Lower Bound Estimation for Low Power High-Level Synthesis†

Lars Kruse, Eike Schmidt, Gerd Jochens, Ansgar Stammermann, Wolfgang Nebel
OFFIS Research Institute, 26121 Oldenburg, Germany

Abstract

This paper addresses the problem of estimating lower bounds on the power consumption in scheduled data flow graphs with a fixed number of allocated resources prior to binding. The estimated bound takes into account the effects of resource sharing. It is shown that by introducing Lagrangian multipliers and relaxing the low power binding problem to the Assignment Problem, which can be solved in $O(n^3)$*, a tight and fast computable bound is achievable. Experimental results show the good quality of the bound. In most cases, deviations smaller than 5% from the optimal binding were observed. The proposed technique can for example be applied in branch and bound high-level synthesis algorithms for efficient pruning of the design space. The estimated lower bound can also be used as a starting point for low power binding heuristics to find optimal or near optimal binding solutions.*

1. Introduction

For most problems in high-level synthesis (HLS) no polynomial time algorithms are known [1]. In order to find optimal or near optimal solutions for this class of problems strategies like branch and bound are applied. A branch and bound algorithm traces a *decision tree* whose leaves represent all possible solutions. Design decisions are made at each internal node while the leaves of the subtree rooted at an internal node are the solutions due to that decision. Given a best solution found during execution of the branch and bound algorithm, a subtree can be pruned if a lower bound estimate of the cost function of all solutions of the subtree is higher than the cost of the current best solution. Tight and fast computable lower bounds therefore improve the run time requirements of such algorithms.

†This work has been funded by the Commission of the European Community as part of the ESPRIT IV programme under contract no. 26796.

This paper addresses the problem of lower bound estimates for low power HLS and related applications. In particular, a lower bound estimation procedure for the power consumption of datapath resources, i.e. registers and functional units (FUs) like adders and multipliers, in scheduled data flow graphs (DFGs) with resource constraints for a given input data stream is given. In the assumed design flow the binding of operations and variables to functional units and registers respectively follows allocation and scheduling. This is a typical flow if, for example, resource constrained scheduling is performed. Conditional branches and loops within a DFG are not considered here. Different bindings produce most probably different datapath activities due to the varying data multiplexing schemes if resources are shared.

The lower bound estimation procedure is not restricted to a specific power cost function of the datapath resources. In this paper we apply two different power metrics: The average Hamming distance of consecutive input vectors (switching activity for short) and a characterization based RTL power model [5]. Most HLS for low power algorithms use the switching activity at the inputs of datapath resources or simple functions thereof as a cost function of the power consumption of the design [2,3,4]. The switching activity is a good indicator of the power requirements and often the only power indicating information available at the higher levels of abstraction as considered here. Accurate RTL power models can be applied in case the resource types (e.g. a CLA scheme for adders etc.) are fixed prior to the binding step [6,7].

The remainder of the paper is organized as follows: section 2 describes the relation of our approach to previous work. In section 3 the representation and calculation of the power cost information is introduced. The new lower bound estimation procedure is presented in section 4. Section 5 shows experimental results and conclusions are drawn in section 6.

2. Previous Work

Lower bound estimation (LBE) techniques are often applied to guide HLS. As examples, the authors of [8] present procedures to estimate lower bounds on the resource requirements from a given DFG with a performance goal. In [9] a technique is described that estimates a lower bound on the performance of schedules from a DFG with resource constraints. To the best of our knowledge, LBE techniques for low power at the higher levels of abstraction are first addressed in [10]. Lower and upper bound estimation procedures are given for scheduled DFGs without resource constraints. This paper extends the work of [10] by improving the bounds if the number of resources is constrained.

Some researchers have addressed HLS for low power problems that are closely related to our work. In [2] the problem of binding the *n* variables of a DFG to *m* registers under the constraint of minimum switching activity at the register inputs is formulated as a max-cost network flow problem. The problem can be solved in $O(mn^2)$. The drawback of this approach, however, is that inter-iteration switching activity cannot be considered. Inter-iteration activity is defined as the switching activity resulting from successive executions of the DFG. For instance, let:

- x^i, $x \in \{a, b, c, d\}$, be the value of the binary variable x at iteration i of the DFG,
- $x|y$ be the concatenation of the binary variables x and y,
- $hd(x, y)$ be the Hamming distance between the values of variables x and y.

Suppose that operations $+_1$ and $+_3$ of the DFG depicted in Fig. 1 are bound to the same adder. One part of the switching activity at the inputs of the adder is:

$$\sum_{i=1}^{T} hd(a^i|b^i, d^i|c^i) \text{, (intra-iteration activity),}$$

where T is the length of the input stream. The inter-iteration part is defined as:

$$\sum_{i=1}^{T-1} hd(d^i|c^i, a^{i+1}|b^{i+1}),$$

e.g. the switching from the values of iteration i to the new ones of iteration $i+1$.

The same authors investigate the problem of binding operations to a fixed number of resources in a functionally pipelined DFG taking inter-iteration effects into account [3]. Due to the inter-iteration constraint the problem can only be transformed to a max-cost multi-commodity network flow problem which is in general not solvable in polynomial time. An integer linear program (ILP) for the problem of binding *n* operations/variables to *m* functional units/registers is formulated in [6]. The inter-iteration effects are considered but no polynomial time algorithm for solving the ILP is given.

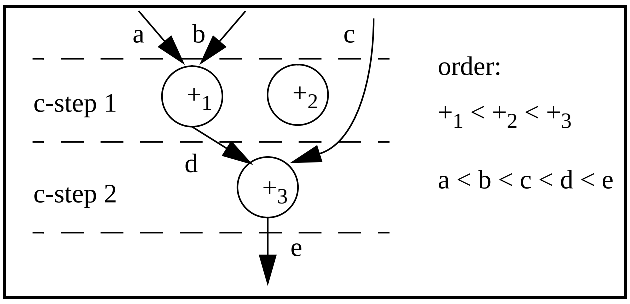

Figure 1. DFG with operation and variable order.

3. Power Cost Function

The power cost function is stored in matrices in order to take all input data correlations into account. Its calculation follows the approach described in [10] that originates from the work presented in [2,3,6]. A square *power cost matrix* (*PCM*) for the variables and for each operation type present in the DFG (e.g. addition, subtraction, multiplication, etc.) is defined. We first describe the formal definition of the *PCM* if switching activity is used as a metric.

In the sequel, we only deal with operations. Variables can be handled in the same way. We index the *n* operations of one operation type according to the execution order of the given schedule. Operations scheduled into the same c-step are called incompatible and are indexed arbitrary.

Switching activity information about operation op_i is stored in column and row i of the *PCM* of the corresponding operation type. An entry $PCM(i, j)$, $i, j \in \{1, \ldots, n\}$ is set to infinity ($+\infty$) if operations op_i and op_j are executed in the same c-step and therefore cannot share a resource. Otherwise if $i < j$ (i.e. op_i is executed before op_j), the entry stores the average Hamming distance between the input vectors of operations op_i and op_j from the same iteration (intra-iteration activity). If $i > j$, $PCM(i, j)$ is set to the average Hamming distance between the inputs of op_i from iteration t and the inputs of op_j from iteration $t+1$ (inter-iteration activity). The elements on the main diagonal $PCM(i, i)$ store the activity at the inputs of operation op_i, e.g. the activity at the inputs of a FU if only op_i is bound to it. Formally:

$$PCM(i, j) = \begin{cases} \infty \ , op_i, op_j \text{ are not compatible} \\ \frac{1}{T}\sum_{t=1}^{T} hd(op_i(t), op_j(t)) \ , i < j \\ \frac{1}{T}\sum_{t=1}^{T} hd(op_i(t), op_j(t+1)) \ , i \geq j \end{cases} \quad (3.1)$$

with:

- $op_i(t)$ the concatenation of the input vectors of operation op_i in iteration *t* of the DFG, and
- *T* the total number of vectors in the input data stream, e.g. the number of iterations of the DFG.

For example, the average switching activity per DFG iteration at the inputs of a resource with operations op_1, op_2, op_3 mapped onto it in that order can now be computed by $SAM(1, 2) + SAM(2, 3) + SAM(3, 1)$ [6]. The Hamming distances can be computed by simulating the DFG with the input stream or by using statistical techniques as for example proposed in [2,3]. Note that two dimensions suffice to store all necessary information.

If we use a more accurate power model as a cost metric, the terms $1/T \sum_{t=1}^{T} hd(op_i(t), op_j(t))$ in (3.1) above are replace by the estimated power consumption for the corresponding resource type and the multiplexed input data streams of the specified operations $op_i(t)$ and $op_j(t)$.

4. Binding for Low Power with Resource Constraints

4.1. Problem formulation

For a given operation type we define the *low power binding problem with resource constraints* as follows:

Given a power cost matrix $(PCM(i, j))_{i, j \in \{1, \ldots, n\}}$ *of n scheduled operations* $op_1, \ldots, op_n$. *Which is the minimum power consumption of the resources if these n operations are bound to m resources.*

An equivalent problem can be stated for binding n variables to m registers. There are

$$\frac{1}{m!} \cdot \left(m^n - \left((m-1)^n \binom{m}{m-1} - \left(\ldots -1^n \binom{m}{1}\right)\ldots\right)\right)$$

possibilities to map n operations onto m resources if all n operations are compatible. The proof is omitted due to space limits. For example, there are more than $4 \cdot 10^{10}$ combinations to map 20 operations onto 4 resources.

The low power binding problem with resource constraints can be expressed as a graph problem by defining an arc labeled directed graph $G(V, A)$ with $V = \{op_1, \ldots, op_n\}$ the set of nodes (one node for each operation) and $A = V \times V$ the set of arcs. Each arc $(op_i, op_j) \in A$ is labeled with a weight $w_{ij} = PCM(i, j)$. The optimization problem is then to cover all nodes with exactly m (node disjoint) cycles with minimum total cost under the constraint that each cycle contains exactly one *backward arc*, e.g. an arc (op_i, op_j) with $i \geq j$. The total cost is the sum of the arc weights of all cycles. Each cycle of a solution to this problem represents one resource while the nodes of a cycle are the operations bound to it. A possible solution of the optimization problem with four operations and two resources is depicted in Fig. 2. $(+_4, +_1)$ and $(+_3, +_3)$ are the two backward arcs.

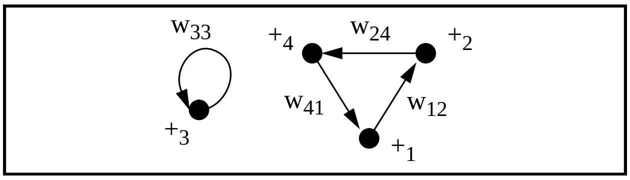

Figure 2. Possible solution of a low power binding problem with resource constraints by covering operation nodes with cycles.

The constraint that each cycle $(op_{k_1}, \ldots, op_{k_p}, op_{k_1})$, $k_1 < \ldots < k_p$ must have exactly one backward arc reflects the precedence constraints of the operations within the schedule of the DFG. Inter-iteration switching activity ($PCM(k_p, k_1)$ in this case) can only occur after all operations $op_{k_1}, \ldots, op_{k_p}$ are executed in one iteration of the DFG on one resource. Loops (op_i, op_i) represent resources with exactly one operation op_i bound to it.

Note that as a by-product, minimizing power based on reducing switching input bits of datapath resources also reduces power in the multiplexers. This is because output activity of multiplexers (which is the input activity of the resources) is a good power model for multiplexers [16].

4.2. Bounding the solution space

We first repeat Theorem 4.1 without proof from [10] which defines a lower bound on the power consumption of the low power allocation and binding problem, i.e. the binding problem without resource constraints:

Theorem 4.1 *A solution of the integer linear program*

$$z = \min \sum_{i, j=1}^{n} w_{ij} x_{ij}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad i = 1, \ldots, n \qquad (4.1.A)$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad j = 1, \ldots, n \qquad (4.1.B)$$

$$x_{ij} \geq 0 \text{ integer variables}$$

provides a lower bound of the low power binding problem (without resource constraints) with power cost matrix $PCM(i, j) = w_{ij}$.

In this ILP, the (binary) variable x_{ij} is associated with arc (op_i, op_j). The solutions to the ILP describe node disjoint cycles covering all nodes. x_{ij} equals 1 in a solution if and only if the corresponding arc belongs to a cycle, otherwise the variable is zero. The n constraints (4.1.A) guarantee that there is exactly one arc incident from node op_i while constraints (4.1.B) insure that exactly one arc leaves node op_j. However, it is not guaranteed that the precedence constraints are fulfilled. Hence a solution to the ILP delivers only a lower bound z on the switching activity and not necessarily the minimum.

The precedence constraints are excluded because otherwise the ILP would not be solvable in polynomial time [6]. As stated in [10] the ILP of Theorem 4.1 can be efficiently solved by the *Hungarian Method* in $O(n^3)$ because it describes the *Assignment Problem* [11].

The following ILP improves the lower bound of Theorem 4.1 with additional constraints on the number of resources:

Theorem 4.2 *A solution of the integer linear program*

$$z = \min\sum_{i,\,j=1}^{n} w_{ij}x_{ij}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad i = 1, \ldots, n \qquad (4.2.A)$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad j = 1, \ldots, n \qquad (4.2.B)$$

$$\sum_{i \ge j} x_{ij} = m \qquad (4.2.C)$$

$$x_{ij} \ge 0 \text{ integer variables}$$

provides a lower bound of the low power binding problem with m *resources and power cost matrix* $PCM(i, j) = w_{ij}$.

Proof: Constraints 4.2.A and 4.2.B are identical to the constraints 4.1.A and 4.1.B respectively and guarantee that all nodes are covered by node-disjoint cycles. 4.2.C insures that exactly m backward arcs are included in a solution of the ILP. However, no constraints exist that forces each cycle to have exactly one backward arc which is a necessary condition of the low power binding problem with resource constraints. The ILP is therefore a relaxation of the original problem and a solution provides a lower bound q.e.d.

As an example, assume that 6 operations must be bound to 3 resources. A possible solution of Theorem 4.2 which also represents a legal binding would be 3 cycles: (op_1, op_5, op_1), (op_2, op_3, op_2) and (op_4, op_6, op_4). op_1 and op_5 are bound to the first resource, and so on. Note that the precedence constraints are fulfilled in each cycle. Another solution of Theorem 4.2 which is not a legal binding could be: $(op_1, op_5, op_4, op_6, op_1)$ and (op_2, op_3, op_2). This solution consists of only 2 cycles wit a total of 3 backward arcs. However, the first cycle has 2 backward arcs (op_5, op_4) and (op_6, op_1). The first cycle therefore violates the precedence constraints.

Instead of solving the ILP of Theorem 4.2 directly, a polynomial time bounded approach is proposed which approximates the ILP based on Lagrangian Relaxation [12], i.e. the original binding problem is relaxed two times. Lagrangian Relaxation explores the fact that for a given ILP (P) of the general form

$$z = \min w^T x$$
$$\text{subject to } Ax = b, Bx = d \qquad (P)$$
$$x \ge 0, x \text{ integer}$$

with w and x vectors of dimension d_1, b being a vector of dimension d_2, and d being a vector of dimension d_3. The set of constraints is partitioned into two sub-sets. A and B are matrices with conformable dimensions describing these two sub-sets.

In Lagrangian Relaxation, a vector y of multipliers is defined and one of the two constraints sub-sets, say $Bx = d$, is moved into the objective function by adding the term $(Bx - d)^T y$:

$$(y) = \min(w^T x + (Bx - d)^T y) \qquad (Q)$$
$$\text{s.t. } Ax = b, x \ge 0, x \text{ integer}$$

y is called vector of *Lagrangian multipliers*. The motivation for doing this is that the modified ILP (Q) is easier to solve.

A solution of (Q) provides a lower bound for (P) for all values of y. This property follows because all feasible solutions of (P) are also feasible for (Q) with the same objective function value. In these cases, the term $(Bx - d)^T y$ is zero because $Bx = d$ is true. The solution of (Q) can become lower than the solution of (P) if $(Bx - d)^T y$ is negative. The best lower bound is found by maximizing $L(y)$, i.e. solving:

$$v = \max_y L(y)$$

The function $L(y)$ is not differentiable but piecewise linear and concave. Therefore, a gradient cannot be computed. However, the maximization can be performed with *subgradient maximization* [14]. k is *subgradient* of L at x iff $k^T(y - x) \ge L(y) - L(x)$ for all y. It can be shown that if $\hat{x}$ minimizes $L(y)$ for some y than $B\hat{x} - d$ is a subgradient of $L(y)$. The *subgradient method* iteratively solves L at points:

$$y_{p+1} = y_p + t_p q_p$$

where q_p is a subgradient of L at y_p and t_p is a suitable step width in that direction ($p \ge 0$, $y_0 = 0$). The Lagrangian multipliers are adapted until the solutions converge to the maximum. Convergence can be guaranteed if the step width t_p is reduced according to $\lim_{p \to \infty} t_p = 0$ and $\sum_{p=0}^{\infty} t_p = \infty$ [14].

Applying the Lagrangian method to the ILP of Theorem 4.2 by relaxing 4.2.C delivers:

$$L(y) = \min(\sum_{i,\,j} w_{ij}x_{ij} - (\sum_{i \ge j} x_{ij} - m)y)$$
$$= \min(\sum_{i,\,j} w_{ij}x_{ij} - \sum_{i \ge j} yx_{ij}) - my$$
$$= \min\sum_{i,\,j} w_{ij}'x_{ij} - my$$

$$with \qquad w_{ij}' = \begin{cases} w_{ij} & \text{if } i < j \\ w_{ij} - y & \text{if } i \geq j \end{cases}$$

subject to 4.2.A and 4.2.B. $L(y)$ can again be solved by the *Hungarian Method* in $O(n^3)$ with the objective function $\min \sum_{i,j} w_{ij}' x_{ij}$ and subtracting the constant my afterwards. Note the similarity to the ILP of Theorem 4.1. The subgradient at iteration p is $q_p = \sum_{i \geq j} \hat{x}_{ij} - m$ if $\hat{x}_{ij}$ is the minimizer of $L(y_p)$, i.e. the number of backward arcs of the solution minus the number of resources.

Fulfilling the conditions $\lim_{p \to \infty} t_p = 0$ and $\sum_{p=0}^{\infty} t_p = \infty$ might result in very slow convergence rates. An approximation is used throughout the experiments instead by dropping the condition $\sum_{p=0}^{\infty} t_p = \infty$. The sequence

$$t_p = \alpha \cdot \frac{U - L(y_p)}{\|q_p\|}$$

is a widely used formula [12] where U is a simple upper bound of $L(y_p)$ and $0 < \alpha < 2$ is a constant which is periodically decreased. We use a simple patching heuristic to estimate U. See [13] for details.

Since we are only interested in an approximation of the solution from below, we perform the iterations only a fixed maximum number of times. This guarantees that the method has polynomial complexity.

5. Experimental Results

The experiments were performed on 3 benchmarks investigating the binding of additions and multiplications: a one dimensional FDCT [15] as a part of a 2D-FDCT with correlated input data of 3 different images I1, I2, and I3 (13 additions and 16 multiplications), a low pass image filter (LPF) applied to the same 3 input images (8 additions, multiplications are not needed), and the Elliptic Wave Filter (EWF) HLS benchmark as specified in [16] with modified coefficient set and a speech signal as input (10 additions and 12 multiplications).

Table 1 shows the deviations of the computed lower bounds from the best binding using the switching activity cost function for a sequential schedule of the FDCT benchmark, i.e. only one addition/multiplication per c-step, depending on the number of allocated resources. The trivial cases of $m = 1$ and $m = n$ are not considered (the first is trivial due to the precedence constraints of the operations). The best bindings were found by exhaustive search. The deviations are clearly below 2% for the addition operations while for the constant multiplications, the deviation increases in a few cases up to 13.4%. The largest deviations occur if the number of available resources equals about half the number of operations. In these cases, the solution space of the binding problem is largest. The ILP of Theorem 4.2 is approximated without significant error by the proposed Lagrangian Relaxation in all experiments. The deviations therefore result from the relaxation of the precedence constraints of the binding problem by the ILP of Theorem 4.2.

Table 1. Deviation of lower bound from best binding in % for sequential schedule of additions and multiplications (FDCT) with switching activity PCM.

# res.	Additions			Multiplications		
	I1	I2	I3	I1	I2	I3
2	0	0	0.2	0	0	0
3	0	0	1.4	0	0	0
4	0.3	0	0	1.9	2.0	2.3
5	0	1.0	0	4.6	3.3	5.1
6	0	0.6	0	6.5	4.4	6.8
7	0.1	1.0	0.3	9.2	5.2	9.2
8	1.3	1.3	0.3	10.1	6.5	12.9
9	0	0	0	9.5	6.2	13.4
10	0	0	0.1	8.1	4.9	12.1
11	0.3	0	0.5	6.7	3.3	9.3
12	0.4	0	0.7	5.9	2.7	7.7
13	-	-	-	4.5	2.0	5.6
14	-	-	-	3.1	1.3	3.8
15	-	-	-	1.5	0.6	2.0

Table 2 presents the results of the same analysis for the low pass filter and the EWF benchmark. The deviations are below 5% in these experiments. In Table 3 the deviations averaged over all resource constraints between the lower bounds and the best bindings are shown if a more accurate power model is applied as a cost function [5]. For comparison, the average deviations for the switching activity cost function are also reported. The results show that the proposed technique delivers tight bounds also for this cost metric.

In order to measure the cpu time requirements, we generated larger *PCMs* with random contents. For $n = 60, m = 5$ and $n = 200, m = 20$ with 10 Lagrange iterations each, the proposed technique required 0.2 s and 8.1 s, respectively (Ultra-Sparc 10, 300 MHz). Solving the ILP of Theorem 4.2 directly with a fast ILP solver [17] took 2.2 s and 101.5 s, respectively. The pro-

Table 2. Deviation in % for sequential schedule of operations of LPF and EWF benchmarks with switching activity PCM.

# res.	LPF I1	LPF I2	LPF I3	EWF ADD	EWF MUL
2	0	0	1.0	0.4	0
3	0	0	3.1	0.7	0
4	0	0	0	0.7	2.1
5	0	0	0	0.7	2.6
6	0	0	0	0.6	4.0
7	0.1	0.1	0	0.7	4.1
8	-	-	-	0.4	3.3
9	-	-	-	0.4	0.6
10	-	-	-	-	0
11	-	-	-	-	0

Table 3. Deviation of lower bound from best binding in %, averaged over all possible resource constraints.

		switching activity	power model
LPF	add	0.2	0.4
FDCT	add	0.3	0.4
	mul	4.9	0.6
EWF	add	0.6	0.5
	mul	1.7	0.3

posed technique is for these cases more than 10 times faster than solving the ILP directly. For comparison: exhaustive search was only feasible upto $n = 16$. It required more than 8 hours to find the optimal binding for $n = 16$, $m = 8$.

6. Conclusion

This paper presented a fast estimation technique that provides tight lower bounds on the power consumption of datapath resources for a given schedule with resource constraints. The low power binding problem under resource constraints was formulated and relaxed to the Assignment Problem with Lagrangian multipliers. A few number of iterations suffice to achieve estimates that are very close to the best possible solution.

The proposed technique can be applied in HLS to reduce the power consumption in datapath components. Typical applications are branch and bound based algorithms or binding heuristics which transform a lower bound solution into a constraint satisfying binding with an additional optimization step.

Future work will be devoted to extend the presented approach to handle multi-functional units and conditional execution of operations in branches and loops.

7. References

[1] G. De Micheli, ”Synthesis and Optimization of Digital Circuits“, McGraw-Hill, 1994

[2] J.-M. Chang, M. Pedram, ”Low Power Register Allocation and Binding“, Proc. of the DAC 1995

[3] J.-M. Chang, M. Pedram, ”Module Assignment for Low Power“, Proc. of the Euro-DAC 1996

[4] E. Musoll, J. Cortadella, ”Scheduling and Binding for Low Power“, Proc. of the ISSS, 1995

[5] G. Jochens, L. Kruse, E. Schmidt, W. Nebel, ”A New Parameterizable Power Macro-Model for Datapath Components“, Proc. of DATE 1999

[6] A. Raghunathan, N.K. Jha, ”An ILP formulation for low power based on minimizing switched capacitance during datapath allocation“, Proc. of the Int. Symposium on Circuits & Systems, 1995

[7] P.E. Landman, J.M. Rabaey, ”Activity-Sensitive Architectural Power Analysis“, IEEE Transactions on CAD of Integrated Circuits and Systems, 15(6), June 1996

[8] S.Y. Ohm, F.J. Kurdahi, N.D. Dutt, ”A Unified Lower Bound Estimation Technique for High-Level Synthesis“, IEEE Transactions on CAD, 16(5), May 1997

[9] M. Langevin, E. Cerny, ”A Recursive Technique for Computing Lower-Bound performance of Schedules“, ACM Transactions on Design Automation of Electronic Systems, 1(4), October 1996

[10] L. Kruse, E. Schmidt, G. Jochens, W. Nebel, ”Lower and Upper Bounds on the Switching Activity in Scheduled Data Flow Graphs“, Proc. of ISLPED 1999

[11] C.H. Papadimitriou, K. Steiglitz, ”Combinatorial Optimization, Algorithms and Complexity“, Prentice Hall, Inc., 1982

[12] M.L. Fisher, ”An Applications Oriented Guide to Lagrangian Relaxation“, pp. 10-21, Interfaces 15(2), March-April 1985

[13] L. Kruse, E. Schmidt, G. Jochens, W. Nebel, ”Low Power Binding Heuristics“, Proc. of PATMOS, pp. 41 - 50, 1999

[14] B.T. Polyak, ”Subgradient Methods: A Survey of Soviet Research“ in C. Lemarechal, R. Mifflin (eds.) ”Nonsmooth Optimization“, pp. 5 - 29, Pergamon Press, Oxford, 1978

[15] K.R. Rao, P. Yip, ”Discrete Cosine Transform - Algorithms, Advantages, Applications“, pp. 378 - 380, Academic Press, Inc., 1990

[16] A. Raghunathan, N.K. Jha, S. Dey, ”High-Level Power Analysis and Optimization“, p. 25, Kluwer Academic Publishers, 1998

[17] M.R.C.M. Berkelaar, Unix manual of lp_solve, Eindhoven University of Technology, 1992