Characterization of CMOS Sequential Standard Cells for Defect Based Voltage Testing

A. Wielgus and W. A. Pleskacz
Institute of Microelectronics and Optoelectronics
Warsaw University of Technology
ul. Koszykowa 75, 00-662 Warszawa, POLAND
wielgus@imio.pw.edu.p, pleskacz@imio.pw.edu.p

Abstract

This paper presents a new characterization methodology of CMOS sequential standard cells for defect based voltage testing. It allows to estimate the probabilities of physical defects occurrences in a cell, describes its faulty behavior caused by the defects and finds the test sequences that detect those faults. Finally, all of found sequences are validated to check their effectiveness in fault covering and the optimal complex test sequence for all detectable faults is constructed. Experimental results for sequential cells from industrial standard cell library are presented.

1. Introduction

Test generation for sequential circuits remains a challenging problem over the last several years. The complexity of this problem is definitely much higher than for combinational circuits [1], [2] due to the presence of the memory elements. As a result, instead of applying a single test vector, one needs a sequence of vectors to detect faults in a circuit. Moreover, the controllability and observability of the internal signals become much worse. There exist many solutions to the sequential ATPG problem [2]-[6] that operate on different levels of abstraction while describing the circuits: on architectural, state transition, RTL or gate level. Nowadays, the hierarchical test generation [5]-[6] is believed to be the most effective technique for sequential circuits testing. As was stated in [5], the hierarchical approach is independent of fault models and thus could acquire the results of any low-level fault analysis. Although the most commonly used is the classical approach to fault modeling, which employs an abstract stuck-at fault model, some new approaches based on layout analysis [7]-[15] have been recently developed and successfully applied to combinational circuits [16]-[18]. The most promising seem to be two approaches, published in [12] and [13]-[15], that propose complete fault characterization of standard cell libraries. However, they still concern combinational circuits only. Characterization of flip-flops, mentioned in [19], was done only on electrical level and did not address layout related properties. In [20] some hardware improvements to flip-flops implementation were proposed in order to make them transparent during test and so to increase the fault coverage.

The authors propose to extend an approach presented in [13]-[15] for sequential standard cells, which is the main contribution of this paper. The considerations are limited to voltage testing of the complete set of the bridging faults. The methodology of sequential standard cells characterization is presented. It uses the following two mechanisms:

- layout analysis based on critical area approach in order to estimate the probabilities of physical defect occurrences,
- transistor-level electrical simulation to identify the actual behavior of a faulty cell and to find a set of test sequences that detect each particular fault.

A solution of the fault covering problem allows therefore to construct an optimal sequence of test vectors which detects all these faults.

The paper is organized as follows. Section 2 gives the detailed description of the proposed methodology of sequential standard cells characterization. In Section 3 the whole procedure is illustrated by example results for standard cells from industrial CMOS library. Section 4 concludes the paper.

2. Cells Characterization Methodology

Proposed methodology of sequential standard cells characterization consists of the following steps:

- estimation of defect occurrence probabilities,

- identification of faulty operation of the cell caused by considered defects,
- generation of test sequences for all testable faults,
- construction of the optimal test sequence.

2.1. Estimation of Defect Occurrence Probabilities

As stated earlier, our considerations in this paper are limited to shorts between conducting regions that seem to be one of the most important sources of faults in CMOS digital circuits. A short is a piece of extra conducting material that connects a pair of separate conducting regions in the integrated circuit. This affects the connectivity of the circuit - two separate electrical nodes become shorted. It is intuitively obvious that probabilities of shorts depend on the layout of the circuit. Adjacent conducting regions are more susceptible to shorts than regions that are separated by a large distance. For each pair of conducting regions representing electrical nodes, the probability of short P_{sh} was calculated. The detailed description of this procedure was given in [13]-[15]. The pair of nodes, for which $P_{sh}=0$, cannot be shorted and was not taken into account.

2.2. Identification of Faulty Operation

For a given cell, all the pairs of nodes with non-zero probabilities of shorts were considered during the next step of characterization. In order to identify the operation of a given circuit with particular shorts, several electrical simulations were performed. They were carried out in a testbench circuit given in Fig. 1 using SPECTRE simulator.

The flip-flop under test (FFUT) was driven through the input buffers. The simulation was carried out twice using “Wired-AND” and “Wired-OR” conditions at inputs. These two models were implemented by using input buffers with MOS transistors of different channel geometries. Typical flip-flops from the standard cells library have two outputs, Q and QN, representing the state of the flip-flop and its negation. In order to avoid ambiguity in classifying circuit responses as logic one or logic zero, the non-inverting buffers, which regenerate the voltage level, were also connected to the outputs. A short itself was implemented in the netlist as a resistor of 100 mΩ, which is the average value of sheet resistance of metals used in the 0.35 μm CMOS process. As the resistances of contacts and vias in this technology are several orders of magnitude higher, the value of the resistor simulating a short is not critical. The cell behavior at the logic level did not depend on the resistance values in the range of 0.01 Ω - 200 Ω.

While taking into account the basic D flip-flop, it was necessary to examine the following cases:

- switching states: 0→1 and 1→0,
- remaining in the same state: 0→0 and 1→1.

The considered defects of the flip-flop layouts from the standard cell library may cause various functional faults. One could observe the following faulty behavior:

- stuck-at faults on one or both outputs,
- clock pulse on the output,
- changes of the active clock edge,
- generation of the additional clock edges,
- oscillations on both outputs,
- other faults described by the faulty truth tables.

The waveforms obtained from the simulation allowed to determine the truth table describing the actual logic function performed by the faulty circuit. As a result a set of faulty truth tables (FTT) of a given standard cell was obtained for all considered shorts, which significantly improves the characterization process for testing purposes.

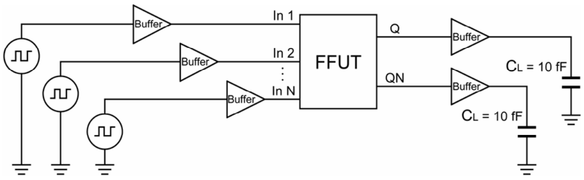


Fig.1. Schematic diagram of a testbench circuit.

2.3. Test Sequences Generation

The results of electrical simulations allowed to identify the sequences of input vectors that should be applied to detect any particular short. All the considered sequences were applied several times in order to start from different initial states and thus to take into account the memory effect of sequential cells. After applying the last vector from a sequence, the voltages on both outputs of a tested faulty circuit were compared with the reference outputs of the correct one. If any of these two pairs did not match, the input sequence was counted as a test sequence of a given short for an appropriate output.

The following sequences, describing state changes of the D flip-flop mentioned in the previous section, were taken into account:

$\rightarrow 0 \quad \rightarrow \quad S(CD) = \{00\ 10\}$,
$\rightarrow 1 \quad \rightarrow \quad S(CD) = \{01\ 11\}$,
$0 \rightarrow 0 \quad \rightarrow \quad S(CD) = \{00\ 10\ 00\ 10\}$,
$1 \rightarrow 1 \quad \rightarrow \quad S(CD) = \{01\ 11\ 01\ 11\}$,
$0 \rightarrow 1 \quad \rightarrow \quad S(CD) = \{00\ 10\ 01\ 11\}$,
$1 \rightarrow 0 \quad \rightarrow \quad S(CD) = \{01\ 11\ 00\ 10\}$.

The first two sequences concern the situation of setting the state independently of the previous one. The last two sequences include simultaneous change of both clock and data inputs. In fact, a particular order of the input values changes leads to one of the two possible sequences, e.g:

$$\begin{tikzcd}
 & S(CD) = \{00\ 10\ 00\ 01\ 11\} \\
S(CD) = \{00\ 10\ 01\ 11\} \arrow[ur] \arrow[dr] & \\
 & S(CD) = \{00\ 10\ 11\ 01\ 11\}
\end{tikzcd}$$

$$\begin{tikzcd}
 & S(CD) = \{01\ 11\ 01\ 00\ 10\} \\
S(CD) = \{01\ 11\ 00\ 10\} \arrow[ur] \arrow[dr] & \\
 & S(CD) = \{01\ 11\ 10\ 00\ 10\}
\end{tikzcd}$$

The last vector in a sequence specifies when one has to compare the output voltages. The authors called such a sequence a **simple test sequence** to emphasis that comparison of voltages is made only once after applying the whole sequence. However, the result of such comparison could depend on the details of clock and data timings. Careful analyses of possible faulty behaviour shows that the moment of changing input voltages and the moment of comparing output voltages could affect the result of test. Because of that, it was necessary to treat the clock input C as a regular input of a circuit during simulations. It allowed to consider some special cases of setting the state of the flip-flop:

- comparison occurs when the clock is high e.g.: $S(CD) = \{01\ 11\}$,
- comparison occurs when the clock is low e.g.: $S(CD) = \{01\ 11\ 01\}$,
- data value changes when the clock is high e.g.: $S(CD) = \{00\ 10\ 11\ 01\}$,
- data value changes when the clock is low e.g.: $S(CD) = \{00\ 10\ 00\ 01\}$.

After collecting the results of test sequences, one could construct a fault table, which describes the connection between all detectable faults and the simple test sequences able to detect them. Based on the fault table and probabilities of defects, the effectiveness of simple test sequences could be calculated. This **effectiveness** is defined as a sum of occurrence probabilities of a set of faults detected by a given sequence. An example of such a fault table is given in Tab. 1 and will be discussed in the next section.

2.4. Construction of the Optimal Test Sequence

One can easily notice that simple sequences may be covered by each other. For instance, having two simple sequences:

$$S_1 = \{00\ \mathbf{10}\},\ S_2 = \{00\ 10\ \mathbf{00}\},$$

S_2 covers S_1. One can combine such simple sequences in a complex one with the extended set of detected faults formed from the original sets. Thus, the effectiveness of the sequence also increases. However, a complex sequence requires the output voltage to be checked more than once. In a presented example the following complex sequence is obtained:

$$S_2 = \{00\ \mathbf{10}\ \mathbf{00}\},$$

where input vectors that require checking the output voltage were bolded in the sequence. The last step is to select the minimal set of complex sequences that cover all detectable faults and construct the optimal test sequence. The algorithm of solution to this fault covering problem looks as follows:

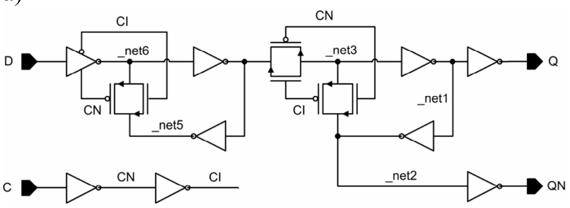
1) modify the fault table taking into account the extended sets of detected faults of all complex sequences;
2) identify such faults, which are detectable by only single sequences, and include those necessary sequences in the final solution;
3) exclude the set of faults detected by the selected sequences from the sets of the remaining sequences and recalculate their effectiveness;
4) select the sequence with the highest effectiveness and include it in the solution set;
5) repeat the last two steps until there are still uncovered faults;
6) construct the optimal complex test sequence from the sequences in the solution set.

The length of the final test sequence could be usually limited due to the partial covering of the building sequences, as shown in Section 3.

3. Example of characterization results

The proposed methodology was used for characterization of a number of flip-flop cells from two industrial standard cell libraries in 0.8 µm and 0.35 µm CMOS technologies. A basic D flip-flop cell (DF1 cell in 0.35 µm CMOS) was selected to illustrate the process of characterization. The rest of results were omitted here due to the lack of space.

a)



b)

C	D	Q_{k-1}	QN_{k-1}	Q_k	QN_k
↑	0	0	1	0	1
↑	1	0	1	1	0
↑	1	1	0	1	0
↑	0	1	0	0	1

Fig. 2. DF1 cell: a) logic diagram, b) truth table.

Fig. 2 shows a logic diagram of D flip-flop and its truth table. There are 91 possible defects of non-zero probabilities. In Tab. 1 a selection of 38 defects are presented together with the fault table for the Q output. The first three columns address the shorts and show their probabilities. In the next column the logic faults caused by the defects are described. A symbol FTT represents faulty operation of the flip-flop which is difficult to describe briefly and therefore is given by the faulty truth table (not presented in the paper). The last sixteen columns show the simple test sequences and mark the defects which were detected by these sequences. There are still nine possible defects that can not be detected with voltage testing. The effectiveness of test sequences was given in Tabs. 2 and 3, for simple and complex sequences respectively (the fault table modified for complex sequences was omitted). Now we can solve the covering problem. The solution set could be formed by two complex sequences:

$$S_{16}(CD) = \{01\ \mathbf{11\ 10\ 00\ 10}\},\quad S_8(CD) = \{00\ \mathbf{10\ 00\ 01}\}.$$

They let us construct the optimum input sequence:

$$S = \{01\ \mathbf{11\ 10\ 00\ 10\ 00\ 01}\},$$

which covers all detectable faults.

It is important to notice that the obtained sequence includes only two clock pulses. Nevertheless, monitoring of the output voltage should be extended in time. As the existing testing equipment allows to strobe the outputs for a period of time [21], the fault tables presented in this paper seem to become useful in determining the limits of this period. They could also help to prepare the appropriate input waveform.

4. Conclusions

In this paper a characterization methodology of CMOS sequential standard cells for defect based voltage testing was presented. It extends an approach used already for combinational cells [13]-[15]. The proposed methodology applies analysis of electrical schematic topology and layout of characterized circuits. It allows to estimate the probabilities of physical defects that may occur in a real integrated circuit and to determine types of logic faults caused by those defects. As a result, a set of faulty truth tables describing faulty operation of a given standard cell is obtained for all considered bridging faults with non-zero probabilities of occurrence. In addition, for each particular fault, the procedure generates a set of test vector sequences, which detect this fault. A fault table collects all found sequences for all considered faults in a cell. The sequences are then validated to determine their effectiveness in fault covering. The proposed algorithm allows to construct the optimal complex test sequence for all detectable faults in a given standard cell. The paper presents the local solution for single sequential standard cells, which form the main building blocks of bigger sequential circuits. Therefore, the presented results are considered to be useful in hierarchical approach to ATPG for sequential circuits.

5. Acknowledgment

The authors are grateful for Prof. Raimund Ubar and Dr. Jaan Raik from Tallinn University of Technology for their valuable feedback. The authors would like also to thank Mr. Tomasz Borejko, Mr. Piotr Borkowski, Mr. Michał Rakowski and Mr. Karol Szacki from Warsaw University of Technology for their help in simulations.
This work was supported in part by the Polish State Committee for Scientific Research under project No. 4 T11B 023 24.

Table1. Probabilities of defects in DF1 cell (D flip-flop) and the fault table for Q output (38 out of 91 defects, Wired-AND conditions)

No.	Short defect	Probability	Logic fault at Q	Simple test sequence <*CD*>															
				00 10	01 11	00 10 00	00 10 11	01 11 01	01 11 10	00 10 00 10	00 10 00 01	00 10 11 01	01 11 01 11	01 11 01 00	01 11 10 00	00 10 00 01 11	00 10 11 01 11	01 11 01 00 10	01 11 10 00 10
1	C/D	1.55E-10	SA0 after switching to 0		1			1	1				1	1	1	1	1		
2	C/Q	5.27E-12	Q=C	1			1	1		1				1	1			1	1
3	C/QN	3.22E-12	Not detected	None															
4	D/Q	1.20E-11	Q=D				1		1		1	1		1	1				
5	D/QN	7.86E-12	Not detected	None															
6	Q/QN	1.48E-10	SA0		1			1	1				1	1	1	1	1		
7	C/CN	5.94E-10	Active falling edge of C									1			1	1		1	
8	C/CI	2.08E-10	Not detected	None															
9	C/_net1	2.84E-11	Q=not(C)		1	1			1		1	1	1			1	1		
10	C/_net2	1.42E-11	Active both edges of C					1						1	1				
11	C/_net3	2.24E-11	Q=C	1			1	1		1				1	1			1	1
12	C/_net4	7.80E-11	Q=C	1			1	1		1				1	1			1	1
13	C/_net5	3.61E-11	SA0		1			1	1				1	1	1	1	1		
14	C/_net6	6.71E-11	SA0		1			1	1				1	1	1	1	1		
15	CN/D	7.10E-10	SA1	1		1	1			1	1	1						1	1
16	CI/D	9.59E-10	SA1 after switching to 1			1				1	1							1	1
17	D/_net1	6.26E-11	Q=not(D)	1	1	1		1		1			1			1	1	1	1
18	D/_net2	3.11E-11	FTT								1	1		1	1				
19	D/_net3	4.98E-11	Q=D				1		1		1	1		1	1				
20	D/_net4	2.93E-10	FTT				1		1			1			1				
21	D/_net5	9.26E-11	FTT	1	1	1		1		1	1		1	1		1	1	1	1
22	D/_net6	1.92E-10	FTT	1	1	1		1		1	1		1	1		1	1	1	1
23	CN/Q	7.36E-12	SA1	1		1	1			1	1	1						1	1
24	CI/Q	9.66E-12	SA1 after switching to 1			1				1	1							1	1
25	Q/_net1	2.56E-10	Q=QN	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
26	Q/_net2	1.49E-10	Not detected	None															
27	Q/_net3	9.46E-11	SA0		1			1	1				1	1	1	1	1		
28	Q/_net4	3.23E-11	SA0		1			1	1				1	1	1	1	1		
29	Q/_net5	2.40E-11	Q=not(CD)		1	1		1	1				1			1	1		
30	Q/_net6	2.12E-11	Oscillations		1						1		1			1	1		
31	CN/QN	4.55E-12	SA0		1			1	1				1	1	1	1	1		
32	CI/QN	6.23E-12	SA1 after switching to 1			1				1	1							1	1
33	QN/_net1	6.75E-11	FTT			1				1	1							1	1
34	QN/_net2	6.28E-11	Not detected	None															
35	QN/_net3	4.85E-11	FTT					1						1	1				
36	QN/_net4	2.12E-11	SA0		1			1	1				1	1	1	1	1		
37	QN/_net5	1.57E-11	Not detected	None															
38	CI/CN	1.23E-09	FTT												1			1	

Table 2. Effectiveness of simple test sequences for DF1 cell

Test sequence <CD>	Number of detected shorts	Effectiveness
S_1 = 00 10	29	5.24E-09
S_2 = 01 11	35	4.64E-09
S_3 = 00 10 00	40	9.19E-09
S_4 = 00 10 11	32	6.50E-09
S_5 = 01 11 01	42	5.25E-09
S_6 = 01 11 10	33	4.56E-09
S_7 = 00 10 00 10	41	8.88E-09
S_8 = 00 10 00 01	41	9.06E-09
S_9 = 00 10 11 01	30	6.68E-09
S_{10} = 01 11 01 11	37	5.43E-09
S_{11} = 01 11 01 00	43	5.22E-09
S_{12} = 01 11 10 00	41	6.21E-09
S_{13} = 00 10 00 01 11	36	5.23E-09
S_{14} = 00 10 11 01 11	37	5.43E-09
S_{15} = 01 11 01 00 10	41	9.93E-09
S_{16} = 01 11 10 00 10	41	8.88E-09

Table 3. Effectiveness of complex test sequences for DF1 cell

Test sequence <CD>	Number of detected shorts	Effectiveness
S_1 = 00 **10**	29	5.24E-09
S_2 = 01 **11**	35	4.64E-09
S_3 = 00 **10 00**	45	9.50E-09
S_4 = 00 **10 11**	35	6.84E-09
S_5 = 01 **11 01**	45	6.16E-09
S_6 = 01 **11 10**	40	5.18E-09
S_7 = 00 **10 00 10**	45	9.50E-09
S_8 = 00 **10 00 01**	51	9.72E-09
S_9 = 00 **10 11 01**	43	8.39E-09
S_{10} = 01 **11 01 11**	45	6.16E-09
S_{11} = 01 **11 01 00**	50	6.42E-09
S_{12} = 01 **11 10 00**	51	7.79E-09
S_{13} = 00 **10 00 01 11**	76	1.32E-08
S_{14} = 00 **10 11 01 11**	69	1.13E-08
S_{15} = 01 **11 01 00 10**	78	1.43E-08
S_{16} = 01 **11 10 00 10**	79	1.46E-08

6. References

[1] T. E. Marchok, A. El-Maleh, W. Maly, J. Rajski, "Complexity Analysis of Sequential ATPG," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol.15, no.11, pp. 1409-1423, Nov. 1996.

[2] Y. Ch. Kim, K. K. Saluja, "Sequential test generators: past, present and future," *Integration, the VLSI Journal*, Volume 26 , Issue 1-2 , pp. 41–54, December 1998.

[3] Kwang-Ting Cheng, "Gate-level test generation for sequential circuits," *ACM Trans. on Design Automation of Electronic Systems (TODAES)*, v.1 n.4, p.405-442, Oct. 1996.

[4] I. Hamzaoglu , J. H. Patel, "Deterministic test pattern generation techniques for sequential circuits," *Proc. of the 2000 IEEE/ACM International Conference on Computer-Aided Design*, November 2000, San Jose, California.

[5] J. Raik, R. Ubar, "Fast Test Pattern Generation for Sequential Circuits Using Decision Diagram Representations," *Journal of Electronic Testing: Theory and Applications* 16, 213–226, 2000.

[6] J. Raik, and R. Ubar. "Sequential Circuit Test Generation Using Decision Diagram Models," *Proc. DATE Conf.*, 1999, pp. 736-740.

[7] P. Nigh, W. Maly, "Layout-driven test generation," *Proc. Int. Conf. on Computer-Aided Design*, pp. 154-157, USA, Nov. 1989.

[8] T. M. Storey, W. Maly, "CMOS bridging fault detection," *Proc. Int. Test Conf.*, pp. 842-851, USA, Sep. 1990.

[9] J. P. Shen, W. Maly, F. J. Ferguson, "Inductive fault analysis of MOS integrated circuits," *IEEE Design & Test*, pp. 13-26, Dec. 1985.

[10] F. J. Ferguson, J. P. Shen, "Extraction and simulation of realistic CMOS faults using inductive fault analysis," *Proc. Int. Test Conf.*, pp. 475-484, USA, 1988.

[11] S. W. Bollinger, S.F. Midkiff, " Test generation for IDDQ testing of bridging faults in CMOS circuits," *IEEE Trans. Computer-Aided Design*, vol. 13, no. 11, pp. 1413-1418, Nov. 1994.

[12] J. Khare, W. Maly, N. Tiday, "Fault characterization of standard cell libraries using inductive contamination analysis (ICA)," *Proc. VLSI Test Symp.*, pp. 405-413, USA, April 1996.

[13] W. A. Pleskacz, D. Kasprowicz, T. Oleszczak, W. Kuźmicz, "CMOS Standard Cells Characterization for Defect Based Testing," *Proc. IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems - DFT01*, pp. 384-392, San Francisco, USA, October 2001.

[14] W. A. Pleskacz, T. Borejko, W. Kuźmicz, "CMOS Standard Cells Characterization for IDDQ Testing," *Proc. IEEE Intl. Symp. on Defect and Fault Tolerance in VLSI Systems*, pp. 390-398, Vancouver, Canada, Nov. 2002.

[15] D. Kasprowicz, W. A. Pleskacz, "Improvement of integrated circuit testing reliability by using the defect based approach," *Microelectronics Reliability (PERGAMON – Elsevier Science)*, vol. 43/6, pp. 945-953, June 2003.

[16] T. Cibakova, M. Fischerowa, E. Gramatova, W. Kuzmicz, W. Pleskacz, J. Raik, R. Ubar, "Hierarchical test generation for combinational circuits with real defects coverage," *Microelectronics Reliability (PERGAMON – Elsevier Science)*, vol. 42/7, pp. 1141-1149, July 2002.

[17] J. Raik, R. Ubar, J. Sudbrock, W. Kuzmicz, W. Pleskacz, "Deterministic Defect-Oriented Test Generation for Combinational Circuits," *Proc. of the 6th IEEE Latin-American Test Workshop*, pp. 325-330, Salvador, Bahia, Brazil, March-April 2005.

[18] J. Raik, R. Ubar, J. Sudbrock, W. Kuzmicz, W. Pleskacz, "DOT: New Deterministic Defect-Oriented ATPG Tool," *Proc. of the 10th IEEE European Test Symposium ETS'05*, pp. 96-101, Tallinn, Estonia, May 2005.

[19] C. Metra, M. Favalli, P. Olivo, B. Ricco, "Testing of resistive bridging faults in CMOS flip-flop," *Proc. European Test Conference* ETC' 93, pp. 530-531, 1993.

[20] M. Sachdev, "Testing Defects in Scan Chains," *IEEE Design & Test of Computers*, vol. 12, no. 4, pp. 45-51, 1995

[21] A. Miczo, "Digital logic testing and simulation," John Wiley & Sons, Inc., 2003