

# Doctoral Symposium: Trade-off Analysis of Thermal-Constrained Scheduling Strategies in Multi-Core Systems

Muhammad Usama Sardar  
Technische Universität Dresden, Dresden, Germany  
muhammad\_usama.sardar@mailbox.tu-dresden.de

## ABSTRACT

The increasing usage of multi-cores in safety-critical applications, such as autonomous control, demands high levels of reliability, which crucially depends on the temperature. The scheduling of tasks is one of the key factors which determine the natural trade-off between system performance and reliability. Commonly used techniques, such as simulation-based on benchmarks, can simulate only a limited number of input sequences of system runs and hardly optimize the performance-reliability trade-off. In order to accurately evaluate the schedulers and provide formal guarantees suitable in early design stages, we use a design flow based on formal methods for a quantitative performance-reliability trade-off analysis. Specifically, we propose to use energy-utility quantiles as a metric to evaluate the effectiveness of a given scheduler. For illustration, we evaluate TAPE, a state-of-the-art thermal-constrained scheduler, with theoretical optimal ones.

## CCS CONCEPTS

• **Hardware** → **Temperature control**; • **Computer systems organization** → *Multicore architectures*; • **Theory of computation** → *Probabilistic computation*; • **Software and its engineering** → *Model checking*.

## KEYWORDS

thermal-constrained scheduling, distributed multi-core systems, energy-utility quantiles, probabilistic model checking

## ACM Reference Format:

Muhammad Usama Sardar. 2020. Doctoral Symposium: Trade-off Analysis of Thermal-Constrained Scheduling Strategies in Multi-Core Systems. In *The 14th ACM International Conference on Distributed and Event-based Systems (DEBS '20)*, July 13–17, 2020, Virtual Event, QC, Canada. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3401025.3406442>

## 1 PROBLEM

The enormous increase in the processor power density [6] has made on-chip temperature a critical design constraint of multi-core

systems. The elevated chip temperatures adversely impact other design constraints, such as reliability, performance and cooling costs [13]. These high temperatures can result in more frequent transient errors and/or even permanent faults [20]. Indeed, industrial studies have demonstrated that a small difference in the operating temperature (order of 10–15C) can result in almost two times difference in the device lifespan [22]. Additionally, studies show that the cooling cost increases super-linearly with the thermal dissipation [7]. Moreover, the static (leakage) power [23] has exponential dependence on the operating temperature, which potentially results in more thermal runaway [16]. Hence, to ensure the reliability, performance, and safety of the multi-core for modern embedded real-time systems like autonomous control [12], thermal-constrained scheduling is crucial. Consequently, the thermal constraints should be accounted for in the scheduling of tasks to the cores.

The capability of formal methods is gaining more and more attention for a quantitative trade-off analysis. *Energy-utility quantiles* [1] provide a relevant trade-off measure in probabilistic systems, e.g., systems where the environment, the workload or the occurrence of errors is modelled probabilistically. For example, a possible instance of an energy-utility quantile is the minimal number of thermal violations to ensure finishing a given number of tasks within a given time horizon with sufficiently high probability. The thermal violations refer to the situation where the temperature of a core is above the critical temperature. *Such properties cannot be determined using simulative approaches*. The core problem is to explore the use of formal methods for a quantitative performance-reliability trade-off analysis.

## 2 STATE OF THE ART

Simulation with existing benchmarks is popular in the embedded system community to analyze the effectiveness of the scheduling strategy. However, most of these approaches can simulate only a *limited number of input sequences* and thus may result in missing critical situations. In turn, this may lead to delays in the deployment of thermal management schemes as happened in the case of Foxton thermal management, which was designed for the Montecito chip [4]. It may also result in poor performance and/or thermally unsafe behaviors or even catastrophic failures at run-time, e.g., vehicle breakdown or smartphone explosion [12].

Exhaustive formal methods such as *model checking* (see, e.g., [2]) are popular for ensuring reliability of critical system parts. In this regard, a few abstract thermal models have been proposed for the formal analysis. Most of these works, e.g., [8–10, 19], ignore the thermal coupling among the cores. The models without incorporating

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).  
*DEBS '20, July 13–17, 2020, Virtual Event, QC, Canada*  
© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 978-1-4503-8028-7/20/07...\$15.00  
<https://doi.org/10.1145/3401025.3406442>

thermal coupling can result in underestimation of the temperature, which accounts for a significant difference in reliability estimation.

### 3 PROPOSED APPROACH

In this section, we present our proposed formal thermal model for 2D/3D multi-core systems as well as describe our workflow and parametric model. Finally, we present a model instance that we implemented for a quantitative analysis using the PRISM model checker. We denote the set of integers  $i, i + 1, \dots, j - 1, j$  by  $[i..j]$  for  $i, j \in \mathbb{N}$ .

#### 3.1 Abstract Thermal Model

For simplicity, we consider a 2D grid layout in which  $N = n \times n$  homogeneous cores are placed, where we refer to a core placed at position  $(i, j)$  via an index  $i + n \cdot j \in [0..N - 1]$ . Similar to the related works [8–10, 19], we consider a discrete-time model where the power states and resulting temperatures of the cores are observed after fixed discrete intervals of time. Intuitively, the change in temperature of a core depends on three major factors: (i) power dissipated by the core, (ii) heat transferred by the core to the ambience, and (iii) heat transferred among the cores. We merge the first two factors into one called *self-heating* and model the last one as thermal coupling.

We now explain the mathematical model by equations that hold for each core with some given index  $i$ . The change in temperature of core  $i$  is given by:

$$\Delta Temp_i = w_{sh} \cdot sh(Temp_i, Power_i) + w_{cpl} \cdot cpl_i(Temp_{[0..N-1]}) \quad (1)$$

where  $\Delta Temp_i$ ,  $Temp_i$  and  $Power_i$  represent the change in temperature, the current temperature, and the current power state (e.g., ON or OFF), respectively, of core  $i$ . The weight  $w_{sh} \in \mathbb{Q}$  is a constant used to represent the scaling for self-heating, which depends on the nature of the material and environmental conditions. The weight  $w_{cpl} \in \mathbb{Q}$  represents the overall scaling of the thermal coupling and is specifically dependent on the conductivity of the material and the sampling interval  $\Delta t$ , as suggested by the general equation of heat conduction [17]. Moreover,  $sh$  and  $cpl$  represent self-heating and thermal coupling functions, as described below. The power dissipated by a core for performing some computation produces heat while the heat transferred by a core to the ambience leads to a drop in the temperature of the core. The overall effect is represented by the self-heating function  $sh$ :

$$sh(Temp_i, Power_i) = pd(Power_i) + amb(Temp_i) \quad (2)$$

where the function  $pd$  models the increase in the temperature of a core due to the power consumed for doing some computation in one time step. For the case of 2-level Dynamic Voltage and Frequency Scaling (DVFS), it is defined as follows:

$$pd(Power_i) = \begin{cases} p_1, & \text{if } Power_i = 1 \\ p_2, & \text{if } Power_i = 0 \end{cases} \quad (3)$$

where  $p_1, p_2 \in \mathbb{Q}_{>0}$  are positive parameters, depending on the microarchitecture, application, and leakage characteristics. The function  $amb$  models the heat transferred to the ambience, inspired by the Newton's law of cooling [3, 17]:

$$amb(Temp_i) = -c(Temp_i - Temp_{amb}) \quad (4)$$

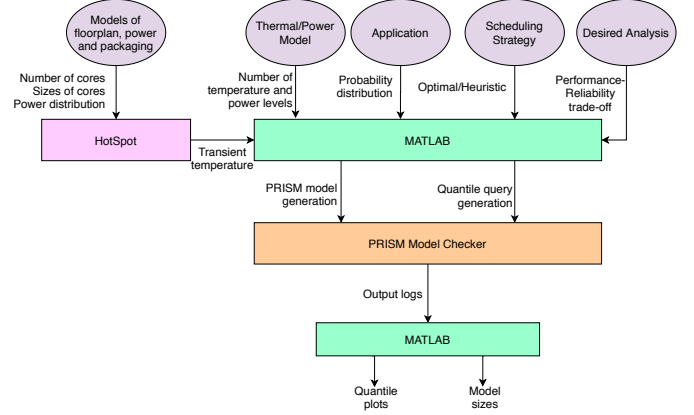


Figure 1: Workflow of the proposed approach

where  $c \in \mathbb{Q}_{>0}$  is a positive parameter, depending on the cooling solution (heat sink and spreader) specifications, and  $Temp_{amb}$  represents the ambient temperature. The negative sign shows that it leads to a drop in the temperature of the core.

The thermal coupling mainly depends on the conductivity of the material, sampling time, the difference of the temperature of the cores, and distance between the cores [17]. The first two factors are modeled by  $w_{cpl}$  while the last two factors vary from core to core and are captured in our model by the thermal coupling function  $cpl_i$ , as described below:

$$cpl_i(Temp_{[0..N-1]}) = \sum_{j \in [0..N-1], j \neq i} \left\{ w_{ij} \cdot (Temp_j - Temp_i) \right\} \quad (5)$$

where while the index  $i$  still represents the index of the core under consideration, the index  $j$  in the sum represents the index of other cores. We model the weights  $w_{ij} \in \mathbb{Q}$  for thermal coupling, i.e., coupling coefficients, by the reciprocal of the Euclidean distance between the two cores in the 2D grid. The validation of the thermal model is presented in [18].

#### 3.2 Workflow

The workflow of the proposed approach, presented in Fig. 1, begins with a popular thermal simulator, HotSpot [21], which requires models of floorplan, packaging and power traces. The floorplan describes sizes and placement of the cores. The user can select parameters for floorplan and packaging based on the system under consideration. Power traces of the application are then given to the thermal simulator to compute transient temperatures. The transient temperatures are then analyzed in our proposed tool developed in MATLAB to find the trends in the behavior of the temperature, which form the basis for the validation of our proposed thermal model. A reasonable continuous-valued thermal model is developed from transient temperatures. In order to find the optimal weights for Eq. (1), we use properties of symmetry with respect to power and memorylessness with respect to the initial temperature and the minimum mean square error criteria to evaluate our discretized thermal model against the continuous-valued one.

For exhaustive formal analysis, the continuous parameters, such as temperature, have to be discretized to a certain number of levels

to analyze the results within a suitable time. In our parametric model, the designer can select various parameters, such as the number of temperature and power levels, number of cores in the system and the scheduling strategy (optimal/heuristic) for the analysis. In case of heuristic analysis, the scheduling criteria is required. Additionally, the designer may choose the probability distribution and its characteristics (e.g., mean) to capture the behavior of the application.

With the above parameters, we generate model variants, i.e., transition systems, Discrete-time Markov Chains (DTMCs) or MDPs in PRISM model checker[11], depending on the desired analysis. The purpose of having states in a transition system is the step-wise behavior induced by the task arrival and scheduling. The probability distributions model the task arrival and the non-determinism is used to model the different choices for task scheduling on the cores. Depending on the desired analysis, we provide flags in our tool for the option whether to include time or thermal violations in the state space and generate properties to be investigated accordingly. The generated formal model along with the property (e.g., quantile query) is input to the probabilistic model checker to perform the analysis (e.g., to compute quantiles for various probability thresholds). The output logs from the probabilistic model checker are input to our tool in MATLAB to generate quantile plots as well as analyze model sizes. Thus, even a designer with minimal prior knowledge of formal methods, can analyze these plots to perform performance-reliability trade-off analysis for the designed scheduler.

### 3.3 Concrete Model

For illustrative purposes, we present the details of the concrete model, instantiated from the parametric model presented above, for the configurations used in our experiment: the system has 9 cores arranged in a 2D grid of  $3 \times 3$ . We chose a granularity of 3 temperature levels, represented as 0, 1 and 2, respectively, for our experiments. We further considered 2-level DVFS, i.e., core power has 2 states (ON or OFF).

Further parameters chosen are  $p_1 = 1$  and  $p_2 = 0$  in Eq. (3), and  $c = 0.25$  in Eq. (4). For brevity, in this work we consider thermal coupling effects from 4 direct neighbors of each core. The cores are considered to be unit distance apart, so that the weights  $w_{ij}$  in Eq. (5) are taken as unity for all the 4 direct neighbors. The selected weights for Eq. (1) for the above parameters by using the minimum mean square error are 1.3 for self-heating ( $w_{sh}$ ) and 0.09 per core per temperature level difference for thermal coupling ( $w_{cpl}$ ). It should be noted that the concrete model is a proof-of-concept, and the realistic parameters can be sourced from a pre-analysis step for a specific application and architecture.

## 4 RESULTS

In this section, we present a refined formal model that reflects the system behavior in practical applications, with additional stochastic information, e.g., on the workload of the system. For including thermal management strategies in our model to distribute workload on the cores, we add non-deterministic choices of the cores to be powered ON as soon as the required amount of workload is apparent. This yields a MDP with probabilistic workload choices

and non-deterministic powering of cores. Our model then does not only pave the way for a best- and worst-case analysis, revealing optimal thermal management policies for the assumed probabilistic workload profile, but can also be used to analyze existing thermal management heuristics. For further details, the reader is referred to the work [18].

### 4.1 Formal Model

Inspired by the work [15] from the literature, we consider that the number of tasks arriving in the multi-core CPU at each time instant follows a Poisson distribution. For a  $N$ -core system, we truncate and normalize the Poisson distribution for  $0 - N$  tasks. We compute the probabilities in our MATLAB-based tool and export to PRISM as constants. Similar to simulation-based analysis [14], we assume that there is no data dependency among tasks, i.e., we consider independent tasks. For brevity, each task is assumed to have an execution time of one time step. Moreover, we assume that each core executes only one task at a time.

For the optimal scheduler, we use non-determinism to capture the possible ways that a controller can influence the behaviour of the system. Since we are interested in determining which cores should be turned ON, the non-determinism is in the selection of a core to run a task and the decision whether to put the tasks in a queue. The optimal scheduler is then computed that resolves all the non-deterministic choices such that the expected values are either maximized or minimized.

For the heuristics, the non-determinism is resolved by the specific schedulers, resulting in a DTMC [2]. In this work, we consider 3 specific schedulers with a defined mapping criteria. For more than 1 core satisfying the mapping criteria, to resolve the non-determinism in each case, we use a specific order, i.e., [0 8 6 2 1 3 5 7 4], where the numbers indicate the indices of the cores, e.g., core 8 is scheduled before core 5. The mapping criteria of considered heuristics are presented below: A popular thermal-aware scheduler, TAPE [5], is based on the economic model and maps the tasks to the core with the criteria  $\max(sellT_i - buyT_i)$ , and in case of multiple cores satisfying this criteria:  $\min(\text{abs}(buyT_i))$ , where  $sellT_i$  and  $buyT_i$  represents  $sell$  and  $buy$  values, respectively, of a core  $i$  at temperature  $T_i$ . For modelling TAPE, we use the same weights as presented in the paper [5]. The reactive thermal-aware schedulers map to the cores with the criteria  $Temp_i < TempThreshold$  and the minimum temperature scheme maps to the coolest core among the currently available cores, i.e.,  $\min(Temp_i)$ .

### 4.2 Comparative Trade-off Analysis

The formal model presented above is parameterized in the queue size and mean value of the number of tasks. For a real-world scenario, the queue size is selected according to the system under consideration and the mean value is selected based on the prior information about the workload. In the following analysis, we consider the mean value of 8.5 and queue size of three to consider as an example. Because of parameterization, any desired values can be analyzed.

We propose to use performance metrics for the evaluation of a thermal-aware scheduler based on *energy-utility quantiles* [1]. Within energy-utility quantiles, two reward structures formalize

quantities of the system and a trade-off condition is posed by putting bounds on the accumulated reward during an execution. Varying one of the bounds and optimizing this value such that the probability mass of paths with the accumulated rewards staying within the bounds exceeds a given threshold provides a trade-off metric that can be computed using probabilistic model checking [1].

**4.2.1 Maximal Time to Thermal Violations.** One of the most important functionalities of a thermal-aware scheduler is to maximize the system's thermal stability, in terms of the time to a certain number of violations. So, we consider the following energy-utility quantile: what is the maximal time the system survives with probability  $p$  until reaching a certain number of thermal violations for some scheduler. Formally, this can be expressed as the following *existential energy-utility upper-bound quantile* [1]:

$$\max \{t : \Pr_s^{\min}(\diamond (Time \leq t \wedge \#Violations \geq v)) \leq p\} \quad (6)$$

where  $v \in \mathbb{N}$  represents a lower bound on the accumulated number of global thermal violations,  $t \in \mathbb{N}$  represents an upper bound on time and  $p \in [0, 1] \cap \mathbb{Q}$  represents the probability threshold. The results for  $v = 35$  are presented in Fig. 2 and show that TAPE and minimum temperature heuristic have the same trade-off characteristics. This is because task migration is currently not integrated in our approach. Both of them perform better than the reactive heuristic, e.g., at a high probability threshold of 0.99.

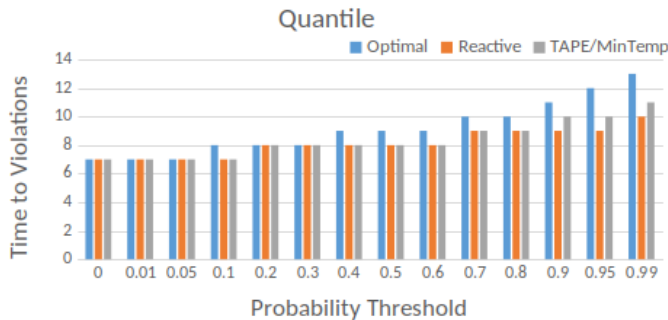


Figure 2: Time to reach 35 global thermal violations

## 5 CONCLUSIONS AND FUTURE WORK

In this work, we presented a formal thermal model for multi-cores incorporating thermal coupling as well as transient temperatures. This is challenging because HotSpot gives only the final temperature instead of the individual components of self-heating and thermal coupling. We proposed a quantitative performance-reliability trade-off analysis, based on quantiles, of thermal-aware scheduling strategies for multi-core systems. The results show that the evaluated scheduler TAPE can be improved with respect to the maximal time to reach a certain number of thermal violations. Our approach thus helps in the evaluation of heuristics. This work opens door for many future directions. First, the framework can be enhanced to include task migration. Second, the evaluation of heuristics for heterogeneous multi-core systems can be very interesting. Moreover, performance evaluation in terms of throughput of a scheduler can also be interesting.

## ACKNOWLEDGMENTS

The author would like to thank Clemens Dubsloff, Sascha Klüppelholz, Christel Baier and Akash Kumar for their contributions in this work.

## REFERENCES

- [1] Christel Baier, Marcus Daum, Clemens Dubsloff, Joachim Klein, and Sascha Klüppelholz. 2014. Energy-utility quantiles. In *NASA Formal Methods*, Vol. 8430. Springer, 285–299. [https://doi.org/10.1007/978-3-319-06200-6\\_24](https://doi.org/10.1007/978-3-319-06200-6_24)
- [2] Christel Baier and Joost-Pieter Katoen. 2008. *Principles of model checking*. MIT Press.
- [3] Louis C Burmeister. 1993. *Convective heat transfer*. John Wiley & Sons.
- [4] D. Dunn. 2005. Intel delays Montecito in roadmap shakeup. *EE Times, Manufacturing/Packaging* (2005).
- [5] T. Ebi, Mohammad Abdullah Al Faruque, and J. Henkel. 2009. TAPE: Thermal-aware agent-based power economy multi-/many-core architectures. In *Computer-Aided Design*. IEEE, 302–309. <https://doi.org/10.1145/1687399.1687457>
- [6] Dennis Gnad, Muhammad Shafique, Florian Kriebel, Semeen Rehman, Duo Sun, and Jörg Henkel. 2015. Hayat: harnessing dark silicon and variability for aging deceleration and balancing. In *Design Automation Conference*. ACM/EDAC/IEEE, 1–6. <https://doi.org/10.1145/2744769.2744849>
- [7] Stephen H Gunther, Frank Binns, Douglas M. Carmean, and Jonathan C. Hall. 2001. Managing the impact of increasing microprocessor power consumption. *Intel Technology Journal*, (2001), 1–9.
- [8] Shafaq Iqtedar, Osman Hasan, Muhammad Shafique, and Jörg Henkel. 2015. Formal probabilistic analysis of distributed dynamic thermal management. In *Design, Automation and Test in Europe*. IEEE, 1221–1224.
- [9] Shafaq Iqtedar, Osman Hasan, Muhammad Shafique, and Jörg Henkel. 2015. Probabilistic Formal Verification Methodology for Decentralized Thermal Management in On-Chip Systems. In *Enabling Technologies: Infrastructures for Collaborative Enterprises*. IEEE, 210–215.
- [10] Muhammad Ismail, Osman Hasan, Thomas Ebi, Muhammad Shafique, and Jörg Henkel. 2013. Formal Verification of Distributed Dynamic Thermal Management. In *Computer-Aided Design* (San Jose, California). IEEE, 248–255.
- [11] M. Kwiatkowska, G. Norman, and D. Parker. 2011. PRISM 4.0: Verification of Probabilistic Real-time Systems. In *Computer Aided Verification*, Vol. 6806. Springer, 585–591. [https://doi.org/10.1007/978-3-642-22110-1\\_47](https://doi.org/10.1007/978-3-642-22110-1_47)
- [12] Youngmoon Lee, Hoon Sung Chwa, Kang G Shin, and Shige Wang. 2018. Thermal-Aware Resource Management for Embedded Real-Time Systems. *Computer-Aided Design of Integrated Circuits and Systems* 37, 11 (2018), 2857–2868.
- [13] Zao Liu, Sheldon X.-D. Tan, Xin Huang, and Hai Wang. 2015. Task Migrations for Distributed Thermal Management Considering Transient Effects. *IEEE Transactions on Very Large Scale Integration Systems* 23, 2 (2015), 397–401. <https://doi.org/10.1109/TVLSI.2014.2309331>
- [14] S. Pagani, J.J. Chen, M. Shafique, and J. Henkel. 2018. *Advanced Techniques for Power, Energy, and Thermal Management for Clustered Manycores*. Springer International Publishing.
- [15] Anuj Pathania, Vanchinathan Venkataramani, Muhammad Shafique, Tulika Mitra, and Jörg Henkel. 2017. Defragmentation of tasks in many-core architecture. *Architecture and Code Optimization* 14, 1 (2017), 2:1–2:21.
- [16] Massoud Pedram and Shahin Nazarian. 2006. Thermal Modeling, Analysis, and Management in VLSI Circuits: Principles and Methods. *Proc. IEEE* 94, 8 (2006), 1487–1501. <https://doi.org/10.1109/JPROC.2006.879797>
- [17] Ralph Remsburg. 2011. *Advanced thermal design of electronic equipment*. Springer.
- [18] Muhammad Usama Sardar, Clemens Dubsloff, Sascha Klüppelholz, Christel Baier, and Akash Kumar. 2020. Performance Evaluation of Thermal-Constrained Scheduling Strategies in Multi-core Systems. In *Computer Performance Engineering*, Marco Gribaudo, Mauro Iacono, Tuan Phung-Duc, and Rostislav Razumchik (Eds.). Springer International Publishing, Cham, 133–147.
- [19] M. U. Sardar, O. Hasan, M. Shafique, and J. Henkel. 2017. Theorem Proving Based Formal Verification of Distributed Dynamic Thermal Management Schemes. *J. Parallel and Distrib. Comput.* 100 (2017), 157–171.
- [20] Jayanth Srinivasan, Sarita V. Adve, Pradip Bose, and Jude A. Rivers. 2005. Lifetime reliability: Toward an architectural solution. *IEEE Micro* 25, 3 (2005), 70–80.
- [21] Mircea R Stan, Kevin Skadron, Marco Barcella, Wei Huang, Karthik Sankaranarayanan, and Sivakumar Velusamy. 2003. HotSpot: A dynamic compact thermal model at the processor-architecture level. *Microelectronics* 34, 12 (2003), 1153–1165.
- [22] Ram Viswanath, Vijay Wakharkar, Abhay Watwe, and Vassou Lebonheur. 2000. Thermal Performance Challenges from Silicon to Systems. *Intel Technology Journal* (2000), 1–16.
- [23] Inchoon Yeo, Chih Chun Liu, and Eun Jung Kim. 2008. Predictive dynamic thermal management for multicore systems. In *Design Automation Conference*. ACM/IEEE, 734–739. <https://doi.org/10.1145/1391469.1391658>