Exact and Approximate Estimation for Maximum Instantaneous Current of CMOS Circuits

Yi-Min Jiang and Kwang-Ting Cheng

Dept. of Electrical & Computer Engineering
University of California, Santa Barbara
CA 93117, USA
e-mail: {jym,timcheng}@bigbend.ece.ucsb.edu

Abstract

We present an integer-linear-programming-based approach for estimating the maximum instantaneous current through the power supply lines for CMOS circuits. It produces the exact solutions for the maximum instantaneous current for small circuits, and tight upper bounds for large circuits. We formulate the maximum instantaneous current estimation problem as an integer linear programming (ILP) problem, and solve the corresponding ILP formulae to obtain the exact solution. For large circuits we propose to partition the circuits, and apply our ILP-based approach for each sub-circuit. The sum of the exact solutions of all sub-circuits provides an upper bound of the exact solution for the entire circuit. Our experimental results show that the upper bounds produced by our approach combined with the lower bounds produced by a genetic-algorithm-based approach confine the exact solution to a small range.

1. Introduction

With increasing demands for high reliability in modern VLSI designs, accurate estimation of the maximum instantaneous current during the design process is becoming essential. Excessive instantaneous current through the power and ground (P&G) nets may result in performance degradation due to large voltage drops along the P&G nets and circuit failures due to electromigration.

For CMOS circuits, instantaneous current is mainly due to signal switching, which, in turn, depends on the input patterns applied to the circuits. To cause signal switching, a two-vector sequence, $V = (v_1,v_2)$, has to be applied at the inputs. One way to find the maximum instantaneous current would be to simulate all possible patterns. For a circuit with n primary inputs, this would require simulation of 4^n patterns. This is practical only for circuits with a small number of primary inputs.

Several approaches have been proposed for maximum instantaneous current estimation [6][5][4]. Kriplani *et al*. [6] present a pattern-independent algorithm (iMax algorithm) to find an upper bound on the maximum instantaneous current. Krstic *et al*. [5] propose a timed ATPG algorithm and a probability-based algorithm to estimate the maximum instantaneous current. In the timed ATPG [5] approach, a set of signals whose simultaneous switching produces high current is assigned transitions and timed ATPG is used to derive test patterns. In the probability-based approach, a set of selected gates is assigned weights based on their possible current contribution at the given time. Next, these weights are propagated backwards to the primary inputs, and the patterns for maximum instantaneous current are derived using these values. A genetic-algorithm-based approach for finding lower bounds for the maximum instantaneous current has been proposed in [4]. This approach applies a genetic algorithm to identify patterns causing high instantaneous current through iteratively generating new patterns for simulation. The new patterns are generated using genetic operations, based on "good" patterns derived in the previous iterations. All techniques, except [6], target finding a lower bound of the exact solution. They are heuristic procedures and the quality of the lower bounds can not be precisely measured. The algorithm proposed in [6] for estimating maximum instantaneous current obtains an upper bound of the exact solution. However, due to the assumption that all signals (primary inputs and internal signals) are uncorrelated, the estimated maximum instantaneous current for most circuits represents a loose upper bound.

In this paper we propose an integer-linear-programming-based technique to obtain the *exact* solutions for the maximum instantaneous current for small circuits, and tight upper bounds for large circuits. We model the problem as an integer linear programming (ILP) problem. Solving the corresponding ILP formulae allows us to obtain the *exact* solutions. However, this approach may not be suitable for large designs because of the large number of variables involved. Therefore, we propose to partition a large circuit into sub-circuits, and then obtain the exact solution of each sub-circuit by solving its corresponding ILP formulae. *Since the worst-case solution for each sub-circuit can be computed, the sum of the worst-case solutions of all sub-circuits corresponds to an upper bound of the worst-case solution for the entire circuit.*

*This work was supported by the National Science Foundation under grant MIP-9503651, California MICRO and Synopsys/EPIC Inc.

The contributions of our ILP-based approach for the estimation of the maximum instantaneous current are twofold. First, the exact worst-case solutions derived by our approach for small circuits can be used to evaluate the estimation quality of other approaches for the maximum instantaneous current. Second, the upper bounds of the worst-case solutions for large circuits, together with the lower bounds derived by other approaches give designers proper guidelines for estimating the exact worst-case solutions. Our experimental results show that our approach produces, on the average, an upper bound on the maximum instantaneous current which is 47% tighter than the one obtained by iMax algorithm [6]. Also, the upper bounds derived by our approach, combined with the lower bounds derived by a genetic-algorithm-based approach [4] confine the exact solutions to a small range. To our knowledge, the ILP-based technique is the first reported methodology for obtaining the exact solution on the maximum instantaneous current.

The rest of this paper is organized as follows. In Section 2, we first introduce the current model used in this paper. We, then, present a set of transformation rules from logic gates to ILP formulae, and propose a transformation rule used for modeling the maximum instantaneous current problem as an ILP problem. In Section 3, we formulate the maximum instantaneous current problem as an ILP problem. The partitioning strategy for large circuits is given in Section 4. Section 5 gives the experimental results. Section 6 concludes the paper.

2. Preliminaries

2.1 Current model

Our methodology for estimating the maximum instantaneous current operates at the gate level. Therefore, we need a gate level current model. For estimating the maximum instantaneous current, we use the current model proposed in [6]. This model assumes that the current drawn from the supply lines during switching of a signal is of a triangular form as shown in Figure 1. The peak current is

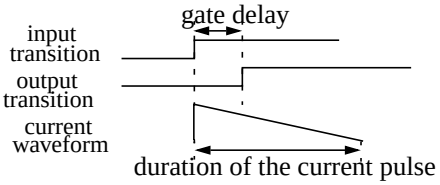


Figure 1: Current model.

assumed to coincide with the transition at the input of the gate. The value of the peak current and the duration of the current pulse are dependent on the gate type and the load capacitance of the gate.

2.2 The transformation rules from logic gates to ILP formulae

To apply the ILP-based approach for estimating the maximum instantaneous current, we derive a set of transformation rules for converting the logic functions of primitive gates into ILP formulae. Using these rules, we transform the logic description of a circuit into a set of integer linear constraints. Then, maximizing instantaneous current corresponds to optimizing an objective function with respect to the set of linear constraints. In the following, we describe our transformation rules.

Rule 1. The logic function of an *m*-input OR gate can be represented as: $y \le x_1+x_2+...+x_m \le m*y$, where x_1, x_2, ..., x_m are the inputs of the gate, and y is the output of the gate. The values of x_1, x_2, ..., x_m and y are limited to 0 and 1.

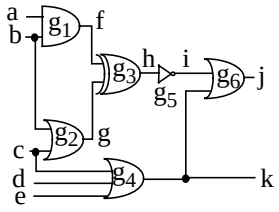
Rule 2. The logic function of an *m*-input AND gate can be represented as: $m*y \le x_1+x_2+...+x_m \le y+(m-1)$, where x_1, x_2, ..., x_m are the inputs of the gate, and y is the output of the gate. The values of x_1, x_2, ..., x_m and y are limited to 0 and 1.

Rule 3. The logic function of an inverter can be represented as: $y = 1-x$, where x is the input of the gate, and y is the output of the gate. The values of x and y are limited to 0 and 1.

Rule 4. The logic function of a two-input XOR gate can be represented as: $\pm(x_1 - x_2) \le y \le x_1+x_2 \le 2-y$, where x_1 and x_2 are the inputs of the gate, and y is the output of the gate. The values of x_1, x_2 and y are limited to 0 and 1.

The following example illustrates the above rules.

Example 2.1 Consider the circuit in Figure 2(a). The integer linear constraints obtained after applying the transformation rules are shown in Figure 2(b).



(a)

$2f \le a + b \le f + 1$	----- g_1
$g \le b + c \le 2g$	----- g_2
$\pm(f - g) \le h \le f + g \le 2 - h$	----- g_3
$k \le c + d + e \le 3k$	----- g_4
$i = 1 - h$	----- g_5
$j \le i + k \le 2j$	----- g_6

(b)

Figure 2: The transformation from logic gates to ILP formula.

For CMOS circuits the current through the supply lines is mainly due to the switching on the signals. Based on the general-delay model, and using the *iMax* algorithm [6] we can obtain all possible switching times for each signal and calculate current contributions of all gates at all times. The instantaneous current at time *t* corresponds to the sum of the current contributions of all gates at time *t*. The instantaneous current at each time instance can be modeled as a single function, and the instantaneous current for all time instances can be represented as a multi-function. Since the ILP package that we use [7] can optimize only a single objective function, we need to transform the optimization of a multi-function into the optimization a

single function. We use the following proposition to perform this transformation.

Proposition 1. For an m-output multi-function with outputs c_1, c_2, ..., c_m, the maximum value can be found by solving the following ILP formulae:

Maximize $k_1+k_2+...+k_m$ (1)

Subject $\alpha_1+\alpha_2+...+\alpha_m = 1$; (2)

$c_i - k_i \le L*(1-\alpha_i)$, $1 \le i \le m$; (3)

$0 \le k_i \le \alpha_i * L$, $1 \le i \le m$; (4)

$k_i \le c_i$, $1 \le i \le m$; (5)

where k_1, k_2, ..., k_m are real numbers, and L is a large positive real number whose value is greater than or equal to any possible value of the c_1, c_2, ..., c_m. The values of α_1, α_2, ..., α_m are limited to 0 and 1.

Proof. Since the values of α_1, α_2, ..., α_m are limited to 0 and 1, to satisfy constraint (2), one α must be set to 1, and others to 0. Constraints (3), (4), and (5) ensure that if α_i is set to 1, then k_i is equal to c_i; otherwise k_i is equal to 0. The four constraints ensure that only one k can have a non-zero value, while others are equal to zero. Therefore, maximizing the objective function (1) results in the maximum value of the multi-function.

3. ILP Formulation for the Maximum Instantaneous Current

Before introducing our ILP formulation for the maximum instantaneous current we define the following notations:

- G is the set of all gates.
- g_i is the gate with index i.
- $g_i(t)$ the output value of g_i at time t.
- T is the set of all time instances.
- $T(g_i)$ is the set of all possible transition times of the output of gate g_i.
- $T_{g_i}(m)$ = 1, if a switching of gate g_i occurs at time m. = 0, otherwise.
- $[T(g_i)]^j$ is the set of switching times of gate g_i such that these transitions contribute to the instantaneous current at time j.
- $I_{g_i}(j)$ is the current at time j contributed by gate g_i.
- $[I_{g_i}(j)]^m$ is the current value at time j contributed by the output switching at time m of gate g_i.
- $I(j)$ is the total instantaneous current at time j (for the entire circuit).

The maximum instantaneous current problem can be formulated as follows:

Maximize $\sum_{j=1}^{|T|} k_j$ (6)

Subject to

$$\sum_{j=1}^{|T|} \alpha_j = 1; \quad (7)$$

$$I(j) - k_j \le L \times (1-\alpha_j), \quad \text{for } 1 \le j \le |T|; \quad (8)$$

$$0 \le k_j \le \alpha_j \times L, \quad \text{for } 1 \le j \le |T|; \quad (9)$$

$$k_j \le I(j), \quad \text{for } 1 \le j \le |T|; \quad (10)$$

$$T_{g_i}(t_l) = g_i(t_{l-1}) \oplus g_i(t_l), \quad \text{for } 1 \le i \le |G|, \ t_{l-1}, t_l \in T(g_i), \ 1 \le l \le |T(g_i)|; \quad (11)$$

$$I_{g_i}(j) = \underset{m}{Max} \left(T_{g_i}(m) \times [I_{g_i}(j)]^m\right), \text{ for } 1 \le i \le |G|, \ 1 \le j \le |T|, \forall m \in [T(g_i)]^j; \quad (12)$$

$$I(j) = \sum_{i=1}^{|G|} I_{g_i}(j), \quad \text{for } 1 \le j \le |T|; \quad (13)$$

where L is a large real number whose value is greater than or equal to any possible value of the instantaneous current at all time instances, and the values of α_1, α_2, ..., α_m are limited to 0 and 1. The objective function (6) states that we are going to maximize the instantaneous current. Constraints (7), (8), (9), and (10) implement the multi-function optimization (see Proposition 1). Constraint (7) states that the maximum instantaneous current appears only at one time instance. Constraint (11) states that the switching of gate g_i at time t_k happens when the output values of the gate is different at times t_k and t_{k-1}. Constraints (12) refers to the instantaneous current at time j contributed by gate g_i. The operation, *Max* can be expressed as integer linear constraints. (This transformation is tedious, and the details are omitted here.) This current corresponds to the maximum of all possible current contributions that gate g_i can have at time j. Constraint (13) states that the instantaneous current for the entire circuit at each time instance is derived by summing up the current contributions of all gates at the corresponding time instance. The optimal solution of the objective function represents the maximum instantaneous current.

4. Partitioning-based approach

The time required for solving the ILP formulae grows rapidly with the increase of the size of the circuit. We propose a partitioning-based approach to obtain upper bounds of the worst-case solutions for larger circuits. This approach partitions a large circuit into sub-circuits, and applies our ILP-based approach for each sub-circuit to obtain the worst-case solution for each sub-circuit. After these worst-case solutions are obtained, the summation of the solutions of all the sub-circuits represents an upper bound of the worst-case solution for the entire circuit. In this section, we investigate the partitioning issues in order to achieve tight upper bounds of the worst-case solutions.

4.1 Modeling the maximum instantaneous current

Our partitioning-based approach for obtaining a tight upper bound on the maximum instantaneous current, in the first step, uses the iMax algorithm [6] to produce an upper bound of the instantaneous current at each time instance. As mentioned before, the bound given by iMax for each time instance is a loose upper bound. Then, all time instances with non-zero upper bound of the instantaneous current are put in the processing list. This list is next sorted in descending order of the corresponding upper bound.

We select the time instance with the highest upper bound from the processing list and extract the part of the circuit which contributes to this upper bound. We then apply a partitioning algorithm K-MAFM [1] to partition the part of the circuit into sub-circuits. The maximum number of gates allowed in each sub-circuit is chosen as 300 in our experiment. Empirically, this value gives tighter upper bounds in a reasonable CPU time for our ILP-based technique. After partitioning the extracted circuit into sub-circuits, we apply the ILP-based approach to each sub-circuit and then sum up the exact solutions for the sub-circuits. The result represents a new, tighter upper bound (tighter than the bound obtained by iMax) of the instantaneous current at the given time because the signal correlations in each sub-circuit are considered. If the upper bounds for some time instances in the processing list are already lower than the newly obtained upper bound, we do not need to process those time instances, and remove these time instances from the processing list. The above process continues by selecting a time instance with the next highest upper bound on the instantaneous current, and ends when the processing list is empty. The maximum value of the new upper bounds at all time instances is referred to a tight upper bound of the maximum instantaneous current of the entire circuit. Figure 3 shows the summary of our algorithm.

```
Perform iMax algorithm to obtain the upper bound of the instantaneous
  current at each time instance;
Put all sorted time instances in the processing list;
While the processing list is not empty
{
  Select the time instance with the highest upper bound of the
    instantaneous current;
  Extract the part of the circuit which contributes to this current;
  Partition the extracted circuit into sub-circuits;
  Apply the ILP-based approach to each sub-circuit to obtain the
    instantaneous current at this time;
  Sum up the instantaneous current at this time of all sub-circuits;
  Update the upper bound of the instantaneous current at this time;
  Update the processing list;
}
```

Figure 3: Summary of the algorithm for estimating the maximum instantaneous current.

5. Experimental Results

To characterize the gate delays for different types of gates and different loads, we have built lookup tables using a transistor-level simulator DelayMill [3]. Also, lookup tables obtained by PowerMill [2] are used for estimating the values of the peak current and the duration of current pulse for different types of gates and different loads. Our experimental results are derived based on these delay and current tables, as well as the gate-level current models shown in Section 2.1. We compare our results for the maximum instantaneous current, to the results obtained by a genetic-algorithm-based approach [4] which produces a lower bound of the solution, and to a random approach which generates a set of weighted random patterns with primary input switching probability of 0.9. The number of input patterns generated by genetic-algorithm-based and random approach is 9600. Also, we compare our results to the results obtained by iMax algorithm [6]. We use a commercial tool LINDO [7] to solve the ILP formulae.

We chose 9 small MCNC benchmark circuits and the 8 largest ISCAS85 benchmark circuits. Table 1 shows the estimated maximum instantaneous current for the 9 small MCNC benchmark circuits. All the instantaneous current values are normalized with respect to the exact solution obtained by ILP. In Table 1, Columns 2-3, 4-5, 6-7, 8-9 show the maximum instantaneous current and normalized values estimated by (1) iMax algorithm, (2) ILP-based approach, (3) genetic-algorithm-based approach (4) random approach, respectively. The values estimated by genetic-algorithm-based and random approaches correspond to lower bounds, and the values estimated by iMax algorithm refer to upper bounds. The exact solution can be derived by our ILP-based approach. The CPU times for the four approaches are reported in Columns 10, 11, 12, and 13, respectively.

The estimated maximum instantaneous current for the 8 largest ISCAS85 benchmark circuits is shown in Table 2. Columns 2-3, 4-5, 6-7, 8-9 show the maximum instantaneous current and normalized value estimated by (1) iMax algorithm, (2) ILP-with-partitioning approach, (3) genetic-algorithm-based approach, and (4) random approach, respectively. All the normalized values are with respect to the values derived by our ILP-with-partitioning approach. Note that the values estimated by iMax algorithm and our ILP-with-partitioning approach correspond to the upper bounds, and the values estimated by genetic-algorithm-based and random approaches correspond to the lower bounds. The CPU times for the four approaches are reported in Columns 10, 11, 12 and 13, respectively.

The experimental results show that the ILP-based approach produces the exact worst-case solution in a reasonable time for small circuits. Also, the ILP-with-parti-

tioning approach provides tighter upper bounds of the worst-case solutions for large circuits as compared to the bounds derived by iMax. Note that the upper bounds derived by our approach are close to the lower bounds derived by a genetic-algorithm-based approach [4]. Therefore, the two bounds confine the worst-case solutions to a small range.

6. Conclusions

We have proposed an ILP-based approach to obtain the exact solutions for the maximum instantaneous current estimation. For large designs, we have proposed a partitioning strategy to obtain tight upper bounds. The experimental results show that in comparison with the lower and upper bounds derived by other approaches, the bounds produced by our approach are much tighter.

References

[1] J. Cong, Z. Li, and R. Bagrodia, "Acyclic Multi-Way Partitioning of Boolean Networks," *Proc. of DAC*, pp. 670-675, June 1994.

[2] EPIC Design Technology, "PowerMill Reference Manual," 1995.

[3] EPIC Design Technology, "DelayMill Reference Manual," 1995.

[4] Y.-M. Jiang, K.-T. Cheng, and A. Krstic, "Estimation of Maximum Power and Instantaneous Current Using a Genetic Algorithm," *Proc. of CICC*, pp. 135-138, May 1997.

[5] A. Krstic and K.-T. Cheng, "Vector Generation for Maximum Instantaneous Current Through Supply Lines for CMOS Circuits," *Proc. of DAC*, pp. 383-388, June 1997.

[6] H. Kriplani, F. N. Najm, and I. N. Hajj, "Pattern Independent Maximum Current Estimation in Power and Ground Buses of CMOS VLSI Circuits: Algorithms, Signal Correlations, and Their Resolution," *TCAD*, pp. 998-1012, August 1995.

[7] LINDO, "User's Manual," LINDO Systems, Inc., 1996.

Table 1: The estimated maximum instantaneous current for 9 small MCNC benchmarks

circuit	maximum instantaneous current								CPU time (sec.)			
	upper bound		exact solution		lower bound							
	iMax		ILP		GA		random		iMax	ILP	GA	random
	(mA)	normal.	(mA)	normal.	(mA)	normal.	(mA)	normal.				
cm42a	14.4	1.21	11.9	1.00	11.9	1.00	11.9	1.00	8	119	33	33
cm163a	51.1	1.37	37.3	1.00	37.0	0.99	24.6	0.66	12	373	29	29
cm85a	47.3	1.17	40.4	1.00	40.4	1.00	27.0	0.67	6	89	39	39
cmb	38.5	1.04	37.0	1.00	34.8	0.94	26.9	0.73	16	370	46	46
cc	63.0	1.41	44.7	1.00	44.7	1.00	40.0	0.89	18	447	43	43
cm150a	72.1	1.35	53.4	1.00	53.4	1.00	45.5	0.85	21	534	52	52
pcler8	79.7	1.12	71.2	1.00	71.2	1.00	39.3	0.55	24	712	72	72
b9	144.7	1.43	101.2	1.00	99.4	0.98	70.4	0.70	43	994	102	102
c8	51.8	1.20	43.2	1.00	43.2	1.00	37.8	0.88	37	432	112	112
average	-	1.26	-	1.00	-	0.99	-	0.77	-	-	-	-

Table 2: The estimated maximum instantaneous current for 8 largest ISCAS85 benchmarks

circuit	maximum instantaneous current								CPU time (min.)			
	upper bound				lower bound							
	iMax		ILP with partitioning		GA		random		iMax	ILP with partition	GA	random
	(mA)	normal.	(mA)	normal.	(mA)	normal.	(mA)	normal.				
C880	124.6	1.11	112.5	1.00	103.2	0.92	82.9	0.74	0.5	38.1	3.5	3.5
C1355	220.3	1.24	177.6	1.00	134.6	0.76	100.9	0.57	0.7	40.5	5.6	5.6
C1908	273.3	1.61	169.9	1.00	136.4	0.80	104.8	0.62	0.9	46.5	6.0	6.0
C2670	223.1	1.21	184.7	1.00	184.3	0.99	130.1	0.70	1.2	88.9	15.1	15.1
C3540	616.4	2.11	292.0	1.00	212.7	0.73	152.1	0.52	1.7	70.7	11.9	11.9
C5315	710.2	1.47	484.0	1.00	335.4	0.69	261.3	0.54	2.3	102.1	22.5	22.5
C6288	1364.2	1.67	816.8	1.00	723.4	0.89	567.2	0.69	3.6	136.4	40.3	40.3
C7552	1138.2	1.37	830.8	1.00	568.9	0.68	449.2	0.54	2.5	123.6	32.8	32.8
average	-	1.47	-	1.00	-	0.81	-	0.62	-	-	-	-