High-Level Power Estimation with Interconnect Effects*

Kavel M. Büyükşahin
ECE Dept. and Coordinated Science Lab.
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801, USA
buyuksah@uiuc.edu

Farid N. Najm
ECE Department
University of Toronto
Toronto, Ontario, Canada M5S 3G4
f.najm@toronto.edu

ABSTRACT
We extend earlier work on high-level average power estimation to include the power due to interconnect loading. The resulting technique is a combination of a RTL-level gate count prediction method and average interconnect estimation based on Rent's rule. The method can be adapted to be used with different place and route engines and standard cell libraries. For a number of benchmark circuits, the method is verified by extracting wire lengths from a layout of each circuit and then comparing the predicted (at RTL) power against that measured using SPICE. An average error of 14.4% is obtained for the average interconnect length, and an average error of 25.8% is obtained for average power estimation including interconnect effects.

Categories and Subject Descriptors
B.7.2 [**Hardware**]: Integrated Circuits—*Design Aids*

General Terms
High-level power estimation, Register transfer level (RTL) power estimation, Interconnect capacitance estimation

1. INTRODUCTION
The rapid increase in design complexity and the need to reduce the design time have resulted in the need for CAD tools that can help make design decisions early in the design process. Delay and power dissipation are two of the most important criteria that have to be taken into account while making these decisions. Thus, there is a need to estimate the power consumption and delay from a design description of the circuit at a high level of abstraction, before the low level details of the circuit have been finalized.

*This research was supported by the Semiconductor Research Corporation, under SRC 97-DJ-484, with customization funds from Texas Instruments Inc. and SRC 99-TJ-682, with customization funds from IBM Corp..

Before the advent of deep submicron technology, it was sufficient to concentrate on the logic gates of the circuit and neglect the effects of the interconnect. The delay of a given circuit was mainly determined by the logic gates comprising it. In deep submicron designs, this is no longer true, because interconnect capacitance has not scaled down as much as gate capacitance. As the capacitive component of the power dissipation is the dominant component of the total power dissipated in a static CMOS circuit, the relative increase of interconnect capacitance with respect to gate capacitance results in an increased importance of interconnect loading for power estimation.

Currently, there are techniques to estimate the power consumption and the delay of circuits given a *register transfer level* (RTL) description [6]. These techniques concentrate on the logic gates of the circuit and do not take the interconnect effects into account. The work presented in this paper is aimed towards incorporating the wire capacitance into the approach presented in [6]. In order to do that, we have to be able to estimate the average interconnect capacitance from the RTL description of a circuit. This requires an estimate of the average interconnect *length*, from a high-level (RTL) design specification.

We propose a method to estimate the average interconnect length given a high-level description for the circuit. To do this, we will make use of the well-known Rent's rule, which is stated as:

$$P = F \cdot G^p \tag{1}$$

where P is the total number of pins (inputs or outputs) of a logic circuit, F is the average number of pins per block (cell or gate) constituting the circuit, and G is the number of blocks (cells or gates) in the circuit. The superscript p in the equation (known as the "Rent's exponent") is an empirical constant taking values $0 < p < 1$. Rent's rule was introduced in 1971 by Landman and Russo [5], and used by many researchers in prediction of interconnection attributes since then (average length, length distribution, number of nets, etc.) [1, 2, 3, 4, 7, 8, 9].

The proposed method is based on a modified form of the average interconnect length estimation method of [8], and a high-level area estimation method introduced in [6]. The next section describes the estimation approach and experimental results are given in section 3, with conclusions to follow in section 4.

2. METHODOLOGY

We propose a method of average interconnect length estimation that involves first estimating the gate count and the Rent's exponent, and which includes a scale factor term that requires characterization of the place and route tool. The different steps involved are described below.

2.1 Estimating gate count

In [6], the authors propose a way to estimate the *minimum* number of gates $\mathcal{A}$ required to implement a Boolean function given the target standard cell library, without actually performing any logic synthesis on the function. The sizes of the essential prime implicants (i.e., number of literals in the cube) of the ON and OFF-sets are used to capture the complexity of a Boolean function, and define a complexity measure based on that. They also show that the RGBFs (Randomly Generated Boolean Functions), and typical VLSI functions have a similar relationship between their complexity measures and their gate counts ("areas"). Based on this, they generate a set of curves for RGBFs, and use them to estimate the area of a typical VLSI function once its complexity measure is computed. We use this method to compute the gate count G.

2.2 Estimating Rent's exponent

Having computed G, and since P is known, then if F is estimated somehow (say, from knowledge of the library as explained below), then one may consider using Rent's rule (1) backwards in order to compute p, as follows:

$$p = \frac{\log P - \log F}{\log G} \tag{2}$$

This, strictly speaking, is not correct and has to be done with care, for the following reason. Central to the application of Rent's rule is a certain partitioning scheme that is conceptually applied to the circuit. If the netlist is partitioned, and then each partition is iteratively partitioned until individual gates are reached, one can visualize a resulting *partitioning tree*. The top level of the tree consists of a single partition that contains the whole netlist and the leaves of the tree are individual gates. In practice, Rent's rule as given above is found to be applicable only to partitions that are lower than level 3 or so in the tree [5]. It is not applicable to partitions in the top three levels or so, so-called "Region II" of the partitioning hierarchy [5]. Lower levels of the tree are referred to as "Region I."

Any value of r that is determined by applying (1) to the whole netlist (i.e., at level 1 of the tree) is thus, at best, an approximation; it is a Region II "estimate." However, a determination of r based on the use of Rent's rule in Region I leads to a value of r that depends on the partitioning algorithm/scheme used [5]. There is no unique partitioning scheme that is the "correct" scheme. In principle, the partitioning scheme should be consistent with the layout algorithm used in the placement/routing tool. Some tools use partitioning, and some don't. Given a commercial place/route tool, it may be impossible to determine what, if any, partitioning scheme is being used.

To resolve this impasse, our approach is to go ahead and estimate r by applying Rent's rule to the whole circuit (Region II), but then to introduce a correction or scale factor to help adapt the average wire length estimation to the specific place/route tool one is working with. In this paper, using Cadence Silicon Ensemble, we found that a simple exponential factor provides a good fit, as will be shown below.

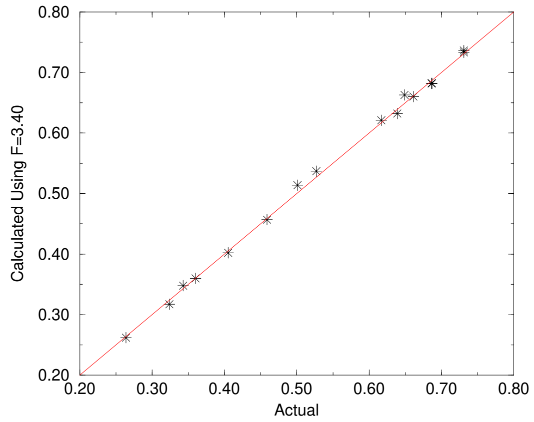

Figure 1: Rent's exponent computed using constant F vs actual Rent's exponent.

It remains to explain how F is to be estimated. This parameter (average number of terminals per gate) is a function of the gate library used and the circuit function. For a specific gate library, this value typically varies in a very limited range, and an average value can be used for all the circuits. As the difference between the actual value of F and the value used is typically small, and this difference is compressed even more by the logarithmic dependence of p on F, the error introduced will be very small. Previous design data can be used to determine the typical range of F values for a specific gate library, and an appropriate choice can be made according to those data. For our case, the actual range of F for the circuits used is 3.12–3.54, and a value of 3.40 is used in estimation. The average absolute error introduced is only 0.43% (assuming G is known exactly). Fig. 1 shows the plot of p computed by using constant F versus the actual p values.

2.3 Average wire length estimation

We have based our method for average interconnect length estimation on the method proposed in [8], where the authors present a closed form equation for average interconnect length in terms of the gate count and the Rent's exponent for the given circuit. They use the results of Donath's earlier work [4] that gives the global distribution of the wire length for a good two-dimensional Manhattan grid as:

$$f(l) = \begin{cases} g/l^{\gamma}; & 1 \leq l \leq L \\ 0; & l \geq L \end{cases} \tag{3}$$

where L is a constant related to the size of the square grid; and γ is a constant characteristic of the logic related to the Rent's exponent through the equation $2p + \gamma \approx 3$. g is a normalization constant that makes the sum of these values over all lengths l equal to 1.

The authors assume that in a hierarchical design, the local wire length distribution can be estimated by a scaled version

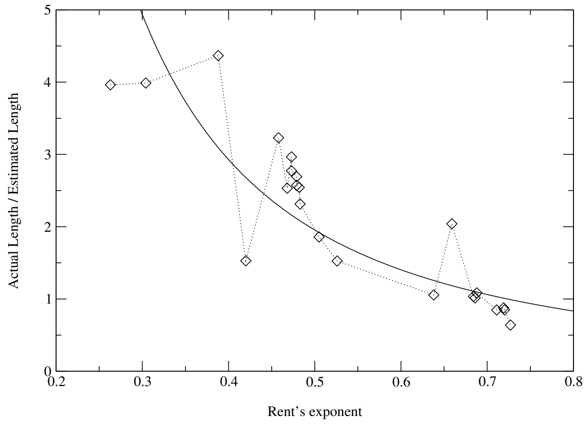

Figure 2: Circuits mapped to generic SIS library.

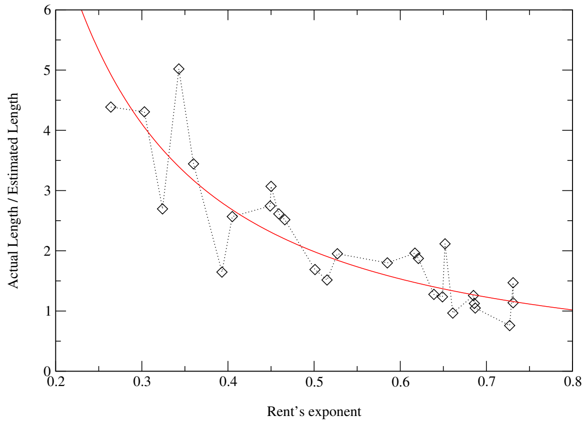

Figure 3: Circuits mapped to Odyssey cell library.

of the global distribution. Hence, it is possible to approximate the curve for global distribution of wires using the peak values of the local distributions, and come up with a wire length distribution of the form given in (3). Combining this assumption with the probability distribution function, they arrive at a closed form formula for the average interconnect length:

$$\mathcal{L}_{\mathcal{S}} = R(p)\,\frac{H(K,p,1)}{H(K,p,2)} \tag{4}$$

with

$$H(K,p,x) = \frac{2^{K(2p-x)}-1}{2^{2p-x}-1}$$

where

$$R(p) = \frac{4R_a(p)+2R_d(p)}{6}$$

and

$$\begin{aligned} R_a(p) &= \frac{(p-1)}{(p+1)}\,\frac{3^{2p+2}-(p+4)\,2^{2p+2}+(4p+7)}{3^{2p+1}-(2p+7)\,2^{2p}+(4p+5)} \\ R_d(p) &= \frac{(p-1)}{(p+1)}\,\frac{4^{2p+1}-3^{2p+2}+3\cdot 2^{2p+1}-1}{4^{2p}-3^{2p+1}+3\cdot 2^{2p}-1} \end{aligned}$$

and where K is computed from G according to

$$G = 4^K$$

The average length computed using (4) is in terms of gate pitches of the standard cell library being used.

2.4 Characterization of the place/route tool

We have used a set of combinational ISCAS and MCNC benchmark circuits to test this method. To get the actual average interconnect length, the circuits are synthesized and mapped to two different libraries using Berkeley's SIS [10] synthesis tool. Silicon Ensemble is used to place and route the circuits, and the average interconnect length is extracted from these layouts.

The results are shown in Figs. 2 and 3. The only difference between the two figures is that two different standard cell libraries were used to produce the layouts from which the interconnect lengths were extracted. Ideally, the points should be clustered around a value of 1 on the vertical axis, but of course they are not. The deviation probably has to do with the character of the placement/routing engine being used, which affects the results for two reasons. One reason (already mentioned above) is that the partitioning scheme used in the place/route engine was not factored into the estimation of Rent's exponent (2). Another reason is that the above equation (4) was derived assuming that the place/route engine uses a quadrature placement algorithm on a Manhattan grid. However, this may not always be the case. Commercial place/route tools use different algorithms and certainly many heuristics in order to find an optimal placement and routing scheme for a given circuit.

Therefore, we consider the observed near-exponential decay in Figs. 2 and 3 to be a result of the algorithms and heuristics used in the specific place/route engine being used. This view is enforced by the fact that this behavior is repeated for different libraries. Therefore, our approach has been to modify the average interconnect estimation with a scale factor of the form:

$$\mathcal{S}(p) = C \cdot p^k \tag{5}$$

where C and k are parameters that are related to the standard-cell library and the place/route tool being used, and which can be obtained through regression analysis. We view the computation of these constants as a characterization of the specific place/route tool and the standard cell library being used.

With this modification, the average interconnect length model becomes

$$\mathcal{L} = \mathcal{S}(p) \cdot \mathcal{L}_{\mathcal{S}} \tag{6}$$

where $\mathcal{L}_{\mathcal{S}}$ is the length computed using (4), and $\mathcal{L}$ is in terms of gate pitches.

In summary, the overall flow of the estimation is as follows:

1. Characterization of the place and route engine:
   - The values for C and k in (5) are obtained by using previous design data.

2. Characterization of the target standard cell library:
   - The curves used by the area (gate count) estimation tool are generated for the specific gate library.
   - The gate-pitch for the library is determined.
   - An appropriate value for F is determined by analyzing previous designs.
3. Estimation of G and p:
   - Using the area estimator, the minimum number of gates required to implement the circuit in the target library is estimated. Using this value, and the value for F determined in step 2, p is estimated.
4. Estimation of average interconnect length:
   - Using (6) with the values of C and k obtained in step 1, an estimate of average interconnect length in terms of gate-pitch is computed. This value is multiplied by the gate-pitch to get the average interconnect length in absolute units.

The first two steps are performed only once for the place and route engine at hand and the target standard cell library. The actual estimation of average wire length is performed in steps 3 and 4.

3. EXPERIMENTAL RESULTS

In this paper, we have addressed the problem of estimating the Rent's exponent and the average interconnect length from a high-level description of a circuit. In this section we will report experimental results for estimation of these two parameters, as well as average power dissipation of the circuit including interconnect effects.

In our experiments, circuits were synthesized and mapped on a subset of the Odyssey cell library using SIS [10]. Silicon Ensemble was used to place and route the circuits; the average interconnect length and capacitance are extracted from these layouts.

The values for C and k of (5) were found to be 0.862 and -1.275, respectively, in the characterization step. The circuits used to do the characterization are given in Table 1 and are different from the circuits used to verify the technique, which are given in Table 2. These circuits were selected from the combinational ISCAS and MCNC benchmark circuits. The gate pitch for the cell library used is 7.20 μm. As can be seen in Table 3, the average error in estimation of average interconnect length is 14.4%. A correlation plot is also given in Fig. 4.

The actual Rent's exponents for the circuits are computed by using the actual values for gate count (G), average number of terminals per gate (F), and the total number of I/O pins (P). G and F are obtained from the gate-level netlist constructed by SIS. The estimated values for Rent's exponent are computed using (2) with the estimated value for G and a single value for F. A reasonable value for F is found to be 3.40 in the characterization step. As can be seen in

Table 1: Benchmark circuits used for characterization.

Circuit	#I/O (P)	#Gates (G))
74181	22	60
alu4	21	401
c1355	73	434
c1908	58	411
c2670	221	425
c3540	72	777
c6288	64	2274
c880	86	254
i7	266	453
i8	214	567

Table 2: Benchmark circuits used to verify the technique (s1196 is the combinational part of the original sequential circuit).

Circuit	#I/O (P)	#Gates (G)
alu2	14	224
apex6	220	451
apex7	79	145
c432	43	113
c499	73	139
cht	83	124
example2	147	170
i6	205	495
s1196	28	349
x3	224	463
x4	159	186

Table 3: Average interconnect length (μm).

Circuit	Actual	Estimated	Error (%)
alu2	40.84	35.24	13.7
apex6	25.93	31.39	21.1
apex7	22.75	24.46	7.5
c432	27.85	28.71	3.1
c499	33.44	26.39	21.1
cht	21.61	26.18	21.2
example2	31.30	30.64	2.1
i6	23.05	28.92	25.4
s1196	41.46	36.55	11.8
x3	27.86	31.99	14.8
x4	24.73	28.94	17.0

Table 4: Rent's exponent.

Circuit	Actual	Estimated	Error (%)
alu2	0.264	0.283	7.2
apex6	0.687	0.688	0.1
apex7	0.639	0.645	0.9
c432	0.527	0.479	9.1
c499	0.617	0.594	3.7
cht	0.649	0.633	2.5
example2	0.731	0.631	13.7
i6	0.661	0.733	10.9
s1196	0.360	0.356	1.1
x3	0.686	0.681	0.7
x4	0.731	0.676	7.5

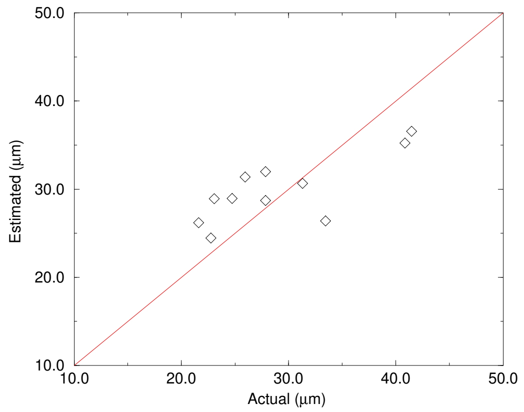

Figure 4: Estimated vs actual average interconnect length.

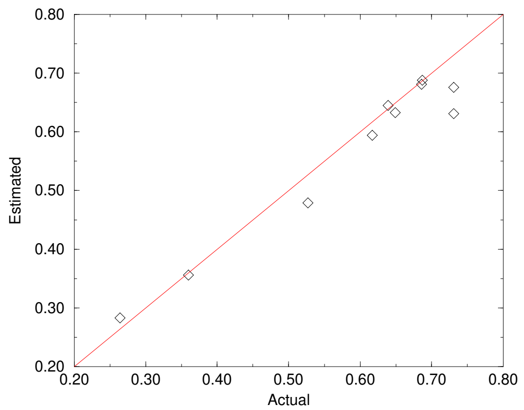

Figure 5: Estimated vs actual Rent's exponent.

Table 4, the average error in estimation of Rent's exponent is 5.2%. A correlation plot is also given in Fig. 5.

In order to verify the average power estimation including interconnect, actual power values were obtained using Star-Hspice. The power was predicted at a high level by starting with the Boolean equations for each circuit and, using [6], predicting the gate count and the average switching activity. The power is then obtained as the product of $0.5V_{dd}^2$ and the following terms: (i) the predicted gate count, (ii) the predicted average switching activity, and (iii) the sum of the average gate output capacitance (computed from the library [6]) and the average interconnect capacitance (estimated as detailed above). As can be seen in Table 5, the average error in estimation of average power dissipation is 25.8%. A correlation plot is also given in Fig. 6.

4. CONCLUSION

We have presented a method for estimating the average interconnect length of a combinational circuit given the high-level description (Boolean equations). This method can be tuned for different place and route engines and standard cell libraries. Estimation is based on Rent's rule, and and a high-level area (gate count) prediction method.

Table 5: Average power dissipation (μW).

Circuit	Actual	Estimated	Error (%)
alu2	17.90	11.22	37.2
apex6	19.23	26.77	39.2
apex7	10.59	8.754	17.3
c432	12.20	13.24	8.5
c499	13.50	11.14	17.4
cht	11.08	10.03	9.4
example2	10.81	13.01	20.3
i6	28.48	21.68	23.8
s1196	17.68	13.08	25.9
x3	20.03	28.22	40.8
x4	12.50	18.06	44.5

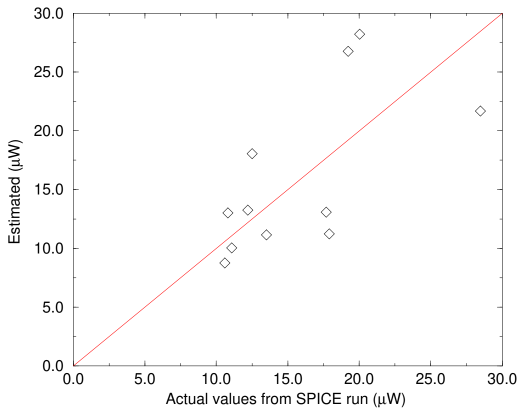

Figure 6: Estimated vs actual average power dissipation.

5. REFERENCES

[1] J. A. Davis, V. K. De, and J. D. Meindl. A stochastic wire length distribution for gigascale integration (gsi). In *Proc. Custom Integrated Circuits Conf.*, pages 145–150. IEEE, 1997.

[2] J. A. Davis, V. K. De, and J. D. Meindl. A stochastic wire-length distribution for gigascale integration (gsi)—part ii: applications to clock frequency, power dissipation, and chip size estimation. *IEEE Trans. Electron Devices*, 45:590–597, March 1998.

[3] W. E. Donath. Placement and average interconnection lengths of computer logic. *IEEE Trans. Circuits and Systems*, CAS-26:272–277, April 1979.

[4] W. E. Donath. Wire length distribution for placements of computer logic. *IBM Journal of Research and Development*, 25:152–155, May 1981.

[5] B. S. Landman and R. L. Russo. On a pin versus block relationship for partitions of logic graphs. *IEEE Trans. Computers*, C-20:1469–1479, December 1971.

[6] M. Nemani and F. Najm. High-level area and power estimation for vlsi circuits. *IEEE Trans. Computer-Aided Design*, 18(6):697–713, June 1999.

[7] D. Stroobandt and F. J. Kurdahi. On the characterization of multi-point nets in electronic

designs. In *Proc. 8th Great Lakes Symposium on VLSI*, pages 344–350. IEEE, February 1998.

[8] D. Stroobandt, H. V. Marck, and J. V. Campenhout. An accurate interconnection length estimation for computer logic. In *Proc. 6th Great Lakes Symposium on VLSI*, pages 50–55. IEEE, March 1996.

[9] A. Y. Tetelbaum. Estimations of layout parameters of hierarchical systems. In *Proc. 27th Southeastern Symposium on System Theory*, pages 353–357, 1995.

[10] *SIS-1.2 Reference Manual*. University of California, Berkeley, 1992.