A Multipath Detection Scheme Using SAT

Mohamed El-Tarhuni
Department of Electrical Engineering
American University of Sharjah, UAE
mtarhuni@aus.edu

Fadi A. Aloul
Department of Computer Science & Engineering
American University of Sharjah, UAE
faloul@aus.edu

Abstract — **This paper presents a new technique for multipath detection in wideband mobile radio systems. An intelligent search algorithm using Boolean Satisfiability (SAT) techniques is used to search for the multipath delays. The proposed algorithm finds the multipath components with high probability and hence allows the system to utilize the inherent diversity in these components to mitigate channel fading. Simulation results indicate that the system can achieve 90% detection or higher of all paths at low signal-to-noise ratio under frequency-selective Rayleigh fading conditions.**

Keywords: *Multipath detection; PN code acquisition; SAT.*

I. INTRODUCTION

Multipath propagation is one of the main challenges in mobile radio communication systems caused by reflection, refraction, and scattering of radio waves as they pass through the wireless channel. Multipath propagation results in receiving multiple copies of the transmitted signal. In narrow band transmission schemes, multipath causes intersymbol interference (ISI) that degrades the bit error rate (BER) performance of the system. On the other hand, if wideband signal transmission is used, such as spread spectrum (SS) or ultra wideband (UWB), multipath propagation can be exploited to improve the system BER performance through the diversity gain from the different copies of the received signal. However, for full utilization of the multipath scenario, it is very important for the receiver to first detect the presence of these multipath components and identify their corresponding parameters (time delay, amplitude, and phase).

In spread spectrum systems, a pseudo random (PN) code is used to spread the message spectrum over a wide bandwidth. At the receiving end, a time-synchronized version of the same PN code is used to de-spread the signal and recover the original message [12]. Synchronization is very crucial for the proper operation of the system. It can be accomplished by searching a range of delays for the correct multipath delays. The uncertainty range represents the possible delays that the signal may have and is related to the channel memory. The delay range is usually specified as cells that are one-chip or one-half of a chip apart. The search of these cells, i.e. finding the cells that have strong energy and hence multipath components, can either be done in a serial or parallel fashion [11, 14, 15, 16].

In serial search, one cell at a time is tested by measuring the signal energy at that cell using a single correlator circuit. If the energy exceeds a preset threshold then the cell is declared as a multipath cell while if the energy is below the threshold then it is declared as a no multipath cell. The search advances to the next cell and the process is continued until all cells in the uncertainty range are tested. The other search strategy uses parallel search where the energies of all cells are calculated simultaneously using a bank of parallel correlators and cells with energy above the threshold are declared as multipath cells. Apparently, serial search is slower compared to parallel search at it takes longer time to search all the cells and find the delays. On the other hand, serial search has a much lower reduced complexity (both hardware and processing).

A common drawback of existing schemes is that in searching for the correct cells they don't utilize the inherent structure of the PN code. These schemes need to search all possible cells in the search window, which could be as large as the length of the PN code, in order to find the correct cells. For example, for a PN code with a length of 2047 chips (generated by 11-stage shift register) the serial and parallel search schemes need to test 2047 cells if the search step is one chip or twice of that if the search step is one-half of a chip. This testing may need to be repeated many times if the multipath components were not detected at the first trial due to noise and fading. In this paper, we propose a PN code acquisition scheme that exploits the structure of the PN code to reduce the number of decisions needed to find correct cells. The proposed scheme is based on using Boolean Satisfiability (SAT) solving to perform intelligent search of the uncertainty region and hence reduce the number of decisions needed to find the correct cells significantly. This is done by searching only PN code phases that result in minimum difference (minimum distance) between the PN code in the received signal and a locally generated PN code.

Recently, Boolean Satisfiability (SAT) have been shown to be very successful in solving complex problems in various Engineering and Computer Science applications. Such applications include: FPGA routing [9] and Power Optimization [3, 10]. SAT has also been extended to a variety of applications in Artificial Intelligence including other well known NP-complete problems such as graph colorability, vertex cover, hamiltonian path, and independent sets [6]. Despite SAT being an NP-Complete problem [5], many researchers have developed powerful SAT solvers that are able of handling problems consisting of thousands of variables and millions of constraints [2, 4, 7, 8]. Briefly defined, the SAT problem consists of a set of Boolean variables and a set of constraints expressed in product-of-sum form. The goal is to identify an assignment to the variables that would satisfy all constraints or prove that no such assignment exists.

Even though in recent years we have seen a surge in the application of SAT techniques to assist in finding solutions to various Engineering problems, very few researchers reported on the use of SAT-based techniques in mobile communication related research. The formulation of the PN acquisition as a SAT instance to develop the search strategy is described in details in this paper. Simulation results indicate that the proposed approach is *complete* and is guaranteed to identify the optimal solution (correct cells).

The reminder of this paper is organized as follows. Sections II and III present the signal model and an overview of SAT, respectively. Section IV describes the proposed scheme and shows how to formulate the PN code acquisition problem as a SAT instance. Simulation results are presented and discussed in Section V. Finally, the paper is concluded in Section VI.

II. System Description

A direct-sequence code division multiple access (DS-CDMA) system is investigated in this paper. We assume that a separate pilot signal is available for use in delay estimation. The transmitted pilot signal from the desired user is given by:

$$s(t) = \sqrt{P}\sum_{i}\sum_{k=0}^{N-1} c_k g(t - iT_b - kT_c) \qquad (1)$$

where P is the transmitted power, c_k is the spreading pseudo random (PN) code of the desired user of ± 1, N is the PN code length which is the same as the number of chips per bit, i.e. $N = T_b / T_c$, T_b is the bit duration, T_c is the chip duration, and $g(t)$ is the chip pulse shape assumed to be a rectangular pulse with duration of T_c.

The radio channel is modeled as a frequency-selective Rayleigh fading channel using a time-varying Finite Impulse Response (FIR) filter. The received signal is given by

$$u(t) = \sum_{l=1}^{L} \beta_l s(t - \tau_l) + n(t) \qquad (2)$$

where L is the number of paths, β_l is the l^{th} path complex coefficient with Rayleigh amplitude and uniform phase distribution over $[0, 2\pi)$, τ_l is the l^{th} path delay that we would like to estimate, and $n(t)$ is an additive white Gaussian noise (AWGN) with zero mean and two-sided power spectral density $N_0/2$ that models the effect of the receiver noise. It is assumed that the number of users is relatively large such that the interference can be modeled as part of the AWGN.

To maximize the signal-to-noise ratio, the received baseband signal is first applied to a chip-matched filter to produce the following signal

$$z[k] = \int_{(k-1)T_c}^{kT_c} u(t)g(t)dt \qquad (3)$$

In conventional acquisition schemes, the output of the chip-matched filter is correlated with a locally generated PN code with different offsets that cover the delay uncertainty region (possibly the whole PN code period) as follows

$$Y[i] = \sum_{k=0}^{N-1} z[k-i]c_k, \qquad i = 0, 1, \ldots, N-1 \qquad (4)$$

where the index i indicates the delay offset. The correlation results in (4) are used to estimate the energy at different delay offsets and a decision is made on the multipath delays based on the highest energy values. It is also common to use a preset threshold where only energy values that exceed the threshold are declared as correct multipath components while others are ignored.

The main objective of the acquisition system is to maximize the probability of detection while minimizing the probability of false alarm. It is also important to minimize the resources needed to achieve PN code acquisition including both hardware and processing dimensions.

III. Boolean Satisfiability

Boolean Satisfiability (SAT) is often used as the underlying model in the field of computer aided designs of integrated circuits. A number of SAT solvers have been proposed and implemented [2, 4, 7, 8]. These solvers employ powerful algorithms that are sufficiently efficient to deal with large-scale SAT problems that typically arise in the Engineering domain. Most of these algorithms claim competitive results in runtime efficiency and robustness.

In SAT, given a formula *f*, the objective is to identify an assignment to a set of Boolean variables that will satisfy a set of constraints. If an assignment is found, it is known as a satisfying assignment, and the formula is called *satisfiable*. Otherwise if an assignment doesn't exist, the formula is called *unsatisfiable*. The constraints are typically expressed in conjunctive normal form (CNF). In CNF, the formula consists of the conjunction (AND) of *m* clauses $\omega_1, \ldots, \omega_m$ each of which consists of the disjunction (OR) of *k* literals. A literal *l* is an occurrence of a Boolean variable or its complement. Hence, in order to satisfy a formula, each of its clauses must have at least one literal evaluated to true.

As an example, a CNF instance $f(a, b, c) = (a + \bar{b}) \cdot (a + b + c)$ consists of 3 variables, 2 clauses, and 5 literals. The assignment $\{a = 0, b = 1, c = 0\}$ leads to a conflict, whereas the assignment $\{a = 0, b = 0, c = 1\}$ satisfies *f*.

Despite the problem being NP-Complete, there have been dramatic improvements in SAT solver technology over the past decade. This has lead to the development of several powerful SAT solvers that are capable of solving problems consisting of thousands of variables and millions of constraints in a few seconds [2, 4, 7, 8].

Recently, SAT solvers [2, 4, 7] have been extended to handle pseudo-Boolean (PB) constraints which are linear inequalities with integer coefficients that can be expressed in the normalized form [2] of:

$$a_1x_1 + a_2x_2 + \ldots + a_nx_n \geq b \qquad (5)$$

where $a_i, b \in Z$ and x_i are Boolean variables. PB constraints can, in some cases, replace an exponential number of CNF constraints. They have been found to be very efficient in expressing "counting constraints" [2]. Furthermore, PB extends SAT solvers to handle *optimization* problems as opposed to only *decision* problems. Subject to a given set of CNF and PB constraints, one can request the minimization (or maximization) of an objective function which consists of a linear combination of the problem's variables. Note that each CNF constraint can be viewed as a PB constraint. For example the CNF constraint $(a + \bar{b})$ can be viewed as the PB constraint $a + \bar{b} \geq 1$. PB constraints represent 0-1 integer linear programming (ILP) inequalities and has introduced many new applications to the SAT domain. Recent studies

Shift Register (SR) bits: *m bits*

depth : $d = 2^m - 1$

S_{11}	S_{21}	S_{31}	$\cdots$	S_{m1}
S_{12}	S_{22}	S_{32}	$\cdots$	S_{m2}
S_{13}	S_{23}	S_{33}	$\cdots$	S_{m3}
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
S_{1d}	S_{2d}	S_{3d}	$\cdots$	S_{md}

Fig. 1. Sample layout of Shift Register bits.

have shown that SAT-based optimization solvers can in fact compete with the best generic ILP solvers [2, 4, 7].

Note that SAT solvers can be implemented in hardware. Several studies proposed the use of FPGA reconfigurable systems to solve SAT problems [1, 13, 17]. Hardware solvers could be a standalone or as an accelerator where the problem is partitioned between the hardware solver and the attached computer using software.

IV. SAT Model for PN Code Acquisition

A hard decision is made on the matched filter output in (3) to get a binary sequence of data, z_b, that represent an estimate of the PN code in the received signal as follows

$$z_b[k] = \begin{cases} 1 & z[k] \geq 0 \\ 0 & z[k] < 0 \end{cases} \tag{6}$$

Although hard decisions are in general not sufficient statistics for estimating the delay, but in the context of the developed SAT model for PN acquisition it would be enough to provide an estimate of the received PN code and hence allows for the SAT search to be implemented as will be discussed later.

In this paper, we are interested in using advanced SAT solvers to find the L multipath delays in the received signal. To illustrate our approach, let us assume a system consisting of *n* data bits, and *m* Shift Register (SR) bits. The depth *d* is equal to $(2^m - 1)$ levels as shown in Figure 1. The objective is to:

- Find all solutions that satisfy the presented constraints. Note that a solution means the initial content of the SR that would result in the same PN code once clocked at the chip rate. Since there are L paths then there will be L initial states each corresponding to one path that would create the PN code by the SR.
- For each solution we compute the number of chips in which the matched filter output z_b is different from the locally generated PN code.
- Select the L solutions that have the smallest difference (error) as the shift register states the correspond to the L delay offset estimates.

Three sets of *variables* are defined for the problem:

- A Boolean variable C_i is defined for each data bit *i*. That is a total of *n* variables. A value of 1 (0) for each variable indicates that the corresponding bit is a 1 (0) in the original source.
- A Boolean variable Q_i is defined for each data bit as the difference between the C_i and the PN code chip. That is a total of *n* variables.
- A Boolean variable S_{ij} is defined for each SR bit *i* at each level *j*. That is a total of $m \times (2^m - 1)$ variables.

Logical Constraint	CNF Constraint
$(x = y)$	$(\bar{x} \vee y) \cdot (x \vee \bar{y})$
$(x = y \oplus z)$	$(\bar{x} \vee y \vee z) \cdot (x \vee \bar{y} \vee z) \cdot$ $(x \vee y \vee \bar{z}) \cdot (\bar{x} \vee \bar{y} \vee \bar{z})$

Table 1. Expressing logical constraints using CNF constraints.

The total number of needed Boolean variables is equal to $2n + m(2^m - 1)$.

The following set of *constraints* are generated:

- **Data Bit Constraints:** For each data bit *i*, its corresponding C_i bit is set to 0 or 1 depending on the feeded data. This can be expressed using a single PB constraint per data bit as follows:

$$C_i = v \quad ; v \in 0, 1 \text{ based on input value}; \ i = 1, \ldots n \tag{7}$$

  That is a total of *n* PB constraints.

- **Initial State Constraints:** The initial SR bits should have at least one bit assigned to 1 to avoid having an all-zero state for the SR. This can be expressed using a *single* PB constraint as follows:

$$\left(\sum_{i=1}^{m} S_{i1} \right) > 0 \tag{8}$$

- **Shifting Constraints:** The shifting within the shift register relation, e.g. $S_{22} = S_{11}, S_{32} = S_{21}, \ldots$, is expressed using the following equality constraint per SR bit:

$$(S_{il} = S_{(i-1)(l-1)}) \quad ; \ l = 2, \ldots, d; \ i = 2, \ldots, m \tag{9}$$

  This results in a total of $(m-1)(2^m - 2)$ equality constraints. Each equality constraint of format $(x = y)$ can be expressed using two CNF constraints as shown in Table 1.

- **Feedback Constraints:** The PN code feedback relation is expressed using the following XOR constraint per initial SR bit:

$$[S_{1l} = S_{p(l-1)} \oplus \ldots \oplus S_{q(l-1)}] \quad ; \ l = 2, \ldots, d \tag{10}$$

  where $p, q \in \{1, \ldots, n\}$ according to the feedback connection of the PN code generator. This results in a total of $(2^m - 2)$ XOR constraints. Each XOR constraint of format $(x = y \oplus z)$ is expressed using four CNF constraints as shown in Table 1.

- **Difference Constraints:** The mismatch between the data and SR bits is expressed as follows:

$$[Q_i = S_{(m)(((i+d-1)mod(d))+1)} \oplus C_i] \ ; \ i = 1, \ldots, n \tag{11}$$

  This results in *n* XOR constraints. As mentioned earlier each XOR constraint can be expressed using four CNF constraints.

Data Input Bits: 0, 1, 0, 0, 1, 0, 0, 0

Constraints:

$C_1 = 0 \quad C_5 = 1$

$C_2 = 1 \quad C_6 = 0$

$C_3 = 0 \quad C_7 = 0$

$C_4 = 0 \quad C_8 = 0$

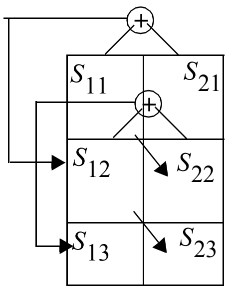
$S_{22} = S_{11}$

$S_{23} = S_{12}$

$S_{12} = S_{11} \oplus S_{21}$

$S_{13} = S_{12} \oplus S_{22}$

$S_{11} + S_{21} > 0$



$Q_1 = S_{21} \oplus C_1$

$Q_2 = S_{22} \oplus C_2$

$Q_3 = S_{23} \oplus C_3$

$Q_4 = S_{21} \oplus C_4$

$Q_5 = S_{22} \oplus C_5$

$Q_6 = S_{23} \oplus C_6$

$Q_7 = S_{21} \oplus C_7$

$Q_8 = S_{22} \oplus C_8$

$min(Q_1 + Q_2 + \ldots + Q_8)$

Fig. 2. An example of a network with 8 data bits and 2 SR bits.

- **Optimization Function:** The final optimization goal is to minimize the sum of Qs for every possible SR initial state. This is expressed using the following PB optimization objective:

$$Min\left(\sum_{i=1}^{n} Q_i\right) \tag{12}$$

4.1 *Illustrative Example*

To further illustrate the formulation in SAT input, lets consider the example in Figure 2 with 8 data bits and 2-stage SR. Hence, the depth d is 3. The SAT problem generates a total of $2 \times 8 + 2(2^2 - 1) = 22$ Boolean variables. The figure displays the needed constraints.

V. Simulation Results

In this paper, we simulated a spread spectrum system with a PN code of length 2047 (11-stage shift register) operating over a three-path Rayleigh fading channel with uniform power delay profile and a normalized Doppler of 10^{-3}. The performance is measured by the probability of detecting one, two, or three multipath components as a function of the signal-to-noise ratio per chip (SNRc). The effect of the duration of the correlation period used in calculating the difference between the locally generated PN code and the received data on the detection probability is also investigated with three values for the correlation period 500, 1000, and 10240 chips. All experiments were performed on an Intel Xeon 3.2 Ghz workstation with 4 GB of RAM. We used the PBS 0-1 SAT-based ILP solver [2] for all experiments.

Figure 3 shows the performance of the SAT based acquisition scheme in AWGN (no multipath). The results indicate that the correct delay can be detected with high accuracy even at very low SNRc. For instance, 90% detection can be achieved with a SNRc of -10, -12, and -25 dB for a correlation period of 500, 1000, and 10240 chips, respectively. Note that performance is improved by increasing the correlation period but at the expense of longer execution time as shown in Figure 4.

The probability of detecting one out of the three paths in a frequency-selective Rayleigh fading channel is shown Figure 5. It is noted that fading causes a degradation of 3 to 6 dB in per-

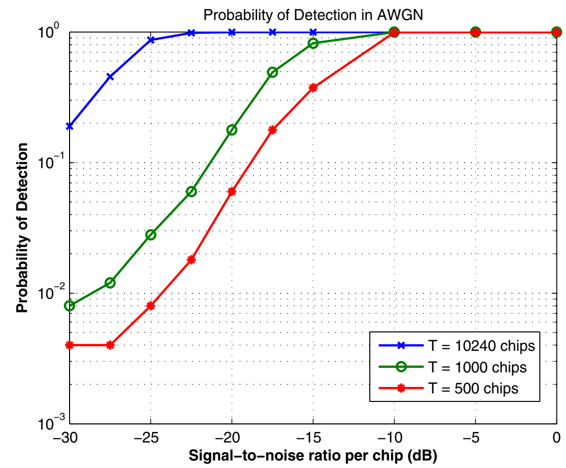


Fig. 3. Probability of detection in AWGN.

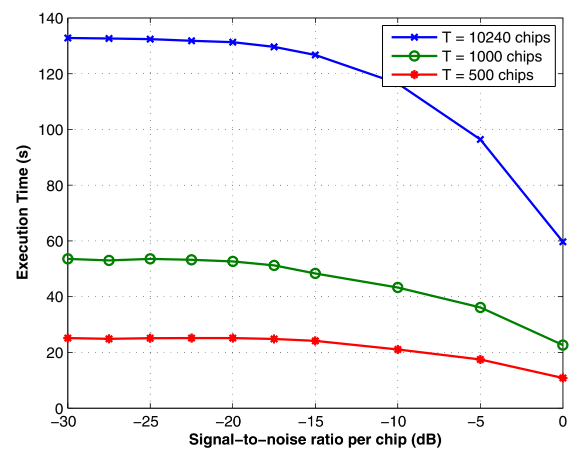


Fig. 4. Search time in AWGN.

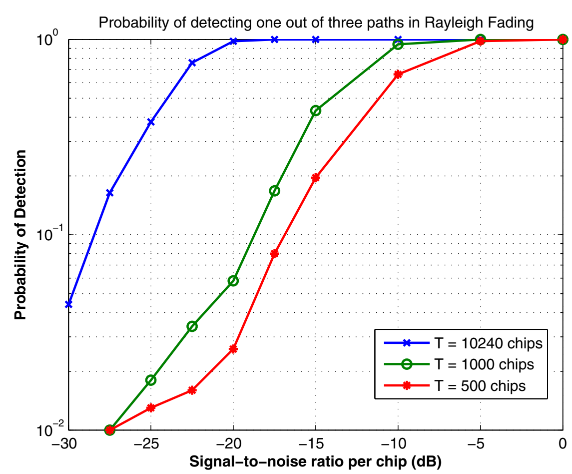


Fig. 5. Probability of detecting one out of three paths.

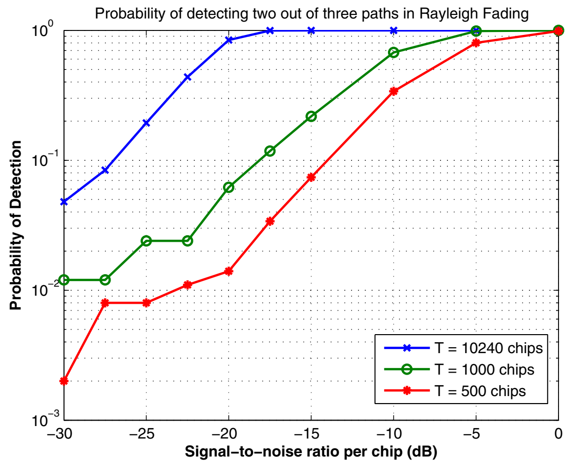


Fig. 6. Probability of detecting two out of three paths.

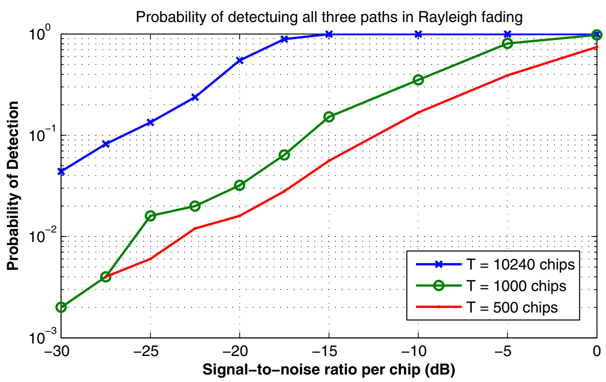


Fig. 7. Probability of detecting all three paths.

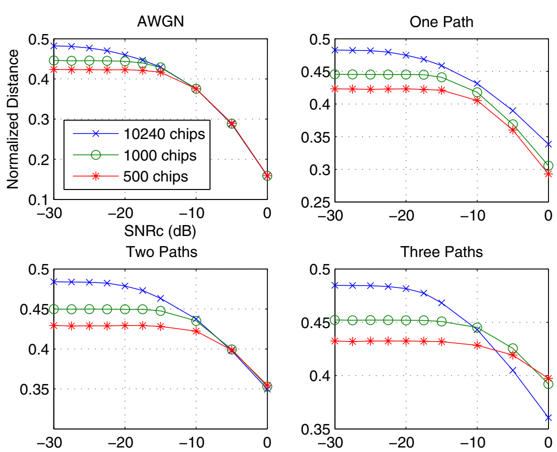


Fig. 8. Normalized distance verses SNRc.

formance compared to the AWGN case. However, this is attributed to the reduction in the path energy as the signal is distributed among the three paths. Again, using a longer correlation period results in better performance. The probability of detecting two out of the three paths is shown in Figure 6, while the probability of detecting all three paths is shown in Figure 7. We notice that the system can detect all paths with high accuracy and hence provide good diversity to the receiver.

The SAT-based algorithm searched for the possible states that match the received signal with the PN code and the states that result in minimum difference are used to find the delay estimate. Figure 8 shows the minimum distance found at different cases normalized by the corresponding correlation period (500, 1000, or 10240 chips). The difference tends to decrease as the SNRc increases because the SAT algorithm is supplied with more reliable data for the search. It is noticed that all multipath cases have similar values for the normalized distance that are worse than those of the AWGN case.

VI. Conclusions

A new multipath detection algorithm using Boolean satisfiability (SAT) techniques has been presented. The SAT-based algorithm uses the deterministic structure of the PN spreading code to perform an intelligent search for the possible propagation delays. Simulation results showed that the proposed scheme was successful in providing correct delay estimates with high reliability over a multipath frequency-selective Rayleigh channel.

References

[1] M. Abramovici, and D. Saab "Satisfiability on Reconfigurable Hardware," in *Proc. of the Int'l Workshop on Field Programmable Logic and Application*, 448-456, 1997.

[2] F. Aloul, A. Ramani, I. Markov, and K. Sakallah, "Generic ILP Versus Specialized 0-1 ILP: An Update," in *Proc. of* ICCAD, 450-457, 2002.

[3] F. Aloul, S. Hassoun, K. Sakallah, and D. Blaauw, "Robust SAT-Based Search Algorithm for Leakage Power Reduction," in *Proc. of* PATMOS, 167-177, 2002.

[4] D. Chai and A. Kuehlmann, "A Fast Pseudo-Boolean Constraint Solver," in *Proc. of Design Automation Conference* (DAC), 830-835, 2003.

[5] S. A. Cook, "The Complexity of Theorem Proving Procedures," in *Proc. of the Symp. on the Theory of Computing*, 151-158, 2004.

[6] N. Creignou, S. Kanna, and M. Sudan, "Complexity Classifications of Boolean Constraint Satisfaction Problems," *SIAM Press*, 2001.

[7] N. Een and N. Sorensson, "An Extensible SAT-solver," in *Proc. of* SAT, 502-508, 2003.

[8] K. Moskewicz, C. Madigan, Y. Zho, and S. malik "Chaff: Engineering an efficient SAT solver," in *Proc. of* DAC, 503-535, 2001.

[9] G. Nam, F. A. Aloul, K. A. Sakallah, and R. Rutenbar, "A Comparative Study of Two Boolean Formulations of FPGA Detailed Routing Constraints," in *IEEE Trans. on Computers*, 53(6), 688-696, 2004.

[10] A. Sagahyroon and F. Aloul, "Using SAT-Based Techniques in Power Estimation," in *Microelectronics Journal*, 38(6-7), 706-715, 2007.

[11] O.-S. Shin and K.B. E. Lee, "Utilization of Multipaths for spread-pectrum Code Acquisition in Frequency-selective Rayleigh Fading Channels," in *IEEE Trans. Comm.*, v. 49, 734-743, April 2001.

[12] M. Simon, J. Omura, R. Scholtz and B. Levitt, "Spread Spectrum Communications," *McGraw-Hill Inc.*, New York, 1994.

[13] I. Skliarova and A. Ferrari "A Software/Reconfigurable Hardware SAT Solver," in *IEEE Trans. on VLSI*, 12(4), 408-419, April 2004.

[14] W. Suwansantisuk and Z. Win, "Multipath Aided Rapid Acquisition: Optimal Search Strategies," in *IEEE Trans. Information Theory*, v. 53, 174-193, January 2007.

[15] W. Suwansantisuk, Z. Win, and L. Shepp, "On the Performance of Wide-Bandwidth Signal Acquisition in Dense Multipath Channels," in *IEEE Trans. Vehicular Technology*, v. 54, 1584-1594, Sep. 2005.

[16] L.-L. Yang and L. Hanzo, "Serial Acquisition Performance of Single-Carrier and Multicarrier DS-CDMA over Nakagami-m fading Channels," in *IEEE Trans. Wireless Communications*, v.1, 692-702, 2002.

[17] P. Zhong, M. Martonosi, P. Ashar, and S. Malik, "Using Reconfigurable Computing to Accelerate Boolean Satisfiability," in *IEEE TCAD*, 18(6), 861-868, 1999.