Decoupling Network Optimization in High Speed Systems by Mixed-Integer Programming

Jai Narayan Tripathi[1], Ashutosh Mahajan[2], Jayanta Mukherjee[1], Raj Kumar Nagpal[3], Rakesh Malik[3], and Nitin Gupta[3]

[1] Department of Electrical Engineering, IIT Bombay, Mumbai, INDIA.

[2] Industrial Engineering & Operations Research, IIT Bombay, Mumbai, INDIA.

[3]TR&D, STMicroelectronics Pvt. Ltd., Greater Noida, India.

Email: [1]{jai, jayanta}@ee.iitb.ac.in, [2]amahajan@iitb.ac.in

[3]{nitin.chhabra, rajkumar.nagpal, rakesh.malik}@st.com

***Abstract*— Power Integrity is maintained in a high speed system by designing an efficient decoupling network. This paper provides a generic formulation for decoupling capacitor selection and placement problem which is solved by mixed-integer programming. A real-world example is presented for the same. The minimum number of capacitors that could achieve the target impedance over the desired frequency range are found along with their optimal locations. In order to solve an industrial problem, the s-parameters data of power plane geometry and capacitors are used for the accurate analysis including bulk capacitors and VRM.**

***Index Terms*— Power Integrity, Power Delivery Networks, Decoupling Capacitors, S-parameters, Mixed-Integer Programming.**

I. Introduction

Power Integrity (PI) is becoming one of the most important issues in high speed systems as the dimensions of the switching devices are reducing continuously. Power Integrity is a term associated with power delivery networks (PDNs) for ensuring the proper quality of power supply in a system. The power supply should be sufficient enough with proper stability and should be supplied with good efficiency, in order to maintain power integrity [1]. In off-chip PDNs, there are various components which have varying impedance profiles in different frequency ranges e.g. VRM, bulk capacitor, power planes, decoupling capacitors etc. Power planes are used for supplying power to the chip through package power nets and decoupling capacitors and have various resonance and anti-resonance peaks in their impedance profile due to cavities. Decoupling capacitors are mid-frequency capacitors that provide low impedance in the range of some hundreds of MHz and are used to damp the cavity mode peaks of power planes [2]. The selection of decoupling capacitors and their positions on the board affect the system performance [3]-[5].

Various researchers have solved the decoupling capacitor placement problem by metaheuristic techniques [6]-[8]. This paper attempts to solve the same problem using a different paradigm: deterministic optimization techniques that are mathematically more rigorous and ensure (under mild assumptions) optimality in a finite number of steps. Specifically, mixed-integer programming [10] is used as the number of capacitors available and the ports where the capacitors should be placed are integers, but their impedance profile is continuous in a frequency range.

II. Problem Formulation

The problem of decoupling capacitors selection and placement in any high speed system can be generically formulated as an integer optimization problem as follows:

$$\min_{x,y} \sum_{p=1}^{P}\sum_{c=1}^{C} x_{p,c} \tag{1}$$

$$s.t.\ F(y^i_{1_f}, y^r_{1_f}, \ldots, y^i_{P_f}, y^r_{P_f}) \leq Z_T\ \forall f \in [0, f^{max}], \tag{2}$$

$$y^r_{p_f} = \sum_{c=1}^{C} \hat{z}^r_{c_f} x_{p,c}\ \forall f \in [0, f^{max}],\ p \in \{1, \ldots, P\}, \tag{3}$$

$$y^i_{p_f} = \sum_{c=1}^{C} \hat{z}^i_{c_f} x_{p,c}\ \forall f \in [0, f^{max}],\ p \in \{1, \ldots, P\}, \tag{4}$$

$$\sum_{c=1}^{C} x_{p,c} \leq K \quad \forall p \in \{1, \ldots, P\}, \tag{5}$$

$$x_{p,c} \in \{0, 1\}\ \forall p \in \{1, \ldots, P\},\ c \in \{1, \ldots, C\}. \tag{6}$$

Here, x is a decision variable with $x_{p,c} = 1$ if capacitor c is placed in port p and 0 otherwise. Thus, x tells us what capacitors should be placed in each port. $y^r_{p_f}$ and $y^i_{p_f}$ are respectively the real and imaginary parts of the total admittance of the capacitors placed at a port p at frequency f. The variable y, that we use to denote the tuple $(y^i_{1_f}, y^r_{1_f}, \ldots, y^i_{P_f}, y^r_{P_f})$, is thus an auxiliary variable, whose value is uniquely determined once we fix x.

The optimization model minimizes the number of capacitors required to achieve the target impedance Z_T. f^{max} is the maximum frequency upto which the board operates. P is the number of ports available on the board where capacitors can be placed, and C is the total number of capacitors available along with their s-parameters files. F is a 'black-box' function which computes the cumulative impedance of the board at a frequency f when the total admittance of each port (due to capacitors placed there) is given. The given parameters $\hat{z}^r_{c_f}$ and

$\hat{z}^i_{c_f}$ are respectively the real and imaginary parts of admittance of a capacitor c at frequency f. K is the maximum number of capacitors which can be placed at any port.

There are several benefits of using a modeling approach like ours. First, there exist state-of-the-art optimization solvers, commercial as well as open-source for solving problems of various types. Second, such a model provides us flexibility to change the model if and when the need arises. Suppose, for instance, a different problem: one may want to know what is the best impedance that can be achieved by using at most T capacitors of any type. Then one can just change the model to

$$\begin{aligned}
&\min_{x,y,z} z \\
&s.t.\ F(y^i_{1_f}, y^r_{1_f}, \ldots, y^i_{P_f}, y^r_{P_f}) \leq z\ \forall f \in [0, f^{max}], \\
&y^r_{p_f} = \sum_{c=1}^{C} \hat{z}^r_{c_f} x_{p,c}\ \forall f \in [0, f^{max}],\ p \in \{1, \ldots, P\}, \\
&y^i_{p_f} = \sum_{c=1}^{C} \hat{z}^i_{c_f} x_{p,c}\ \forall f \in [0, f^{max}],\ p \in \{1, \ldots, P\}, \\
&\sum_{c=1}^{C} x_{p,c} \leq K \quad \forall p \in \{1, \ldots, P\}, \\
&x_{p,c} \in \{0,1\}\ \forall p \in \{1, \ldots, P\},\ c \in \{1, \ldots, C\}.
\end{aligned}$$

While the model has changed, the same algorithmic procedure can be applied with little modifications in setting up the problem. Third, one can study the mathematical structure of these models and attempt to solve these models exactly to optimality.

For the sake of clarity, we explain our methodology using only the first model. In Section V we present results obtained from both the models.

III. Optimization Procedure

Solving the optimization model is difficult because of two reasons.

1) We need an infinite number of variables $y^r_{p_f}, y^i_{p_f}$ as the set of frequencies is dense. Similarly, the constraints (2)-(4) are infinite in number, one for each frequency.
2) It is not easy to write the function F in an algebraic form that is convenient to solve using available integer programming solvers.

We tackle the first difficulty by considering only a finite number of frequencies in the interval $[0, f^{max}]$. By taking a sufficiently large number of discrete frequency points spaced appropriately in the interval, we can ensure that the impedance does not exceed the threshold Z_T at any frequency.

We address the second difficulty by taking linear approximations of the black-box function F. This approximation follows from the Taylor-series expansion of F. Given a point $\hat{y}$, the function F can be approximated in its neighborhood by

$$F(y) \approx F(\hat{y}) + \nabla F(\hat{y})^T (y - \hat{y}). \tag{7}$$

So we replace the intractable constraint (2) in our model by the linear constraint

$$F(\hat{y}) + \nabla F(\hat{y})^T (y - \hat{y}) \leq Z_T. \tag{8}$$

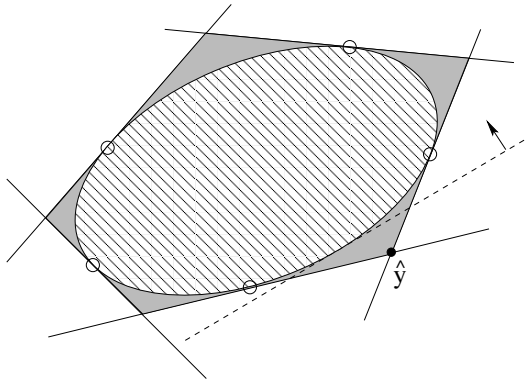

Fig. 1. The feasible region enclosed by a nonlinear constraint is shown by the shaded region. The linear approximation obtained at five different points includes the grey area that is not feasible. If the optimal point in grey area lies outside the feasible region, then we can add a linearization inequality (8) shown by the dotted line to cut it off.

Applying this idea at only one point ($\hat{y}$) results in a rather weak approximation. Hence, we add the constraint (8) at several different points. Selection of these points is an important step of our procedure that we will describe in the next section. The approximate model that we obtain in this way is an instance of mixed-integer linear programming (MILP). Even though MILPs are NP-hard in general, there are several fast and robust solvers available to solve them to optimality in reasonable time [11].

If F is a convex function, then the above approximation is also an under-estimator of F, i.e.,

$$F(\hat{y}) + \nabla F(\hat{y})^T (y - \hat{y}) \leq F(y). \tag{9}$$

In this case, constraint (8) is a relaxation of the original constraint (2). This useful property can be exploited to develop algorithms that guarantee optimal solution in a finite number of steps. We refer the readers to a recent survey [12] for more details. There is no such guarantee when the function is nonconvex, as is the case in our problem. In order to make our algorithm robust towards nonconvexity, we implement a dynamic procedure to add and remove the above approximations from our model depending upon the progress. We describe it next.

IV. Algorithm

Our procedure is based on the outer-approximation procedure developed by [13]. We start by including only a few linear inequalities of the model. In particular, we drop constraints (2) from our model. This model is now a linear problem with integer variables. It is solved by an MILP solver. Suppose the solution obtained is $(\hat{x}, \hat{y})$. If it satisfies all the nonlinear constraints (2), then we are done. Otherwise, there is some frequency f for which we have $F(\hat{y}^i_{1,f}, \hat{y}^r_{1,f}, \ldots, \hat{y}^i_{P,f}, \hat{y}^i_{P,f}) > Z_T$. We add to our model a linear approximation inequality (8) for this frequency f. Figure 1 explains this scheme pictorially. The point $\hat{y}$ clearly does not satisfy this new inequality. So this new inequality ensures that we do not obtain $\hat{y}$ when we solve the MILP in the next iteration. This iterative procedure is continued until we find a point that satisfies all nonlinear

inequalities or our MILP does not return any solution. The latter can happen if the problem has no integer solution (i.e. we provide a very low value of impedance Z_T) or if the MILP solver reaches a time limit.

Algorithm 1 depicts all the steps of our method. As we keep adding new inequalities, the effort required to solve the MILP increases in every iteration. Moreover, as our nonlinear function may not be convex, some of these inequalities may wrongly cut off some solution points. To overcome these difficulties, we call a routine 'cleanOldCons' in every iteration. Linear approximations that were generated a fixed number of iterations ago are removed if they are not active at the current iterate $(\hat{x}, \hat{y})$. More sophisticated versions of this routine can be developed for more efficiency.

Choose a discrete set I of frequencies from $[0, f_{max}]$.
Initialize $\hat{x}_{p,c} \leftarrow 0,\ p = 1, \ldots, P,\ c = 1, \ldots, C$.
Initialize $\hat{y}_{p,f} \leftarrow 0,\ p = 1, \ldots, P,\ f \in I$.
Initially create optimization model M with $\hat{x}, \hat{y}$.
Add objective function (1) and constraints (5), (6) to M.
foreach f *in* I **do** add constraints (5), (6) to M.
Initialize $iter \leftarrow 0$.
while $iter < MaxIter$ **do**
  $iter \leftarrow iter + 1$
  $newcons \leftarrow 0$
  foreach $f \in I$ **do**
    if $F(\hat{y}^i_{1_f}, \hat{y}^r_{1_f}, \ldots, \hat{y}^i_{P_f}, \hat{y}^r_{P_f}) > Z_T$ **then**
      Add constraint (8) for this frequency to M.
      $newcons \leftarrow newcons + 1$
  if $newcons == 0$ **then**
    STOP.
  Solve M using an MILP solver.
  if M *is infeasible* **then**
    STOP.
  else
    Update $\hat{x}, \hat{y}$ to the solution of M.
    cleanOldCons()

Algorithm 1: MILP Approximation Algorithm

V. Implementation

The maximum current of the system is 40 mA, supply voltage is 1.2 V and the tolerance is 3%. So, the target impedance for the system is 0.95Ω. The frequency range of interest for the analysis is 200 MHz which was calculated from the power spectral density (PSD) of the current required at the package pin. Thus, the impedance after placing decoupling capacitors should be lesser than 0.95 Ω at all the frequencies lesser than 200 MHz. The optimal number of capacitors (N) needed to achieve this, their names in the capacitor bank and their optimum locations are to be found.

The cumulative z-parameter matrix of a board loaded with decoupling capacitors can be given by the following formula [4].

$$Z_{eff} = (Z^{-1} + Z^{-1}_{decap})^{-1} \tag{10}$$

Here, the z-parameters matrix of the board is Z, and Z_{decap} is the diagonal matrix in which the diagonal elements are the impedance of the decoupling capacitors on the ports corresponding to the diagonal elements. All non-diagonal elements of Z_{decap} are zero. The problem with the above formula arises when the decoupling capacitors on the board are lesser than the available number of ports. In that case, one or more number of diagonal elements of matrix Z_{decap} are zero, which will not allow calculations of the inverse. To avoid this problem, we used y-parameters for that step particularly.

The analysis is carried out for a high speed serial link board. The extracted s-parameters file was having 39 ports. There are 39 ports in the board from which 4 ports are reserved, one for VRM and 3 for bulk capacitors, while one of the ports is the observation port where the package pin is connected. Thus, there are 34 ports available while defining or initializing the ports for decoupling capacitors. There are hundreds of capacitors (800 for this study) used for creating the bank. The capacitive effects of a capacitor are dominated by the inductive effects after a certain frequency called resonance frequency. After this resonance frequency, a capacitor starts behaving like an inductor, and at this frequency only resistive effects are effective. *rlc* models of capacitors may be very inaccurate at some frequencies [9]. Thus for this case study, s-parameters data is used.

Even if we suppose that each available port can have at most one capacitor from the capacitor bank for meeting the target impedance of the system, the total number of possible combinations is $800^{34} \approx 10^{99}$ which are impossible to enumerate using available computing systems.

A. Function and Gradient Evaluation

In every iteration of our algorithm, we need to evaluate the nonlinear function F (or Z_{eff}) in the formula (10). We also need gradient of F with respect to y variables for adding the linearization constraints (8). We obtained these gradients by method of finite differences. We perturb the point $\hat{y}$ and re-evaluate the function. Gradient is estimated by taking the ratio of the change in the function value and the change in a component of $\hat{y}$. Since there are P components of y we have to evaluate the function P times. This step can thus be time-consuming. In our experiments, we found that in the early stages of the algorithm, the time was comparable to the time spent by MILP solver.

B. Choice of the MILP solver

Branch-and-cut is a well established method for solving large scale MILP problems [13]. The algorithm starts by solving a linear programming relaxation obtained by ignoring any integrality constraints. If the solution satisfies integrality constraints, we are done. Otherwise, the relaxation is tightened by either adding new inequalities (cuts) or by branching. Modern solvers deploy several additional auxiliary techniques like presolving, primal heuristics and symmetry exploitation to enhance the speed of the algorithm. There are several implementations of branch-and-cut available: both commercial and open-source [11]. We have used IBM-CPLEX version

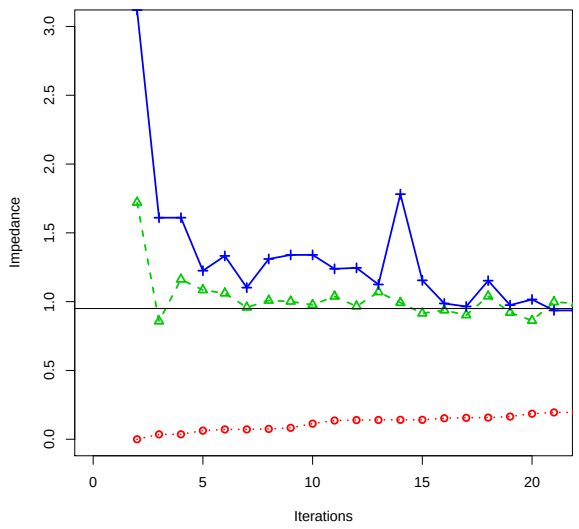

Fig. 2. A comparison of the results from the two optimization models.

12.5 in our experiments. It is capable of using multiple cores sharing a common memory.

C. Results

We implemented our algorithm in MATLAB and ran it on a computer with 64GB RAM and four Intel Xeon 2.3 GHz cores. We solved both models described in Section II. We considered the system of 34 available ports and 800 capacitors described above. The frequency range $[0, 200]$ MHz was approximated by selecting 401 equally spaced points. The resulting MILP initially had 27200 integer variables and 27302 linear constraints. With every iteration, the number of constraints increases as we keep adding more and more linearization inequalities. We chose $K = 5$ for the second model.

The behaviour of the two models is compared in Figure 2. The solid line denotes the impedance (in Ω) at a given iteration from the first model. The dashed line denotes the same for the second model. The dotted line is objective function value of the second model. The horizontal line is the target value. We observe that the second model provides solutions with smaller impedance values as compared to the first model. This behaviour is along expected lines because we are explicitly minimizing the impedance value while allowing upto five capacitors. The first model on the other hand always provides solutions that use fewer capacitors (2-3). The two models thus provide us a good set of solutions satisfying two desirable properties. The dotted line denotes the lower bound on the impedance when the number of capacitors is fixed to five. It increases monotonically because we are adding constraints in each iteration. This curve increases until it reaches one of the other two curves. The intersection point is theoretically the best possible solution. The gap between the curves shows that it may be possible to further reduce the impedance value.

We were able to find several solutions with impedance in range of $0.8\Omega - 0.95\Omega$. One such solution is shown in the table I.

TABLE I

LOCATION OF CAPACITORS FROM THE CAPACITOR BANK ON THE BOARD

Port	26	27	29	33	35
Capacitor	350	141	141	350	348

VI. CONCLUSION

We have developed an optimization model for a highly combinatorial problem of determining the optimal selection and placement of capacitors in a power delivery network. The linearization based algorithm that we deploy makes it possible to solve this problem. Our relatively simple implementation can be refined to make it faster and more accurate, which is something we will pursue in our future research. We also want to try nonlinear branch-and-bound methods for this problem, as nonlinear solvers are usually designed to handle nonconvex problems in a more robust fashion. Also, we have not used any existing algorithmic techniques to obtain the gradients [14], which can be faster and more accurate. Application of this approach to other combinatorial problems in design of high-speed systems is also possible.

REFERENCES

[1] Z. Mu, "Power Delivery System : Sufficiency, Efficiency, and Stability", 2008 9^{th} International Symposium on Quality Electronic Design, pp. 465-469, March 2008.

[2] M. Swaminathan and A. Ege Engin, Power Integrity Modeling and Design for Semiconductors and Systems, Prentice Hall, 2008.

[3] T. Hubing, "Effective strategies for choosing and locating printed circuit board decoupling capacitors", International Symposium on Electromagnetic Compatibility, pp.632-637, Aug. 2005.

[4] Kai-Bin Wu et al, "Optimization for the Locations of Decoupling Capacitors in Suppressing the Ground Bounce by Genetic Algorithm", Progress In Electromagnetic Research Symposium 2005, pp. 411-415, Hangzhou, China, August 2005.

[5] Jun Chen, Lei He, "Efficient In-Package Decoupling Capacitor Optimization for I/O Power Integrity", *IEEE Transaction on Computer Aided Design of Integrated Circuits and Systems*, Vol. 26 No. 4, April 2007.

[6] S. Kahng, "GA-Optimized Decoupling Capacitors Damping Power Bus' Cavity-Mode Resonances", *IEEE Microwave and Wireless Component Letters*, Vol.16, No.6, June 2006.

[7] J. N. Tripathi, R. K. Nagpal, N. K. 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", 2012 13^{th} International Symposium on Quality Electroic Design, pp. 525-528, 19-21 March 2012, Santa Clara, USA.

[8] J. N. Tripathi, N. K. Chhabra, R. K. Nagpal, R. Malik, and J. Mukherjee, "Damping the Cavity-Mode Anti-Resonances' Peaks on a Power Plane by Swarm Intelligence Algorithms", 2012 IEEE International Symposium on Circuits And Systems, pp. 361-364, Seoul, South Korea.

[9] J. N. Tripathi, N. K. Chhabra, R. K. Nagpal, R. Malik, and J. Mukherjee, "Power Integrity Analysis and Discrete Optimization of Decoupling Capacitors on High Speed Power Planes by Particle Swarm Optimization", 14^{th} International Symposium on Quality Electroic Design, March 2013, Santa Clara, USA.

[10] G. L. Nemhauser, L. A. Wolsey, "Integer and Combinatorial Optimization", John Wiley & Sons, Inc., 1988.

[11] J. T. Linderoth, A. Lodi, "MILP Software", In J. J. Cochran et al, editors, Wiley Encyclopedia of Operations Research and Management Science, John Wiley & Sons, Inc., 2010.

[12] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, A. Mahajan, "Mixed-integer nonlinear optimization", Acta Numerica, 22:1-131, May, 2013.

[13] M. A. Duran, I. E. Grossmann, "An outer-approximation algorithm for a class of mixed-integer nonlinear programs", Mathematical Programming, 36(2):307-339, 1982.

[14] A. Griewank, A. Walther, "Evaluating derivatives: Principles and techniques of algorithmic differentiation", second edition, SIAM, Philadelphia, 2008.