

Proofs of the CP 2005 Paper “Towards an Optimal CNF Encoding of Boolean Cardinality Constraints”

Carsten Sinz

WSI for Computer Science
University of Tübingen, Germany

Theorem 1. $LT_{SEQ}^{n,k}$ is a clausal encoding of $\leq k(x_1, \dots, x_n)$ requiring $\mathcal{O}(n \cdot k)$ clauses and $\mathcal{O}(n \cdot k)$ auxiliary variables.

Proof. The proof that $LT_{SEQ}^{n,k}$ is a clausal encoding of $\leq k(x_1, \dots, x_n)$ is accomplished in two steps: First, we show that setting $k + 1$ variables (or more) of the x_i 's to true leads to a contradiction. Then we demonstrate that by setting exactly k of the variables x_i to true all auxiliary encoding variables $s_{i,j}$ get uniquely determined values. From the second part it follows that fixing k variables to true does not lead to a contradiction, and thus setting less than k variables does so, too.

For the first part, assume a variable assignment $\alpha : \{x_1, \dots, x_n\} \rightarrow \mathbb{B}$ with $k + 1$ variables set to 1, and let $\alpha^* : \text{Var}(LT_{SEQ}^{n,k}) \rightarrow \mathbb{B}$ be any extension of α . (Note, that we have $k < n$ by construction.) This α is obviously not a model of $\leq k(x_1, \dots, x_n)$, so we have to show that α^* is not a satisfying assignment of $LT_{SEQ}^{n,k}$. To do this, let the indices of the x_i 's that are set to 1 be sorted in ascending order, resulting in the vector (i_1, \dots, i_{k+1}) . We now show by induction that $\alpha^*(s_{i_j, j}) = 1$ holds for all $1 \leq j \leq k+1$ and therefore, in particular, $\alpha^*(s_{i_k, k}) = 1$: the case $j = 1$ is obvious by a clause of either type A or C.¹ The inductive step employs clauses of type D or F to first derive $\alpha^*(s_{i_{j+1}-1, j}) = 1$ from $\alpha^*(s_{i_j, j}) = 1$. Then by a clause of type E we can show that $\alpha^*(s_{i_{j+1}, j+1}) = 1$. Having $\alpha^*(s_{i_k, k}) = 1$, we can infer $\alpha^*(s_{i_{k+1}-1, k}) = 1$ by clauses of type D or E, which leads to a contradiction with $\alpha^*(x_{i_{k+1}}) = 1$ by clauses G or H. Setting more than $k + 1$ variables to true generates even further restrictions and thus is contradictory, too.

Now, for the second part, assume a variable assignment $\alpha : \{x_1, \dots, x_n\} \rightarrow \mathbb{B}$ with k variables set to 1 (thus satisfying the cardinality constraint), and let $\alpha^* : \text{Var}(LT_{SEQ}^{n,k}) \rightarrow \mathbb{B}$ be an arbitrary extension of α as above. Once again, let the indices of the x_i 's that are set to 1 be sorted in ascending order, resulting in the vector (i_1, \dots, i_k) . With a similar inductive argument as above we can show that for all j with $1 \leq j \leq k$ we have $\alpha^*(s_{p, j}) = 1$ if $p \geq i_j$ (by clauses A, C, E and G) and $\alpha^*(s_{p, j}) = 0$ if $p < i_j$ (by clauses B, D, F and H). Thus we know that α^* is a satisfying assignment of $LT_{SEQ}^{n,k}$. As the encoding is monotone in the x_i 's, we also obtain satisfying assignments by using this α^* if less than k variables are set to true.

The upper bounds on the number of clauses and variables can be read off directly from the encoding given on top of page 7.

¹ We classify different clauses of the encoding $LT_{SEQ}^{n,k}$ by letters A to H depending on which line they appear in the compilation on top of page 7 (A means first line, H last).

Theorem 2. $\text{LT}_{\text{PAR}}^{n,k}$ is a clausal encoding of $\leq^k(x_1, \dots, x_n)$ requiring at most $7n - 3\lfloor \log n \rfloor - 7$ clauses and $2n - 2$ auxiliary variables.

Proof. The prove that $\text{LT}_{\text{PAR}}^{n,k}$ is a clausal encoding of $\leq^k(x_1, \dots, x_n)$ is rather technical and not very enlightening. It relies on the correctness of the constituents of the whole circuit, i.e. the parallel counter and the comparator, which we assume the reader can verify himself.

The upper bounds on the number of clauses and variables were justified in the paper on pages 10 and 11.