On the Correlation between Resource Minimization and Interconnect Complexities in High-Level Synthesis

Shantanu Dutt[1], Xiuyan Zhang[1] and Ouwen Shi[2]

[1]ECE Dept., University of Illinois at Chicago and [2]Cadence Design Systems

Abstract

As the technology node of VLSI designs advances to sub-10 nm, two interconnect-centric metrics of a circuit, the interconnect complexity (either number of interconnects or wirelength/WL) and congestion, become critically important across all design stages alongside conventional resource or function-unit (FU)-centric metrics like area/number-of-FUs and leakage power. High Level synthesis (HLS), one of the earliest and most impactful design stages, rarely monitors interconnect metrics, which makes their recovery at later stages very difficult. HLS algorithms and tools typically perform FU-centric minimization via operation scheduling, module selection (S&MS) and binding. As a consequence, it mostly overlooks interconnect-based metrics. In this paper, we explore whether this can adversely affect interconnect metrics, and in general explore the correlation between FU-centric optimization in S&MS, and the resulting interconnect metrics co-optimized (along with FU metrics) in the later binding stage(s). For this purpose we develop a probabilistic analysis for post-scheduling binding to estimate interconnect metrics, and verify its accuracy by comparison to empirical results across different scheduling techniques that generate different degrees of FU optimization. Based on both empirical and analytical results we predict how interconnects metrics will pan out with different degrees of FU optimization. Finally, based on our analysis, we also provide suggestions to improve interconnect metrics for whatever FU optimization degree an available S&MS technique can achieve.

1. Introduction

Interconnect optimization is of make-or-break significance in VLSI designs in the sub-10nm regime. Since the density of transistors per unit area is increased dramatically, the complexity of interconnects per unit area, as well as overall, become key metrics that need to be made tractable in order to obtain feasible/routable designs. The earliest design stage of High Level Synthesis (HLS) is the first place to estimate the usage of resources and obtain interconnect information. Performing optimization in HLS is generally more effective in determining the quality of the final design, compared to the later stages such as routing and placement. Early HLS functions, like operation scheduling and module selection (including voltage assignment for different operations), are high-impact processes for optimizing functional-unit (FU)-centric metrics like number of FUs, power (mainly leakage, but also dynamic), and area. Once a scheduling and module selection (S&MS) solution is obtained, performing FU and register binding generates the final HLS design with information on needed interconnects and steering logic (muxes and demuxes) between FUs.

Due to considerations of runtime efficiency and algorithm design complexity, most existing HLS design flows/tools that synthesize complete HLS designs use a series of techniques. Each of these performs a different function and targets different sets of metrics. Techniques for S&MS focus on FU-centric metrics (numbers of FUs used, FU power, area, etc.), and techniques for FU/register binding focus on one or more of FU-centric and interconnect-centric metrics (number of interconnects and congestion among FUs, registers, and muxes/demuxes, their area, and dynamic power). There are some techniques like [1] that estimate interconnect metrics by incorporating floorplanning into the FU binding process. However, the complexity of the algorithm as well as the resulting quality improvement vis-à-vis not using floorplanning is of some concern. For an efficient flow, it is thus necessary to solve the complete HLS problem in a sequence of phases, as alluded to above. However, the more interconnect-aware later binding phases would prima facie seem to be constrained by a truncated problem space determined by the solution of the earlier FU-centric S&MS phase. The natural question that then arises is whether this truncation impedes interconnect optimization in the later HLS phases. This would be the case if FU-based minimization fundamentally is in conflict with interconnect optimization. On the other hand, this truncation would not be of any significant concern if the two optimizations were strongly correlated. The former case would call for new algorithms that combine S&MS and the later binding phases to co-optimize both FU and interconnect-based metrics; this would be a significant challenge, and would result in much higher time complexities than in the separated phases of current HLS flows. If the latter is true, then we can be assured that FU and interconnect based metrics are being both properly optimized in current HLS flows that afford significant time efficiencies due to the separated phases; the only improvements needed here could be better optimization algorithms for each phase without the need to combine them. We also explore whether there are any other aspects of the S&MS design, including a tipping point in the degree of FU minimization, that adversely or beneficially affect subsequent interconnect-aware minimization.

Thus the goal of the paper is to provide an overview to designers of how FU-based optimization of early stage HLS impacts interconnect complexity in later stages, and what could be done to avoid any detrimental effect (alternatively, to obtain beneficial effects) of the former on the latter. The main contributions of the paper are: **1)** A qualitative analysis of the correlation between FU minimization and interconnect complexity in HLS (*Sec. 3*). **2)** A probabilistic model of post-

S&MS binding, which includes as an input variable the degree of FU optimization, to determine more generally than obtainable from empirical data, the correlation between FU and interconnect optimizations in HLS (*Sec.4*). The theoretical foundation of this analysis is established, as is its accuracy for real designs via corroboration from empirical data. Such an analysis allows us to estimate interconnect metrics at different degrees of FU minimization to draw more comprehensive conclusions without needing to perform S&MS and binding for a large number of benchmarks. **3)** An empirical exploration of the correlation between FU minimization and interconnect complexity (*Sec. 5*). **4)** Based on the above analyses, we suggest: a) FU "white-spacing" for congestion reduction, and b) a balanced (pre-binding) operation allocation to FUs at any FU optimization degree (i.e., for any S&MS algorithm) in order to obtain better interconnect optimization in the binding stages.

2. An HLS Example and Evaluation Framework

S&MS provides a partial HLS solution for a given data flow graph (DFG). Since after S&MS, every operation (op) has been assigned to a particular time slot, we can obtain the datapath (interconnect structure) by performing FU and register bindings. Figure 1(b) illustrates a scheduling solution of the simple DFG in Fig. 1(a) for the MR-LCS problem (minimize resources/FUs given a latency constraint) with a latency constraint L of 5 clock cycles (cc's), where for simplicity, both adders and a multipliers have a delay of 1 cc. Also, for simplicity, no module selection is performed (i.e., there is only one design for each functional type). The corresponding FU binding solution is shown in Fig. 1(c). After performing scheduling and FU binding (S&B), we can easily determine the interconnects needed among different FUs and finalize the interconnect design by allocating/binding registers and muxes/demuxes. The detailed datapath design for the example S&B solution is shown in Fig. 1(d), where for each FU input, there is a dedicated register bank (this is also our assumed configuration in the register binding solutions in our experimental results). The datapath has a reasonable degree of interconnect sharing between different parent-child op pairs (e.g., between op pairs (op_1, op_5) and (op_3, op_6).

Note that, there are different techniques to bind FUs and allocate registers like [3], and thereby to also synthesize the mux/demuxes at each FU port. Since in this paper, we explore the correlation between the degree of FU-centric optimization of S&MS and the degree of subsequent interconnect-aware optimization achieved by FU and register binding, we vary the techniques for the former to realize varying degrees of FU-centric optimization, but by necessity (for an apples-to-apples comparison between these varying degrees of FU-centric optimization) keep the latter techniques the same. Given the S&MS solution, the technique we use for FU binding [3] is optimal for the number of FUs of each type, and within this solution space, it heuristically minimizes the numbers of interconnects needed (see Sec. 5-2). The register binding technique we use [3] is also optimal for the number of registers, given the FU binding solution. It also determines the mux/demux sizes, and thus, interconnect congestion at each FU port. In other words, we use well-known effective techniques for the latter interconnect-complexity determining phase of HLS, and thus the conclusions we obtain on the aforementioned correlation should be reliable and general.

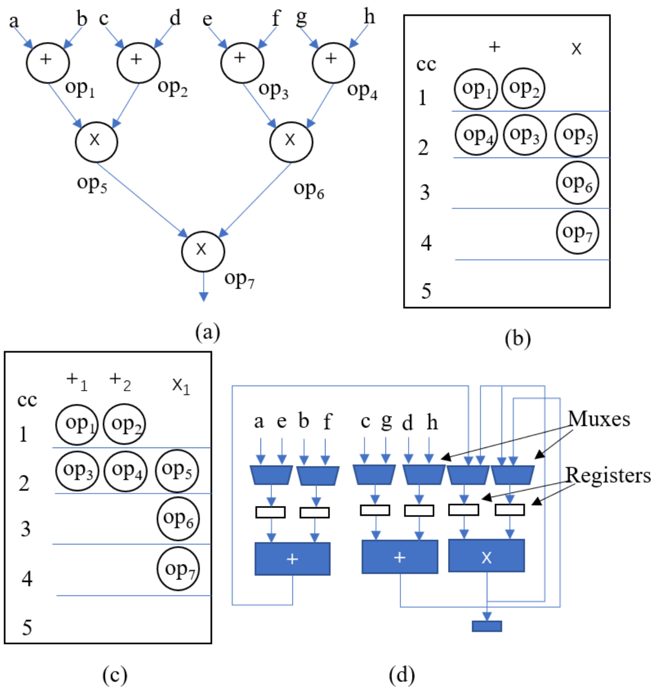

Figure 1: (a) An example DFG. (b) Scheduling and (c) Binding solutions for a latency constraint L = 5 cc's. (d) The corresponding datapath design.

In this paper, we evaluate the interconnect complexity of a given datapath/ design in HLS by the following interconnect-centric metrics.

1) The number of FU-to-FU interconnects in the datapath, denoted by n_w.
2) Interconnect congestion metrics based on the complexities/sizes of the steering logic at an FU), which includes the maximum mux/demux size, denoted by Q, and the average mux/demux size in the design, denoted by Q_{av}. Note that each m-to-1 mux (1-to-m demux, where m > 2) has a tree-structure construction with $2m-1$ basic 2-to-1 muxes (1-to-2 demuxes).

Next, we qualitatively discuss multiple correlations between FU minimization and interconnect complexity.

3. Correlation between FU Minimization and Interconnect Complexity—A Qualitative Analysis

Let $F(S)$ be the total number of FUs allocated in an S&MS solution S for a given DFG. We define the average operation utilization rate $our(S)$ as the average number of ops executed per FU within the latency period L of S. For a S&MS solution S with k ops, $our(S)$ is:

$$our(S) = \frac{k}{F(S)} \qquad (1)$$

Furthermore, $our(S)$ also represents the degree of interconnect sharing based on the data dependency in the DFG for the following reason. Consider two S&MS solutions S_1 and S_2 for the same DFG with k ops. If $F(S_1) < F(S_2)$, $our(S_1) > our(S_2)$. Thus for a set of ops that are executed on the same FU F_i in S_1, due to $F(S_1) < F(S_2)$, the probability of these ops doing data transfer with a particular FU F_j is $1/F(S_1)$ assuming uniform distribution of ops among FUs of the corresponding function type (more realistically, since an interconnect-aware binder is going to cluster ops on an FU A whose children or parents are in a small subset of FUs with which A has/will have interconnects—determined dynamically as the binding process proceeds—the probability will be non-uniform but still inversely proportional to some monotonically increasing function $g(F(S_1))$, e.g., a Gaussian probability distribution with a mean inversely proportional to $F(S_1)$). This is more than the corresponding probability $1/F(S_2)$ (or $1/g(F(S_2))$) in S_2. As a result, two FUs F_i and F_j with an interconnect between them has more data transfers on an average than the same interconnect in S_2. Hence, since the total number of data transfers = number of arcs in the DFG, is the same for both S_1 and S_2, fewer interconnects are needed in S_1 than in S_2.

As far as congestion goes, *our*(S) also tells us that more ops are packed in an FU in a solution with fewer FUs. This means that there is a higher probability that two communications from A will also have a higher probability of having functionally dissimilar children ops, leading potentially to a greater fanout from A. Whether the actual fanout from A increases or decreases for a solution with a smaller number of FUs depends on the relative values of the following two probabilities: (i) probability p_s of two communications out of A with functionally-same child ops, sharing an interconnect (which increases with fewer FUs in a solution and thus has a reduction effect on fanout), and (ii) the probability p_d of the communications having functionally dissimilar children (which also increases with fewer FUs, but has an enlarging effect on the fanout). A symmetric argument applies to the fanin of an FU. Thus based on the relative values of the aforementioned probabilities, the congestion Q and average congestion Q_{av} can either increase or decrease as the number of FUs used increases for a particular DFG.

4. Correlation between FU Minimization and Interconnect Complexity—A Probabilistic Analysis

We develop a probabilistic model for the MR-LCS problem to estimate interconnect-related metrics of an interconnect-unaware binding process that attempts to bind ops of each functional type uniformly across all FUs of that type. We assume interconnect-unaware binding for simplicity, and in order to capture the main correlation between the number of FUs implied by an S&MS solution and interconnect metrics. Our empirical results also show that interconnect-unaware binding results in only 10 % more interconnects on the average compared to interconnect-aware binding. Our analysis should thus also hold, albeit somewhat approximately, for interconnect-aware binding (see Sec. 5-2).

For a function type α, let the number of ops in the DFG be n_α, with an average out-degree (in-degree) to (from) ops of function type-β of $d^o_{\alpha,\beta}$ ($d^i_{\alpha,\beta}$).

$$d^o_{\alpha,\beta} = \frac{n_\alpha}{F_\alpha(S)}, \quad d^i_{\alpha,\beta} = \frac{n^o_{\alpha,\beta}}{n_\alpha} \qquad (2)$$

where $n^i_{\alpha,\beta}$ is the number of type-β ops with inputs from type-α ops, and $n^o_{\alpha,\beta}$ is the number of type-β ops with outputs to type-α ops.

Let the number of type-α FUs implied by an S&MS solution S be $F_\alpha(S)$; the average number of ops bound to a type-α FU is $\frac{n_\alpha}{F_\alpha(S)}$. We use a uniform probability density function; so the probability p that a type-α op is bound to a particular (type-α) FU is $1/F_\alpha(S)$. In future work we will explore other probability density functions like Gaussian.

The average out-degree (number of output data transfers) from a type-α FU to type-β FUs is $D^o_{\alpha,\beta}$:

$$D^o_{\alpha,\beta} = \frac{n_\alpha}{F_\alpha(S)} \cdot d^o_{\alpha,\beta} \qquad (3)$$

The average in-degree (number of input data transfers) from type-β FUs to a type-α FUs is $D^i_{\alpha,\beta}$:

$$D^i_{\alpha,\beta} = \frac{n_\alpha}{F_\alpha(S)} \cdot d^i_{\alpha,\beta} \qquad (4)$$

Let $P_\beta(m, k, r, p)$ be the probability that m type-β ops are bound to exactly k out of r type-β FUs with a base probability of p of a type-β op for being bound to any FU.

$$P_\beta(m,k,r,p) = C^k_r \cdot \Big(\sum_{i=1}^{U} C^i_m \cdot P_\beta(i,1,1,p) \cdot P_\beta(m-i,k-1,k-1,p)\Big) \qquad (5)$$

where C^k_r is "k Choose r". Note that once set, p is a constant throughout the recursion. $U = min\{m\text{-}k\text{+}1, V_\beta\}$, and V_β is the upper bound on the number of type-β ops that can be bound to a single type-β FU, and can be estimated as $min\{(L/((1+\phi)\cdot d_\beta), DF(\beta)\}$, where: (a) ϕ is the fractional fragmentation (we use ϕ = 0.25), d_β is the delay of a type-β FU, and L the latency constraint; (b) $DF(\beta)$ is a non-empty FU distribution factor (to guarantee that no FU fills up with too many ops so that some FUs of that type are empty. It can be formulated as:

$$DF(\beta) = \min\left\{n_\beta - F_\beta(S) + 1, \frac{(1+\gamma)n_\beta}{F_\beta(S)}\right\} \qquad (6)$$

where γ is a uniform-distribution deviation factor in the range [0, 1], and $\gamma = 0.25$ in our experiments. We first prove the following fundamental result about the correctness of our probability analysis before proceeding further (readers may choose to skip the proof without missing pertinent information for understanding the subsequent analysis).

Theorem 1: The probability space defined by $P_\beta(m, k, r, p)$ in Eqn. 5 is a valid one.

Proof Outline: The first term in Eqn. 5, $\sum_{i=1}^{U} C^i_m \cdot P_\beta(i,1,1,p)$ corresponds to the probability of allocating i ops to a designated FU. Since this number varies from 1 to U, this means that in the 2nd term $P_\beta(m\text{-}i, k\text{-}1, r\text{-}1, p)$ of Eqn. 5, each of the remaining (k-1) FUs will have the chance to have the same number (and same subsets) of ops as the designated FU does across all possible patterns of ops to FU bindings among k FUs. Thus the first two terms exhaustively account for all possible ways of distributing the m ops among all k FUs (so that each FU has at least one op bound to it). Also, for each binding of i ops to the designated FU, the second term accounts for all ways of distributing the remaining (m-i) ops among exactly the remaining (k-1) FUs (and hence the 3rd parameter of the 2nd term is also (k-1), as there is no choice but to use all (k-1) remaining FUs—note also that the choices of different subsets of k out of r FUs is captured in the C^k_r term outside the summation expression, and within the summation and its two probability terms, we have an exact subset of k FUs to distribute the m ops among.

We illustrate the above arguments with an example. Let m = 5 and k = 3; then, ignoring V_β for simplicity, U = 3. Let us focus on a specific number of ops, say, 3, to see if Eqn. 5 accounts for patterns of distribution in which each FU can be bound to 3 ops. The 1st term accounts for 3 ops for the designated FU, since *i* ranges from 1 to 3. For *i* = 1 for the 1st term, the 2nd term is $P_\beta(4, 2, 2, p)$, which recursively is $=C_2^2 \cdot (\sum_{i=1}^{3} C_4^i \cdot P_\beta(i,1,1,p) \cdot P_\beta(4-i,1,1,p))$.

So now the "second" designated processor in this sub-expression can be bound to 3 ops, as *i* again ranges from 1 to 3. Further, again for *i* = 1 in the recursive term, the 2nd term of the sub-expression is $P_\beta(3, 1, 1, p)$, which exactly accounts for binding 3 ops to the 3rd FU (in fact all possible subsets of 3 ops, as the C_4^1 = 4 ways of selecting 1 op for the 2nd FU also implies the same number of ways (which is also C_4^3 — analytically, not by coincidence) for selecting the 3-op subsets for the 3rd FU. These are within the subexpression, in which within the "outer" $C_m^1 = C_5^1 = 5$ choices of 1 op for the 1st FU, and thus within the $C_5^1 = C_5^4$ patterns of distributing 4 ops among 2 FUs. Thus the total number of 3-op subsets bound to the 3rd FU that are accounted for in Eqn. 6 is $C_5^4 \cdot C_4^3 = C_5^3 =$ all 3-op subsets among the original set of *m* = 5 ops. A similar analysis shows that Eqn. 5 accounts for the distribution of all possible C_5^3 3-op subsets for the 1st and 2nd FUs. Since 3 ops was a generic number we chose, the above analysis applies to any number *i* between 1 and *U* and thus Eqn. 5 accounts for all *i*-op subsets bound to each of the *k* processors without repeating any; it thus does not undercount or overcount any *i*-op subset in its distribution/binding to any of the *k* FUs. Note also that the definition of $P_\beta(m, k, r, p)$ is that exactly *k* FUs be used for binding the *m* ops, and no less, and thus for correctness *U* can be no more than *m*-*k*+1, which is also taken into consideration. This proves the theorem. □

Boundary Conditions for Eqn. 5: (a) $P_\beta(m, k, r, p) = 1$ if $m = 0$; (b) $P_\beta(m, k, r, p) = 0$ if ($m > 0$ and any of $k, r = 0$) or ($k > r$) or ($m < k$) or (ceiling(m/k) > U_β); (c) $P_\beta(m, 1, 1, p) = p^m$; (d) $P_\beta(m, 1, r, p) = C_r^k p^m = r{\cdot}p^m$ if $m \le U_\beta$, else $P_\beta(m, 1, \mathrm{r}, p) = 0$ (this is also obtained from Eqn. 5 and the other boundary conditions).

The average number of interconnects $W_{\alpha,\beta}^o$ ($W_{\alpha,\beta}^i$) from (into) a type-*α* FU to (from) type-*β* FUs is:

$$W_{\alpha,\beta}^o = \sum_{k=1}^{Y^o} k \cdot P_\beta\left(D_{\alpha,\beta}^o, k, F_\beta(S), \frac{1}{F_\beta(S)}\right)$$

$$W_{\alpha,\beta}^i = \sum_{k=1}^{Y^i} k \cdot P_\beta\left(D_{\alpha,\beta}^i, k, F_\beta(S), \frac{1}{F_\beta(S)}\right) \qquad (7)$$

where $Y^o = min\{D_{\alpha,\beta}^o, F_\beta(S)\}$ and $Y^i = min\{D_{\alpha,\beta}^i, F_\beta(S)\}$. Furthermore, the average number $W(S)$ of interconnects corresponding to S&MS solution *S* is:

$$W(S) = \sum_\alpha \sum_\beta F_\alpha(S) \cdot W_{\alpha,\beta}^o \qquad (8)$$

where *α* = *β* is allowed. Then, the average max input and output congestions corresponding to S&MS solution *S*, $Q_o(S)$ and $Q_i(S)$, respectively, are:

$$Q_o(S) = \max_\alpha \sum\nolimits_\beta W_{\alpha,\beta}^o$$

$$Q_i(S) = \max_\alpha \sum\nolimits_\beta W_{\alpha,\beta}^i \qquad (9)$$

Moreover, the average max congestion corresponding to S&MS solution *S* is $Q(S) = max\{Q_o(S), Q_i(S)\}$, and the average congestion of *S* is $Q_{av}(S) = W(S)/F(S)$.

5. Experimental Results

5.1 Scheduling Techniques for Obtaining Varying Degrees of FU Optimization

We obtain empirical correlation results for the MR-LCS HLS problem for 11 DFGs in [9] by varying the degree of FU-centric optimization in the initial scheduling stage via using the following different well-known algorithms ranging from the seminal/classical and low- to medium quality (LS, FDS, SA) to the state-of-the-art approximate and high quality (FALLS) to an optimal formulation with exponential complexity (ILP): list scheduling (LS) [5], force-directed scheduling (FDS) [7], a simulated-annealing-based technique (SA) [8], FALLS [3], and ILP [6]. We note that for keeping the complexity of our empirical and probabilistic analysis tractable, we do not perform module selection (e.g., [4]) here, but believe that the conclusions we derive should hold when module selection is performed. We obtain a wide range of the degree of FU-centric optimization by plotting the FU-centric results (# of FUs, total area of FUs) along with interconnect-centric metrics (# of interconnects, congestion) of the above scheduling algorithms followed by a common interconnect-aware variation (described in Sec. 5-2) of optimal FU-binding and register-allocation techniques (both using the left-edge algorithm [2]). All techniques were implemented in C++, and all runs were performed on an Intel Core i7-4710HQ processor at 2.5 GHz with 16 GB RAM.

5.2 Interconnect-Aware Binding

After obtaining different scheduling solutions via the techniques listed above, we use interconnect-aware variations of the optimal techniques [2] for FU binding and register allocation to determine interconnect-centric metrics: numbers of interconnects and congestion. For brevity, we describe here only the FU binding technique. This is a modified interconnect-aware left-edge binding technique (Int-LE) for minimizing the numbers of interconnects without changing the FU allocation results provided by a scheduling (or S&MS) solution (i.e., the optimality of the left-edge binding algorithm [2] for the number of FUs is retained in Int-LE). The core idea of this technique is to proceed chronologically by cc's, and in cc *t*, among all ops scheduled in *t*, determine the best (op = *u*, available FU = *F*) pair such that *F* has the maximum of the sum of: (1) existing fanin connections from all FUs that the parent ops of *u* have been bound to, and (2) existing fanout connections to all possible FUs that the child ops of *u* can be bound to. We then bind this (*u*, *F*) pair, and update the interconnects and mux/demux sizes are of all affected FUs. The process is repeated to bind the next best operation-FU pair in cc *t*, and so forth until all such pairs are bound in this cc. The binding then proceeds to ops scheduled in cc *t*+1. Thus, Int-LE achieves the goal of FU-to-FU interconnect minimization.

5.3 Correlation Evaluation

Table 1 shows the number of allocated FUs, numbers of interconnects (*W*), max congestion (*Q*), average congestion (Q_{av}), and area results for LS, FDS, SA, FALLS, and 0/1-ILP.

FALLS reduces the total number of allocated FUs by an average of 14.8% to 49.3% compared to LS, FDS, and SA. Similar are the results for FU area reduction. FALLS has the same number of FUs compared to the optimal 0/1-ILP with a 0.2% optimality gap in FU area.

Figures 2-4 plot W, Q and Q_{av} for the different scheduling techniques (coupled with the aforementioned Int-LE binder) and our probabilistic model across 11 DFGs; for the probabilistic model, the assumed number of FUs for each DFG is the average of the number of FUs allocated by the different techniques. As seen in these plots, except for 2 data points out of 33 (one for Q for DFG *inter.* in Fig. 3, and one for Q_{av} for DFG *write* in Fig. 4), our probabilistic model tracks the empirical results obtained by the better scheduling techniques quite accurately (a 94% accuracy rate). Having established the significant degree of accuracy of our probabilistic model, we can use its results, that we obtain across many more number of FU points than given by the scheduling techniques, to draw reliable conclusions and guidelines.

Figure 5 plots W and Q_{av} across many FU points (FU optimization degrees) for the largest DFG *mat-inv*. It also includes the plot lines for these metrics obtained from the scheduling techniques' results (coupled with the aforementioned Int-LE binder). As can be seen, the probabilistic model's estimates track the empirical results well. The corresponding plot lines are similar for other DFGs. The main conclusions that we can reliably draw here are:

(a) The number of interconnects increase as the number of allocated FUs increase for the same DFG. This also tracks our qualitative analysis in Sec. 3.

(b) The average congestion decreases as the number of allocated FUs increase for the same DFG. This shows that among the two conflicting probabilities p_s and p_d (see Sec. 3), the fanout-increasing probability p_d clearly has the dominating effect. This also provides the following design guideline: If congestion and thus routability is a limiting factor in a design, this can be alleviated by increasing the number of FUs (FU-based "white-spacing") used either globally, for a particular functional type, or in local regions of the chip wherever the congestion is acute; this, of course, has to be followed by re-scheduling ops on all the FUs (a variation of the scheduling algorithms for the ML-RCS problem—minimizing latency given the number of FUs as resource constraints—could be used for this purpose).

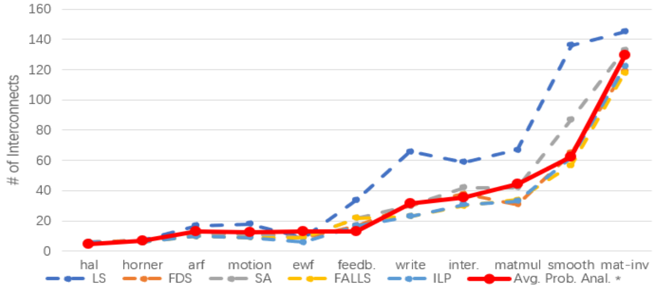

Figure 2: Number of interconnects W for multiple scheduling techniques followed by Int-LE binding, and our probabilistic model for 11 DFGs with increasing sizes from left to right. The solid line is the probabilistic analysis, and the dashed lines are for different scheduling techniques.

Another relevant phenomenon is seen in the FDS and SA results in Table 1. Even though the number of FUs allocated in the two techniques are almost the same (average difference across DFGs of 3.6%), the difference in their W metric is significant—about 15%. We believe that the answer to this conundrum is as follows. Figures 6-7 plot the probabilistic model's estimates of W along with the empirical W results of FDS (Fig. 6) and SA (Fig. 7). In these plots, the numbers of FUs for each DFG assumed for the probabilistic model's estimates are those that are allocated by the corresponding scheduling technique. While the analytical results track the empirical ones from both SA and FDS, it is closer to the FDS results. The reason for this is that the probabilistic model assumes a uniform distribution in the binding of ops to FUs (of the same functionality), which leads to a well-balanced ops occupancy across all FUs of the same functionality. FDS's main goal is to balance the number of ops executing in each cc in order to minimize the maximum number of FUs executing in any cc, and thereby to minimize the number of FUs. This temporal balance can be shown to translate to a spatial balance (i.e., in the number of ops bound to the FUs of the same functionality). This explains the closer accuracy of our probabilistic analysis in estimating W for FDS than for SA, even though the numbers of FUs allocated in each technique are similar. Thus the significant difference in interconnect complexity between FDS and SA is most probably due to spatial balance in FDS and some lack thereof in SA, which does not have this goal. This leads to our final conclusion/guideline:

(c) For any number of FUs allocated for a DFG (i.e., at any FU optimization degree), a balanced ops to FU distribution, will generally provide a reduced number of interconnects compared to a relatively unbalanced distribution.

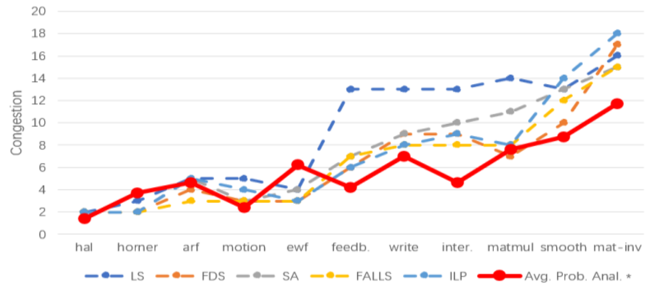

Figure 3: Max congestion Q for multiple scheduling techniques followed by Int-LE binding, and our probabilistic model for 11 DFGs.

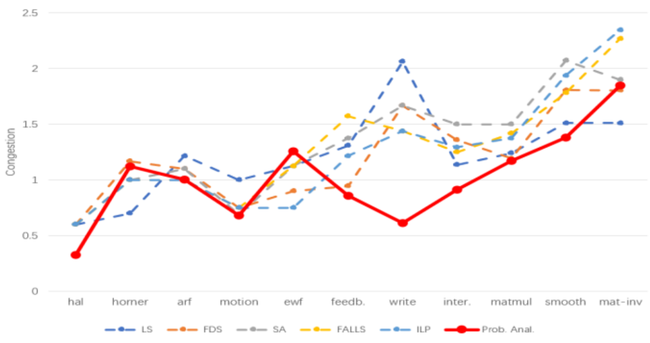

Figure 4: Average congestion Q_{av} for multiple scheduling techniques followed by Int-LE binding, and our probabilistic model for 11 DFGs.

6. Conclusions

In this paper, we explored the correlation between the degree of FU minimization (achieved via multiple scheduling techniques) and some important interconnect-centric metrics for the MR-LCS HLS problem. We also developed a probabilistic model to estimate these interconnect metrics that is quite accurate for the better performing scheduling techniques. The empirical and analytical (probabilistic model) results support our qualitative analysis for the expected correlation: the number of interconnects *W* will decrease for the most part with increased degree of FU-centric optimization, but the average congestion Q_{av} will increase. Furthermore, based on both empirical and analytical results, we propose what we hope are useful guidelines to designers to improve *W*, *Q*, and Q_{av} for any scheduling and binding technique.

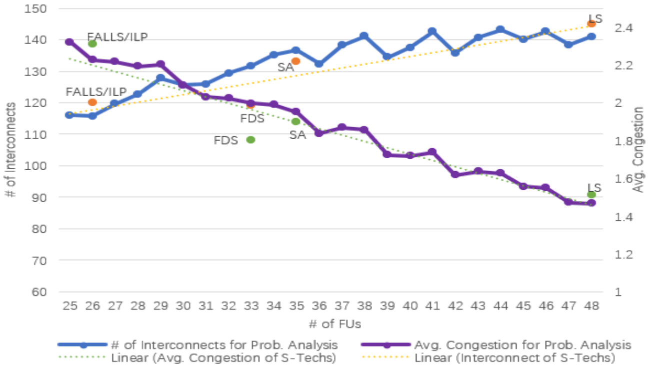

Figure 5: Interconnect metrics for DFG *mat-inv* versus different numbers of allocated FUs for different scheduling techniques followed by Int-LE binding, and a range of assumed number of allocated FUs for our probabilistic model.

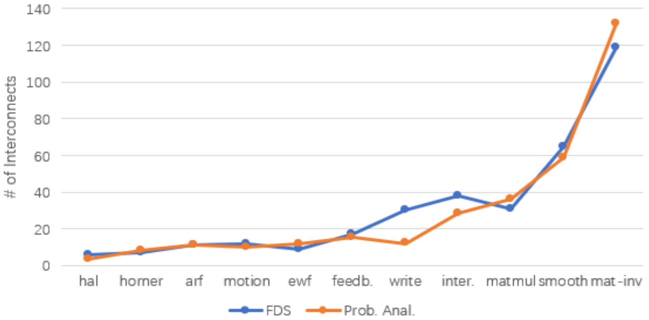

Figure 6: Number of interconnects *W* for FDS plus Int-LE binding, and our probabilistic model for 11 DFGs.

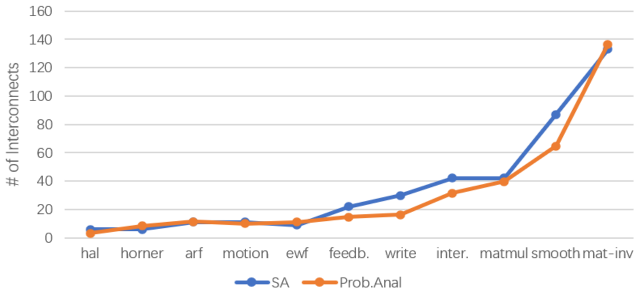

Figure 7: Number of interconnects *W* for SA plus Int-LE binding, and our probabilistic model for 11 DFGs.

Acknowledgements

This research was funded by NSF grant CCF- 2035610.

References

[1] R. Kastner, et al. ”Layout driven data communication optimization for high level synthesis.” In *Proc. Design Automation & Test in Europe Conference*, 2006, pp. 1–6.

[2] A. Hashimoto and S. Stevens, “Wire routing by optimizing channel assignment within large apertures,” *8th DAC workshop*, 1971, pp. 155–169.

[3] S. Dutt and O. Shi, “A fast and effective lookahead and fractional search based scheduling algorithm for high-level synthesis,” in *Proc. Design Automation & Test in Europe Conference*, 2018, pp. 31–36.

[4] X. Zhang, O. Shi, J. Xu, and S. Dutt, “A power-driven stochastic-deterministic hierarchical high-level synthesis framework for module selection, scheduling, and binding,” *Journal of Low Power Electronics*, Vol. 15, No. 4, Dec. 2019, pp. 388–409.

[5] A. M. Sllame and V. Drabek, “An efficient list-based scheduling algorithm for high-level synthesis”, in *Proc. Euromicro Symposium on Digital Systems Design*, 2002, pp. 316–323.

[6] C.-T. Hwang, J.-H. Lee and Y.-C. Hsu, "A formal approach to the scheduling problem in high level synthesis," *IEEE Trans. on CAD*, Vol. 10, 1991, pp. 464–475.

[7] P. Paulin and J. Knight, "Force-directed scheduling for the behavioral synthesis of ASICs," *IEEE Trans. on CAD*, Vol. 8, No. 6, 1989, pp. 661–679.

[8] J. A. Nestor and G. Krishnamoorthy, "SALSA: a new approach to scheduling with timing constraints," *IEEE Trans. on CAD*, Vol. 12, No. 8, Aug. 1993, pp. 1107–1122.

[9] ExPRESS benchmark.
http://express.ece.ucsb.edu/benchmark

Table 1: Average number of FUs (# of FUs), number of interconnects (W), max congestion (Q), average congestion (Q_{av}), and area results across different scheduling techniques coupled with the int-LE binder.

DFG	Size (Node, Arc)	LS					FDS					SA					ILP					FALLS				
		# of FU	W	Q	Qav	Area	# of FU	W	Q	Qav	Area	# of FU	W	Q	Qav	Area	# of FU	W	Q	Qav	Area	# of FU	W	Q	Qav	Area
hal	11, 8	5	6	2	0.6	394	5	6	2	0.6	394	5	6	2	0.6	394	5	6	2	0.6	394	5	6	2	0.6	394
horner	18, 16	5	7	3	0.7	394	3	7	2	1.2	238	3	6	2	1.0	238	3	6	2	1.0	238	3	6	2	1.0	238
arf	28, 30	7	17	6	1.2	559	5	11	5	1.1	394	5	11	5	1.1	394	5	10	5	1.0	394	5	10	5	1.0	394
motion	32, 29	9	18	6	1.0	715	8	12	4	0.8	633	8	11	4	0.7	633	6	9	4	0.8	477	6	9	4	0.8	477
ewf	34, 47	4	9	4	1.1	312	5	9	3	0.9	394	4	9	4	1.1	312	4	9	3	0.8	312	4	6	3	1.1	312
feedb.	53, 50	13	34	15	1.3	1008	9	17	7	0.9	697	8	22	7	1.4	623	7	22	6	1.2	550	7	17	7	1.6	550
write	106, 88	16	66	14	2.1	1237	9	30	9	1.7	697	9	30	9	1.7	697	8	23	8	1.4	614	8	23	10	1.4	614
inter.	108, 104	26	59	16	1.1	2035	14	38	10	1.4	1091	14	42	10	1.5	1247	12	30	9	1.3	935	12	31	9	1.3	935
matmul	109, 116	27	67	16	1.2	2090	13	31	8	1.2	1018	14	42	11	1.5	1100	12	34	8	1.4	944	12	33	9	1.4	944
smooth	197, 196	45	136	14	1.5	3567	18	65	12	1.8	1736	21	87	13	2.1	1632	16	57	14	1.9	1247	16	62	14	1.8	1247
mat-inv	333, 354	48	145	20	1.5	3768	33	119	19	1.8	2576	35	133	15	1.9	2750	26	118	18	2.3	2059	26	122	17	2.3	2044
Avg	92.3, 99.7	18.6	51.3	10.5	1.2	1462	11.1	31.4	7.4	1.2	897	11.5	36.3	7.5	1.3	911	9.5	29.45	7.2	1.2	742	9.5	29.5	7.5	1.3	741
FALLS % Improv.		49.3%	42.4%	29.3%	-5.9%	49.3%	14.8%	5.8%	-1.2%	-6.9%	17.4%	17.5%	18.5%	0.0%	2.2%	18.7%	0.0%	-0.3%	-3.8%	-3.6%	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%