MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

# Stochastically Approximated Multiobjective Optimization of Dual Input Digital Doherty Power Amplifier

Niu, S.; Ma, R.; Manjunatha, K.

TR2017-163 November 2017

**Abstract**

In this work, we propose a novel adaptive control on digital Doherty Power Amplifier (DPA) for efficiency andgain enhancement. Unlike traditional DPA design, we propose Simultaneously Perturbated Stochastic Approximation (SPSA) optimization algorithm by considering phase difference, power distribution and gate voltage parameters of main and peak amplifiers in order to achieve optimal performance. The optimal performance is defined as good linearity, Power Added Efficiency (PAE) and gain. We mainly investigate the efficiency and gain improvements since the linearity can be separately addressed by traditional DPD techniques. Even though we found that the cost function of optimization exists in several local optimums, they are very close, and thereby local minimum is not of great concern in this optimization. In addition, we approximate the DPA circuit model for the fast verification using regression model. Finally, we exam the algorithm in SystemVue and ADS co-simulation environment.

*IEEE International Workshop on Computational Intelligence and Applications*

# Stochastically Approximated Multiobjective Optimization of Dual Input Digital Doherty Power Amplifier

Sufeng Niu[1,3], Koushik A.M[1,4], Rui Ma[1] , Koon Hoo Teo[1] , Shintaro Shinjo[2] and Yuji Komatsuzaki[2]
[1]Mitsubishi Electric Research Lab, Cambridge, MA, USA
[2] Mitsubishi Electric Corporation, Kamakura, Japan
[3] Clemson University, SC, USA
[4] The Univerisity of Alabama, AL, USA
rma@merl.com

***Abstract*—In this work, we propose a novel adaptive control on digital Doherty Power Amplifier (DPA) for efficiency and gain enhancement. Unlike traditional DPA design, we propose Simultaneously Perturbated Stochastic Approximation (SPSA) optimization algorithm by considering phase difference, power distribution and gate voltage parameters of main and peak amplifiers in order to achieve optimal performance. The optimal performance is defined as good linearity, Power Added Efficiency (PAE) and gain. We mainly investigate the efficiency and gain improvements since the linearity can be separately addressed by traditional DPD techniques. Even though we found that the cost function of optimization exists in several local optimums, they are very close, and thereby local minimum is not of great concern in this optimization. In addition, we approximate the DPA circuit model for the fast verification using regression model. Finally, we exam the algorithm in SystemVue and ADS co-simulation environment.**

***Index Terms*—Doherty Power Amplifier, adaptive control, SPSA.**

## I. Introduction

Traditional Doherty Power Amplifier (DPA) offers high efficiency over limited RF bandwidth. In order to resolve efficiency and bandwidth bottleneck while maintaining the simple DPA configuration, we propose a digital DPA [1] based adaptive control to optimize its efficiency, gain and linearity.

Traditional DPA (Fig. 1-(top)) design which is based on single-end input configuration consists of an analog power splitter, tuned phase aligner, carrier PA at class-AB and peak PA at class-C mode, and output combiner. In order to improve DPA performance, the designer needs to manually tune the circuit parameters, and the tuning process only specifies for fixed parameters such as input power, frequency, etc. While in the practical scenarios, the optimal control parameters vary for different inputs and circuit states, which is a common problem in pure analog based design.

Unlike traditional DPA, digital DPA (Fig. 1-(bottom)) is programmable so that it can reduce circuit tuning complexity for designer and can fully take into account circuit impairments. Therefore, digital DPA is flexible providing better performance compared with analog DPA. In our work, we propose a dual input digital DPA based on online optimization technique to adaptively tune the control parameters of DPA. To the best of our knowledge, this work reports the first online learning based optimization of dual input DPA with the state-of-the-art performance.

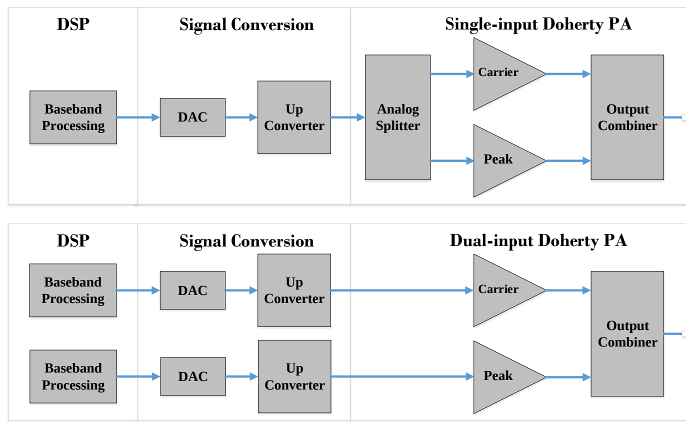


Fig. 1: Single input traditional DPA (top) and, Dual-input digital DPA (bottom)

## II. Related Work

There are many research works that have been done towards optimizing the performance of the DPA. Authors in [2][3], using tradition DPA proposed an uneaven power distribution mechanism where more input power was delivered to the peaking amplifier rather than the main amplifier for optimized linear power operation and envelope tracking[3] to increase the load modulation efficiency. Besides, work [4] proposed asymmetric power distribution for improved linearity, efficiency and peak envelope power by increasing the size of the peaking amplifier transistor and its conduction angle. In a recent work [5], a two stage DPA with an optimized current for the peak amplifier by making use of two stage opearion capability of the DPA. In addition, works [6] and [7], proposed one DPD configuration for dual-input nonlinearity: main PA utilize 2D Generalized Memory Polynomial (GMP) based DPD model. The efficiency-optimized function is a polynomial function, and the coefficients are extracted via sweeping both amplitudes of the RF and envelop input signals. Since traditional analog DPA requires cumbersome configurations, [2], [3], [4],and [5] use the LUT-based method in DSP to compensate the phase misalignment of main PA and peak PA. Recently, a machine learning approach, Multiobjective Bayesian Optimization for Active Load Modulation for Doherty amplifier is proposed in work [8] which can optimize the DPA's matching networks to align the desired and the realized impedance trajectories both at saturation and power backoff from $1.5GHz$ to 2.4 $GHz$. This work still considers single input DPA in their work. To achieve highly efficient operation of RF power amplifier requires normally driving PA to its saturation, which unfortunately, results in distortion of signal. In our project, we focus on dual-input DDPA with aim to achieve optimal PAE, Gain and linearity.

## III. Proposed model

Our proposed approach is to adaptively optimize the input parameters on the fly. As we will show in later sections that different performance goals contain the trade-off, our algorithm still offers high flexibility and efficiency to obtain a variety of optimal solutions based on design constraints. Additionally, our method is based on black-box optimization, and thereby it suits for a variety of PA without engineering tuning.

### A. RF configuration

As we mentioned previously, our RF circuit configuration doesn't require hand-tuned phase shifter and analog splitter. Our design goal is to maximize the PAE, while maintaining high gain and good linearity. Traditional digital linearity is achieved through Digital Pre-Distortion (DPD). We decoupled the problem into linearity and efficiency, since the linearity can be resolved by DPD, which is out of scope of this paper.

We have four control ports: power ratio, phase alignment, main gate bias, peak gate bias, illustrated as follows:

*1)* Power ratio, $\alpha$*:* Power ratio controls the amount of input power distribution between both main PA branch and peak PA branch as follows,

$$P_{main} = \alpha \cdot P_{in} \quad W; \quad P_{peak} = (1-\alpha) \cdot P_{in} \quad W \tag{1}$$

Where $P_{in}$ is the input power to the DPA. The ratio could control the input power of peak PA to be completely off during the low power region to prevent unnecessary power leakage

*2) Phase alignment, $\phi$:* phase alignment between main PA path and peak PA path is critical which affects overall performance (gain, linearity,and PAE), but phase varied based on many known and unknown factors, such as gate bias, temperature, bandwidth, and modulation format. Hence, parameter $\phi$ defines the phase difference between the main PA and peak PA input signals. Note that the control ports contains phase alignment component, and running phase shift on RF frequency is very challenging on digital circuit. We therefore implement the phase shift issue at the baseband signal as follows:

$$y_m(t) = Re\{A_m \cdot e^{j(2\pi f_m t + \phi_m)}\} \tag{2}$$

Where $A_m$, $f_m$, and $\phi_m$ are the amplitude, frequency and phase of the based band signal. Assuming that the carrier signal follows the pattern $y_c = Re\{A_c \cdot e^{j(2\pi f_c t + \xi_c)}\}$ with carrier frequency $f_c$ and initial phase $\xi_c$, the modulated signal can be expressed as:

$$y_{RF} = y_c \cdot y_m = Re\{A_m \cdot A_c exp^{j(2\pi(f_c+f_m)t+\xi_c+\phi_m)}\} \tag{3}$$

*3) Main gate bias, $Vgs_m$ and Peak Gate bias,$Vgs_p$:* gate bias adjusts the amplifier performance over temperature as well as RF signal. Gate bias, together with power ratio, $\alpha$, decides the output contribution ratio for main PA and peak PA. Note that bias configuration is also affecting phase alignment.

According to ADS simulation, the optimal control ports only determined by input configurations, such as input power level, carrier frequency, etc. This is caused by the simplified model from EDA tools, but in real cases there exist other factors affecting the circuit dynamics (ex: temperature).

Fortunately, some latent factors such as temperature varied slowly in time, and in small time window we can treat them as constants so that these dynamics can be ignored. If we take into account the dynamics in real time, this becomes a typical control problem. Besides, we assume that the circuit model are unknown, then, the problem is actually a reinforcement learning. Solving reinforcement learning on DPA circuit is not trivial due that state space and action space are both in continue space, and DPA circuit normally running at RF frequency, which is very difficult for digital circuit to control in real time.

If we ignore the minor dynamics, the optimal control parameters only depends on the input configurations, which becomes an optimization problem. To compensate these minor dynamics, we simply re-estimate parameters of the circuit in a temporal window. This method is simple due to gradient-free approach but effective.

The optimization problem can be defined as the optimal control parameter, $u^*$ with loss $J(u^*)$ defined as:

$$u^* = \arg\min_{u \in U} J(u) \tag{4}$$

Where we use $J(u)$ to represent DPA objective output since it does not have explicity mathmatic model, details are described in Section III-C. While the updates rule to find $u$ based on:

$$u_{n+1} = u_n - \lambda_n \cdot \hat{g}_n(u_n) \tag{5}$$

Where $\lambda_n$ is the learning rate which decays in every iteration. In our DPA circuit, we relaxed our problem that we assume the function which projects control ports to PAE and gain remains unknown, and thereby $\hat{g}_n(u_n)$ is an estimated gradient. Recall that Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm [9][10], in order to estimate the gradients (we assume control ports form a p-dimensional vector), we have:

$$\hat{g}_n(u_n) = \begin{bmatrix} \frac{J(u_n+c_n\delta_0)-J(u_n-c_n\delta_0)}{2c_n\delta_0} \\ \frac{J(u_n+c_n\delta_1)-J(u_n-c_n\delta_1)}{2c_n\delta_1} \\ \vdots \\ \frac{J(u_n+c_n\delta_p)-J(u_n-c_n\delta_p)}{2c_n\delta_p} \end{bmatrix} \tag{6}$$

And $\delta$ is random perturbation vector that control the search directions. Simply replacing the $\hat{g}_n(u_n)$ in (5) with (6) will find the local optimal.

Baysian optimization is an alternatives compared with SPSA algorithm. [8] utilizes Baysian optimization to search the PA design varaibles, while our work falls into the adaptive control problem. Unlike design variables in [8], control ports needs to be re-estimated for certian periods, requiring efficient computation whereas baysian optimization is restricted to problems of moderate dimensions [11]. Additionally, the underlying DPA dynamics are affected by many hidden factors and thus the control ports are not stationary whereas common Bayisan optimizer assume parameteric prior like Gaussian process to be stationary.

### B. Local Optima and Global Optima

It turns out that the DPA optimization problem is highly non-convex and NP-hard to solve. In order to visualize convexity of the object function, for intuitive purpose, we plot the cost function on PAE as optimization goal shown in Fig. 2. Because that the control port is a 4-dimensional vector, we use TSNE [12] for dimension reduction while keeping original high dimensional data distribution. In our ADS sweep setup, we set very coarse step size for less computation time and to cover wide range of control parameters. In Fig. 2, the dots represent the cost function and red dots are the optimal points in which PAE ranges between *60%* to *70%*.

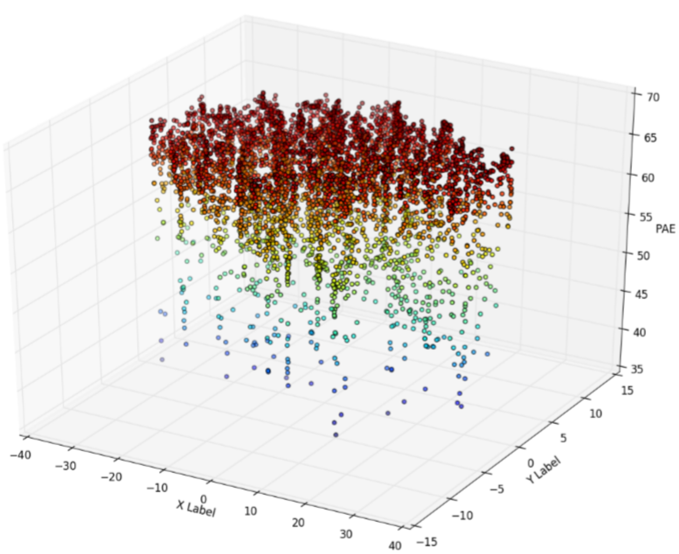


Fig. 2: Objective function of our DPA model based on ADS swept results: X label and Y label are dimension reduced control ports, they don't have any physical meaning but used for visualization purpose.

As shown in Fig. 2 that the DPA optimization is highly non-convex, but it contains very large amount of the local minimums and they are very close. It is also interpretable that peak gate bias is coupled with peak branch phase (For example, while changing the peak gate bias that affects peak branch phase, phase shifter should adaptively align to keep high PAE) and power ratio is correlated with peak gate bias, which also affects main gate bias. All these coupled relations lead the optimization problem to multiple solutions, and thereby cost function is non-convex.

Even though gradient based optimization algorithm would easily fall into the local optimum, adding randomness can overcome such issue. In our algorithm, by using multiple random initialization helps us to obtain multiple optimal points, but as we will show many optimal are close, in terms of the objective function.

### C. Optimization Algorithm

**Algorithm 1** SPSA based optimization of digital DPA

```
Input: θ = [Vgs_m, Vgs_p, φ, α]; % Initial control parameters
Input: θ_L = [Vgs_mL, Vgs_p,L, φ_L, α_L]; % Lower bound
Input: θ_U = [Vgs_mU, Vgs_p,U, φ_U, α_U]; % Upper bound
Input: c, λ_0 γ and γ_1; %  perturbation parameters
Output: θ* % Optimal control parameters
while adaptation==True do
    while converge==False do
        k++
        clip θ between θ_L and θ_U
        c_k = c / kγ
        λ_k = λ / (λ_0 + k)^α
        Δ = Bernouli(1, p) ; % Bernouli perturbation
        θ_+ = theta + c_k · Δ; % +ve perturbation
        θ_- = theta − c_k · Δ; % -ve perturbation
        Determine C(θ_-) and C(θ_+)
        Calculate: g = (C(θ_+) − C(θ_-)) / (2·c_k·Δ)
        Update: θ = θ − λ_k·g
    end while
    Obtain optimal control parameter, θ*
end while
```

The optimization procedure for digital DPA is shown in Algorithm 1. In the previous section, our optimization was with respect to single objective function, PAE. Whereas, here we set the optimization goal as multi-task optimization, which optimize amplifier gain and PAE simultaneously. Gain typically varies from *10 dB* to *16 dB*, while the PAE numerically ranges from *30%* to *70%*. In addition, we set an additional factor $\omega$, a weight factor to decide the importance of each parameter of the objective function. The weight factor could be either fixed or variable type. The varied weight offered a flexible way for objective function control parameter based on design specifications and requirements. For example, when user sets $\omega$ as function of input power, the optimization could prefer high PAE in low input power range, while high gain in high input power range. In our current scenario, we have set $\omega = 0.75$ as we performed our experiment at input power $Pin = 28dBm$. In general, it can be set as $\omega = \frac{Pin}{Pin_{max}}$.The optimization equation with modified objective function is defined as follows,

$$\begin{aligned} &\textbf{max}\ C(\theta) = \omega * Gain + (1-\omega) * PAE \\ &\textbf{s.t:}\quad \theta_L \le \theta \le \theta_U \end{aligned} \tag{7}$$

where $\theta$ is the vector of control parameters defined as, $\theta = [Vgs_m, Vgs_p, \phi, \alpha]$ which is bounded by upper and lower bounds $\theta_U$ and $\theta_L$ respectively.

For SPSA, proper initialization of control parameters is critical to search within the limit to achieve convergence speed and optimal control performance (PAE, gain). We use random initialization of parameters which could overcome the local optima and can stay close to the global optima. We also consider that there exists circuit dynamics but to simplify the problem, we assume that these dynamics are slowly changing, for example, temperature of the amplifier doesn't change rapidly. Re-estimating the circuit parameters periodically is the most simple and effective way to keep the DPA working in optimal condition.

## IV. Benchmark and Results

We simulate the adaption algorithm in ADS EDA co-simulated with SystemVue. To verify the algorithm, we use a regression model to approximate DPA model. Then based on this approximated model, we optimize the control parameters. After verifying the algorithm, we co-simulate it in ADS and SystemVue. More detailed setup is described in later part of the section. Finally, we illustrate how to implement the algorithm in the Test Bench environment for real devices.

### *A. Approximated Regression Model*

Approximation of the optimization algorithms serves two main purposes: A) quick verification of the optimization algorithm. B) before operate on the real device, instead of directly optimizing in real time test-bench, we can perform adaption on the regression model[13] and we can take the optimized parameters and use those parameters in real time test-bench. We experimented several regression techniques based on DPA swept data shown in Table I.

| ***Regression Accuracy*** | | |
|---|---|---|
| **Algorithm** | **Standardization** | **Accuracy (MSE)** |
| *Polynomial + L2 norm* | No | 0.012 |
| *Polynomial + L2 norm* | Yes | 0.012 |
| *Neural Network (6-layer)* | No | 0.057 |
| *Neural Network (6-layer)* | Yes | 0.00027 |

TABLE I: Regression model performance and comparison

We swept DPA parameters in ADS and stored them as regression training/testing set. The training and testing dataset is based upon cross-validation for better generalization purpose. Note that for dumped data, input is a continous wave(CW) signal running at *3.6 GHz*, and input power increases steadily from *22 dBm* to *29 dBm*. We adopted Polynomial regression via scikit-learn Python package, and we found that when the polynomial order is set to 3, the accuracy is reasonably good (we set the PAE and gain as label). Higher order will cause overfitting. L2 normalization is used as prior to normalize the input data. The accuracy is measured based on Mean Squared Error (MSE) (loss criterion is also MSE). Additionally, standard neural network based regression, which is implemented by 6-layer feedforward neural network with rectifier activation using Tensorflow [13] shows worse results than the polynomial regression. With the pre-processing by standardization, which normalize the feature with the same mean and variance, significantly improves the prediction accuracy.

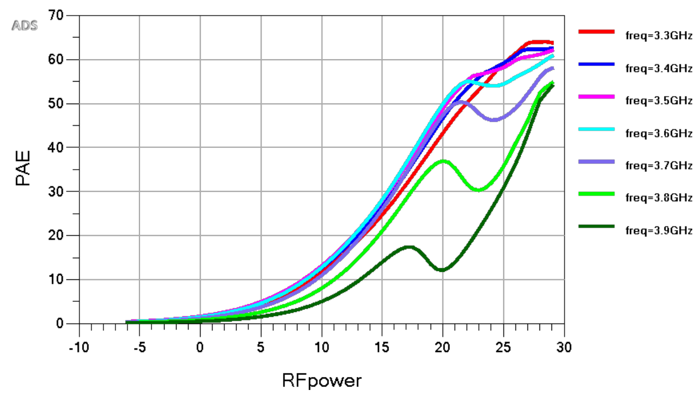


Fig. 3: Single-ended DPA PAE for different frequency

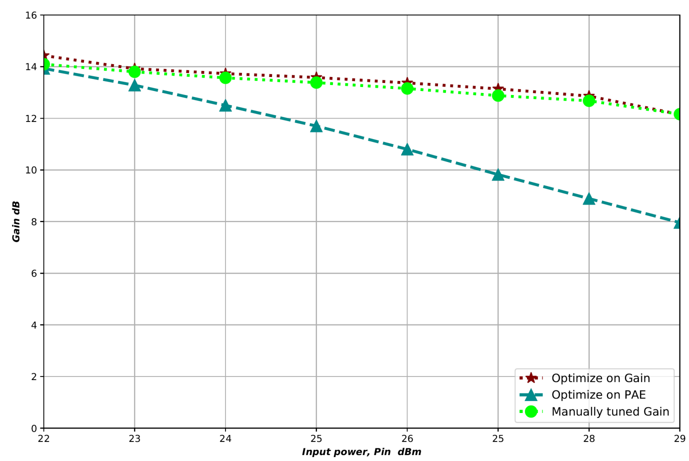


Fig. 4: Gain performance when we use different optimization criterion.

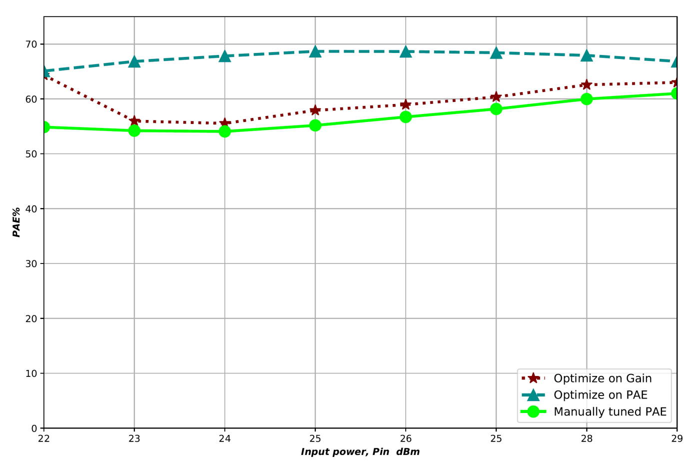


Fig. 5: PAE performance when we use different optimization criterion.

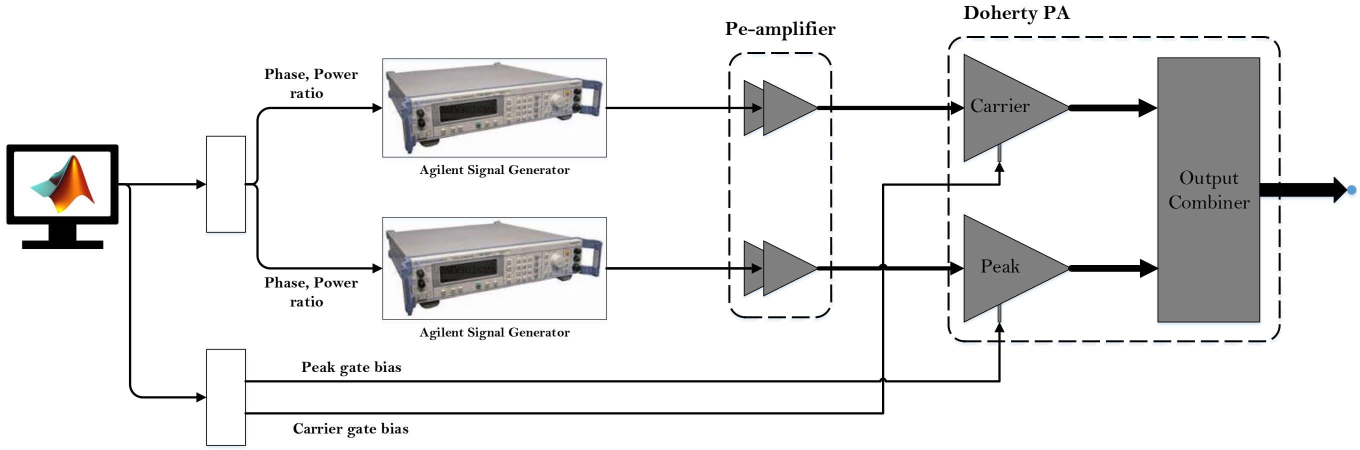


Fig. 6: Testbech setup.

Based on the regression model, we perform the Algorithm 1 to achieve optimized PAE. Our comparison base line is a single ended analog DPA design. The parameters are manually tuned to obtain the optimal PAE for 3.6 *GHz*. Therefore, the performance dropped when operating other frequencies shown in Fig. 3 (non-adaptive traditional design).

In our design, instead of engineering tuning, we quickly run the optimization. Again, the origin curve performance is tuned specified for *3.6 GHz* with non-adaptive property. Running in other frequency is much worse than the baseline, while our algorithm is able to adapt different frequencies and input power. We found that PAE and gain is also trade-off shown in Fig. 4 and 5 (input power from *22 dBm* to *29 dBm*): when we optimize only on PAE, gain decrease heavily shown as red line. Instead, if we optimize only on gain, the results is slightly better than engineering tuned results in terms of PAE and gain. Therefore, in our design, we set our cost function to as multi-parameter optimization as we mentioned in Algorithm 1. Furthermore, we can set the constrained optimization: Maximize the PAE while keep gain at certain level.

### *B. SystemVue and ADS co-simulation*

We also setup the design for SystemVue and ADS co-simulation environment. In the SystemVue, we use Matlab script to implement the Algorithm 1, sampled at 15.36 *MHz* phase, the system gradually converge to minimize the loss function (shown in Fig. 6). In SystemVue setup, we have one source and split into two branches. Then, we feedback two branches output into the control module (optimization), which mimic the digital DPA approach. Figures 7 and 8 shows the co-simulation results where the PAE is achieved around 70% and Gain is achieved around *16 dB* with the input source running at *26 dBm* with *3.6 GHz*. The results illustrate the algorithm convergence in less than 1000 iterations, becoming stable.

Additionally, the hyper parameters in Algorithm 1 needs to be handled carefully: we choose $\gamma_1$ to be 0.606 and $\gamma$ to be 0.101 according to [10] and the parameter $c$ to be 1 since during the optimization step $[C(\theta_+) - C(\theta_-)]$ will create the large value for PAE term. We found that $c$ could be set as 0.1 if we only optimize on gain.

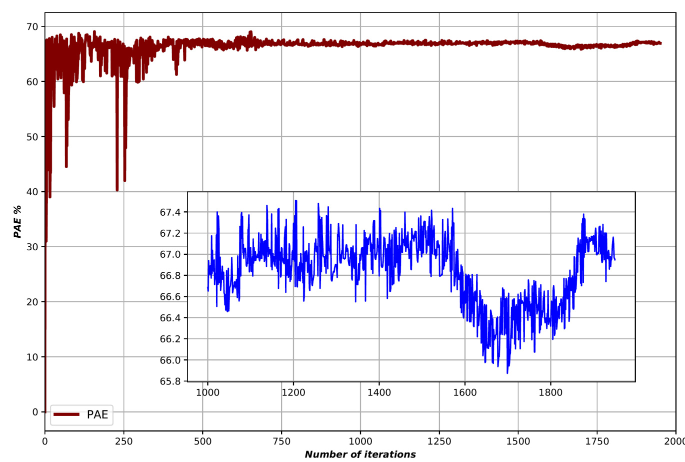


Fig. 7: ADS SystemVue co-simulation: Achieved PAE under SPSA optimization

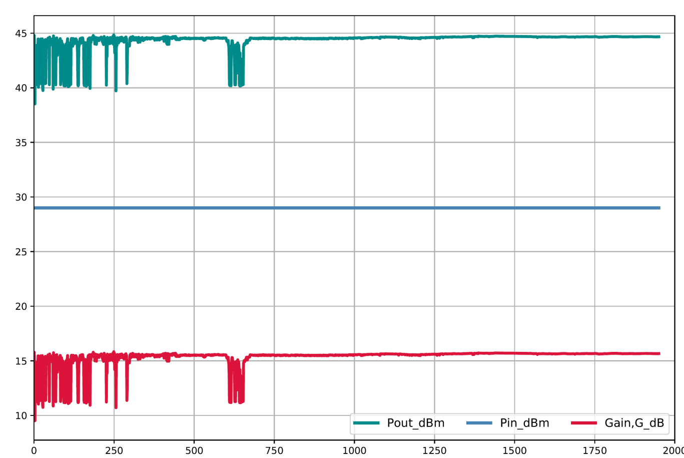


Fig. 8: ADS SystemVue co-simulation: Achieved Gain under SPSA optimization.

### C. Testbench implementation

Figure 6 shows the testbench setup when implementing the co-simulation flow on real PA. We replace the ADS simulation block with the physical DPA. When running the testbench, it contains training phase and prediction phase, and we describe each one as following.

*1) Training phase:* During the training phase, the input is set as CW signal which swept input power and frequency. The swept configurations are further processed by Algorithm 1 to optimize the DPA performance via control ports. When Algorithm 1 estimates the control ports for optimal PAE and gain, the optimal control parameters could be recorded by the Look-Up Table (LUT) at each step, where step size represents the granularity of different configurations (input power, frequency). The training resources (training time, LUT memory size, etc.) will be larger but more precise if the LUT step size is smaller. In our case, we set the step size as 1 for each input power (*0 dBm* to *30 dBm*, 30 steps), and frequency as 0.1 (*3.3 GHz* to *3.9 GHz*, 7 steps). Therefore, the LUT for optimization is needed to run the Algorithm 1 for 210 iterations. Note that this process only consumes longer computation time at the initial stage since all parameters are randomly initialized and future estimation are carried on from the previous optimal values, the convergence speed will be much faster.

*2) Prediction phase:* After the training phase, the pre-trained control ports is already stored in LUT. As we mentioned in training phase, our experiment has 210 parameters ($\approx 3.3$ *MB*) to store, which is small enough for modern DSP/FPGA/ASIC devices. The computation in prediction phase is also light due to control on-chip SRAM based on input configurations. All context switch for control ports is adapted by baseband signal frequency, which is desirable for digital circuit.

## V. Conclusion and Future Work

In this paper, we present a novel adaptive DPA design to enhance the PAE and gain. We take into account several control factors not only limited on phase alignment, but also gate bias and power distribution ratio at the source. Our performance shows significant improvement in the gain and PAE as compared with single-ended DPA design. And our implementation is generalizable for all different types of DPA. We summarize our future works as follows: 1) Implementation of the Optimization technique in Real-time testbench. 2) Reinforcement learning agent as controller to control the dynamics.

## References


[1] R. Darraji, F. M. Ghannouchi, and O. Hammi, "A dual-input digitally driven doherty amplifier architecture for performance enhancement of doherty transmitters," *IEEE Trans. on Microwave Theory and Techniques*, vol. 59, no. 5, pp. 1284–1293, 2011.

[2] J. Kim, *et al.*, "Optimum operation of asymmetrical-cells-based linear doherty power amplifiers-uneven power drive and power matching," *IEEE Trans. on Microwave Theory and Techniques*, vol. 53, no. 5, pp. 1802–1809, 2005.

[3] J. Choi *et al.*, "Optimized envelope tracking operation of doherty power amplifier for high efficiency over an extended dynamic range," *IEEE Trans. on microwave theory and techniques*, vol. 57, no. 6, pp. 1508–1515, 2009.

[4] J. Kim, B. Fehri, S. Boumaiza, and J. Wood, "Power efficiency and linearity enhancement using optimized asymmetrical doherty power amplifiers," *IEEE Trans. on Microwave Theory and Techniques*, vol. 59, no. 2, pp. 425–434, 2011.

[5] H. Lee *et al.*, "Optimized current of the peaking amplifier for two-stage doherty power amplifier," *IEEE Trans. on Microwave Theory and Techniques*, vol. 65, no. 1, pp. 209–217, 2017.

[6] R. Darraji and F. M. Ghannouchi, "High efficiency doherty amplifier combining digital adaptive power distribution and dynamic phase alignment," in *14th Annual Wireless and Microwave Technology Conf.(WAMICON)*. IEEE, 2013, pp. 1–3.

[7] R. Darraji and F. Ghannouchi, "Digital doherty amplifier with enhanced efficiency and extended range," vol. 59, no. 11. IEEE, 2011, pp. 2898–2909.

[8] P. Chen, J. Xia, B. M. Merrick, and T. J. Brazil, "Multiobjective bayesian optimization for active load modulation in a broadband 20-w gan doherty power amplifier design," *IEEE Trans. on Microwave Theory and Techniques*, vol. 65, no. 3, pp. 860–871, 2017.

[9] J. C. Spall, "An overview of the simultaneous perturbation method for efficient optimization," *Johns Hopkins apl Technical Digest*, vol. 19, no. 4, pp. 482–492, 1998.

[10] J. Spall, "Stochastic optimization," in *Handbook of computational statistics*. Springer, 2012, pp. 173–201.

[11] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2016.

[12] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[13] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.