Path Sensitization in Critical Path Problem *

Hsi-Chuan Chen David H.C. Du

Department of Computer Science
University of Minnesota
Minneapolis, MN 55455

Abstract

Since the delay of a circuit is determined by the delay of its longest sensitizable paths (such paths are called critical paths), the problem of estimating the delay of a circuit is called critical path problem. One important aspect of the critical path problem is to decide whether a path is sensitizable. Several path sensitization criteria have been proposed in previously proposed critical path algorithms. However, they are often presented in different forms and it is hard to compare with each other. In this paper we propose a path sensitization criterion according to a general framework. Other path sensitization criteria can also be presented in the same framework. Therefore, they can be compared with each other.

1. Overview of Critical Path Problem

One important requirement of circuit design is the long path timing constraint, which requires the actual delay of a circuit to be bounded by a constant τ - usually the clock period. A path which is never activated by any primary input vector is referred to as a non-sensitizable path or a false path. On the other hand, paths which can be activated by at least one primary input vectors are referred to as sensitizable paths. Since the signal at each primary output of a circuit will become valid no later than the length of the longest sensitizable paths, the actual delay of the circuit is defined as the length of the longest sensitizable paths in the circuit. Those longest sensitizable paths are considered to be the critical paths to the circuit. The problem of estimating the length of the critical paths is, thus, referred to as critical path problem.

Recently, several critical path algorithms have been proposed to improve the accuracy of estimating the actual delay of a circuit [1, 2, 4, 6, 7]. Each of them bases on a path sensitization criterion to determine whether a path is sensitizable or not. Different algorithms are based on different path sensitization criteria and, thus, may have different estimations. Using a path sensitization criterion, the estimated delay of a circuit may be longer than or shorter than the actual delay of the circuit. In the former case (the estimated circuit delay is greater than the actual circuit delay), if the estimated circuit delay is also less than or equal to τ, the long path timing requirement is met. In the latter case, even though the obtained circuit delay is less than or equal to τ, the long path timing requirement may or may not be met. Therefore, from timing verification point of view, a criterion is considered to be "correct" if its estimated circuit delay is never less than the actual circuit delay. Certainly, a criterion is considered to be more accurate if its estimation is closer to the actual circuit delay.

Since all path sensitization criteria are described in different forms, it is hard to directly compare them and evaluate their accuracies. It is also very important to decide whether a proposed criterion is correct. This paper presents a framework such that various path sensitization criteria can be compared in a unified way. We first give some basic definitions to be used in the rest of this paper. Then a framework is presented. A path sensitization criterion corresponding to the framework is also proposed. The proposed path sensitization criterion is an exact criterion (i.e., it achieves 100% accuracy). We present several previously proposed path sensitization criteria in the same framework.

2. Definitions

A combinational circuit is composed of simple gates and leads. Each lead connects the output of a gate to an input of another gate. The delay of gate G and lead f are denoted by $d(G)$ and $d(f)$.

A *path* $P = (f_0, G_1, f_1, \cdots, G_{m-1}, f_{m-1})$ in a circuit is an alternating sequence of leads and gates. Lead f_0 connects a primary input to gate G_1 and lead f_{m-1} connects gate G_{m-1} to a primary output. Lead f_i, $1 \le i \le m-2$, connects gate G_i to gate G_{i+1}. The length of P is the sum of the delays of all the gates and leads of P, and is denoted by $d_p(P)$. Partial path $(f_0, G_1, ..., f_{k-1})$ is denoted by P_k. The length of P_k is denoted by $d_p(P_k)$.

A logic value is the *controlling value* to a gate if the logic value at an input to the gate independently determines the value at the output of the gate. The controlling value to gate G is denoted by $c(G)$. For examples, $c(G)$ = "0" if G is an AND gate or a $NAND$ gate, and $c(G)$ = "1" if G is an OR gate or a NOR gate. The *non-controlling value* to gate G, denoted by $n(G)$, is the complementary value of $c(G)$. For exam-

*This work was supported in part by NSF Grant MIP-9007168

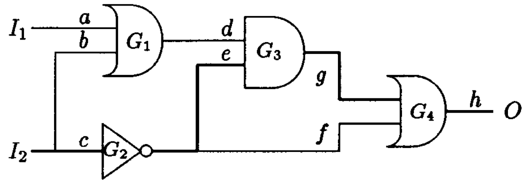

Figure 1: Stable time computation under $v = [0, 1]$

ples, $n(G)$ = "1" if G is an AND gate or a $NAND$ gate, and $n(G)$ = "0" if G is an OR gate or a NOR gate.

Applying a primary input vector to a circuit at time $t = 0$, the values at the inputs and the output of each gate will become stable sooner or later. Let v be a primary input vector. The logic values stabilized at gate input f and gate output G under v are called the *stable values* of f and G and are denoted by $sv(f, v)$ and $sv(G, v)$. The times, when f and G become stable under v, are called the *stable times* of f and G and are denoted by $st(f, v)$ and $st(G, v)$.

Input f is said to be a *controlling input* to gate G under v if $sv(f, v) = c(G)$. On the other hand, f is said to be a *non-controlling input* to gate G under v if $sv(f, v) = n(G)$.

Given both the stable value and the stable time of each input to gate G under v, we describe how $st(G, v)$ is computed as follows [4]. If one or more controlling inputs to G exist under v, $st(G, v)$ is determined by the earliest controlling input. Hence, the earliest controlling input is considered to dominate G and $st(G, v)$ is equal to the stable time of the earliest controlling input plus $d(G)$. On the other hand, if all the inputs to G are non-controlling inputs under v, $st(G, v)$ is determined by the latest input. In this case, the latest input is considered to dominate G and $st(G, v)$ is equal to the stable time of the latest input plus $d(G)$. In summary, let f be an input to gate G. Input f is considered to dominate G under v if either f is the earliest controlling input, or f is the latest input when all inputs to G are non-controlling inputs.

Path P is defined to be *sensitizable* under v if each input f_i, $0 \leq i \leq m - 2$, dominates its succeeding gate G_{i+1}. A path is defined as a sensitizable path if there is at least one primary input vector that sensitizes the path.

In Figure 1, let each lead delay and each gate delay be one time unit. Let $v = [0, 1]$ be the primary input vector applied to the circuit at time $t = 0$. Since b is the only controlling input, it dominates G_1 and $st(G_1, v) = st(b, v) + d(G_1) = 2$. So, $st(d, v) = st(G_1, v) + d(d) = 3$. Input c is the only input to gate G_2. Therefore, $st(G_2, v) = 2$, $st(e, v) = 3$, and $st(f, v) = 3$. Input e dominates G_3 because it is the only controlling input to G_3. So, $st(G_3, v) = st(e, v) + d(G_3) = 4$ and $st(g, v) = 5$. Both inputs to G_4 are non-controlling inputs. Therefore, g, which is the latest input, dominates G_4. Hence, $st(G_4, v) = st(g, v) + d(G_4) = 6$ and $st(h, v) = 7$. The delay of the circuit under v, which is $st(h, v)$, equals 7. As defined, path $(c, G_2, e, G_3, g, G_4, h)$ is sensitized by v and is a sensitizable path in the circuit.

3. SENV- An Exact Path Sensitization Criterion

A path is defined to be sensizable if it can be sensitized by at least one primary input vectors. Therefore, determining the sensitizability of a path is equivalent to determining the existence of primary input vectors which sensitize the path. It will be very helpful to develop a criterion which is capable of computing the set of primary input vectors that sensitize the path. In this section, we will focus on finding all the primary input vectors that sensitize a given path. Let $P = (f_0, G_1, \cdots, G_{m-2}, f_{m-1})$ be any given path. We shall denote the set of primary input vectors which sensitize P as SENV(P). If SENV(P) is empty, path P is a false path. Otherwise, P is a sensitizable path. In the following, we will propose a path sensitization criterion to compute SENV(P). The criterion is thus denoted by SENV.

In order to help the derivation of SENV, several sets of primary input vectors are defined first. The set of primary input vectors allowing input f to be a controlling (non-controlling) input is defined as $CV(f)$ ($NV(f)$ respectively). The set of primary input vectors allowing f to be a controlling input with stable time no earlier than time t is defined as $GEV_c(f, t)$. The set of primary input vectors allowing f to be a non-controlling input with stable time no later than time t is defined as $LEV_n(f, t)$. Clearly, $CV(f)$ and $NV(f)$ are mutually exclusive. $GEV_c(f, t)$ is a subset of $CV(f)$ and $LEV_n(f, t)$ is a subset of $NV(f)$. Also, if time t_1 is earlier than time t_2, $GEV_c(f, t_1)$ is a superset of $GEV_c(f, t_2)$ and $LEV_n(f, t_1)$ is a subset of $LEV_n(f, t_2)$.

Suppose that SENV(P_{k-1}) is available, which is the set of primary input vectors sensitizing partial path P_{k-1}. We start to compute the set of primary input vectors sensitizing partial path P_k which is denoted by SENV(P_k). Clearly, we can partition SENV(P_{k-1}) into two mutually exclusive sets: SENV$_c$(P_{k-1}) and SENV$_n$(P_{k-1}). Each primary input vector in SENV$_c$(P_{k-1}) sets f_{k-2} to be a controlling input with stable time $t = d_p(P_{k-1})$. On the other hand, each primary input vector in SENV$_n$(P_{k-1}) sets f_{k-2} to be a non-controlling input with stable time $t = d_p(P_{k-1})$. That is, SENV$_c$(P_{k-1}) = SENV(P_{k-1}) $\cap CV(f_{k-2})$ and SENV$_n$(P_{k-1}) = SENV(P_{k-1})$\cap$ NV(f_{k-2}).

In order for primary input vector v to allow f_{k-2} to dominate G_{k-1}, v has to set f_{k-2} to be either the earliest controlling input or the latest input when all inputs to G_{k-1} are non-controlling inputs. Therefore, if f_{k-2} is a controlling input under v, each other input to G_{k-1} must be either a non-controlling input or a controlling input with stable time $t \geq d_p(P_{k-1})$. On the other hand, if f_{k-2} is a non-controlling input under v, each other input to G_{k-1} must be a non-controlling input with stable time $t \leq d_p(P_{k-1})$. That is, if v is in SENV(P_k), v must be in either of the following two sets. The set of inputs to G_{k-1} except f_{k-2} is denoted by $SD(f_{k-2})$ (i.e., side inputs of f_{k-2}).

$$\mathrm{SENV_c(P_{k-1})} \bigcap \left(\bigcap_{f \in SD(f_{k-2})} \bigcup_{GEV_c(f, d_p(P_{k-1}))}^{NV(f)} \right) \quad (1)$$

$$\mathrm{SENV_n(P_{k-1})} \bigcap \left(\bigcap_{f \in SD(f_{k-2})} \mathrm{LEV_n(f, d_p(P_{k-1}))} \right) \quad (2)$$

In other words, SENV(P_k) is the union of both Equations 1 and 2. As mentioned, SENV$_c(P_{k-1})$ in Equation 1 can be replaced by SENV$(P_{k-1}) \cap CV(f_{k-1})$ and SENV$_n(P_{k-1})$ in Equation 2 by SENV$(P_{k-1}) \cap NV(f_{k-1})$. Initially, SENV$_c(P_1) = CV(f_0)$ and SENV$_n(P_1) = NV(f_0)$. Thus, by recursion SENV(P_k) can be expressed as follows.

$$\bigcap_{i=0}^{k-2} \left(\begin{array}{c} CV(f_i) \bigcap \left(\bigcap_{f \in SD(f_i)} \bigcup_{GEV_c(f, d_p(P_{i+1}))}^{NV(f)} \right) \\ \bigcup \\ NV(f_i) \bigcap \left(\bigcap_{f \in SD(f_i)} LEV_n(f, d_p(P_{i+1})) \right) \end{array} \right)$$

Path P is sensitizable if SENV(P_m) is not an emptyset.

Let P be the path $(c, G_2, e, G_3, g, G_4, h)$ in Figure 1. Partial paths $P_1 = (c)$, $P_2 = (c, G_2, e)$, $P_3 = (c, G_2, e, G_3, g)$, and $P_4 = (c, G_2, e, G_3, g, G_4, h) = P$. Initially, SENV($P_1$) consists of all primary input vectors and is equal to $\{[x, x]\}$. Note that x represents logic "Don't Care". Since G_2 has only one input, SENV(P_2) =SENV(P_1) = $\{[x, x]\}$. Gate G_3 has two inputs d and e. Therefore, $SD(e) = \{d\}$. The length of partial path P_2 (i.e., $d_p(P_2)$) is equal to 3. For side input d, $NV(d) = \{[x, 1], [1, x]\}$, $GEV_c(d, 3) = \{[0, 0]\}$, and $LEV_n(d, 3) = \{[x, 1], [1, x]\} = NV(d)$. For e, the last input of P_2, $CV(e) = \{[x, 1]\}$ and $NV(e) = \{[x, 0]\}$. So, SENV(P_3) =SENV(P_2) $\cap ((CV(e) \cap (NV(d) \cup GEV_c(d, 3))) \cup (NV(e) \cap LEV_n(d, 3)))$ is equal to $\{[x, 1], [1, 0]\}$. The length of P_3 is equal to 5. For side input f, $NV(f) = \{[x, 1]\}$, $GEV_c(f, 5) = \emptyset$, and $LEV_n(f, 5) = \{[x, 1]\} = NV(f)$. For g, the last input of P_3, $CV(g) = \{[1, 0]\}$ and $NV(g) = \{[x, 1], [0, 0]\}$. So, SENV($P_4$) =SENV($P_3$) $\cap ((CV(g) \cap (NV(f) \cup GEV_c(f, 5))) \cup (NV(g) \cap LEV_n(f, 5)))$ is equal to $\{[x, 1]\}$. That is, P can be sensitized by both primary input vectors $[0, 1]$ and $[1, 1]$ and it is a sensitizable path.

4. SENV$_{loose}$- A Looser Criterion

Some criteria are not general path sensitization criteria and only concern about the sensitizations of long paths. They do not care whether the sensitizations of short paths are misclaimed. Note that a path is considered as a short (long) path if it is shorter (longer) than a critical path. However, it is important to guarantee to claim at least one critical paths as sensitizable (in order not to under-estimate) and to reduce the probability to claim a long path as sensitizable (in order to increase the accuracy). Thus, in order to make the comparison easier, we also show a way to "loose" the criterion SENV. We shall denote the loosed criterion by SENV$_{loose}$. As long as a path sensitization criterion is only concerned with critical paths, we have found that the requirement of being the latest non-controlling input can be relaxed. Instead we only need to require all other inputs to G_{i+1} to be non-controlling inputs too. That is, we can replace $LEV_n(f, d_p(P_{i+1}))$ by $NV(f)$ without over-estimating the length of critical paths. Even though SENV$_{loose}$ is looser than SENV, we show that it achieves the same estimation of the critical path length as SENV [3]. The criterion SENV$_{loose}(P)$ can be expressed as follows.

$$\bigcap_{i=0}^{m-2} \left(\begin{array}{c} CV(f_i) \bigcap \left(\bigcap_{f \in SD(f_i)} \bigcup_{GEV_c(f, d_p(P_{i+1}))}^{NV(f)} \right) \\ \bigcup \\ NV(f_i) \bigcap \left(\bigcap_{f \in SD(f_i)} NV(f) \right) \end{array} \right)$$

5. Accuracy Comparison

In the path sensitization criterion proposed in [1] (denoted by BenV), a path P is claimed to be sensitizable if there exists at least one primary input vector which sets all the side inputs of P to be non-controlling inputs. Thus, we can describe this criterion as follows:

$$\mathrm{BenV(P)} = \bigcap_{i=0}^{m-2} \left(\bigcap_{f \in SD(f_i)} \mathrm{NV(f)} \right)$$

Obviously, BenV(P) $\subseteq$ SENV$_{loose}(P)$ for any path P, and thus the estimated delay of a circuit by BenV may be shorter than or equal to that by SENV$_{loose}$. This implies the possibility of under-estimation by criterion BenV.

In the criterion proposed by Du et al [4] (denoted by DuYenV), two static timing variables ,$max(f)$ and $min(f)$, are precomputed for each input f. The variable $max(f)$ ($min(f)$) is computed as the length of the longest (shortest) partial path from any primary input to f. Clearly, the stable time at f will never be later than $max(f)$ and will never be earlier than $min(f)$. Path P is considered by DuYenV to be sensitizable, if for each f_i of P, the following two conditions are satisfied.

1. If there exists any side input f of f_i with $min(f) > d_p(P_{i+1})$, f_i has to be a controlling input.

2. For each side input f of f_i, if $max(f) < d_p(P_i)$, f has to be a non-controlling input.

We can represent DuYenV(P) as follows.

$$\bigcap_{i=0}^{m-2}\left(\begin{array}{c} CV(f_i)\bigcap\left(\bigcap_{f\in SD(f_i),\, max(f)<d_p(P_{i+1})} NV(f)\right) \\ \bigcup \\ NV(f_i)\bigcap\left(\begin{array}{c}\bigcap_{f\in SD(f_i),\, max(f)<d_p(P_{i+1})} NV(f) \\ \bigcap \\ \bigcap_{f\in SD(f_i),\, min(f)>d_p(P_{i+1})} \emptyset\end{array}\right)\end{array}\right)$$

It can be shown that SENV(P) $\subseteq$ DuYenV(P) [3]. Therefore, DuYenV is possible to over-estimation the delay of a circuit.

The criterion proposed by Perremans et al [7] (denoted by PerrV), is quite similar to DuYenV. A dynamic timing variable, $dmax(f)$, is computed for each input f during the execution of timing verification. The variable $dmax(f)$, representing an upper bound to the latest stable time of lead f, is initialized to equal $max(f)$ and is decreased as soon as it is assured that there is no sensitizable partial path terminating at f of length equivalent to the current $dmax(f)$. Path sensitization criterion PerrV contains two conditions:

1. If f_i of P is a non-controlling input, each side input f in $SD(f_i)$ must be a non-controlling input too.
2. If f_i of P is a controlling input, side input f of f_i must be a non-controlling input when $dmax(f) < d_p(P_{i+1})$.

Criterion PerrV(P) can be expressed as follows.

$$\bigcap_{i=0}^{m-2}\left(\begin{array}{c} CV(f_i)\bigcap\left(\bigcap_{f\in SD(f_i),\, dmax(f)<d_p(P_{i+1})} NV(f)\right) \\ \bigcup \\ NV(f_i)\bigcap\left(\bigcap_{f\in SD(f_i)} NV(f)\right)\end{array}\right)$$

It can be shown that $\text{SENV}_{loose}(P) \subseteq$ PerrV(P) [3]. This implies the possibility of over-estimating the delay of a circuit by criterion PerrV.

Path P is considered to be a viable path [6] (sensitizable path in our terminology) under v if and only if for each f_i of P and for each side input f of (f_i), either one of the following conditions holds.

1. f is a non-controlling input.
2. f is a controlling input and $st(f, v) \geq d_p(P_{i+1})$.

We can express ViableV(P) as follows.

$$\bigcap_{i=0}^{m-2}\left(\bigcap_{f\in SD(f_i)} \begin{array}{c} NV(f) \\ \bigcup \\ GEV_c(f, d_p(P_{i+1}))\end{array}\right)$$

Obviously, SENV(P) $\subseteq$ ViableV(P). This shows the possibility for ViableV to over-estimate the delay of a circuit. However, it is proved that ViableV can achieve the same estimation of the circuit delay as SENV [3].

We summarize the comparison results in the following.

1. regarding the sensitization of a path P:
   - BenV(P) $\subseteq$ SENV(P) $\subseteq$ $\text{SENV}_{loose}(P)$ $\subseteq$ ViableV(P);
   - SENV(P) $\subseteq$ DuYenV(P);
   - $\text{SENV}_{loose}(P) \subseteq$ PerrV(P);
2. regarding the estimation of the actual delay of a circuit:
   - BenV $\preceq$ SENV $=$ SENV_{loose} $=$ ViableV $\preceq$ DuYenV, PerreV.

6. Conclusion

The results presented in this paper represent our effort to understand critical path problem. A framework which allows various previously proposed path sensitization criteria to compare with each other in a unified way is presented. An exact and a looser path sensitization criteria based on the framework are also proposed.

References

[1] J. Benkoski and et al, *Timing Verification Using Statically Sensitizable paths*, IEEE Transactions on Computer-Aided Design, cad-9 (1990), pp. 1073–1084.

[2] D. Brand and V. Iyengar, *Timing Analysis using Functional Analysis*, tech. report, IBM Thomas J. Watson Research Center, 1986.

[3] H. Chen and D. Du, *Path Sensitization in Critical Path Problem*, in ACM International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems, 1990.

[4] D. Du, H. Yen, and S. Ghanta, *On the General False Path Problem in Timing Analysis*, in 26th Design Automation Conference, 1989, pp. 560–566.

[5] K. Keutzer, S. Malik, and A. Saldanha, *Is Redundancy Necessary to Reduce Delay?*, in 27th Design Automation Conference, 1990, pp. 228–234.

[6] P. McGeer and R. Brayton, *Efficient Algorithms for Computing the Longest Viable Path in a Combinational Network*, in 26th Design Automation Conference, 1989, pp. 561–567.

[7] S. Perremans, L. Claesen, and H. DeMan, *Static Timing Analysis of Dynamically Sensitizable Paths*, in 26th Design Automation Conference, 1989, pp. 568–573.