

Efficient Construction of the Dual Span Program

Serge Fehr

May 12, 1999

Abstract

We consider monotone span programs as a tool for representing, we will say *computing*, general access structures. It is known that if an access structure \mathcal{A} is computed by a monotone span program \mathcal{M} , then the dual access structure \mathcal{A}^* is computed by a monotone span program \mathcal{M}^* of the same size. We will strengthen this result by proving that such an \mathcal{M}^* not only exists, but can be efficiently computed from \mathcal{M} .

1 Introduction

Monotone span programs, introduced by Karchmer and Wigderson in [KW93], are a model of computation, based on linear algebra, for computing monotone functions. Since there is a natural one-to-one correspondence between monotone functions $\{0, 1\}^n \rightarrow \{0, 1\}$ and access structures over the set $\mathcal{P} = \{1, \dots, n\}$, every access structure \mathcal{A} can be represented, we will say *computed*, by a monotone span program \mathcal{M} .

Every access structure \mathcal{A} has a natural dual access structure \mathcal{A}^* . This concept was first defined in [SJM91] and found various occurrences like e.g. in partial knowledge proofs [CDS94] or general-adversary multi-party computation [CDM99].

The following question naturally arises. Given a monotone span program \mathcal{M} of reasonable size computing an access structure \mathcal{A} , does there exist a monotone span program \mathcal{M}^* of reasonable size computing the dual access structure \mathcal{A}^* , and, if yes, can it be efficiently computed? The first part of the question has been answered in the affirmative in [Gál95], we will show in the following that also the second part can be answered by yes.

2 Definitions and Basic Properties

Let n be some positive integer and Γ , a set of subsets of $\mathcal{P} = \{1, \dots, n\}$.

Definition 1 Γ , is called an access structure over \mathcal{P} , if it is closed under taking supersets, i.e. if $A \in \Gamma$, $B \supset A \Rightarrow B \in \Gamma$.

The set $\Gamma^* = \{A \mid A^c \notin \Gamma\}$ is called the dual access structure to Γ .¹

Let Γ , be an access structure over $\mathcal{P} = \{1, \dots, n\}$. Further, let K be some field, M a $(d \times e)$ -matrix over K , $\varphi : \{1, \dots, d\} \rightarrow \{1, \dots, n\}$ a (surjective) function and ε a vector in K^e .

Definition 2 The quadrupel $\mathcal{M} = (K, M, \varphi, \varepsilon)$ is called a monotone span program, MSP for short, with labeling φ and target vector ε . The j -th row of M is said to be labeled by k if $\varphi(j) = k$. The MSP \mathcal{M} is said to compute the access structure Γ , if

$$A \in \Gamma \iff \varepsilon \in \text{im} M_A^T$$

where the matrix M_A consists of the rows of M which are labeled by a number in A .²

If $\varepsilon \in \text{im} M_A^T$ holds for some $A \subseteq \mathcal{P}$, then we say that \mathcal{M} accepts A . The size of \mathcal{M} is d , the number of rows of M .

The claims of the following proposition are known and/or easy to verify. We therefore omit the proof.

Proposition 2.1 Let $\mathcal{M} = (K, M, \varphi, \varepsilon)$ be a MSP computing an access structure Γ . Then the following holds.

1. It is easy to transform \mathcal{M} into a MSP, computing the same access structure Γ , of equal size and with target vector $(1, 0, \dots, 0)$.
2. For any $A \subseteq \mathcal{P}$, $\varepsilon \notin \text{im} M_A^T \Leftrightarrow \exists \mathbf{k} : M_{A^c} \mathbf{k} = \mathbf{0}, \langle \mathbf{k}, \varepsilon \rangle = 1$.
3. Deleting a column of M (and the corresponding entry of ε), which can be expressed as a linear combination of the other columns, does not change the access structure computed by the MSP. Therefore, we can always assume that $e \leq d$.

¹It is easy to see that Γ^* indeed is an access structure.

²Also for a vector $\mathbf{v} = (v_1, \dots, v_d)$ we let \mathbf{v}_A be the vector consisting of the entries v_j with $\varphi(j) \in A$.

3 Existence

As already mentioned, the following result is proven in [Gál95].

Theorem 1 *Let $\mathcal{M} = (K, M, \varphi, \varepsilon)$ be a MSP computing some access structure \mathcal{A} . Then there exists a MSP $\mathcal{M}^* = (K, M^*, \varphi, \varepsilon^*)$ of the same size computing the dual access structure \mathcal{A}^* .*

Even though the proof given in [Gál95] is constructive, the construction is not efficient.

4 Efficient Construction

We now state and prove the main result of this report.

Theorem 2 *Let $\mathcal{M} = (K, M, \varphi, \varepsilon)$ be a MSP computing some access structure \mathcal{A} . Then a MSP $\mathcal{M}^* = (K, M^*, \varphi, \varepsilon^*)$ of the same size, computing the dual access structure \mathcal{A}^* , can be efficiently computed. Furthermore, M and M^* satisfy $M^T M^* = \varepsilon \varepsilon^{*T}$.*

Proof: Let d and e be the number of rows and columns of the matrix M (whose columns are wlog linear independent) and assume that the target vector ε is $\varepsilon = (1, 0, \dots, 0) \in K^e$. Let \mathbf{v}_0 be a solution of the linear equation system $M^T \mathbf{x} = \varepsilon$ and $\mathbf{w}_1, \dots, \mathbf{w}_{e-d}$ a basis for $\ker(M^T)$. Set $M^* = [\mathbf{v}_0, \mathbf{w}_1, \dots, \mathbf{w}_{e-d}]$ and $\varepsilon^* = (1, 0, \dots, 0) \in K^{e-d+1}$. Note that M^* is a $d \times (d \Leftrightarrow e + 1)$ -matrix which fulfills $M^T M^* = E$ where E 's first column equals ε and all other entries are zero, hence $E = \varepsilon \varepsilon^{*T}$. Further, every solution of $M^T \mathbf{x} = \varepsilon$ is a linear combination of the columns of M^* in which the first column, \mathbf{v}_0 , occurs exactly once.

We will show now that the MSP $\mathcal{M}^* = (K, M^*, \varphi, \varepsilon^*)$ computes \mathcal{A}^* . Consider a set $A \in \mathcal{A}^*$. So there exists a vector $\boldsymbol{\lambda}$ with $\boldsymbol{\lambda}_{A^c} = \mathbf{0}$ and $M^T \boldsymbol{\lambda} = \varepsilon$. Therefore, $\boldsymbol{\lambda}$ must be of the form $\boldsymbol{\lambda} = M^* \mathbf{k}$ with the first entry of \mathbf{k} being one. But since $M_{A^c}^* \mathbf{k} = \boldsymbol{\lambda}_{A^c} = \mathbf{0}$ and $\langle \mathbf{k}, \varepsilon^* \rangle = 1$, A^c is not accepted by \mathcal{M}^* .

Consider now a set A such that A^c is not accepted by \mathcal{M}^* . This means that ε^* is not in the span of the rows of $M_{A^c}^*$ or, equivalent, there exists a vector \mathbf{k} with $M_{A^c}^* \mathbf{k} = \mathbf{0}$ and $\langle \mathbf{k}, \varepsilon^* \rangle = 1$. If we set $\mathbf{a} = M^* \mathbf{k}$, then $\mathbf{a}_{A^c} = \mathbf{0}$ and hence $M_A^T \mathbf{a}_A = M^T \mathbf{a} = M^T M^* \mathbf{k} = E \mathbf{k} = \varepsilon$. Therefore $A \in \mathcal{A}$.

Hence, $A \in \mathcal{A}$, if and only if A^c is not accepted by \mathcal{M}^* . \square

Acknowledgments

We would like to thank Ronald Cramer for many interesting and helpful discussions concerning this and other topics and for his support.

References

- [CDM99] Ronald Cramer, Ivan Damgård, and Ueli Maurer. General Secure Multi-Party Computation from any Linear Secret-Sharing Scheme. In preparation, 1999.
- [CDS94] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In Yvo G. Desmedt, editor, *Advances in Cryptology—CRYPTO '94*, volume 839 of *Lecture Notes in Computer Science*, pages 174–187. Springer-Verlag, 21–25 August 1994.
- [Gál95] A. Gál. *Combinatorial Methods in Boolean Function Complexity*. PhD thesis, University of Chicago, 1995.
- [KW93] Maurizio Karchmer and Avi Wigderson. On span programs. In *8th Annual Conference on Structure in Complexity Theory (SCTC '93)*, pages 102–111, San Diego, CA, USA, May 1993. IEEE Computer Society Press.
- [SJM91] G.J. Simmons, W.A. Jackson, and K. Martin. The geometry of shared secret schemes. *Bulletin of the Institute of Combinatorics and its Applications*, 1:71–88, 1991.