Passivity Enforcement using Incomplete Complex Frequency Hopping

Yi Qing Xiao, Muhammad Kabir, and Roni Khazaka

Department of Electrical and Computer Engineering, McGill University, Montréal, Québec, Canada, H3A 0E9
Email: yi.q.xiao@mail.mcgill.ca, muhammad.kabir@mail.mcgill.ca, and roni.khazaka@mcgill.ca

***Abstract*—In this paper, an efficient passivity enforcement algorithm for S-parameter based macromodels is proposed. The approach is based on the perturbation of the imaginary eigenvalues of the Hamiltonian Matrix. The CPU cost savings are obtained by selective computation and perturbation of a subset of the imaginary eigenvalues. The proposed approach is shown to be efficient compared to existing state of the art methods.**

***Index Terms*—Passivity Enforcement, Hamiltonian matrix, complex frequency hopping, Arnoldi algorithm, singular values, scattering.**

I. Introduction

An important requirement when modeling passive networks is ensuring the passivity of the macromodel. This is required in order to guarantee the stability of the simulation once terminations are added [1]. A number of approaches, such as [2–5], have been proposed for enforcing the passivity of a macromodel. An important class of such methods relies on the perturbation of the imaginary eigenvalues of the Hamiltonian matrix [2]. Such an approach can however be very CPU expensive due to the need to compute the eigenvalues of the typically very large Hamiltonian matrix. A solution to this problem was proposed in [6] which uses spectral transformation, the Arnodi process [7] and a frequency hopping scheme in order to compute all the imaginary eigenvalues of the Hamiltonian matrix without the need to compute its complete set of eigenvalues. This approach was later improved in [8], but can still be computationally expensive for cases where a large number of imaginary eigenvalues are present. In this paper, we propose an approach that is similar to that of [2] in its use of spectral transformation, however we use a modified Arnoldi process and frequency hopping scheme in order to efficiently compute a critical subset of the imaginary eigenvalues of the Hamiltonian rather than computing the full set. As is shown in the examples, this results in a CPU cost improvement for the passivity enforcement algorithm.
[1]

II. Problem Formulation

Consider a non-passive Linear Time Invariant (LTI) system in the form,

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t)\\ \boldsymbol{y}(t) &= \mathbf{C}\boldsymbol{x}(t) + \mathbf{D}\boldsymbol{u}(t),\end{aligned} \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{x}(t) \in \mathbb{R}^n$ is a vector of state variables. $\boldsymbol{u}(t) \in \mathbb{R}^p$ and $\boldsymbol{y}(t) \in \mathbb{R}^p$ are the incident and reflected power at the ports respectively. $\mathbf{B} \in \mathbb{R}^{n\times p}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$ and $\mathbf{D} \in \mathbb{R}^{p\times p}$ complete the system where n is the order and p is the number of ports. The transfer function of system (1) is:

$$\mathbf{G}(j\omega) = \mathbf{C}(j\omega\mathbf{I}_n - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \tag{2}$$

where $\mathbf{I}_n$ is an identity matrix of size n. $\mathbf{G}(j\omega)$ representes the S-parameter of the network, and a necessary and sufficient condition for passivity is that the singular values $\sigma(j\omega)$ of $\mathbf{G}(j\omega)$ remain below 1 at all $\omega \in \mathbb{R}$. System (1) describes a passive structure and our goal is to efficiently perturb this system in order to enforce passivity. In order to do so, we use an approach based on the perturbation of the imaginary eigenvalues of the Hamiltonian matrix [2] defined as,

$$\mathcal{H} = \begin{bmatrix}\mathbf{A} - \mathbf{B}\mathbf{T}^{-1}\mathbf{D}^T\mathbf{C} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\\ \mathbf{C}^T\mathbf{T}^{-1}\mathbf{C} & -\mathbf{A}^T + \mathbf{C}^T\mathbf{D}\mathbf{R}^{-1}\mathbf{B}^T\end{bmatrix} \tag{3}$$

where $\mathbf{T} = \mathbf{D}\mathbf{D}^T - \mathbf{I}_p$, $\mathbf{R} = \mathbf{D}^T\mathbf{D} - \mathbf{I}_p$. The imaginary eigenvalues of $\mathcal{H}$ have the property that the magnitude of every one of them is the angular frequency at which a singular value curve crosses the threshold 1 [9]. Perturbing the imaginary eigenvalues appropriately can eliminate the frequency regions over which $\sigma(j\omega) > 1$, enforcing passivity. The computation of the eigenvalues and eigenvectors of the large matrix $\mathcal{H}$ is CPU intensive. Our goal is to improve the CPU cost by proposing a perturbation scheme that uses only a subset of the imaginary eigenvalues, and a method similar to the approach in [6] for selectively computing these eigenvalues and their respective eigenvectors.

III. Frequency Hop Review

As proposed in [6], to approximate the closest eigenvalues to a hop point θ, we compute the shifted $\mathcal{H}$

$$\mathcal{H}_\theta = \mathcal{H} - \theta \times \mathbf{I}_{2n} \tag{4}$$

The Arnoldi process is then applied on $\mathcal{H}_\theta^{-1}$ using a Krylov subspace size of $k \ll 2n$ to approximate the eigenvalues of $\mathcal{H}$ which are closest to θ. Note that an efficient algorithm is available for computing the inverse of $\mathcal{H}_\theta$ [6]. In order to be able to define the circular region covered by each hop point, the N_λ closest eigenvalues to each hop point θ are forced to converge using an explicit restart and deflation scheme [6]. To compute all the imaginary eigenvalues, hop points are applied in a binary search pattern over the bandwidth of the system until the entirety of the bandwidth has been covered [6]. An improved version to [6] first utilizes a $\sigma(j\omega)$ scan to roughly delimit the non-passive regions, frequency region over which $\sigma(j\omega) > 1$, and then apply the frequency hop binary

[1]The authors would like to thank Natural Sciences and Engineering Research Council of Canada (NSERC) and the Regroupement Stratégique en Microsystèmes du Québec (ReSMiQ) for supporting this project.

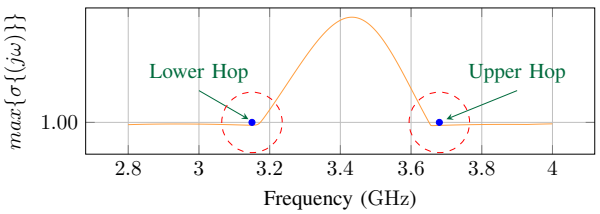

Fig. 1: Hop Point Selection Scheme

	Ex 1	Ex2
Syst. Order (n)	1502	1900
Num. of ports (p)	78	128
Num. of img. eig.	88	280
Num. of non-pass reg.	12	3

TABLE I: Simulation Example Details

	Ex 1			
Algorithm	B			A
(k,N_λ)	$(60, 6)$	$(60, 12)$	$(100, 12)$	/
Num of hops	108	74	74	53
Num of iter	8	8	8	7
Total time (s)	317.4	417.5	325.6	208.0
RMS error	6.25e-3	6.25e-3	6.25e-3	6.83e-3

TABLE II: Ex 1 Simulation Results

search only within these regions [8]. However, the improved algorithm can still be very expensive for cases of large number of ports and large number of imaginary eigenvalues.

IV. Proposed Passivity Enforcement Method

In this paper, we propose an algorithm where a shifted Arnoldi process similar to [6] is used to compute a subset of the imaginary eigenvalues of $\mathcal{H}$ instead of the full set.

A. Eigenvalue Selection at a Frequency Hop

The eigenvalues selected to be retained at a given hop are chosen as follows:

1) The approximated eigenvalues are ordered in ascending order of distance to θ.
2) The eigenvalues are retained from the start of the list up to the one before the first non-convergent eigenvalue.
3) The effective search radius is defined as the distance between θ and the furthest convergent eigenvalue on the retained list.

In the event the very first eigenvalue of the list is non-convergent, we would force its convergence using the explicit restarts process presented in [6] and retain only one eigenvalue.

B. Selection of Hop Points

In this paper, at each iteration of the enforcement algorithm, we only perturb the imaginary eigenvalues closest to the boundary of the passivity violation regions. To select the hop points, the non-passive regions are roughly delimited through a frequency sweep, and their borders' precision is refined by applying singular value sampling in binary search pattern. Then, we place the hops right outside the regions' borders. The selection scheme can be observed on Fig. 1. For simplicity, we perform the sweep using linearly distributed sampling points over the band, though more robust sweeps can be utilized [8].

C. Selection of the Order of the Krylov Subpsace

As the imaginary eigenvalues of $\mathcal{H}$ indicate frequency crossing points between $\sigma(j\omega)$ curves and 1, change in number of $\sigma(j\omega) > 1$ between consecutive frequency sampling points provides the minimum number of imaginary eigenvalues between the sampling points. Thus, by scanning the $\sigma(j\omega)$ sampling points near a hopping point, we obtain the minimum number of nearby imaginary eigenvalues d. We apply the heuristically chosen $k = 5 * d$ to encourage convergence of all nearby imaginary eigenvalues of θ. The range of allowed k is set as $[80, 150]$. The lower bound is chosen to avoid having non-convergent closest eigenvalues to θ. An upper bound to the Krylov subspace is chosen to avoid an increase in the Arnoldi process cost.

D. Perturbation of Imaginary Eigenvalues

Once the imaginary eigenvalues have been computed through the process described through sections IV-A, IV-B, and IV-C, they are perturbed in the direction according to the slope sign of the respective $\sigma(j\omega)$ curve crossing them. The $\sigma(j\omega)$ slope curve can be determined using the eigenvector of the corresponding eigenvalue through which the curve crosses 1 [2]. The perturbation amount is set as a factor of the effective search radius of the respective hop point where the eigenvalues are approximated. The whole process repeats iteratively until no non-passive regions are detected, and at that point a final brute force computation of the eigenvalues of $\mathcal{H}$ is performed to ensure passivity of the perturbed system.

V. Simulation Results

Two examples are presented to illustrate the proposed method, and their specifications are listed on Table I. Both models are Lowener Matrix models purposely generated at lower order to obtain non-passive behavior [10]. We compare two algorithms for passivity enforcement. Algorithm A is the proposed algorithm described in section IV. Algorithm B is similar to the one presented in [6], but with binary search applied only within non-passive regions as suggested in [8]. Note that in both cases we use a simple linear frequency sweep with refinement to identify the non-passive regions, although a more efficient adaptive approach can also be used [8]. For Algorithm B, three combinations of parameter pair (k,N_λ) are used to provide more insight [6]. It should be noted that for both algorithms, a full $\mathcal{H}$ matrix eigenvalue computation is performed at the final iteration to ensure there are no imaginary eigenvalues left. The overall simulation results are presented on Table II for Ex 1 and on Table III for Ex 2.

From Table II, we observe that the Algorithm A utilized nearly 110 s less computation time compared to the best

	Ex 2			
Algorithm	B			A
(k,N_λ)	$(60,6)$	$(60,12)$	$(100,12)$	/
Num of hops	/	60	59	21
Num of iter	/	5	5	6
Total time (s)	/	552.5	477.7	254.6
RMS error	/	1.22e-4	1.22e-4	1.44e-4

TABLE III: Ex 2 Simulation Results

case scenario of Algorithm B, (k, N_λ)=$(60, 6)$. This can be attributed to a combination of factors such as fewer hops required, difference of 1 iteration required, and no restarts process for the Algorithm A. We also observe that for Algorithm B, when (k, N_λ)=$(60, 12)$ is applied, though the number of hops required for convergence is less than the case of (k, N_λ)=$(60, 6)$, the computation time rose by $\sim$100 s. This difference is attributed to the fact that $N_\lambda = 12$ is a large burden imposed of a small Krylov subspace of size 60 which introduces the need for many restarts. However, when we use parameters (k, N_λ)=$(100, 12)$, though almost the same number of hops were required, the computation time is lower than the case of (k, N_λ)=$(60, 12)$, because a larger Krylov subspace provided better convergence and required a smaller number of restarts.

Transient simulations are performed using the equivalent SPICE netlist system to Ex 1's non-passive model and passive models obtained using passivity enforcement Algorithm A and Algorithm B. The simulations were performed by inputting a pulse with 1 V amplitude, 6 ns width, and 100 ps rise and fall time. The voltage at port 1 is plotted on Fig. 3 and we observe that all three models have very similar response.

For Ex 2, Algorithm B with parameters (k, N_λ)=$(60, 6)$ encountered the scenario where a hopping point happens to land too near an imaginary eigenvalue during the binary search. We detect this scenario when the Arnoldi process is unable to converge after more than 60 restart attempts. This scenario occurred readily in Ex 2 because the third non-passive region of this model has 248 imaginary eigenvalues, some of which are very densely packed. When a small number of eigenvalues N_λ are retained at each hop, the binary search will converge to a dense cluster of eigenvalues leading a hop point to be very close to an eigenvalue. To account for the random factor due to randomly chosen initial vector at the start of each Arnoldi process, we ran the perturbation Algorithm B 10 times in succession for Ex 2, all of which encountered this scenario when parameters were set at (k,N_λ)=(60,6). This suggest that an adaptive choice of k and N_λ could be useful for Algorithm B. In Algorithm A, because the hop points are selected outside the non-passive regions, such scenario does not occur.

VI. Conclusion

In this paper, an incomplete frequency hopping technique was proposed to efficiently compute and perturb a subset of the imaginary eigenvalues of the Hamiltonian matrix that are close to the edges of the passivity violation region at each iteration. The proposed approach is shown to be robust and efficient using numerical examples.

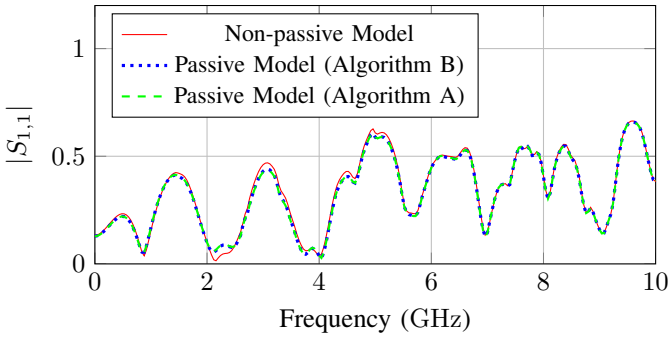

Fig. 2: Ex 1 $|S_{1,1}|$ Plot

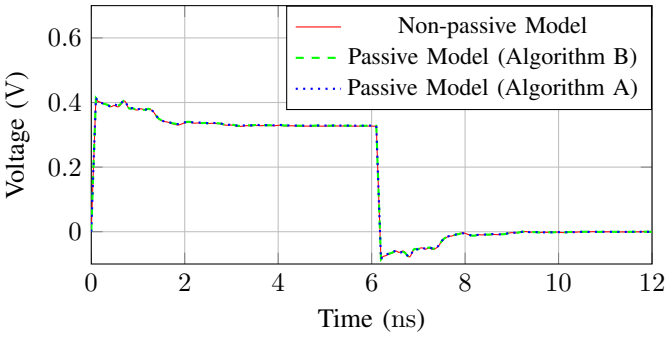

Fig. 3: Ex 1's transient plot.

References

[1] R. Achar and M. Nakhla, "Simulation of high-speed interconnects," *Proceedings of the IEEE*, vol. 89, no. 5, pp. 693 –728, may 2001.

[2] S. Grivet-Talocia, "Passivity enforcement via perturbation of hamiltonian matrices," *IEEE Trans. Circuits Syst.*, vol. 51, no. 9, pp. 1755 – 1769, 2004.

[3] B. Gustavsen, "Fast passivity enforcement for s-parameter models by perturbation of residue matrix eigenvalues," *IEEE Trans. Adv. Packag.*, vol. 33, no. 1, pp. 257–265, 2010.

[4] G. C. Calafiore, "Subgradient techniques for passivity enforcement of linear device and interconnect macromodels," *IEEE Trans. Microw. Theory Tech.*, vol. 60, no. 10, pp. 2990–3003, 2012.

[5] C. P. Coelho, J. Phillips, and L. M. Silveira, "A convex programming approach for generating guaranteed passive approximations to tabulated frequency-data," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 23, no. 2, pp. 293–301, 2004.

[6] S. Grivet-Talocia and A. Ubolli, "On the generation of large passive macromodels for complex interconnect structures," *IEEE Trans. Adv. Packag.*, vol. 29, no. 1, pp. 39–54, Feb 2006.

[7] Y. Saad, *Numerical methods for large eigenvalue problems*. SIAM, 2011.

[8] S. Grivet-Talocia, "An adaptive sampling technique for passivity characterization and enforcement of large interconnect macromodels," *IEEE Trans. Adv. Packag.*, vol. 30, no. 2, pp. 226–237, May 2007.

[9] S. Boyd, V. Balakrishman, and P. Kabamba, "A bisection method for computing the h-infinity-norm of a transfer matrix and related problems," *Mathematics of Control, Signals, and Systems*, vol. 27, no. 101, pp. 207–219, 1989.

[10] M. T. Kassis, M. Kabir, Y. Q. Xiao, and R. Khazaka, "Passive reduced order macromodeling based on loewner matrix interpolation," *IEEE Trans. Microw. Theory Tech.*, vol. 64, no. 8, 2016.