of finding a minimum Horn DNF for a given Horn function can be stated as follows:

HORN MINIMIZATION (HM)

Instance: A Horn DNF $\mathcal{F}$ and a positive integer q.
Question: Is there a DNF equivalent to $\mathcal{F}$ which contains at most q terms?

The main result of this paper proves that the above problem is NP-complete.

Theorem 1 *The Horn Minimization problem is NP-complete.*

First, after introducing the necessary terminology in the next section, we shall prove that HM belongs to NP. In the rest of the paper we then show that the known NP-complete problem of finding a Hamiltonian path in a cubic graph (see e.g. [3, 4]) reduces to a special case of HM (in polynomial time), thus establishing the NP-completeness of HM.

The reduction will use the following correspondence between graphs and Horn functions. Let $G = (V, E)$ be an arbitrary (undirected) graph on n vertices with m edges, and let us identify each vertex of G with a Boolean variable x_i, i.e. $V = \{x_1, x_2, ..., x_n\}$. Throughout this paper we shall assume that G is simple, i.e. it has no loops or parallel edges. Let us associate with the graph G a pure Horn function f_G defined by the following DNF expression

$$\mathcal{F}_G = (\bigvee_{x_i \in V} v\overline{x}_i) \vee (\bigvee_{(x_i, x_j) \in E} x_i x_j \overline{v}), \tag{2}$$

1 Introduction.

Given a Boolean function, the problem of finding its representation of a minimal size (either with respect to the number of literals or to the number of terms) is NP-hard. One way to prove it is to realize that satisfiability testing (a known NP-complete problem, see e.g. [3]) is a special case of this minimization problem. However, there are important subclasses of Boolean functions for which the complexity of minimization is still not known. The subclass studied in this paper is the class of pure Horn functions (sometimes also called definite Horn functions). Due to the fact that for Horn functions the satisfiability problem is solvable in linear time [2] (as opposed to the NP-completeness of satisfiability for general Boolean functions, see e.g. [3]), Horn functions constitute a well known and widely studied subclass of Boolean functions.

The problem of finding a minimal representation of a given pure Horn function arises and seems to be an important problem in several application areas. One such instance is the design of propositional expert systems ("knowledge bases") [6, 7]. Another area in which the above problem was studied is the area of relational databases, namely the question how to utilize functional dependencies among attributes (see e.g. [1]).

The main contribution of this paper is to show that the problem of minimizing the number of terms in disjunctive normal form (DNF) representations of Horn functions is NP-complete. This results is obtained by showing that the minimization for an even more specialized subclass, the class of cubic pure Horn formulae, is already NP-complete.

This result means that in the case of "knowledge base optimization", finding a minimum representation of a given set of rules (i.e. finding as small as possible set of rules equivalent to the given one) is an intractable task even in the special case when the given rules are all proper implications, i.e. when the rule base corresponds to a pure Horn function. For relational databases the presented result shows the intractability of finding optimal decomposition into "smaller" data sets (relations) defined on subsets of attributes connected by functional dependencies.

The intractability of the similar task of minimizing the number of literal occurences in disjunctive normal form representations of general Horn functions has recently been established in [7].

The paper is organized as follows: In the next section we first introduce the basic Boolean terminology and then we define a special class of pure Horn functions that correspond to undirected graphs ("graph functions"). Also in Section 2 the main results of the paper are stated. Basic properties of the "graph functions" are shown in Section 3. Section 4 contains several technical lemmas characterizing minimum DNFs of the "graph functions". The proof of the main result is presented in Section 5.

2 Basic Definitions and Main Results.

A *Boolean function* f of n variables $x_1, \ldots, x_n$ will be considered as a mapping $\{0,1\}^n \to \{0,1\}$. The variables $x_1, \ldots, x_n$ and their negations $\overline{x}_1, \ldots, \overline{x}_n$ are called *positive* and *negative*

ABSTRACT

Boolean functions given in disjunctive normal form representation are considered in this paper. For a given function such a representation is typically not unique and different representations may consist of different number of terms. The problem of minimizing a Boolean formula ammounts to finding its equivalent representation which has the minimum possible number of terms. This problem is a well-known hard problem, where the difficulty stems largely from the fact that the satisfiability problem for general Boolean formulae in disjunctive normal forms is a NP-complete problem. There are however, special classes of Boolean formulae for which satisfiability is known to be polynomial, like Horn formulae arising in artificial intelligence or in relational database applications. The complexity of the minimization problem of Horn formulae was not known so far.

In this paper we show that, given a Horn formula, finding a disjunctive normal form equivalent to it and having the minimum possible number of terms is NP-complete.

DIMACS Technical Report 94-7
March 1994

On the Complexity of Horn Minimization

by

Endre Boros [1,2]
RUTCOR
Rutgers University
P.O.Box 5062
New Brunswick, New Jersey 08903
email : boros@rutcor.rutgers.edu

Ondřej Čepek [3,4]
RUTCOR
Rutgers University
P.O.Box 5062
New Brunswick, New Jersey 08903
email : cepek@rutcor.rutgers.edu

[1]Permanent Member
[2]This work was partially supported by the Office of Naval Research (grants N0001492F1375 and N0001492F4083) and by the Air Force Office of Scientific Research (grant F49620-93-1-0041).
[3]Affiliated Graduate Student Member
[4]This work was partially supported by the Office of Naval Research (grants N0001492F1375 and N0001492F4083) and by the Air Force Office of Scientific Research (grant F49620-93-1-0041).

DIMACS is a cooperative project of Rutgers University, Princeton University, AT&T Bell Laboratories and Bellcore.

DIMACS is an NSF Science and Technology Center, funded under contract STC–91–19999; and also receives support from the New Jersey Commission on Science and Technology.