

# Cost-Effective Decap Selection for Beyond Die Power Integrity

Yi-En Chen\*, Tu-Hsiung Tsai\*, Shi-Hao Chen<sup>†</sup> and Hung-Ming Chen\*

\*Institute of Electronics, National Chiao Tung University and SoC Center, Hsinchu, Taiwan

<sup>†</sup>Global UniChip Corp., Hsinchu, Taiwan

Email: gn190187@gmail.com; thebear325@gmail.com; hockchen@guc-asic.com; hmchen@mail.nctu.edu.tw

<sup>1</sup> Abstract— In designing reliable power distribution networks (PDN) for power integrity (PI), it is essential to stabilize voltage supply to devices on chip. We usually employ decoupling capacitor (decap) to suppress the noise generated by the switching of devices. There have been numerous prior works on how to select/insert decaps in chip, package, or board to maintain PI, however optimal decap selection is usually not applicable due to design budget and manufacturability. Moreover, design cost is seldom touched or mentioned. In this research, we propose an efficient methodology “PDC-PSO” to automatically optimizing the selection of available decaps. This algorithm not only takes advantage of particle swarm optimization (PSO) to stochastically search the design space, but takes the most effective range of decaps into consideration to outperform the basic PSO. We apply this to three real package designs and the results show that, compared to the original decap selection by rules of thumb, our approach could shorten the design period and we have better combination of decaps at the same or lower cost. In addition, our methodology can also consider package-board co-design in optimizing different operation frequencies.

## I. INTRODUCTION

As the semiconductor manufacturing technology advances, the noise margin of chip is much lower than before, and a small voltage ripple might cause the devices on chip malfunction. The authors in [6][17][15] show that the fluctuation of voltage would reduce the operation frequency, and the relationship of voltage and operation frequency is almost linear. Thus Power Integrity (PI) becomes more and more important, and it is about delivering clean power from voltage supplier to chip. Power distribution network (PDN) usually consists of Voltage Regulator Module (VRM), interconnections and capacitors of PCB, package, and chip[11]. If the PDN is not well-designed, noise generated by devices on chip switching would exceed the tolerable range, and it might cause the Signal Integrity (SI), Electro-Magnetic Interference (EMI) problems and make the chip working incorrectly[12].

In PDN design, Decoupling Capacitor (decap) insertion is a common method to reduce voltage fluctuation. A decap acts as a temporary current pool and provides the low-noise return path for signals. However, it also acts as an inductor at the frequency higher than its self-resonance due to the intrinsic equivalent series inductance (ESL) decreasing its ability. Therefore, a good PDN usually includes several decaps to cover the targeted frequency range and to make the PDN robust. How to efficiently optimize the type, location and number of decaps to save cost and make PDN robust is critical in chip, package and PCB design[19]. Fig.1 shows that in a real package design, the engineers manually choose 16 decaps to meet the PDN specification. However, it could meet the same specification with only 5 decaps optimized by our program and the saving cost is very significant.

There are researches about decap selection optimization, such as [10] [14], but they are manual rather than automatic

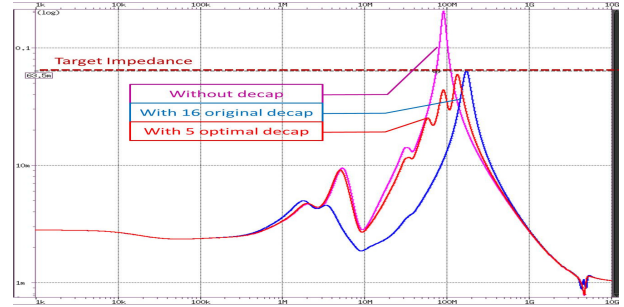


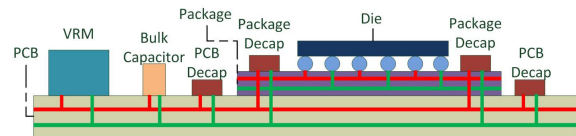
Fig. 1. With the specification that target impedance is  $0.0635\Omega$ , our algorithm could reduce the total decaps from 16 to 5.

optimizations. In [4][20] the authors use simulated annealing (SA) algorithm to choose the best location and type of decaps. However, compared to other stochastic algorithms like particle swarm optimization (PSO), SA is relatively ineffective and inefficient. Although PSO is applied to decap selection optimization problem in [16], it suffers from a problem that the result of decap selection is not commonly used in the industry. Since the values of capacitance, resistance and inductance are decided by PSO algorithm and such decap might not exist in the industry, it would be expensive to manufacture, or the design of package or PCB does not have enough area to for each selection.

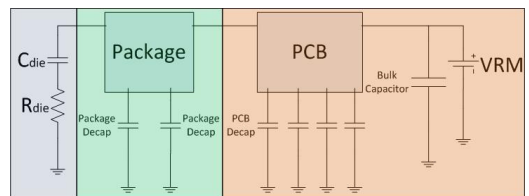
In this paper, we introduce an efficient algorithm named “Preferred Decap Choice Particle Swarm Optimization (PDC-PSO)” to optimize the decap combination for PDN design automatically. The constraints like type, amount, location, and cost of decaps could be taken into account to avoid over-design, thus this PDC-PSO algorithm is practical in real design. We blend the concepts mentioned in [5][18] in our decap optimization problem.

The rest of this paper is arranged as follows. Section II shows the preliminaries and objectives. Section III describes the basic PSO and our algorithm. The experimental results in real designs are shown in Section IV and we conclude this paper in Section V.

## II. PRELIMINARIES AND OBJECTIVES



(1) Cross view of PDN system



(2) Lumped model of PDN system

Fig. 2. PDN system includes chip, package, PCB, and VRM. (a) is the cross view of PDN system. (b) is the equivalent lumped model of PDN system.

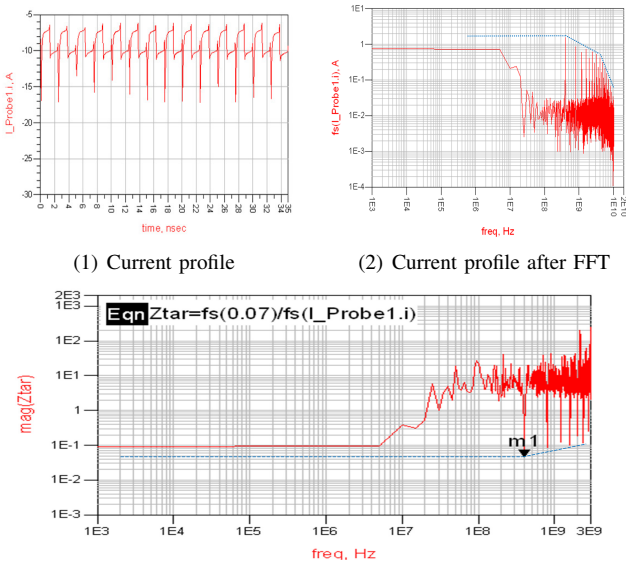


Fig. 3. To obtain the target impedance, we get the current profile in time domain (1) measured at the device on chip, and using FFT to translate it into frequency domain spectrum (2). Finally we derived the tolerable impedance at different frequency (3) from (2) and Eq(1), and the blue line is the target impedance.

### A. Power Distribution Network Model

The PDN includes VRM, decaps, and the interconnections of power grid on PCB, package, and die as shown in Fig.2. The voltage sent by VRM to chip will be derated by the resistance and inductance of the PDN interconnection. The fluctuation of voltage at the pads on chip may harm the circuit noise margin and cause those devices on chip malfunction[9]. Therefore, we have to shrink the fluctuation of voltage within an acceptable range to ensure the robustness of PDN.

### B. Target Impedance

To accurately estimate the target impedance, in this research we apply the approach in [8] to get the real current profile and then use fast fourier transform (FFT) to translate the time-domain current spectrum to frequency-domain current spectrum. Fig.3 is an example of current profile, it is measured from the devices on chip and it records the change of current as the devices switch. After the current profile is translated by using FFT, it would be a frequency-domain spectrum and represents the compositions of current distributed in every frequency, as shown in Fig.3(2). The peak is usually the operation frequency (clock frequency) and the PDN impedance should be below target impedance on this frequency. Since the regular switching of devices would draw the current and lead to regular voltage drop, this could be regarded as a recurrent noise, and it is the main source of noise.

$$V(f) = I(f)Z \quad (1)$$

We use Eq(1) to translate the frequency-dependent current spectrum to the frequency-dependent impedance spectrum, where  $V(f)$  is the allowed voltage ripple, and then we obtain the target impedance as the blue line in Fig.3(3).

### C. Objectives

In real designs, the most important criterion is to work correctly, so in order to ensure the PDN is stable, the first objective function in our algorithm is defined as

$$\min \int_{f_L}^{f_U} \text{penalty}(f) \times p(f) \quad (2)$$

where  $f_L$  and  $f_U$  are the lower and upper bound of interesting frequency respectively.  $p(f)$  is the part of PDN impedance exceeding the target impedance and  $\text{penalty}(f)$  is the penalty at each frequency. When we meet the target impedance, cost becomes the next important criterion in industry. When there are  $G$  decap combinations which could make the PDN impedance meet the target impedance, we use the following equation to choose the minimum cost one from those combinations:

$$\begin{aligned} \min \sum_{i=0}^M \text{cost}_i^g \times \text{decap}_i^g \\ \text{subject to } 0 \leq g \leq G \end{aligned} \quad (3)$$

The  $\text{cost}_i^g$  and  $\text{decap}_i^g$  denote the retail price and the decap used in  $i_{th}$  port in  $g_{th}$  combination and  $M$  is the number of predefined ports for decap insertion.

## III. METHODOLOGY

### A. Particle Swarm Optimization (PSO)

To implement PSO algorithm in our problem, we regard the entire solution space as a multi-dimension grid, and each predefined decap insertion port corresponds to a dimension. The specification-matched decaps of each port form the coordinates, as Fig.4 shows. In the beginning, there would be  $P$  particles

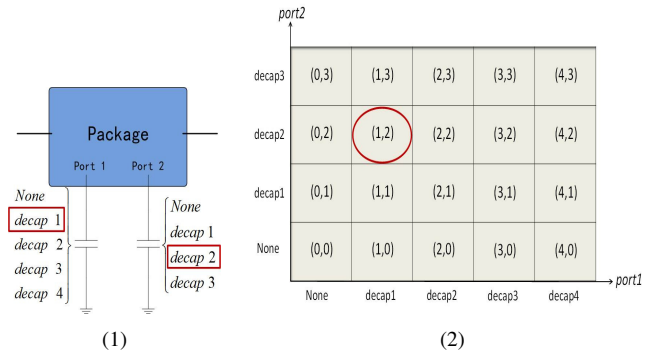


Fig. 4. An example showing the meaning of solution space in PSO. (1) is a package design with two predefined decap insertion ports, and there are several specification-matched decaps which could be chosen in each port. (2) is the discrete PSO solution space we map. If a particle is on (1,2), it means we choose decap1 for port1 and decap2 for port2. Besides, “None” means there is no decap placed in this port.

generated, and distributed randomly in the entire discrete solution space. Each particle is assigned a velocity randomly and that represents a solution to the optimization problem. Next, we calculate the fitness of all particles. Fitness is calculated according to the objective function(Eq(2)) in the optimization problem, and usually the lower fitness represents better solution. After the fitness calculation of all particles is performed, each particle memorizes its own fitness as its  $p_{best}$ , and best fitness of those particles is defined as global best  $g_{best}$ . If there are the particles whose fitness is 0, the  $p_{best}$  and  $g_{best}$  would be decided by objective function(Eq(3)). After that, particles adjust their position by the following equations:

$$v_i^{t+1} = \omega(t)v_i^t + p_1 r_1 (p_{best}^t - x_i^t) + p_2 r_2 (g_{best}^t - x_i^t) \quad (4)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (5)$$

where  $x_i^t$  is the position of  $i_{th}$  particle in  $t_{th}$  iteration, and  $v_i^t$  is the velocity of  $i_{th}$  particle in  $t_{th}$  iteration.  $p_{best}^t$  is the best solution ever found by  $i_{th}$  particle till  $t_{th}$  iteration, and  $g_{best}^t$  is the best solution ever found by all particles till  $t_{th}$  iteration.  $r_1$  and  $r_2$  are randomly number distributed between [0, 1].  $\omega$  is the “inertia”, and  $p_1$  and  $p_2$  are the coefficients of acceleration.

## B. Preferred Decap Choice (PDC)

As [14] shows, to reduce the impedance in the specified frequency range, using the combination of different decaps to make PDN impedance meet the target impedance is more effective than using the combination of the same decaps. Fig.5 demonstrates how we define a “Preferred” decap. When using the same total amount of decaps, if we use more “Preferred” decap, we could make the PDN impedance meet the target impedance easier. Therefore, the optimal solution of decap selection usually includes several “Preferred” decaps, we want the particles in PSO to search the area around the location with more “Preferred” decaps to find the optimal solution.

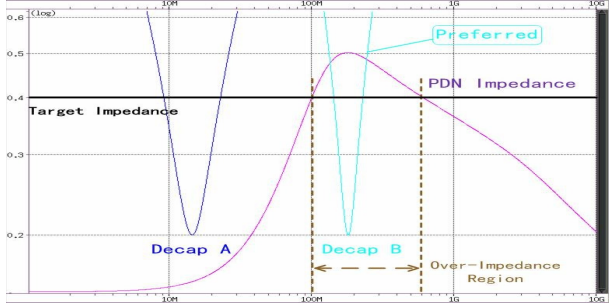


Fig. 5. To choose an effective decap from DecapA and DecapB, since the resonance frequency of DecapB is within the over-impedance region, it is more effective and we mark it as “Preferred”.

## C. PDC-PSO

The basic PSO usually chooses the decap whose self-resonance is not at the non-meeting target impedance frequency, and it wastes time to search the solution consisting of those decaps. In [5], the authors show that in the PSO algorithm, there would be a better result if  $p_2$  is less than “1” and  $p_1$  is between [4, 10], and  $p_1$  should decrease and  $p_2$  should increase as the number of iteration increases. Therefore, we would set parameter  $p_{1max}$ ,  $p_{1min}$ ,  $p_{2max}$  and  $p_{2min}$  to define the boundaries of  $p_1$  and  $p_2$ . We give more information about which decap should be chosen to make the PSO algorithm have higher chances to find the optimal solution.

In our algorithm, each particle has its own coefficients of acceleration  $p_1$  and  $p_2$ , and when a particle moves to a better location and updates  $pbest$  or  $gbest$ , the particle would check how many decaps it chooses are marked as “Preferred”, and renew its  $p_1$  and  $p_2$  according to Eq(6)(7)(8)(9).

$$p_{1new} = p_1 + \phi_1 \quad (6)$$

$$p_{2new} = p_2 + \phi_2 \quad (7)$$

$$\phi_1 = (p_{1max} - p_1) \times \frac{N_{Local}}{N_{Local} + N_{Global}} \quad (8)$$

$$\phi_2 = (p_{2max} - p_2) \times \frac{N_{Global}}{N_{Local} + N_{Global}} \quad (9)$$

where  $p_{1max}$  and  $p_{2max}$  are the user defined upper bound of  $p_1$  and  $p_2$ .  $N_{Local}$  is the amount of “Preferred” decaps used in  $pbest$  location, and  $N_{Global}$  is the amount of “Preferred” decaps used in  $gbest$  location. In the beginning of PDC-PSO, we set  $p_1$  for  $p_{1max}$ , and  $p_2$  for  $p_{2min}$ .

Since we know that the coefficients of acceleration influence the PSO significantly, we let every particle have its own  $p_1$  and  $p_2$ , and we know they are relative to  $pbest$  and  $gbest$  respectively, so we renew those coefficients when its  $pbest$  or  $gbest$  is updated. When the  $pbest$  or  $gbest$  location of a particle uses more “Preferred” decaps, we know that it has higher possibility that the global optimal solution is nearby. Eq(6) and Eq(8) demonstrate that if the  $pbest$  of a particle uses more “Preferred” decaps,  $p_1$  would be increased so that this

particle would tend to search the area around  $pbest$ . Similarly, if  $gbest$  uses more “Preferred” decaps than  $pbest$ , according to Eq(7) and Eq(9), the coefficient  $p_2$  would be larger than  $p_1$ , and that would make the particle tend to search the area around  $gbest$ . To avoid our algorithm using the decaps whose resonance frequencies are at the same over-impedance region and being trapped to local optimal solution, we set maximum capacity for each over-impedance region to prevent there are too many “Preferred” decaps in the same over-impedance region. The maximum capacity is decided by user. Once the amount of “Preferred” decaps used in  $pbest$  or  $gbest$  location is more than the maximum capacity, it would increase the maximum capacity in  $N_{Local}$  or  $N_{Global}$ , as Eq(10) shows.

For an over – impedance region,

if  $\#Decap_{Preferred} > Capacity$ , (10)

$$N_{Local(Global)} = N_{Local(Global)} + Capacity$$

Since we map the solution space into a multi-dimension grid, the particle must be on the first index of the dimension to let no decap be placed in the port. If the decap library of port is very large, the probability of a particle being on the first index would be small. Thus we add more none-decap-insertion locations, which means there is no decap placed in the port, in each dimension to increase the probability that particle would find the lower cost solution. In the process of PDC-PSO, if the best solution of particles let the PDN impedance meet the target impedance, when particles move to new locations, it would compare the cost of the decap combinations and set the lower cost one as its best solution. By those modifications in our algorithm, the particles would not be trapped in local best and we have more probability to find the global optimal solution.

## IV. EXPERIMENTAL RESULT

We implement our algorithm with C++ language and apply it to three package designs. We use HSPICE to get the PDN impedance. The package, PCB SPICE models are extracted by SIwave[1] for all cases, and the chip is modelled by a resistor ( $R_{chip}$ ) and a capacitor ( $C_{chip}$ ). The information of the three cases is shown in Table I respectively.

TABLE I  
INFORMATION OF ALL CASES. T-I MEANS THE TARGET IMPEDANCE.

Case Information			
	Case-1	Case-2	Case-3
Process	40nm	28nm	28nm
# ports	2	5	16
Op frequency	800Mhz	100Mhz	200Mhz
Supply voltage	1.5V	0.7V	0.9V
Voltage tolerance	10%	10%	10%
T-I(Performance)	0.75	0.01	0.036
T-I(cost)	0.9	0.1	0.07

### A. Cost Driven Decap Selection

We first show how to achieve lower design cost in decap selection. We slightly relax the target impedance of each case to let some of the predefined ports be empty and still could make PDN meet the target impedance, and apply algorithms to find the minimum cost decap combination. We set each decap cost to 1, that is, using less decaps stands for the lower cost. The cost of each decap could be set by users, and our algorithm would take that into consideration according to Eq(3).

In case 1, we could use only 1 decap to meet target impedance, and in case 2 and 3 we only use 2 and 5 decaps. This result and Fig.1 show that the manual and careless decap selection usually causes over-design, and our algorithm is

TABLE II

PEAK-TO-PEAK VOLTAGE FLUCTUATION COMPARISON. OUR METHODOLOGY COULD SUPPRESS THE SYSTEM VOLTAGE FLUCTUATION MORE EFFECTIVELY THAN RULES OF THUMB IN ALL CASES, AND MAKE THE PDN MORE ROBUST.

Case	Operation Frequency	Without Decap	With Original Decaps	Our Methodology	Improvement of Original Decaps(%)	Improvement of Our Methodology(%)
Case-1(2-port)	800MHz	338mv	253mv	247mv	25.15	26.92
Case-1(2-port)	90MHz	374mv	330mv	278mv	11.76	25.67
Case-2(5-port)	100MHz	724mv	1120mv	386mv	-54.7	46.69
Case-3(16-port)	200MHz	270mv	371mv	179mv	-37.4	33.7

effective in decap selection while maintaining the PDN stable and taking the cost into consideration.

### B. Optimizing Decap for Voltage Fluctuation Reduction

We run the SSN simulation to verify the decap combination for voltage fluctuation reduction, and the results are shown in Table II and Fig.6. We found that the improvement of Case-1 is minimal, it is because the PDN impedance at operation frequency is below the target impedance already. That means if there is no other noise, the PDN system without decap is stable. However, besides the operation frequency, there are still many noise occurred anywhere unexpectedly. To prevent the unexpected noise from causing the PDN system unstable, we should be conservative and make the entire frequency range meet the target impedance.

Although using both the decap combinations selected by rules of thumb and our program could maintain the PDN system within the specification 300mV at operation frequency, there might be unexpected noise in low, middle, and high frequency. Therefore, we measure the voltage fluctuation when there is a noise coming from PCB or chip at 90MHz in Case-1. The manual selection is not effective to suppress the noise, and our result could still keep the PDN system voltage fluctuation under 300mV, as shown in Table II.

Another problem we should take care is that sometimes the performance of PDN with decaps is worse than PDN without decaps since the anti-resonance might occur at the noisy frequency. As Fig.6 shows, the decaps selected by rules of thumb cause the voltage fluctuation larger than the original design without decaps. Therefore, choosing decap should consider its own characteristic rather than rules by thumb, otherwise we may obtain the PDN system worse than the original design.

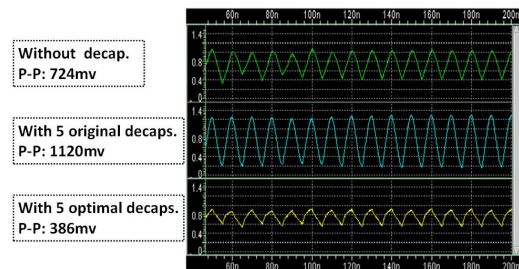


Fig. 6. The comparison of Case-2 for original and optimal decap combination in time domain on 100MHz (operation frequency). Compared to decaps chosen by hands, decaps chosen by our methodology could improve the system voltage fluctuation greatly. P-P means peak-to-peak.

## V. CONCLUSION

A well-designed PDN is essential for high speed system. To maintain the power integrity, adding decaps is an effective way. Since the more decaps would cost more money and area, how to choose decaps becomes a critical issue. In this paper, we introduce an efficient algorithm named “PDC-PSO” to optimize the type and location of decaps automatically. The results show that, compared to the decaps chosen by rules of thumb, our algorithm could effectively shrink the voltage fluctuation at pads

on chip within the tolerable range at the same or lower price in a relatively short execution time.

## REFERENCES

- [1] Ansys. <http://www.ansys.com/>.
- [2] Murata manufacturing co. <http://www.murata.com/>.
- [3] E. Bogatin. *Signal and Power Integrity - Simplified (2nd Edition)*. Prentice Hall, 2009.
- [4] J. Chen and L. He. Efficient in-package decoupling capacitor optimization for i/o power integrity. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 26(4):734–738, 2007.
- [5] J. Chen and L. He. Experimental analysis of acceleration coefficient in particle swarm optimization algorithm. *Computer Engineering*, 36(4), 2010.
- [6] R. Heald, K. Aingaran, C. Amir, M. Ang, M. Boland, P. Dixit, G. Gouldsberry, D. Greenley, J. Grinberg, J. Hart, T. Horel, W.-J. Hsu, J. Kaku, C. Kim, S. Kim, F. Klass, H. Kwan, G. Lauterbach, R. Lo, H. McIntyre, A. Mehta, D. Murata, S. Nguyen, Y.-P. Pai, S. Patel, K. Shin, K. Tam, S. Vishwanthaiha, J. Wu, G. Yee, and E. You. A third-generation sparc v9 64-b microprocessor. *IEEE Journal of Solid-State Circuits*, 35(11):1526–1538, 2000.
- [7] J. Kennedy and R. Eberhart. Particle swarm optimization. In *IEEE International Conference on Neural Networks, 1995. Proceedings.*, volume 4, pages 1942–1948 vol.4, 1995.
- [8] D. Lai. Achieve optimized power delivery using adaptive target impedance. <http://www.ansoft.com/firstpass/pdf/AchieveOptimizedPowerDelivery.pdf>.
- [9] H. Li, Z. Qi, S. Tan, L. Wu, Y. Cai, and X. Hong. Partitioning-based approach to fast on-chip decap budgeting and minimization. In *Design Automation Conference*, pages 170–175, 2005.
- [10] S. Nabil, A. El-Rouby, and A. Hussin. A complete solution for the power delivery system (pds) design for high-speed digital systems. In *International Conference on Design Technology of Integrated Systems in Nanoscale Era*, pages 179–183, 2009.
- [11] I. Novak. *Frequency-Domain Characterization of Power Distribution Networks*. Artech House Publishers, 2007.
- [12] Y. Shi, J. Xiong, C. Liu, and L. He. Efficient decoupling capacitance budgeting considering operation and process variations. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 27(7):1253–1263, 2008.
- [13] L. Smith, R. Anderson, D. Forehand, T. Pelc, and T. Roy. Power distribution system design methodology and capacitor selection for modern cmos technology. *IEEE Transactions on Advanced Packaging*, 22(3):284–291, 1999.
- [14] L. D. Smith. Frequency domain target impedance method for bypass capacitor selection for power distribution systems. *DesignCon*, 2006.
- [15] M. Swaminathan and A. Ege Engin. *Power Integrity Modeling and Design for Semiconductors and Systems*. Academic Internet Publishers, 2007.
- [16] J. Tripathi, R. Nagpal, N. Chhabra, R. Malik, and J. Mukherjee. Maintaining power integrity by damping the cavity-mode anti-resonances’ peaks on a power plane by particle swarm optimization. In *International Symposium on Quality Electronic Design (ISQED)*, pages 525–528, 2012.
- [17] A. Waizman, O. Vikinski, and G. Sizikov. Cpu power delivery impedance profile resonances impact on core fmax. In *IEEE Electrical Performance of Electronic Packaging*, pages 119–122, 2006.
- [18] Z. Wu and J. Zhou. A self-adaptive particle swarm optimization algorithm with individual coefficients adjustment. In *International Conference on Computational Intelligence and Security*, pages 133–136, 2007.
- [19] H. Yu, C. Chu, and L. He. Off-chip decoupling capacitor allocation for chip package co-design. In *Design Automation Conference*, pages 618–621, 2007.
- [20] H. Zheng, B. Krauter, and L. Pileggi. On-package decoupling optimization with package macromodels. In *Custom Integrated Circuits Conference*, pages 723–726, 2003.