FPGA Placement by *Thermodynamic Combinatorial Optimization*

Juan De Vicente[1], *Juan Lanchares*[2], *Román Hermida*[2]

(1) *E.T.S.I.A.N*
Arturo Soria, 287, Madrid (Spain)
email: juan.vicente@etsian.mde.es

(2) *Dpto. Arquitectura de Computadores y Automática.*
E.S. Informática. Univ. Complutense de Madrid (Spain)
email: {julandan,rhermida}@dacya.ucm.es

Abstract

In this paper, the placement problem on FPGAs is faced using Thermodynamic Combinatorial Optimization (TCO). TCO is a new combinatorial optimization method based on both Thermodynamics and Information Theory. In TCO two kinds of processes are considered: microstate and macrostate transformations. Applying the Shannon's definition of Entropy to microstate reversible transformations, a probability of acceptance based on Fermi-Dirac statistics is derived. On the other hand, applying thermodynamic laws to reversible macrostate transformations, an efficient annealing schedule is provided. TCO has been compared with Simulated Annealing (SA) on a set of benchmark circuits for the FPGA placement problem. TCO has achieved large time reductions with respect to SA, while providing interesting adaptive properties.

1. Introduction

Placement is one of the most time consuming tasks of integrated circuit physical design. The placement quality affects both the area and speed of circuits. In the case of FPGAs where wiring resources are strongly limited, placement becomes a key issue for routing success. Placement is an NP-complete problem, thus new challenges arise from the continuous growth in the number of logic elements contained into commercial FPGAs. Since the solution space grows exponentially with the size of the circuit, it is necessary to resort to efficient combinatorial optimization algorithms. *Simulated Annealing (SA)* is a combinatorial optimization method standing out for the quality of solution it provides in solving placement problems [1]. This algorithm simulates the annealing process performed in the industry to achieve the lowest energy ground state of solids. Thus, *SA* algorithm consists of heating the system at high temperature, then lowering the temperature slowly until no further changes occur. One of theoretical foundations of SA is that the system must be close to equilibrium throughout the annealing process. To achieve such goal the algorithm demands a careful adjustment of the annealing schedule parameters. Moreover, *SA* becomes a method whose best results depend upon the skills or experience of practitioners.

In this paper, we first review Simulated Annealing algorithm, highlighting several successful annealing schedules. Following, we present *Thermodynamic Combinatorial Optimization* (*TCO*). *TCO* is a combinatorial optimization method derived from Thermodynamics and Information Theory. In this method the optimization is performed by means of reversible processes meeting thermodynamic laws. A useful feature of reversible processes is precisely that their intermediate states are also equilibrium states. Moreover, a reversible process improves the efficiency of any other process performed irreversibly. In *TCO* two kinds of processes are considered: microstate and macrostate transformations. Applying Shannon's definition of Entropy to microstate reversible transformations, a probability of acceptance based on Fermi-Dirac statistics is derived. On the other hand, applying thermodynamic laws to reversible macrostate transformations, an efficient annealing schedule is provided.

TCO has been compared with a common *SA* implementation for a set of benchmark circuits in the FPGA placement problem. *TCO* has provided interesting time reduction with respect to *SA* while obtaining high quality solutions.

2. FPGA placement problem

FPGAs [2] are user-programmable integrated circuits providing flexibility and reconfiguration advantages for supporting the design and production of digital systems. An FPGA is basically composed of a bidimensional array of Configurable Logic Blocks (CLBs) to support the logic and storage elements of circuits, and programmable input/output blocks (IOBs) at the perimeter of the device to provide off-chip interconnections.

The FPGA placement problem consists of assigning a netlist of virtual CLBs and IOBs (circuit components) to specific CLBs and IOBs on the FPGA. Placement must be optimized so that the circuit can be routed with the available resources and signal delays meet timing constraint [3][4][5][6][7][8].

The quality of placements is ussualy estimated by the *Bounding Box (BB)* cost function applied to all nets. *BB* approximates the cost of routing a net by the perimeter of the rectangle that encloses it. Thereby, *BB* estimates the total wire length and hence reduces delays and congestion in the optimization process. A sligth variation of this cost function [9][10] has been chosen throughout the rest of this paper to compare the efficiency of the combinatorial optimization algorithms. The functional form of this cost function is

$$Cost = \sum_{n=1}^{N_{nets}} \frac{q(n)}{C_{av}} \left[bb_x(n) + bb_y(n) \right]$$

where bb_x and bb_y denote the horizontal and vertical spans of the bounding box for each net respectively, q(n) is a factor to compensate the underestimated area of multiterminal nets [11], and C_{av} is the average channel capacity (in tracks).

3. Placement by Simulated Annealing

Simulated Annealing [12] is based on the analogy that can be found between the resolution of combinatorial optimization problems and the problem of determining the lowest-energy ground state of solids. A similar process to the annealing of solids can be simulated in solving combinatorial optimization problems. In this case, solutions of the combinatorial optimization problem replace the physical states of the system, while cost of solutions so does with energy states. From these equivalences, three operators or functions simulate the annealing process:

1) An operator for generating a new solution by a local transformation from the current one. This operator must be designed in such a way that the solution space is efficiently sampled (e.g. narrowing the magnitude of transformations as the search progresses may improve the performance).

2) A probability of accepting a new solution given by

$$P = \begin{cases} 1 & \Delta C \leq 0 \\ e^{-\frac{\Delta C}{T}} & \Delta C > 0 \end{cases} \qquad (1)$$

where T is the temperature and ΔC is the cost variation. According to Equation 1, transformations with cost-decrement are always accepted, while transformations with cost-increment are accepted with a probability given by the Boltzmann factor. This definition for the probability of acceptance allows the algorithm to escape from local minima.

3) The annealing schedule or method of determining the initial temperature and its cooling. According to Equation 1, the temperature parameter controls the probability of acceptance. Since an annealing process is carried out, a high value of temperature is initially selected. Such high initial temperature maintains the probability close to one, and hence almost all movements are initially accepted. In order to provoke the convergence of the algorithm to the global optimum, temperature must be lowered slowly according to the search advance. Also, at each temperature, the simulation must proceed long enough for the system to reach a steady state. The annealing schedule [13][14] is a key issue for getting success in the optimization, and it is usually connected to the distribution density used in generating new solutions. There are some theoretical studies based on Markov chains that prove the asymptotic convergence of SA to the optimal solution for some specific annealing schedule [15][16][17]. Thus, the logarithmic annealing schedule ($T=T_o/\ln(t)$), along with Gaussian-Markovian distribution as the generation function, has been proven to asymptotically reaching the global optimum [18]. Nevertheless, such asymptotic requirement predicts infinite transformations to guarantee the global optimum solution. In practice, real life problems are time constrained, and a common goal is to reach good solutions within the available time.

To speed up the annealing process with respect to the logarithmic annealing schedule, different functional forms have emerged. For example, the original annealing schedule ($T_{i+1}=cT_i$, $0 <c< 1$) where the temperature drops exponentially with the number of temperature updates, or *Fast Annealing* (*FA*) [19] which uses the Cauchy distribution for the generation function, and the relation $T=T_o/t$ for the temperature decrease. In spite of the success of such annealing schedules, they lack for some variable connecting the temperature to the problem or cost function being optimized. Thus, the unique link between the annealing schedule and the problem is a set of parameters to be adjusted experimentally. In this way, they usually demand a large number of empirical studies to fine-tune the annealing schedule parameters to specific problems or cost functions. Along with the above annealing schedules there are others aiming to provide practical adaptive properties as *Generalized Simulated Annealing* [20], *Adaptive Simulated Annealing* [21], or *Natural Optimization* [22].

In spite of the arduose efforts performed in providing an efficient annealing schedule valid for a wide range of problems, it remains as an open question and much experimentation must still be performed.

4. *Thermodynamic Combinatorial Optimization*

One of the theoretical bases of SA consists of reaching equilibrium at each temperature. In order to achieve this goal, the annealing schedule of SA goes along a sequence of temperatures for cooling the system, while a number of rearrangements are attempted to recover equilibrium at each one, i.e., the system does not achieve equilibrium if temperature is lowered too quickly. A main concern of SA practitioners is how slowly the cooling should be to avoid the search gets stuck in local minima. That is, how large temperature steps should be, and how many moves should be performed to recover equilibrium at each temperature. Both parameters the magnitude of temperature drops and the number of movements performed at each temperature are usually adjusted experimentally for each cost function.

Thermodynamic Combinatorial Optimization (*TCO*) provides a method to perform the cooling close to equilibrium missing out on experimental adjustments. *TCO* is derived from the first and second thermodynamic laws applied to reversible processes [23]. A key feature of reversible processes is just that the intermediate states of a transformation are also equilibrium states. In this section, we deduce both the probability of acceptance and the annealing schedule, constrained to the optimization process is reversible.

A combinatorial optimization problem can be formulated from a cost function C defined on a solution space N. The optimization problem consists of finding the solution of minimum cost C within the solution space N. In [17], a relationship between Simulated Annealing and Statistical Physics is settled. In a similar way, aiming to establish a link between Thermodynamics and Statistical Physics on one hand, and Thermodynamic Combinatorial Optimization on the other, we differentiate among the concepts of macrostate, microstate and state.

In *TCO* scope we first define the concept of macrostate associated to the temperature *T*, as a random variable X_T taking values in the solution space *N* with a probability distribution

$$P\{X_T = x_i\} = p_i \qquad x_i \in N$$

in such a way that

$$\sum_{i=1}^{n} p_i = 1 \tag{2}$$

where n is the number of possible configurations in N. The probability distribution concerns to statistics approach, since macrostate properties can be derived from p_i values. Thus, the cost C_T of the macrostate X_T is equal to the average or expected cost, that is

$$C_T = \sum_{i=1}^{n} c_i \cdot p_i \tag{3}$$

where c_i is the cost of the solution x_i. The entropy S_T associated to the macrostate X_T can be expressed as

$$S_T = \sum_{i=1}^{n} p_i \cdot s_i \tag{4}$$

Note that according to Information Theory [24], the entropy may also take the form

$$S_T = -\sum_{i=1}^{n} p_i \cdot ln(p_i) \tag{5}$$

In second term, we define the concept of microstate as every value x_i that can take the random variable X_T together with its associated probability (p_i). Thus, a microstate is a possible solution, and hence has a measurable cost c_i. On the other hand, although we are unfamiliar with the concept of the entropy of a microstate, it can also be defined. Thus, matching Equations 4 and 5, the entropy of a microstate can be expressed as

$$s_i = -ln(p_i) \tag{6}$$

Finally, we define the concept of state as the current microstate or solution examined in the opmization process. A state is characterized by its cost c_i. A transformation between two states A and B is subjeted to the probability of acceptance P_{AB}.

From these definitions, *Thermodynamic Combinatorial Optimization* consists of two superimposed processes: local and global optimizations. The first optimization focuses on local transformations between microstates, while the second consists of a path through a series of macrostates. Two main results can be derived from the application of thermodynamic laws and information theory to these optimization processes. Firstly, we are going to derive an expression for the probability of accepting a transformation between two microstates meeting the thermodynamic laws and Shannon's definition of entropy. Following, an annealing schedule based on thermodynamic laws is provided.

Probability of acceptance

During the local search new microstates will be generated to submit to the acceptance test. According to Thermodynamics, the efficiency of a reversible process improves any other transformation perfomed irreversible. Therefore, a priority goal of *TCO* is to perform transformations in a reversible way, i.e., all local transformations performed between two microstates are demanded to meet the second thermodynamic law for reversible processes performed at temperature T. Thus, a local transformation attempted between the microstates x_A and x_B should meet Equation 7.

$$s_B - s_A = \frac{Q_R}{T} \tag{7}$$

where s_A and s_B are the entropy of the microstates *A* and *B* respectively, Q_R the heat interchanged in the process, and T the temperature.

On the other hand, the first law of Thermodynamics for systems that only interchange heat with surroundings and do not perform work takes the form

$$u_B - u_A = Q \tag{8}$$

where *u* is the *internal energy*.

Therefore, a reversible process suffered by a system at temperature T that only interchange heat with surroundings should meet

$$u_B - u_A = T \cdot (s_B - s_A) \tag{9}$$

Making the equivalence between *internal energy* (*u*) and *cost* of a solution (*c*) for combinatorial optimization problems Equation 9 becomes

$$c_B - c_A = T \cdot (s_B - s_A) \tag{10}$$

Substituting the entropy of a microstate given by Equation 6, we have

$$s_B - s_A = -ln \frac{p_B}{p_A} \tag{11}$$

where p_A and p_B are the probabilities of the microstates x_A and x_B respectively. Now, note that starting from the microstate x_A, the probability p_B of reaching the microstate x_B will be P_{AB}. In the same way, the probability p_A of remaining in the microstate x_A will be equal to 1- P_{AB}. Substituting these values in Equation 11 we have

$$s_B - s_A = -ln \frac{P_{AB}}{1 - P_{AB}} \tag{12}$$

and Equation 10 takes the form

$$c_B - c_A = -T \cdot ln \frac{P_{AB}}{1 - P_{AB}} \tag{13}$$

Finally, finding P_{AB} in Equation 13 we obtain

$$P_{AB} = \frac{1}{1 + e^{\frac{\Delta C}{T}}} \tag{14}$$

Therefore, local transformations submitted to the acceptance test given by Equation 14, meet the first and second thermodynamic laws defined for reversible processes. Equation 14 has the form of *Fermi-Dirac* distribution. Note that appliying such probability of acceptance not all transformations with cost decrease are always accepted. In practice, aiming to benefit from all cost-decrement transformations, the Boltzmann distribution (Equation 1) can be applied instead.

Thermodynamic annealing schedule

In *SA*, the temperature T of the probability of acceptance is strictly controlled in order to keep the system close to equilibrium. Unlike *SA*, in *TCO* a temperature schedule meeting thermodynamic laws is derived. As stated above, *TCO* performs a superimposed global optimization along with the local one. This global process consists of a path through a series of macrostates. The ends of this path can be easily identified. At the begining (failing knowledge about the system), we will be interested in searching throughout the solution space (N), i.e., starting from an initial macrostate X_{T_0} where all configurations have the same probability. That is

$$P\{X_T = x_i\} = \frac{1}{n} \qquad \forall x_i \in N \tag{15}$$

The entropy for the initial macrostate X_{T_0} can be expressed as

$$S_{T_0} = -\sum_{i=1}^{n} \frac{1}{n} \cdot ln(\frac{1}{n})$$

therefore

$$S_{T_0} = ln(n)$$

Intuitively, the initial macrostate can be considered as an indefinite macrostate where all solutions have the same probability to be selected. Such macrostate has maximum entropy. As the process goes on, not only the cost but also entropy must decrease. Thus, the optimization process loses uncertainty and gets certainty or information. The final goal of the search process is to achieve an optimum solution. Let N_{opt} be the set of optimum solutions, and let n_{opt} be the cardinal of this set, the final macrostate X_{Tf} should meet

$$P\{X_{Tf} = x_i\} = \begin{cases} \frac{1}{n_{opt}} & x_i \in N_{opt} \\ 0 & x_i \notin N_{opt} \end{cases}$$

In this way, the average cost matches up to the optimum value

$$C_{Tf} = \sum_{i=1}^{n_{opt}} \frac{1}{n_{opt}} \cdot C_{opt} = C_{opt}$$

and its entropy is minimum (maximum information)

$$S_{Tf} = -\sum_{i=1}^{n_{opt}} \frac{1}{n_{opt}} \cdot \ln(\frac{1}{n_{opt}})$$

When a single optimum solution exists, the final entropy becomes

$$S_{Tf} = -1 \cdot ln(1) = 0$$

Note that entropy decrease is necessary. Thus, although the optimum solution can be reached early in the optimization process, it may be lost in few iterations if entropy remains high (uncertainty).

Once the initial and final macrostates are identified, we are interested in finding an efficient path between both. That is, the optimization process must go through the shortest path. According to Thermodynamics, a reversible process performed between two macrostates X_A y X_B yields better than any other irreversible process performed between them. A goal must be to move from the initial macrostate X_{T0} to final macrostate X_{Tf} by a reversible process meeting the first and second laws (Equation 16).

$$T = \frac{C_B - C_A}{S_B - S_A} \tag{16}$$

Since macroscopic space is continuous, Boltzmann statistics can be applied to macrostate transformations, then

$$S_B - S_A = \ln(P_{AB})$$

Moreover, to perceive a change from macroscopic point of view, it is not enough appliying the first and second laws to a single microstate transformation. Instead, due the probabilistic definition of macrostate, Equation 16 must be met by any compound transformation. Therefore, assume that the transition between the macrostates X_A and X_B is performed by means of k iterations. Let ΔC_i be the cost variation of the local transformation tried at iteration i, let T_i be the temperature at which the transformation was tried, and let P_i be its probability of acceptance. The entropy variation will be

$$S_B - S_A = ln\prod_{i=1}^{k} P_i$$

therefore

$$S_B - S_A = \sum_{i=1}^{K} ln\, P_i \qquad (17)$$

On the other hand, let $M_{accepted}^{k}$ be the set of transformations accepted at iteration k, the cost variation will be

$$C_B - C_A = \sum_{i \in M_{accepted}^{k}} \Delta C_i \qquad (18)$$

Substituting equations 17 and 18 in 16, the macrostate transformation performed until iteration k will be reversible if T meets Equation 19. Since the value of T given by 19 will be applied at next iteration, it can be denominated T_{k+1}.

$$T_{k+1} = k_A \cdot \frac{\sum_{i \in M_{accepted}^{k}} \Delta C_i}{\sum_{i=1}^{K} ln\, P_i} \qquad (19)$$

The parameter k_A has been introduced in Equation 19 to control the run-time quality tradeoff. Since the optimization time is proportional to this parameter, the value of k_A can be easily adjusted to the time available ($t_{available}$) according to Equation 20.

$$k_A = \frac{t_{available}}{t_{K_A=1}} \qquad (20)$$

where $t_{k_A=1}$ is the run time measured for k_A=1. Regarding initial temperature, its value will depend on the initial configuration or solution. In *TCO* two operational modes can be considered: normal and adaptive. When the initial configuration is generated randomly, a normal mode can be applied. In this case, the initial temperature must be selected in such a way that initially almost all movements are accepted. For example, it can be calculated in a similar way to [25] according to the following equation

$$T_o = -\frac{\overline{|\Delta C|}}{\ln(P)}$$

where $\overline{|\Delta C|}$ is the average of the absolute value of the cost variation obtained for an initial sequence of random transformations, and *P* is a high acceptance probability close to one. On the other hand, whenever the optimization process does not start from an initial random solution (i.e., it starts from a halfway solution), the adaptive mode should be applied. In this case, the initial temperature must be selected to a very low value. Then, after some updating movements, the temperature will match up to the quality of this initial solution.

Finally, some special cases in applying Equation 19 must be remarked. Thus, while entropy variation remains zero, the temperature must be set to T_0 in order to avoid singularities. On the other hand, at the beginning of the optimization process the temperature defined according to Equation 19 may take negative values. To force cost minimization, the situation above must be avoided. Therefore, if the cost of the current solution is larger than the cost of the initial solution, the temperature should be set to T_0. According to above remarks, Equation 19 can be rewritten as

$$T_{k+1} = \begin{cases} T_0 & if \quad \sum_{i=1}^{k} \ln(P_i) = 0 \quad or \quad \sum_{i \in M_{accepted}^{k}} \Delta C_i \geq 0 \\ k_A \cdot \dfrac{\sum_{i \in M_{accepted}^{k}} \Delta C_i}{\sum_{i=1}^{k} \ln(P_i)} & other\ cases \end{cases} \qquad (21)$$

In this section *Thermodynamic Combinatorial Optimization* method has been presented. TCO consists of two superimposed optimizations: local and global. The local optimization performs microstate transformations within a macrostate defined by the temperature T. The global optimization process observes a set of local transformations as a macrostate reversible transformation. Unlike the local process, in the global optimization the probability P_{AB} given by Equation 14 is not interpreted as the probability to move from one microstate to another. Instead, P_{AB} is reinterpreted as the probability to move from a macrostate X_A with its cost and entropy values (C_A *and* S_A), to another macrostate X_B with another set of values (C_B *and* S_B). In this way, whenever the acceptance test is applied, the macrostate of the system is modified. For example, when a transformation is rejected, the uncertainty decreases since there are fewer changes. Thus, both the entropy decreases, and the macrostate of the system changes.

5. Experimental results

Thermodynamic Combinatorial Optimization has been implemented to perform the placement of a set of benchmark circuits on an FPGA with *Bounding Box* (*BB*) cost function. In order to appreciate the behavior of *TCO*, it has been compared with *VPR* (*Versatile Place&Route*) a fine-tuned SA placement tool [9][10].

Figure 1 depicts the optimization development with *TCO* and *SA* for the circuit seq.net and k_A=300. Horizontal axis represents the number of iterations or movements evaluated, while vertical axis shows the cost of solutions.

Note that *TCO* drops faster than both *SA* does, while a similar cost is obtained.

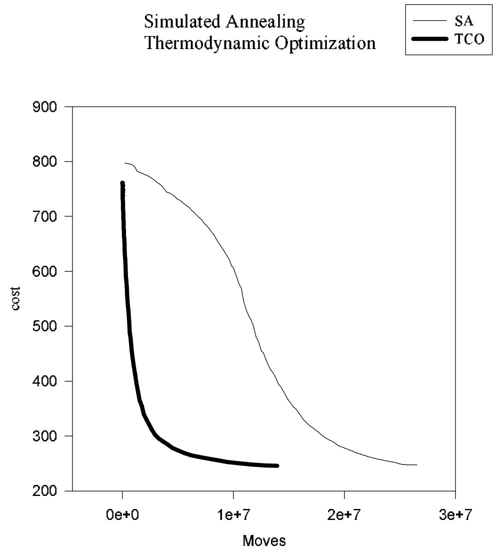

Figure 1. *Cost/Movements relationship for the placement of a circuit in an FPGA with different methods: SA and TCO.*

Figure 2 shows the relationship between the cost and temperature decrease represented in a logarithmic scale. Note that both graphs have similar behavior. *TCO* provides a simple interpretation of this phenomenon, i.e., each temperature has an associated macrostate, and hence an average cost.

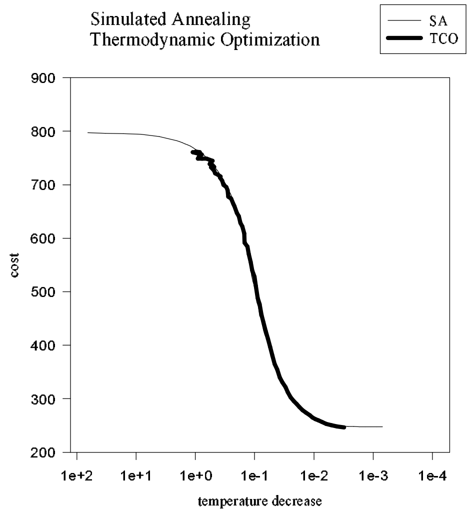

Figure 2. *Cost/Temperature* ***e****volution for the placement with SA and TCO*

In order to appreciate different behaviors in both methods, we must refer to the temperature/time relationship. In Figure 3, horizontal axis represents the number of movements evaluated, while vertical axis the temperature T. Remark that the temperature with *TCO* is tuned with the development of the optimization, and hence only usefull movements are performs at each temperature.

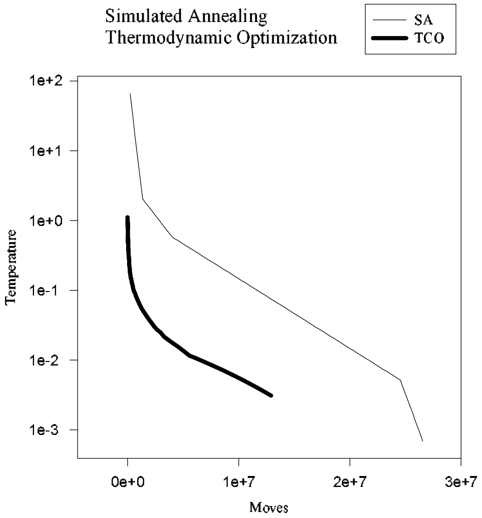

Figure 3. *Temperature/moves relationship for the placements performed by SA and TCO*

We have expanded the comparison between both methods on a set of benchmark circuits (Table 1).

Table 1. *Benchmark circuits: CLBs, IOBs and demanded FPGA size*

CIRC	*#CLBs*	*#I/O*	*FPGA size*
tseng	1047	174	33x33
ex5p	1064	71	33x33
apex4	1262	28	36x36
misex3	1397	28	38x38
diffeq	1497	103	39x39
alu4	1522	22	40x40
seq	1750	76	42x42
apex2	1878	41	44x44
s298	1931	10	44x44
dsip	1370	426	54x54
bigkey	1707	426	54x54
frisc	3556	136	60x60
spla	3690	62	61x61
ex1010	4598	20	68x68
pdc	4575	56	68x68
s38417	6406	135	81x81
s38584.1	6447	342	81x81
clma	8383	144	92x92

Table 2 shows the values of the cost function *BB* optimized by *SA* and *TCO,* as well as the number of movements needed by both methods for these circuits. The last two columns show the relative improvement of *TCO* with respect to *SA* measured as follows:

$$Q = 100 \cdot \frac{Cost(NO) - Cost(SA)}{Cost(SA)} \qquad (Quality)$$

$$T = 100 \cdot \frac{Moves(NO) - Moves(SA)}{Moves(SA)} \qquad (Time)$$

Remark that negative values of Q and T imply a *TCO* improvement with respect to *SA* and vice versa. Note that while similar quality solution is provided by both methods, *TCO* reduces appreciably the time demanded by *SA*.

Table 2. *Comparison between SA and TCO for the placement problem on a set of benchmarks*

CIR.	Cost (Bounding Box)			Moves (x10^6)		Q	T
	Init.	*SA*	*TCO*	*SA*	*TCO*	%	%
tseng	420	93	93	15.5	6.7	0.0	-57
ex5p	427	162	162	13.8	6.2	0.0	-55
apex4	503	180	179	16.3	8.1	-0.5	-50
misex3	588	188	188	18.8	9.7	0.0	-48
diffeq	670	146	145	22.5	11.6	-0.7	-48
alu4	612	192	191	21.2	7.3	-0.5	-66
seq	804	247	246	26.7	12.9	-0.4	-52
apex2	912	267	266	28.6	12.7	-0.4	-56
s298	750	203	203	29.3	21.6	0.0	-26
dsip	895	171	171	28.5	22.1	0.0	-22
bigkey	1085	187	187	35.7	19.5	0.0	-45
frisc	2288	516	516	72.4	52.3	0.0	-28
spla	2373	608	604	72.8	41.1	-0.7	-44
ex1010	3325	655	655	97.6	56.4	0.0	-42
pdc	3230	869	869	97.2	72.3	0.0	-26
s38417	5837	674	666	166	112	-1.1	-32
s38584.1	5616	647	647	173	118	0.0	-32
clma	7957	1390	1389	240	148	-0.1	-38

6. Conclusion

In this paper *TCO*, a new combinatorial optimization method has been presented. *TCO* is similar to simulated annealing method, but both the probability of accepting local transformations and the temperature schedule, are directly derived from the thermodynamic laws and Shannon's definition of entropy. Unlike most simulated annealing strategies, *TCO* adapts automatically to different cost functions while providing high performance.

TCO has been applied to FPGA placement, improving the results provided by *VPR*, a fine-tuned SA tool for this problem.

References

[1] N.A. Sherwani, "Algorithms for VLSI Physical design automation," Kluwer Academic Publishers, 1999.

[2] J. Rose, A. El Gamal, A.Sangiovanni-Vincentelli, "Architecture of FPGAs," Proc. of IEEE, July 1993, pp.1013-1029.

[3] K. M. Hall, "An r-dimensional quadratic placement algorithm", Management Science, 17, 219-229, 1970.

[4] M.A. Breuer, "A Class of Min-Cut Placement Algorithms," Proceedings DAC, pp. 284-290, 1977.

[5] N. R. Quinn, M. A. Breuer, "A force directed component placement procedure for printed circuit boards", IEEE Trans. Circuits and Systems, pp. 377-388, June 1979.

[6] C. Sechen, and A. Sangiovanni-Vincentelli,"The TimberWolf placement and routing package," in Proc. Custom IC Conf. (Rochester, NY), 1984, pp. 522-527.

[7] J. P. Cohoon, and W. Paris, "Genetic Placement," Proc. IEEE ICCAD, pp.422-425, 1986.

[8] J.M. Kleinhans, G. Sigl, F.M. Johannes, K.J. Antreich, "GORDIAN: VLSI Placement by Quadratic Programming and Slicing Optimization", IEEE Transactions on CAD of ICs and Systems , Vol. 10, NO. 3, 1991, pp. 356-365.

[9] V. Betz and J. Rose, "VPR: A new Packing, Placement and Routing Tools for FPGA Research," FPL97, http://www.eecg.toronto.edu

[10] V. Betz and J. Rose, "Effect of the Prefabricated Routing Track Distribution on FPGA Area-Efficiency", IEEE Trans. on VLSI, 1998.

[11] C.E. Cheng, "RISA: Accurate and Efficient Placement Routability Modeling," DAC, 1994, pp. 690-695.

[12] S. Kirkpatric, C.D. Gelatt, and M.P. Vecchi,"Optimization by simulated annealing," Science, vol. 220, no. 4598, pp. 671-680, May 1983.

[13] D.F. Wong, H.W. Leong, C.L. Liu, "Simulated Annealing for VLSI Desing", Kluwer Academic Publishers,1988.

[14] R.H.J.M. Otten, L.P.P.P. van Ginneken, "The Annealing Algorithm", Kluwer, 1989.

[15] M. Lundi, a. Mess, "Convergence of an annealing algorithm", Math. Programming 34, 1986, 111-124.

[16] B. Hajec, "Cooling schedules for optimal annealing", Math. of Operations Research 13, 1988, pp. 311-329.

[17] E. Aarts, J. Korst, "Simulated Annealing and Boltzmann Machines, A Stochastic Approach to Combinatorial Optimization and Neural Compupting", John Wiley & Sons, 1989.

[18] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution and the Bayesian restoration in images", IEEE Trans. Patt. Anal. Mac. Int. 6 (6), 721-741, 1984.

[19] H. Szu, R. Hartley, "Fast simulated annealing", Physics Letters A 122, 1987, pp.157-162.

[20] M. E. Johnson I. O. Bohachevsky, M. L. Stein, "Generalized simulated annealing for function optimization", Technometrics, 28(3), 1986.

[21] L. Ingber, "Adaptive simulated annealing (ASA): Lesson learned," Control and Cybernetics, 25, pp.33-54 (1996).

[22] J. De Vicente, J. Lanchares, R. Hermida, "Adaptive Placement by Natural Optimization", Proc. IEEE International Workshop on Rapid System Prototyping (RSP), Paris (France), Jun 2000, pp 188-193.

[23] H.B. Callen, "Thermodynamics", John Willey&Sons, 1960.

[24] C.E. Shannon, "A mathematical theory of communication", I & II. Bell Syst. Tech. J., 27, pp. 379-423, 623-656, 1948.

[25] D. F. Wong, C. L. Liu, "A New Algorithm for Floorplan Design", Proc. 23rd ACM/IEEE Design Automation Conference, 1986, pp.101-107.