國立臺灣大學電機資訊學院電信工程學研究所

碩士論文

Graduate Institute of Communication Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

應用於快速合成微波及毫米波積體電路之新式演化式演算法

Novel Evolutionary Algorithms for Fast Synthesis of
Microwave and Millimeter Wave Integrated Circuits

陳柏翰

Po-Han Chen

指導教授：王暉 博士

Advisor: Huei Wang, Ph.D.

中華民國 102 年 6 月

June, 2013

# 國立臺灣大學碩士學位論文
# 口試委員會審定書

## 應用於快速合成微波及毫米波積體電路之新式演化式演算法
## Novel Evolutionary Algorithms for Fast Synthesis of Microwave and Millimeter Wave Integrated Circuits

本論文係 陳柏翰 君（R00942088）在國立臺灣大學電信工程學研究所完成之碩士學位論文，於民國 102 年 6 月 26 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿（簽名）
（指導教授）

＿＿＿＿＿＿＿＿＿＿＿　　＿＿＿＿＿＿＿＿＿＿＿

＿＿＿＿＿＿＿＿＿＿＿　　＿＿＿＿＿＿＿＿＿＿＿

所　　長　＿＿＿＿＿＿＿＿＿＿＿（簽名）

# 誌謝

　　能完成這篇論文，首先特別感謝我的指導教授王暉老師；老師處理事情的效率的及對研究的嚴謹都是我學習的榜樣。老師不僅鼓勵我投稿論文於國際發表，並且不厭其煩地與我來回地修訂論文與提出獨到的見解，使我有能夠出國發表的機會。老師讓我能夠自由的做想做的研究，使我能夠專心的鑽研電路合成演算法的主題，讓這篇論文得以順利完成。

　　再來要感謝電機所的于天立老師，儘管我不是老師的指導學生，老師都很熱心的與我討論演算法的問題，並且給我很有幫助的研究方向與意見，使我能夠把電路合成演算法成功的完成；演化式計算的內容也在老師的幫忙批改下更加的完整。還有老師能夠當我投稿期刊論文的共同作者也是我莫大的榮幸。

　　謝謝各位口試委員們的建議與不同觀點，提醒了一些我忽略的細節，使得本論文內容更完善。感謝台灣積體電路製造股份有限公司以及中央研究院天文所提供的學術研究晶片製作，與國家奈米實驗室的晶片量測服務。

　　還有要感謝實驗室的學長同儕學弟們(竟然沒學妹)。謝謝高瑞智學長的一路的以來的指導，跨領域的研究真的很辛苦，當初若沒有學長的鼓勵與肯定，我的演算法老早就流產了，學長真的超級厲害，祝學長能夠找到理想的工作 or 當教授也不錯！？還有當初七月才一起進實驗室的諦騰和光聖，過去一起努力補上落後進度的情景還歷歷在目，有幾次和諦騰一起搭車到中央找各自的女朋友令人懷念，祝你早日賺大錢就可以不用再省了！光聖就像大哥一樣的穩重對大家都很好，祝你研究順利博班可以早日畢業！峻安人真 nice 每次被我捶都不會罵我，而且都會幫我很多忙，非常的可靠不愧為年薪百萬男！麻吉博士是實驗室的開心果，你讓大家都很開心，祝你工作順利。大老闆姜博祝你的事業一路衝天發發發，到時候可以捐錢蓋一棟姜博館在台大！祝福學弟們能夠早日找到研究的主題順利畢業，祝鋒哥能夠像大仁哥一樣最後能夠取得她的芳心，葉闆神人寫出商用模擬軟體，柏羽學長感情順利。

　　最後要感謝我的家人無怨無悔的支持，讓我能夠拿到這個碩士學位。爸就算工作很忙也會抽空來替我加油打氣，常常帶著非常多的水果給我補充營養，我同

學都以為我家是開水果攤哈哈！可愛的小密蔻大老遠北上來找我吃飯每次都讓我很感動(偷哭)，乎乾啦！在嚴寒的研究生活中，老接不時地關心我令我備感溫暖，還常常有驚奇小禮物，有老接真好！哥一直都是我學習的榜樣，在我徬徨時給我指點。還有我的女朋友領翎，謝謝你一路來的陪伴與支持。

柏翰　　2013/06/20

# 中文摘要

　　無線通訊晶片在智慧型手機等高科技產品上扮演著不可或缺的角色。隨著技術快速演進，工程師如何在短時間內設計出高效能的晶片是決定著自身及產業競爭力的關鍵。本篇論文以演化式計算實作微波及毫米波電路的匹配網路自動合成及最佳化，實現了商用模擬軟體缺乏的電路合成功能。在平行計算的加速之下，可在數秒內輕易地完成匹配網路的最佳設計。

　　論文的第一部分說明如何使用先進的演化式計算去合成匹配網路。有別於過去在微波領域發表過的文獻，我們深入探討匹配網路合成的特性並結合電腦科學領域的先進技術，針對匹配網路設計特別實作出更快且精確的合成演算法，使工程師能夠在不需分析匹配網路架構下設計出高效能的電路。最後以電腦實驗統計提出的演算法的優異效能，如放大器及濾波器電路合成。

　　第二部分展示用提出的演算法實作的兩個寬頻放大器。首先是增益頻寬第一個涵蓋整個 D-band 的 65-nm CMOS 寬頻放大器，我們針對級間網路設計一個最佳化方法來決定所需的放大器級數，並且合成各個寬頻的匹配網路，此外也探討了改善疊接式放大器匹配的方法。而第二個電路是使用 0.1-μm GaAs pHEMT 製程實作的寬頻 Ka-band 功率放大器，由於最佳化的匹配網路使得增益、反射損耗、輸出功率及效率能夠有良好的寬頻表現。

關鍵字 —— 電路合成，電腦輔助設計及最佳化，電路設計自動化，寬頻匹配技術，寬頻 D-band 放大器，寬頻功率放大設計，演式計算，基因遺傳演算法，演化策略。

# ABSTRACT
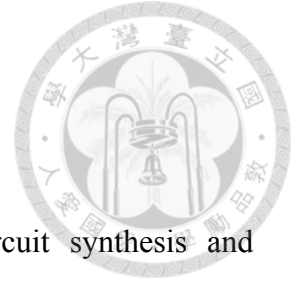
This thesis presents a novel algorithm for microwave circuit synthesis and optimization, which is not presented in the current commercial circuit-simulation software. With the help of our proposed algorithm, the time for the design of matching networks can be reduced to only a few seconds.

Our proposed algorithm is based on the real-coded expended compact genetic algorithm (rECGA) and evolutionary strategy (ES), which combines advantages of linkage learning from genetic algorithm (GA) and exploration ability from ES for robust global optimizations. Due to the specific design of the proposed algorithm for matching network synthesis, the possibility of immature convergence is reduced and the optimum matching network can be found rapidly. Besides, the proposed algorithm can simultaneously find many sub-optimal circuits with different topology, which makes the design flexible in choosing the desired circuit architecture. In order to validate the proposed algorithm, several experiments have been performed and analyzed, and the results show that our algorithm outperforms the previously published works.

Two monolithic microwave integrated circuits (MMICs) are also implemented using our algorithm. The first one is a 110-180 GHz broadband amplifier in 65-nm CMOS process, which is the first CMOS amplifier covering the full D-band. The second circuit is a two-stage Ka-band power amplifier in 0.1-μm GaAs pHEMT process, which performs broadband response of gain, return loss, output power and high power-added efficiency (PAE).

Index Terms —Circuit synthesis, Computer-aid design (CAD), Electronic design automation (EDA), Optimization method, broadband matching technique, broadband amplifier, evolution computation, genetic algorithm (GA), evolutionary strategy (ES).
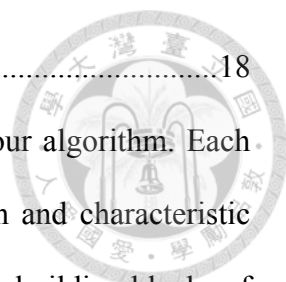
# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# Chapter 1　Introduction

## 1.1　Motivation

Optimization of matching networks is a necessary step to design a high performance microwave circuits. Unfortunately, the problem to synthesize an optimal matching network for a given circuit specification can be reduced to the NP-complete problems [1]. Therefore, enumeration is still the only way to ensure the global optimality so far. However, the design of microwave circuits usually has real-time constraints, and thus it is desired to have an algorithm to fast find a reasonable solution or possibly the global optimum instead of enumeration.

If the time for the design of matching networks can be reduced to only a few seconds, the work efficiency of design engineers can be enhanced significantly. Since the computing capability of modern computer is powerful and the parallel computations are well developed, it is worth putting more effort on computer-aid design (CAD) and electronic design automation (EDA). Therefore, a robust algorithm is presented in this thesis, which can fast and accurately synthesize the optimum matching network, and several experiments have been performed to show that our proposed algorithm outperforms previously published works.

## 1.2 Literature Survey and Background

### 1.2.1 Circuit Synthesis Algorithms

There are various techniques for circuit optimization, such as methods based on calculus and derivative-free meta-heuristic [2]. Fig. 1.1 shows the category of usually used search algorithms. Most parts of realistic optimization problems have multiple local optima, where the methods based on steepest descent easily fall in one of local optimum. Evolutionary computation is well-known for global optimization with a derivative-free meta-heuristic and population-based optimization; one of the characteristics is to solve problems with multiple local optima.
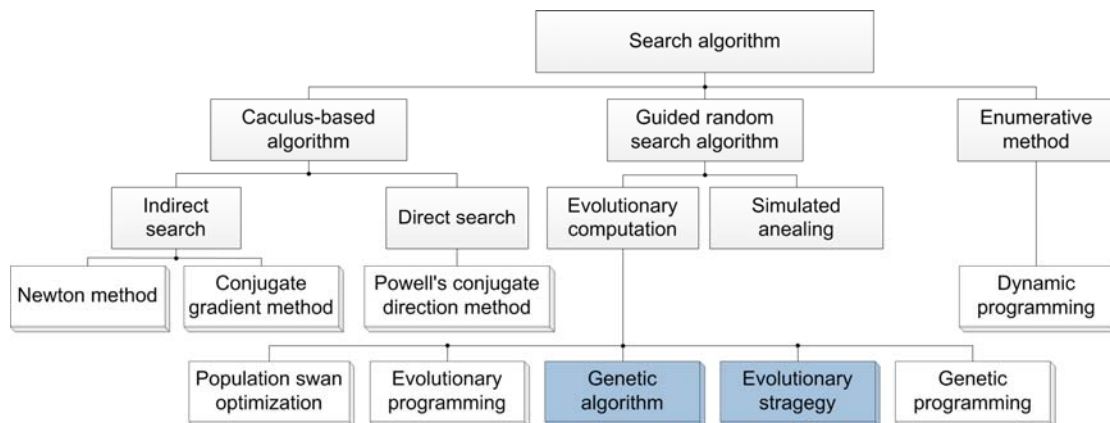
Fig. 1.1. Category of usually used search algorithms.

Genetic algorithm (GA) is one of evolutionary algorithms [3], which is search heuristic inspired from natural evolution. Due to its reliable and accurate optimization ability, GA has been widely used in science and engineering problems [4]-[5]. There are also many applications of microwave circuits that used GA to optimize performance [6]-[10]. GA is used with 2.5 D electric-magnetic (EM) simulation with extraordinary metal layout to produce specific responses [6]. GA is employed to generate special

line-segment circuits in 2D and 3D respectively [7]-[8]. Nevertheless, [6]-[8] need large time for computation and are not suitable for general circuit design under real-time constraints. GA is applied for filter synthesis with distributed components, but the algorithm is too simple to be robust [9].

However, the GAs used in most existing works on microwave circuit optimizations are equivalent to simple genetic algorithm (SGA). The operators of SGA do not consider the dependency among variables and may cause solution candidates to converge immaturely. Modern GA designs include learning genetic linkages and mixing via building blocks [11]. Several papers have empirically verified that considering the dependencies and relations among genes of GA greatly enhance the optimization performance [12]-[15].

Evolutionary strategy (ES) is another efficient stochastic search algorithm [16]. The basic idea behind ES is to generate offspring by perturbing solution candidates; the ES mechanism usually consists of elitic selection and self-adaptive mutation [17]. Contrary to GA that focuses more on crossover, the development of ES centers more on mutation, which lead ES usually converge to the optimum in monotonic functions faster than GA.

Due to the periodicity characteristic in distributed components, there are many local optima in solution space and make circuit synthesis difficult. According to the no-free-lunch theorem [18], specific algorithm for particular problem has better search ability and outperforms the general search algorithms. Therefore, a novel synthesis algorithm is developed to overcome this difficulty instead of only employing SGA. The proposed algorithm is based on rECGA and ES, which combines advantages of linkage learning from GA and the exploration ability of ES for robust global optimization.
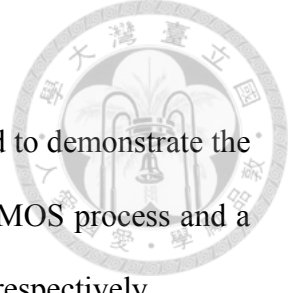
## 1.2.2  Broadband Amplifier

Two broadband amplifiers have been designed and implemented to demonstrate the proposed algorithm. They are a 110-180 GHz amplifier in 65-nm CMOS process and a Ka-band medium power amplifier in 0.1-μm GaAs pHEMT process respectively.

Due to the wide data bandwidth and higher image resolution, there are many system applications in D-band (110-170 GHz), such as the biomedical imaging systems and short-term high data rate communications. However, it is difficult to design the CMOS amplifiers beyond 100 GHz, due to the gain limitation of the transistor and losses of on-chip passive components in CMOS process. For instance, a 6-stage amplifier is presented [28], but the small signal gain is low. Even the high gain can be achieved in D-band, the bandwidth is narrow and gain significantly degrades around 170 GHz due to the moderate $f_{max}$ and $f_T$ of transistors in CMOS process [29]-[32].

Table 1.1 summarizes previously published CMOS D-band amplifier. It is obvious that there is no amplifier whose bandwidth can actually cover the full D-band. Therefore, our design target is to simultaneously achieve the full D-band bandwidth and desirable gain performance by applying the circuit synthesis algorithm.

Table 1.1.  Summary of the previously published D-band Si-based amplifier.

| Reference | Ref.[28] 2008 RFIC | Ref.[29] 2010 ASSCC | Ref.[30] 2009 ISSCC | Ref.[31] 2012 IMS | Ref.[32] 2008 RFIC |
|---|---|---|---|---|---|
| Process | 65-nm CMOS | 65-nm CMOS | 65-nm CMOS | 65-nm CMOS | 0.13-μm SiGe |
| Topology | 6-Stage Single-Ended LNA | 3-Stage Differential Amplifier | 3-Stage Single-Ended Amplifier | 4-Stage Single-Ended PA | 5-Stage Single-Ended Amplifier |
| Frequency (GHz) | 140 | 144 | 150 | 140 | 165 |
| 3-dB Bandwidth (GHz) | 10 | 33* | 27 | > 30 | 15 |
| Gain (dB) | 8 | 20.6 | 8.2 | 15 | 14 |
| dc Power (mW) | 63 | 102 | 25.5 | 115.2 | 135 |
| Chip Size (mm²) | 0.06 ** | 0.21 | 0.41 | 0.38 | 0.14 |

**\* : Gain > 10 dB bandwidth, not 3-dB bandwidth.**

**\*\*: Estimate from chip micrographs (not including test pads).**

Power amplifier (PA) plays an important role in wireless transmitter; the growing demands in satellite communication in Ka-band (26.5-40 GHz) have motivated the need for broadband power amplifiers design. One of the design challenges of power amplifier is performing broadband response; there are several techniques to achieve this target, such as balanced amplifiers [33], distributed amplifier [34] and synthesized transformer [35]. Although [33]-[34] can obtain wide bandwidth, the large chip size is large and power-added efficiency is degraded due to on-chip large passive components. Table 1.2 summarizes the previously published GaAs HEMT Ka-band power amplifier.

Table 1.2. Summary of the previously published Ka-band power amplifier in GaAs HEMT process.

| Reference | Ref.[33] 2001 SSICT | Ref.[34] 2005 APMC | Ref.[35] 2012 MTT | Ref. [36] 2006 EUMC |
|---|---|---|---|---|
| Process | 0.25-μm GaAs HEMT | 0.15-μm GaAs HEMT | 0.15-μm GaAs HEMT | 0.15-μm GaAs HEMT |
| Topology | 2-Stage Balanced | Distributed | Output Equivalent Coupled Resonator | 4-Stage Single-Ended |
| 3-dB Frequency (GHz) | 17-36 | 4-37 | 17-35 | 35-42 |
| Gain (dB) | 11-14 | 15 | 9-12 | 26-28 |
| dc Supply (V) | 5 | 5 | 4 | 5 |
| $P_{sat}$ (dBm) | 23-24 | 20-23 | 22.5-23.5 | 26 |
| $OP_{1dB}$ (dBm) | 21.5-23 | 19-21 | 21-22 | 24 |
| PAE$_{@Peak}$ (%) | 22-35 | 15 | 30-40 | 14 |
| Chip Size (mm$^2$) | 4.32 | 3 | 1.5 | 3.5 |

## 1.3 Contributions

This thesis presents a novel circuit synthesis algorithm which combines advantages of linkage learning from GA and exploration ability from ES. In order to validate the proposed algorithm, several experiments have been conducted and analyzed, and the results show that the proposed algorithm indeed outperforms the previously published

works, since it can find the better optimum solution with the fewer function evaluations. Besides, the proposed algorithm can simultaneously find many sub-optimal circuits with different topology, which make the design flexible in choosing the desired circuit architectures.

Two broadband amplifiers have been designed and implemented to demonstrate the proposed algorithm. The 110-180 GHz amplifier is the first amplifier covering full D-band in 65-nm CMOS process; we propose an optimization method to determine the minimum number of cascade stage and an impedance transformation technique to achieve desirable gain performance and wide bandwidth. The Ka-band 0.1-µm GaAs pHEMT power amplifier performs broadband output power and high power-added efficiency (PAE) due to the synthesized broadband matching networks.

Therefore, our algorithm can help microwave-circuit engineer to design matching networks within only a few seconds, thus the work efficiency of engineer or IC design-house can be enhanced significantly.

## 1.4    Thesis Organization

Chapter 2 presents the detail of our circuit synthesis algorithm. Several experiments such as transistor matching and dual-band filter design have been performed to compare our proposed algorithm with previously published works.

Chapter 3 presents the circuit synthesis instances. The design of full D-band 65-nm CMOS amplifier and broadband 0.1-µm GaAs pHEMT power amplifier will be demonstrated.

Chapter 4 summarizes the results of previous chapter and illustrates the future works.

# Chapter 2   Novel Evolutionary Algorithms for Fast Synthesis of Microwave Matching Network

This chapter presents a novel circuit synthesis algorithm based on real-coded expended compact genetic algorithm (rECGA) and evolutionary strategy (ES), which combines advantages of linkage learning from genetic algorithm (GA) and exploration ability from ES. Due to the specific design of the proposed algorithm for matching network synthesis, the possibility of immature convergence is reduced and the optimum matching network can be found rapidly. Besides, the proposed algorithm can simultaneously find many sub-optimal circuits with different topology, which makes the design flexible in choosing the desired circuit architectures.

## 2.1   Difficulties in Microwave Matching Networks Synthesis

### 2.1.1   Formalization of Microwave Matching Networks Synthesis

The matching network design can be reduced to the nonlinear least-square problem. The operating frequency is the independent variable, while the electric parameters of each circuit component are the adjustable variables, and the dependent variable is the frequency response of the network. Finding the nonlinear least-square error solution is equivalent to achieve the desirable frequency response.

Fig. 2.1 illustrates the flow of matching network synthesis. Each unknown block (symbol '?') is one of the distributed components. Although the set of possible components can include lump or other components, we only apply distributed

7

components for instruction. Given an sampling data of target frequency response of $\Gamma(f)$, the first step is to transform terminated impedance ($Z_T$) to the $\Gamma_{in}(f)$, followed by calculating the square error between transformed response and target response as

$$\sum_i |\Gamma(f_i) - \Gamma_{in}(f_i)|^2 \qquad (2.1)$$

The synthesis algorithm should find the nonlinear least-square solution under pre-defined constraints, such as the order of network ($N$), limits of transmission-line electrical lengths ($EL$) and characteristic impedances ($Z_0$); while the $[a_i, b_i]$ and $[c_i, d_i]$ denote the range of $EL$ and $Z_0$ which belong to component $i$ respectively. The more point we samples, the more accurate the global optimum is; however, redundant sampling will cause more computation overhead.
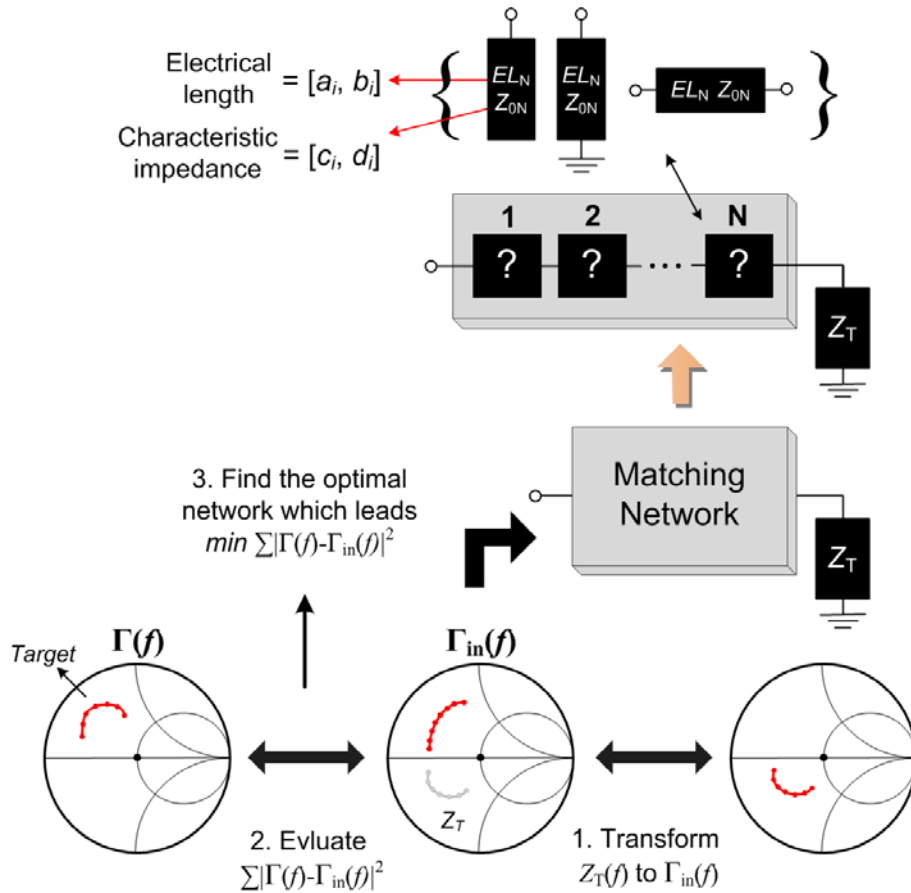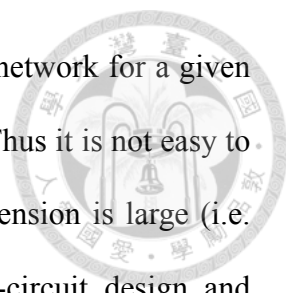


Fig. 2.1. Flow charts of matching network synthesis. This problem can be reduced to nonlinear least-square problem; finding the nonlinear least-square error solution is equivalent to achieve the desirable frequency response.

Unfortunately, the problem to synthesize an optimal matching network for a given circuit specifications can be reduced to NP-complete problems [1]. Thus it is not easy to find global optimum under real-time constraint when problem dimension is large (i.e. the order of network is high). However, the periods of integral-circuit design and hybrid-circuit design are not long enough to use naïve search algorithm, such as enumeration (brute-force method). Therefore, it is desired to have an algorithm to fast find a reasonable solution or possibly the global optimum instead of enumeration.

However, for the global optimum, the algorithm output sometimes is not the best choice due to circuit design consideration. For example, the global optimum may contain more than two shunt-connected components join on a node, and this topology is impractical for layout in general. Another example on microwave circuit design is that the global optimum may contain one short stub, which causes that engineer need to add additional decoupling capacitor to prevent dc bias from short-circuited to ground. Although simply transforming each consideration into constraints of algorithm can handle this problem, an algorithm which can simultaneously find many sub-optimal circuits with different topology is actually more practical.

## 2.1.2 Naïve Algorithm

Although the computation effort of enumeration is too large to use, it is a good choice when problem dimension is very small (small network order). Therefore, we discuss this naïve algorithm and correspondent time complexity before investigating another practical algorithm.

The enumeration for circuit synthesis can be outlined as following pseudo code

$$N \begin{cases} \quad \text{for } Type_1 \subset \{\text{short stub, straight line, open stub}\} \\ \qquad\qquad\qquad\qquad \vdots \\ \quad \text{for } Type_N \subset \{\text{short stub, straight line, open stub}\} \end{cases}$$

$$N \begin{cases} \quad \text{for } EL_1 \subset [a_1,\ b_2] \\ \qquad\qquad \vdots \\ \quad \text{for } EL_N \subset [a_N,\ b_N] \end{cases}$$

$$N \begin{cases} \quad \text{for } Z_{01} \subset [c_1,\ d_2] \\ \qquad\qquad \vdots \\ \quad \text{for } Z_{0N} \subset [c_N,\ d_N] \end{cases} \tag{2.2}$$

$$\text{circuit} = \text{compute\_response}(Type_1, \cdots, Type_N, EL_1, \cdots, EL_N, Z_{01}, \cdots, Z_{0N})$$

$$\text{optimum} = \arg\min(\text{optimum.square\_error, circuit.square\_error})$$

where the parameters are defined in Fig. 2.1. This naïve algorithm simply enumerates all possible component type and electrical parameters to find the global optimum.

For an example, if the range of $EL$ and $Z_0$ for all components is $[a, b]$ and $[c, d]$ with unit electrical parameter step (i.e. $EL \in a, a+1, \cdots, b$ and $Z_0 \in c, c+1, \cdots, d$) respectively, the time complexity $T(N)$ can be derived as

$$T(N) = \Theta((3 \times (b-a) \times (c-d))^N) \tag{2.3}$$

where $\Theta$ is an asymptotic notation [37]. Replace with usually used parameters such as $[a, b] = [0, 100]$, $[c, d] = [40, 100]$ and $N = 4$, the time complexity become

$$\begin{aligned} T(N) &= \Theta((3 \times (100-0) \times (100-40))^4) \\ &= \Theta(10^{16}) \end{aligned} \tag{2.4}$$

It is obvious that this computation cannot be completed in a reasonable time even if commercial cluster with parallel computation is applied. But if we only want to synthesize a network with a few order and parameters like $N = 2$, $[a, b] = [0, 100]$ and $[c, d] = [50, 50]$, the time complexity is acceptable to employ enumeration

$$\begin{aligned} T(N) &= \Theta((2 \times (100-0))^3) \\ &= \Theta(10^4) \end{aligned} \tag{2.5}$$

However, it is necessary to use another algorithm instead of enumeration for general

purpose applications. Evolutionary computation is well-known for global optimization with a derivative-free meta-heuristic and population-based optimization; one of the characteristics is to accurately solve problems with multiple local optima like the circuit synthesis.
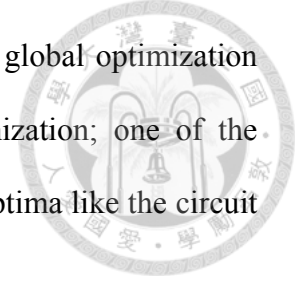
### 2.1.3　Multimodal and Deceptive Characteristics of Circuits Synthesis

However, due to the periodicity characteristic in distributed components, there are many local optima (i.e. multimodal) in solution space and make circuit synthesis difficult. For instance, Fig. 2.2 shows two artificial circuit synthesis problems, which are to find a 60-Ω transmission line (a distributed component) that causes the most same frequency response of target. The objective function is defined as

$$F_{objective} = \frac{1}{1 + \sum_i |\Gamma(f_i) - \Gamma_{in}(f_i)|^2} \tag{2.6}$$

Maximizing the objective function is equal to minimize the total square error. Although these problems are trivial, they can show the multimodal characteristic and variable dependency between circuit type and electrical parameters. Due to the small problem dimension, the enumeration can be employed to search entire solution space. Figs. 2.3 and 2.4 show solution space of these two problems. Each local optimum is labeled with correspondent component type and electrical length. In problem 1 defined in Fig. 2.2, the 180° straight line, 180° open stub and 85° short stub have a similar impedance-transformation result to the answer (straight line with 10° electrical length). In the problem 2, the 90° short stub and 0-180° of straight line have a similar impedance-transform result to the answer (open stub with 10° electrical length).

All the directions radiated from the global optimum point to other local optima (deceptive). These multimodal and deceptive characteristics make circuit synthesis

11

problem difficult to solve. Therefore, it is beneficial to use niching technique to find

global optimum with clues from local optima, and keep solution diversity to enhance

exploration.
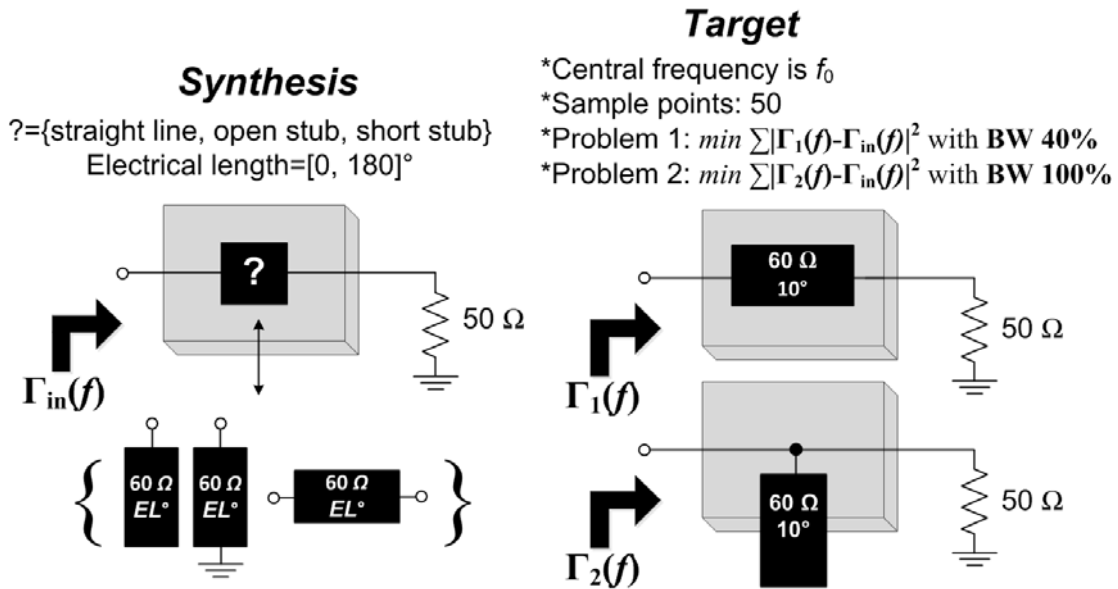


Fig. 2.2.  Two artificial problems to show multimodal and deceptive characteristics of circuit synthesis.
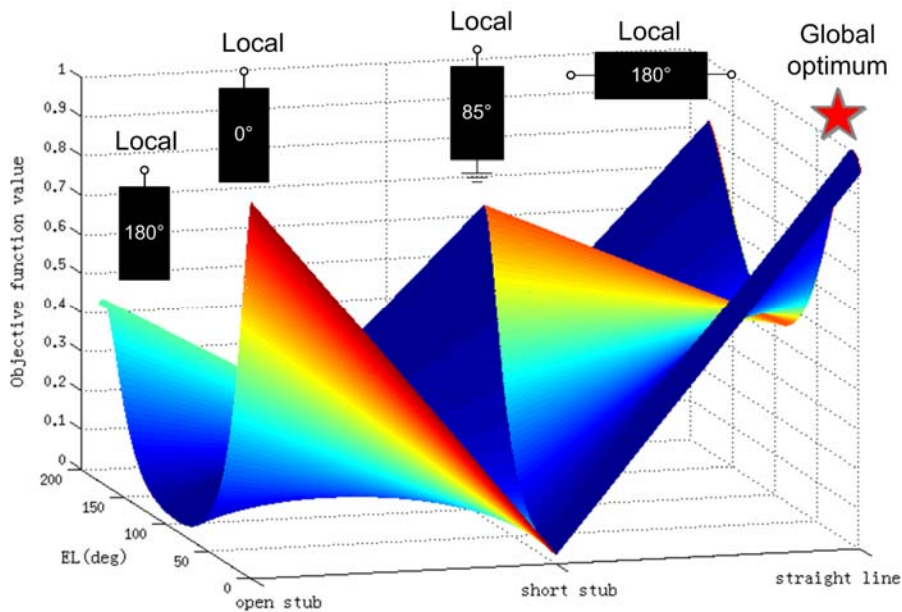


Fig. 2.3.  Solution space of problem 1 described in Fig. 2.2.
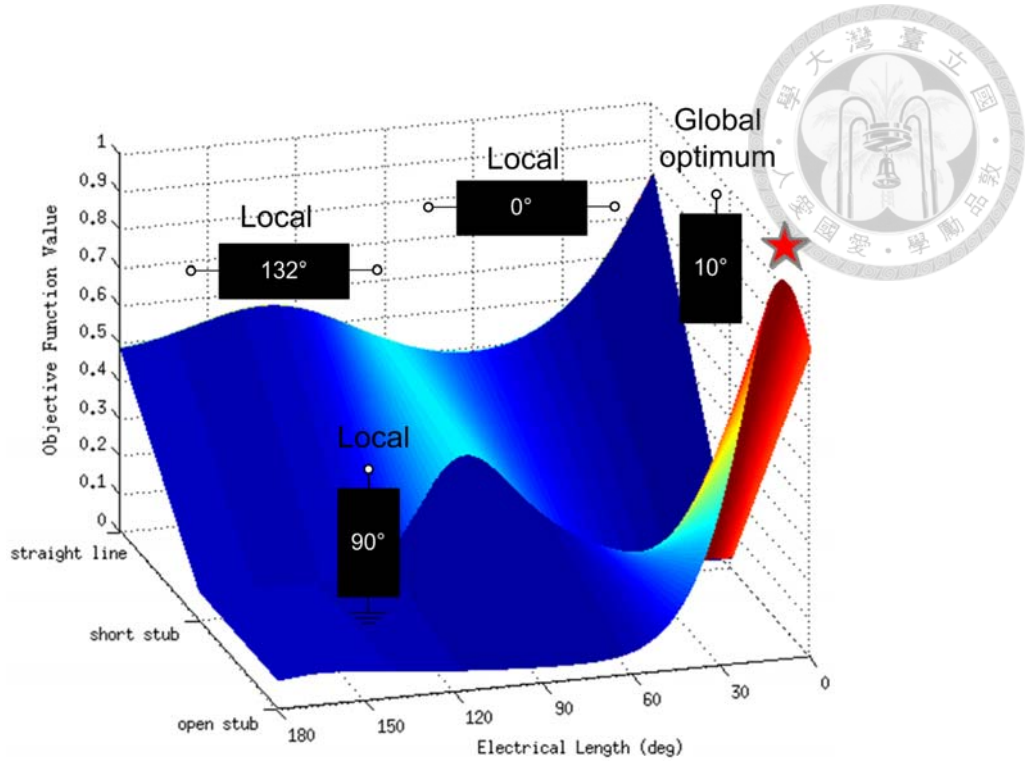
Fig. 2.4. Solution space of problem 2 described in Fig. 2.2.

## 2.2   Genetic Algorithm with Linkage Learning Technique

Crossover is an important operator in GA, because it is used for generating new solution candidates and exploiting the search space. An inadequate crossover operator causes local optima easily take over solution candidates; this phenomenon usually occurs when SGA is used to solve deceptive problems [20]-[21]. Therefore, this section introduces the real-coded expended compact genetic algorithm (rECGA), which employs the linkage learning techniques from GA to accurately solve the deceptive problem.

In order to describe the differences between SGA and rECGA, consider a deceptive function defined as

$$
F(x_1, \cdots, x_n) = \begin{cases} 2nu, & \text{if } x_i < u / \lambda \text{ for } i = 1, \cdots, n \\ \sum_{i=1}^{n} x_i, & \text{otherwise} \end{cases} \tag{2.7}
$$

where $x$, $u \in R$ and $x$ is restricted in $[0, u]$, both $x$ and $u$ are non-negative. Fig. 2.5 shows the solution space of the deceptive function $F$ with $u = 100$, $n = 2$ and $\lambda = 4$. This function has a local optimum located at $(x_1, x_2) = (u, u)$ and a global optimum located at $(x_1, x_2) = (0, 0)$. Then we define a difficult artificial optimization problem based on the deceptive function $F$ as

$$Maximize \ G(\underline{x_1, \cdots, x_4}, \ \underline{x_5, \cdots, x_8}, \cdots, \ \underline{x_{m-3}, \cdots, x_m})$$
$$G(x_1, \cdots, x_m) = \sum_{i=1}^{m/4} F(x_i, \cdots, x_{i+3}), \ \ m \bmod 4 = 0 \text{ and } \lambda = 4 \tag{2.8}$$

Assume both SGA and rECGA are blind to this problem structure initially. The uniform or one-point crossover can be used for SGA, and these operators cause each variable $x_i$ easily takeover by $u$ and be trapped into local optimum $G(u, \cdots, u)$, since it cannot figure out the dependencies between each variable in $F$, that is, some variables should be grouped and cannot be separated after the crossover, which leads to inadequate crossover, as shown in Fig. 2.6.

Unlike SGA, rECGA uses minimum description length (MDL) to find out the dependencies between variables [22]. Therefore, rECGA can use building-block-wise crossover [15] instead of simple operators like uniform and one-point crossover, thus it can solve each sub-problem $F$ and maximize $G$. Fig. 2.7 algorithmically outlines the rECGA.

Fig. 2.8 shows the experiment results of solving optimization deceptive problem $G$. The parameter in both SGA and rECGA are $n = 50000$, 100000 total function evaluations and tournament selection pressure $s = 30$ without mutation operator. Due to linkage learning technique, rECGA can accurately find global optimum, while the SGA easily trapped into the local optima when problem dimension is increased.

The deceptive solutions space shown in Fig. 2.5 is similar to the Figs. 2.3 and 2.4.

According to this experiment, employing rECGA rather than using SGA for circuit synthesis is beneficial, since SGA will lead to inadequate crossover when solution space contain variable dependencies.
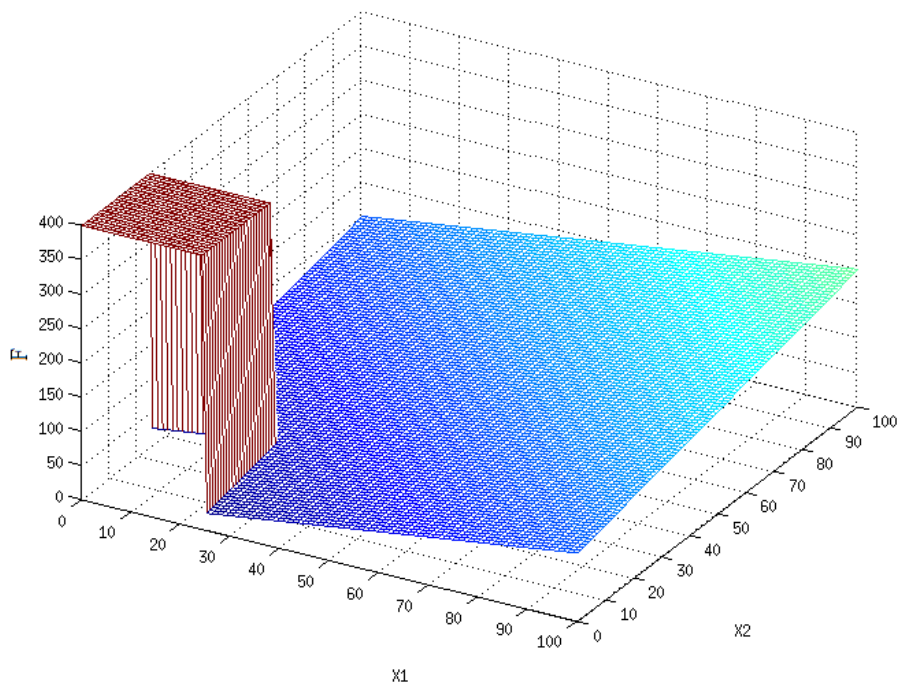


Fig. 2.5. Solution space of the deceptive function $F$ with $u = 100$ and $\lambda = 4$. This function has a local optimum located at $(x_1, x_2) = (u, u)$ and a global optimum located at $(x_1, x_2) = (0, 0)$.
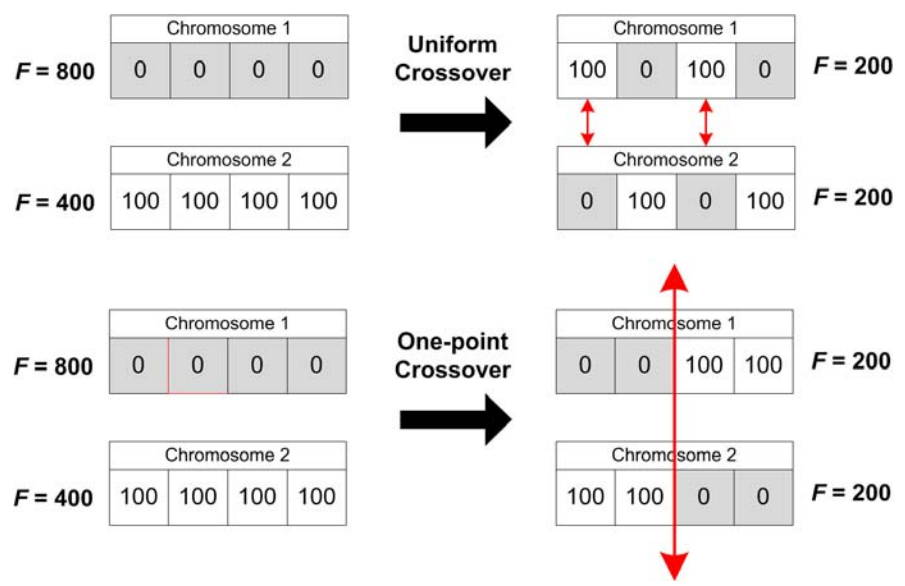


Fig. 2.6. Simple crossover operator makes solution candidates (chromosome) difficult to maintain good variables patterns.
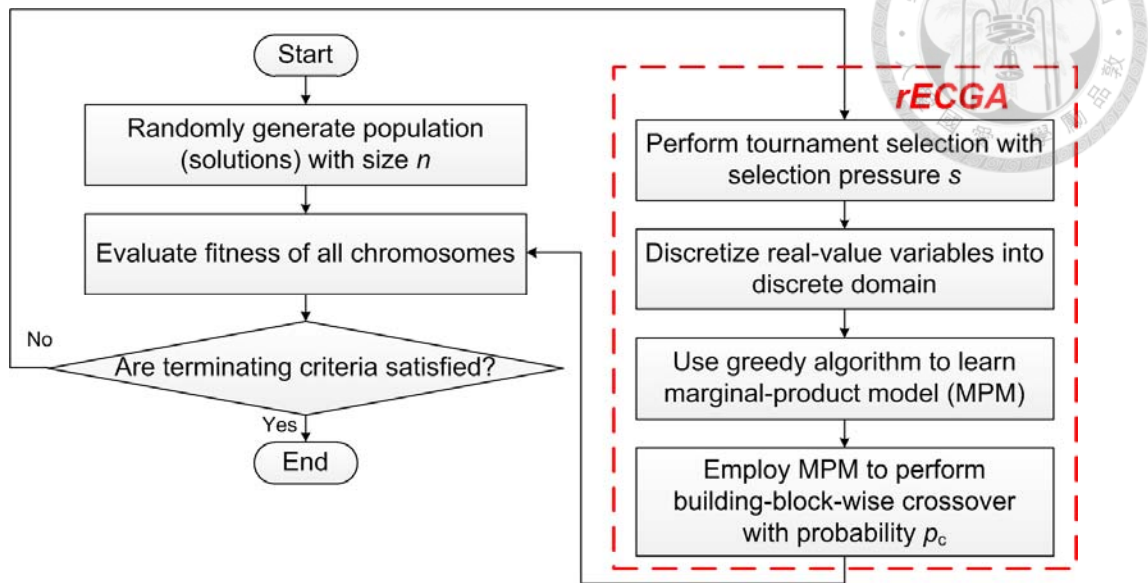
Fig. 2.7. Flow chart of real-coded expended compact genetic algorithm (rECGA). The procedures enclosed by dash line are main operation of rECGA.
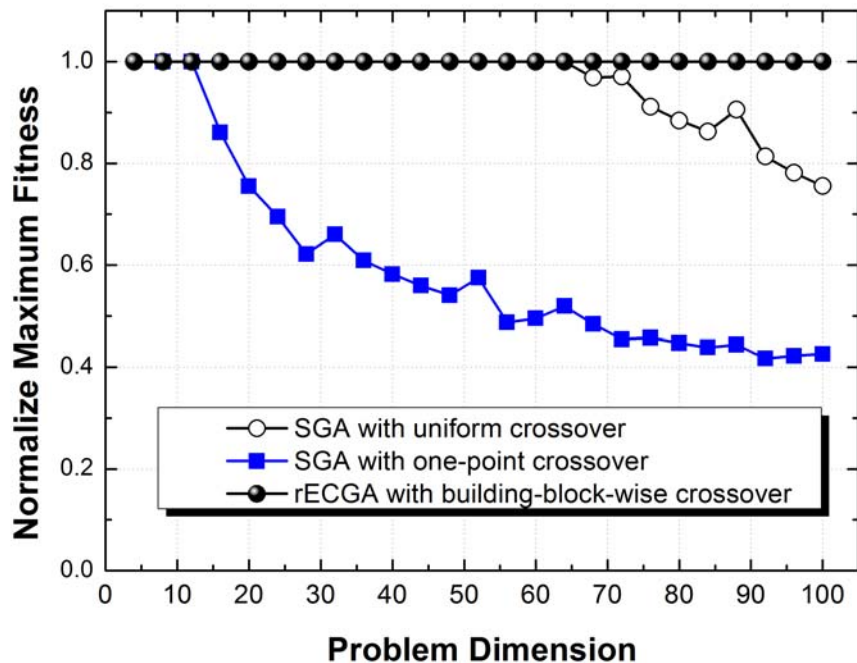


Fig. 2.8. Experiment results of solving artificial problem $G$ of rECGA and SGA. The parameter in both SGA and rECGA are $n = 50000$, 100000 total function evaluations and tournament selection pressure $s = 30$ without mutation operator.

## 2.3    Algorithm Design

### 2.3.1    Design of Overall Architecture

According to the no-free-lunch theorem in search algorithm [18], if the algorithm can be more specifically designed for a particular problem, the search ability is enhanced, with the sacrifices of the performance on other problems that we do not care. Therefore, our proposed algorithm is designed especially for circuit synthesis, and is anticipated to outperform other general search algorithms on this type of problems.

Since the solution space of circuit synthesis performs multimodal and deceptive characteristics, our proposed algorithm combines rECGA and ES with special niching method to handle these difficulties instead of only to use SGA. The rECGA performs a more accurate crossover operation when problem has the deceptive characteristic, and generates good solution patterns among large solution space; while the niching method keeps diversity of population by maintaining a few individuals to represent the information for each local optimum, and uses localized competition to exploit solutions. ES enhances exploration on each solution maintained by niching method and leads global optimum to outstand other local optima. In a simple word, rECGA takes charge of global search of solutions while ES is responsible for fast local search.

Fig. 2.9 illustrates the flow chart of the proposed circuit synthesis algorithm. The procedures of rECGA and ES are the cores of this algorithm, and they are highlighted in the figure. The details of each procedure in this flow chart are presented accordingly.
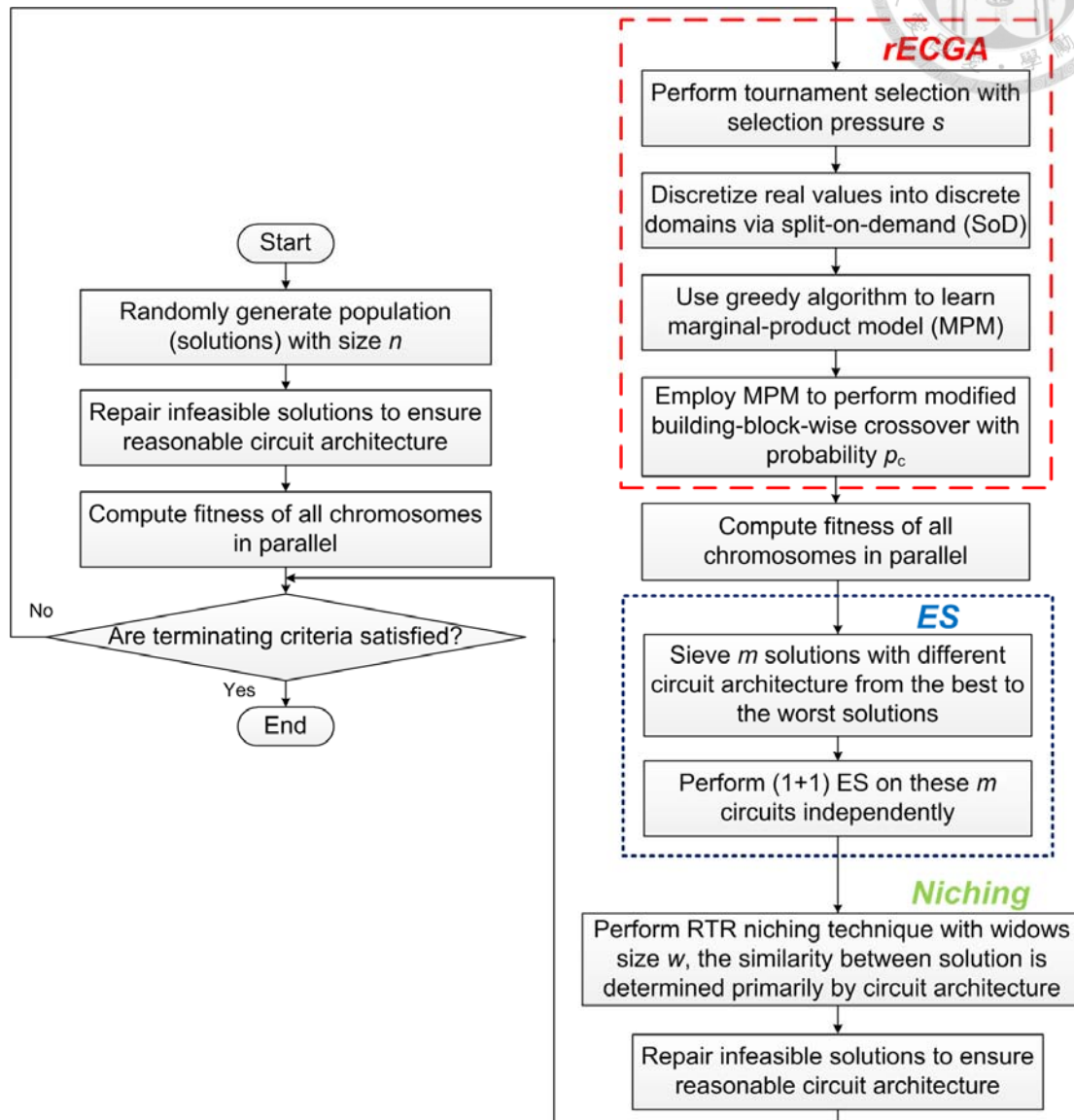
Fig. 2.9. Flow chart of our proposed circuit synthesis algorithm. The procedures of rECGA and ES are the cores of this algorithm, and they are highlighted in the figure. In a simple word, rECGA takes charge of global search of solutions and ES is responsible for fast local search.

### 2.3.2 Transformation between Circuits and Building Blocks of GA

In order to perform rECGA, it is beneficial to transform the circuit to the building blocks (BBs) of GA; this ideal is novel in the circuit optimization algorithm. Fig. 2.10 shows how to encode circuit to chromosomes (candidate solutions represented in GA) and transform the components to primitive building blocks of GA before performing linkage learning. Each electrical parameter of component has pre-defined realistic upper and lower bounds. A special circuit type called *empty* is used to eliminate redundant components [9] and it can ensure minimal order of optimal circuit. The relation between terminology of GA and circuit is shown in Table 2.1.
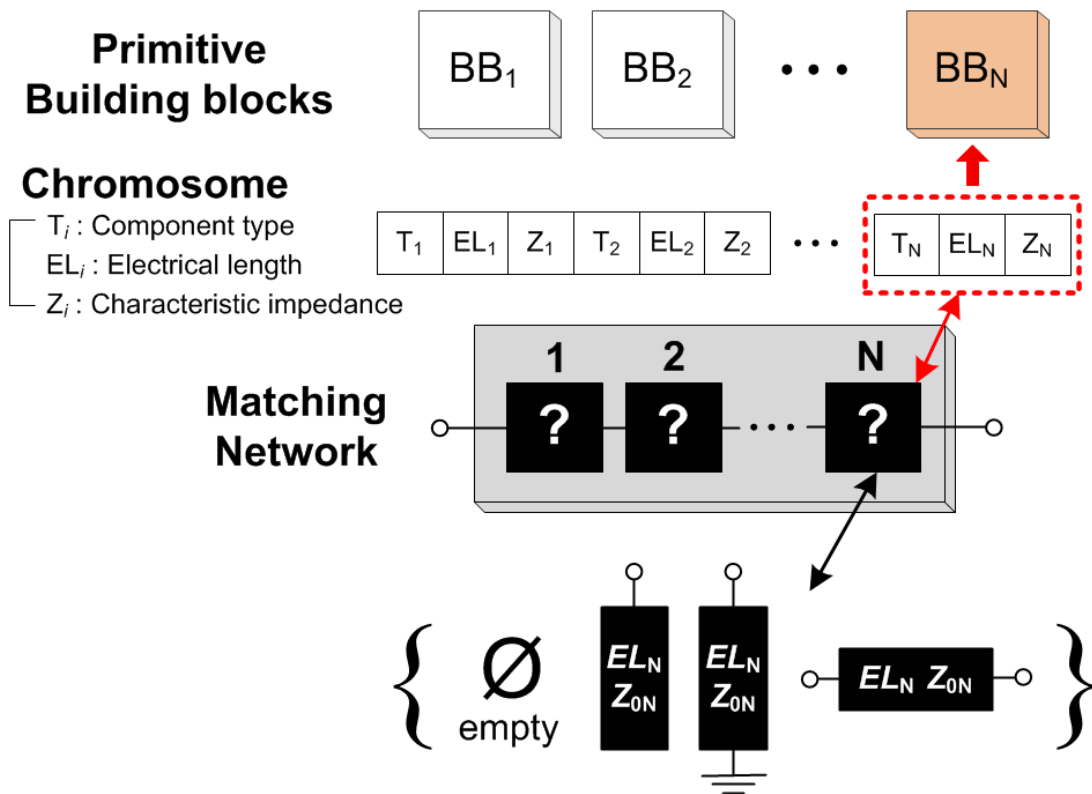


Fig. 2.10. Novel representation schemes of matching network in our algorithm. Each distributed component consists of type, electrical length and characteristic impedance, and we group these parameters into primitive building blocks of GA before performing linkage learning.

Table 2.1. Terminology synopsis.

| GA | Circuit |
|---|---|
| Gene | Circuit type<br>Electrical length (°)<br>Characteristic impedance (Ω) |
| Allele | Value of gene |
| Building Blocks | A Sub-circuit has at least one component |
| Chromosome | A matching network |
| Population | Set of matching network |

### 2.3.3 Computation of Circuit Response

In order to get the fitness of a chromosome (matching network) defined in Eqn. (2.6), the Eqn. (2.1) should be evaluated first; the $\Gamma(f_i)$ is the given sampling data, thus the only variable need to compute is $\Gamma_{in}(f_i)$. Our computation method is succinct: for each frequency, calculating individual tow-port $S$-matrix of each component first, and employing signal-flow graph to connect all two-port $S$-matrix and obtain $S$-parameters of the matching network. The computation effort of this approach is lower than using nodal analysis.

According to the basic definition of two-port S-matrix [38], we derive the general formula of $S$-parameters of each distributed component as

$$[S_{open\ stub}(f)] = \begin{bmatrix} \dfrac{-Z_{ref}}{Z_{ref} + 2Z_0 \dfrac{-1}{\tan(EL\dfrac{f}{f_0}\dfrac{\pi}{180})}} & S_{21} \\ 1 + S_{11} & S_{11} \end{bmatrix} \tag{2.9}$$

$$[S_{short\ stub}(f)] = \begin{bmatrix} \dfrac{-Z_{ref}}{Z_{ref} + 2Z_0 \tan(EL\dfrac{f}{f_0}\dfrac{\pi}{180})} & S_{21} \\ 1 + S_{11} & S_{11} \end{bmatrix} \tag{2.10}$$

$$[S_{straight\ line}(f)] = \begin{bmatrix} \dfrac{Z_0(1+\Gamma_0 e^{-j2EL\frac{f}{f_0}}) - Z_{ref}(1-\Gamma_0 e^{-j2EL\frac{f}{f_0}})}{Z_0(1+\Gamma_0 e^{-j2EL\frac{f}{f_0}}) + Z_{ref}(1-\Gamma_0 e^{-j2EL\frac{f}{f_0}})} & S_{21} \\ \dfrac{(1+S_{11})(1+\Gamma_0)}{(1+\Gamma_0 e^{-j2EL\frac{f}{f_0}})e^{jEL\frac{f}{f_0}}} & S_{11} \end{bmatrix} \tag{2.11}$$

$$\Gamma_0 = \frac{Z_{ref} - Z_0}{Z_{ref} + Z_0} \tag{2.12}$$

where *EL* denotes electrical length of transmission line, $Z_0$ denotes characteristic impedance, $Z_{ref}$ represents terminated impedance and $f_0$ denotes reference frequency.

Fig. 2.11 shows signal flow graph of cascading two *S*-matrixes and the correspondent *S*-parameters are

$$S_{11} = S_{11A} + \frac{S_{21A}S_{12A}S_{11B}}{1 - S_{22A}S_{11B}} \tag{2.13}$$
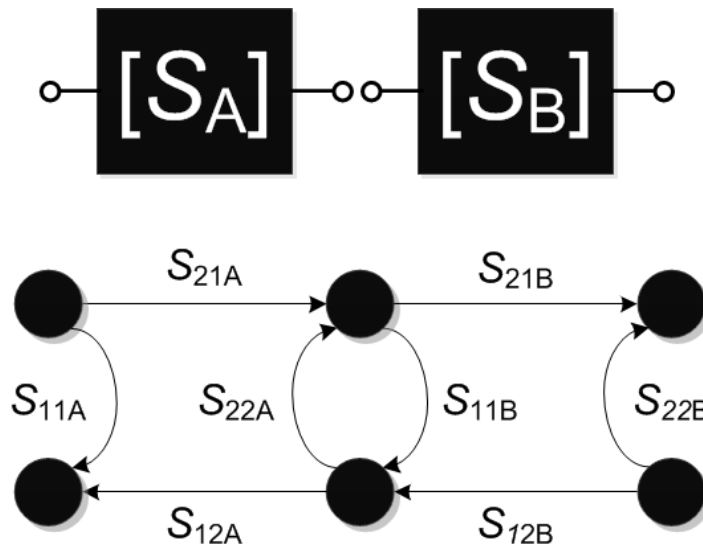


Fig. 2.11. Signal flow graph used to evaluate connected *S*-matrixes.

$$S_{21} = \frac{S_{21A}S_{21B}}{1 - S_{22A}S_{11B}} \qquad (2.14)$$

$$S_{12} = \frac{S_{12A}S_{12B}}{1 - S_{11A}S_{22B}} \qquad (2.15)$$

$$S_{22} = S_{22B} + \frac{S_{21B}S_{12B}S_{22A}}{1 - S_{22A}S_{11B}} \qquad (2.16)$$

Cascading the two circuits, the $Z_T(f_i)$ can be transformed to $\Gamma_{in}(f_i)$ as

$$\Gamma_{in} = S_{11} + \frac{S_{21}S_{12}\dfrac{Z_T(f_i) - Z_{ref}}{Z_T(f_i) + Z_{ref}}}{1 - S_{22}\dfrac{Z_T(f_i) - Z_{ref}}{Z_T(f_i) + Z_{ref}}} \qquad (2.17)$$

An example to evaluate the $\Gamma_{in}(f_i)$ of a given 4th-order matching network is shown in Fig. 2.12. The first step is to cascade the *S*-matrix of each circuit components in a bottom-up approach, followed by transforming the terminated impedance to the $\Gamma_{in}(f_i)$.
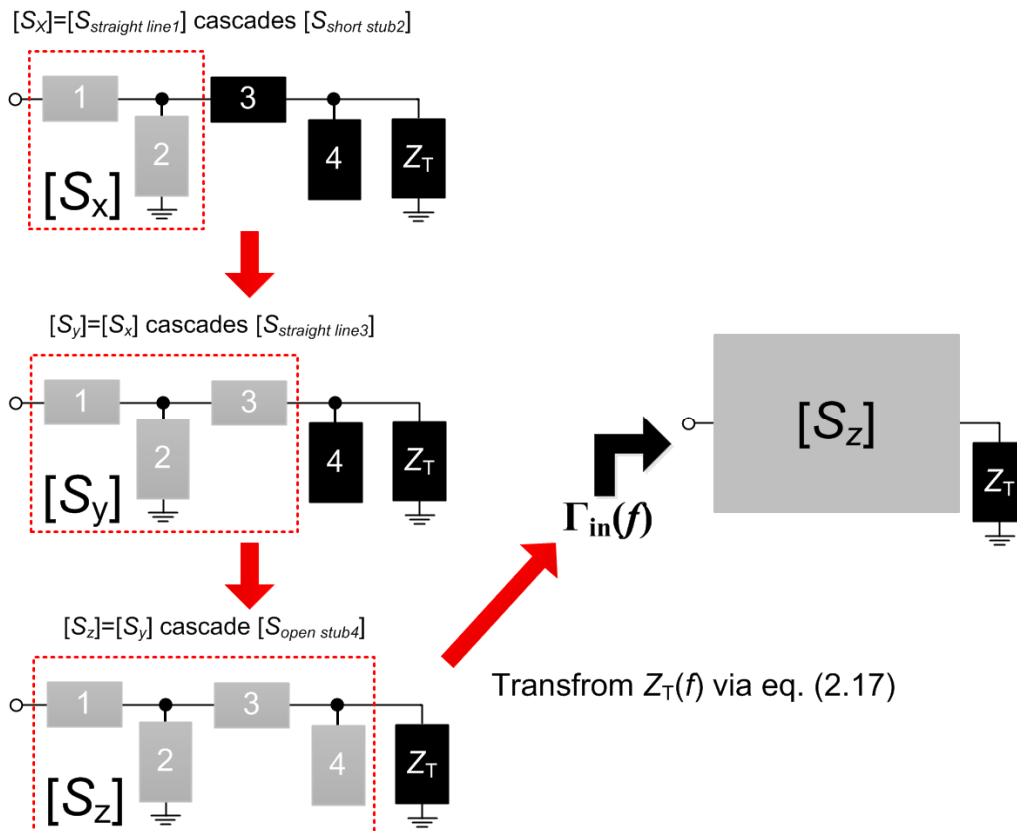


Fig. 2.12. Example of calculating circuit response of a 4th-order matching network.

## 2.3.4  Implementation of Circuit Synthesis Algorithm

The first step in Fig. 2.9 is to randomly generate $n$ solution candidates (population with size $n$) under boundaries of electrical parameters. To ensure reasonable circuit architecture for realistic fabrication, we set constraints to all solutions and repair infeasible solutions, such as to avoid multiple shunt-connected components ($\geq 3$) joining in a node. For instance, a solution for $4^{th}$-order matching network contains 4 short stubs is difficult to layout in general. The following procedure is to evaluate the fitness of each feasible solution, which is the most time-consuming step. Fortunately, we can adopt functional decomposition and compute in parallel described in next section.

In rECGA procedures, tournament selection is adopted due to its efficiency and stability [39]. In tournament selection, the best solution always survives, thus the noise effect of random number generator is reduced. The selection pressure $s$ should be selected carefully to avoid immature convergence. In general, the more difficult the problem is, the larger the selection pressure $s$ should be.

Discretization procedure is used to transform problems from the continuous domain to the discrete domain for performing linkage learning. Fig. 2.13 illustrates flow of discretization procedure, where $b$ denotes the interval value of bins, $L$ denotes the chromosome length and $h(i)$ denotes the length of code vector for gene$_i$. The first step is to use discretization method to construct code vectors with different number of bins per gene, followed by discretizing each component parameter such as type, $EL$ and $Z_0$ according to the code table. Previous research has shown that split-on-demand (SoD) discretization method outperforms fixed-height-histogram and fixed-width-histogram to discretize the continuous domain [23]-[24], since SoD automatically determines the number of bins to fit problem characteristic. Fig. 2.14 shows the flow of SoD, where the $\gamma$ denotes split rate; the smaller $\gamma$ causes the larger number of bins. An example which
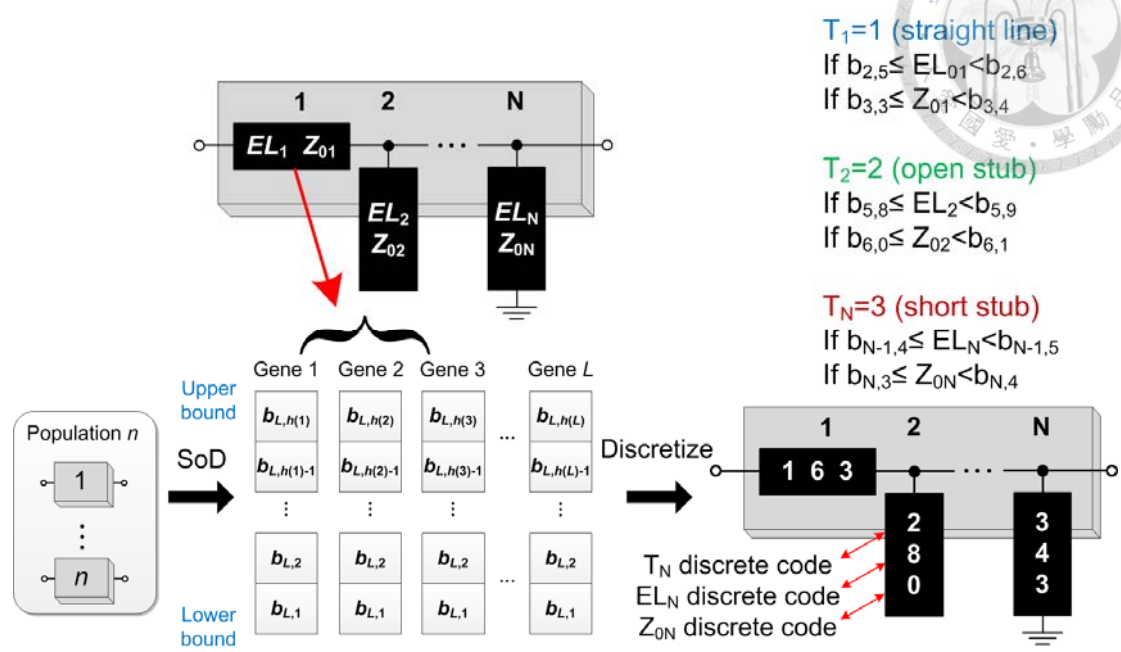
Fig. 2.13. Flow of discretization procedure. Where the $b$ denotes the interval value of bins, $L$ denotes the chromosome length and $h(i)$ denotes the length of code vector for $gene_i$. The first step is to use split-on-demand (SoD) to construct code table with different number of bins per gene, followed by discretizing each component parameter such as type, $EL$ and $Z_0$ according to the code table.
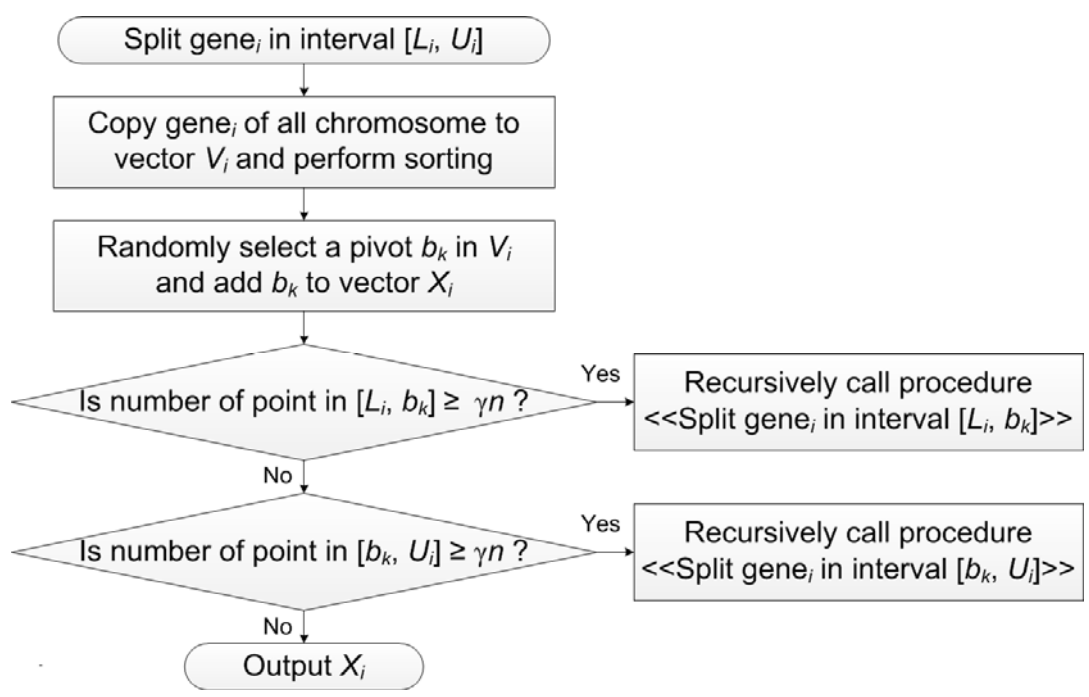


Fig. 2.14. Flow chart of SoD.

illustrates the generation of code table by SoD is shown in Fig. 2.15. For each electrical parameter, the first step is to collect correspondent parameter values from population, followed by sorting collected values, and finally employ SoD to generate code vector $X_i$ for discretization of $gene_i$. Each real value represented by $gene_i$ maps to specific discrete code according to the correspondent interval value in code vector $X_i$.



| Dimension 1 (*Type* 1-3) | | Dimension 2 (*EL* 0-100°) | | Dimension 3 ($Z_0$ 40-110 Ω) | |
|---|---|---|---|---|---|
| Interval | Code | Interval | Code | Interval | Code |
| 1-1.5 | 0 | 0-30 | 0 | 40-80 | 0 |
| 1.5-2.2 | 1 | 30-50 | 1 | 80-100 | 1 |
| 2.2-3 | 2 | 50-65 | 2 | 100-110 | 2 |
| | | 65-100 | 3 | | |
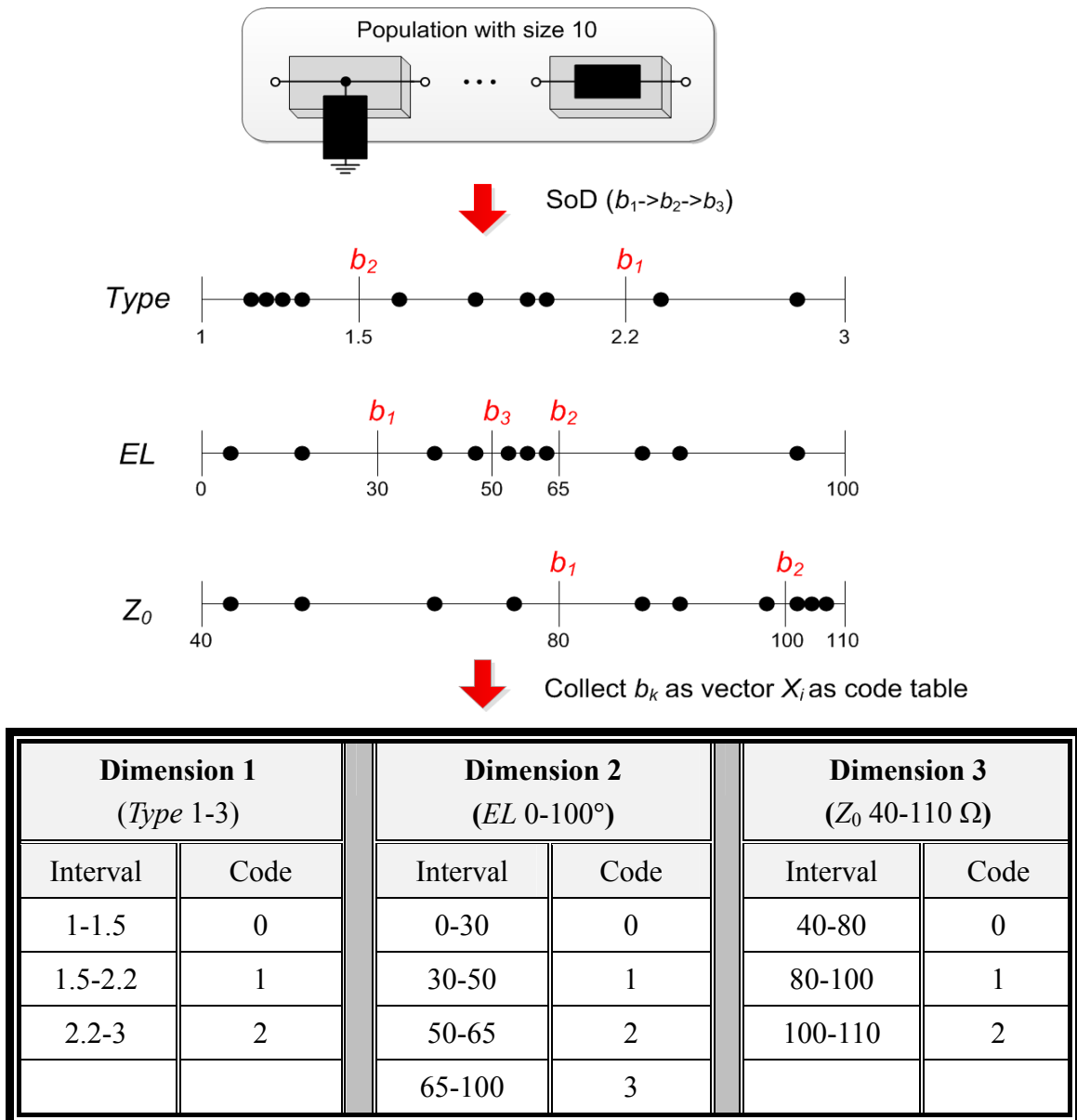
Fig. 2.15. Example of SoD with population size $n = 10$ and split rate $\gamma = 0.5$. The $b_k$ denotes the interval value of bins choose in iteration $i$ and $X_i$ denotes the vector containing all $b_k$ of $gene_i$. Each electrical parameter value represented by $gene_i$ maps to a specific discrete code according to the correspondent interval value in code vector $X_i$.

25
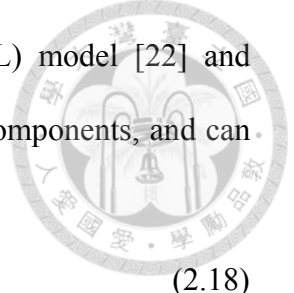
The rECGA is based on minimum description length (MDL) model [22] and greedy algorithm [37]. The MDL metric is the summation of two components, and can be expressed as

$$MDL = D_{Model} + D_{Data} \tag{2.18}$$

$$\text{Model Complexiy } D_{Model} = \log_2 n \sum_{I} (\prod_{i}^{|BB(I)|} \chi_i) \tag{2.19}$$

$$\text{Compressed Population Complexity } D_{Data} = n \sum_{I} E(M_I) \tag{2.20}$$

$$E(M_I) = -p \log_2 p \tag{2.21}$$

where $n$ denotes population size, $|BB(I)|$ denotes length of $I^{th}$ building blocks, $\chi$ denotes cardinality of gene in $BB(I)$, $p$ denotes statistic probability of $BB(I)$ and $E(M_I)$ represents Shannon entropy [25] of $I^{th}$ marginal distribution. Fig. 2.16 shows how to evaluate MDL for a given marginal-product model (MPM) [12]; the first step is to count the probability of circuit patterns sampled from each building block, followed by using this statistic probability to calculate result of Eqns. (2.19)-(2.21).
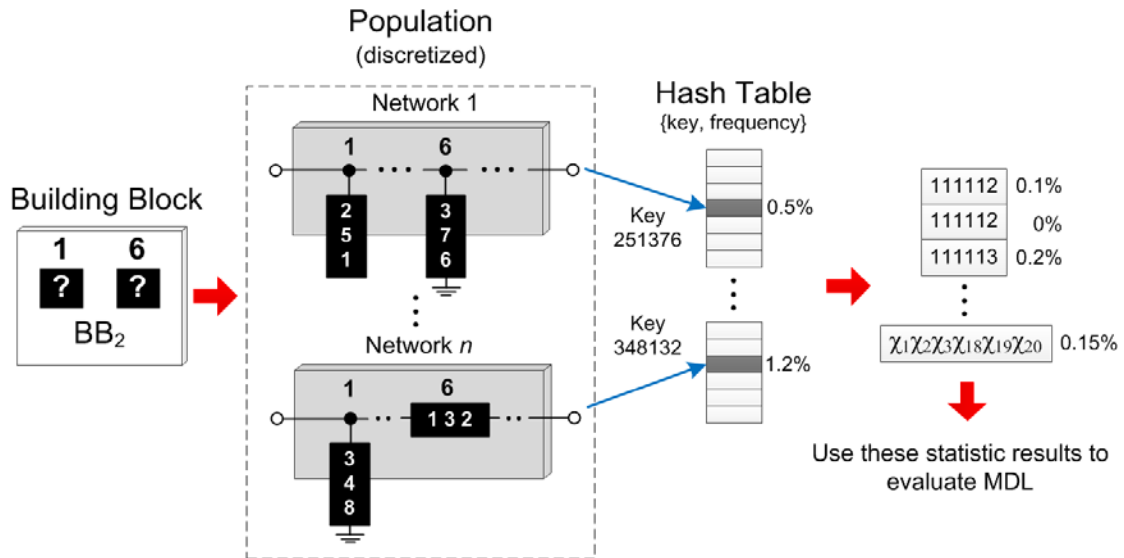


Fig. 2.16. An example to show how to computing the MDL. The first step is to count the probability of circuit pattern, followed by using this statistic probability and evaluating eqs (2.19)-(2.21).

MPM is the classification of building blocks which denotes what circuit components should be grouped as a sub-circuit, because promising solutions in population have the same sub-circuit pattern according to the MDL model. The MPM with minimal MDL represents the optimal classification of building blocks. It is necessary to use another search algorithm to find a optimum MPM, and the greedy algorithm is efficient and suitable for this problem. Fig. 2.17 illustrates the flow of MPM learning.
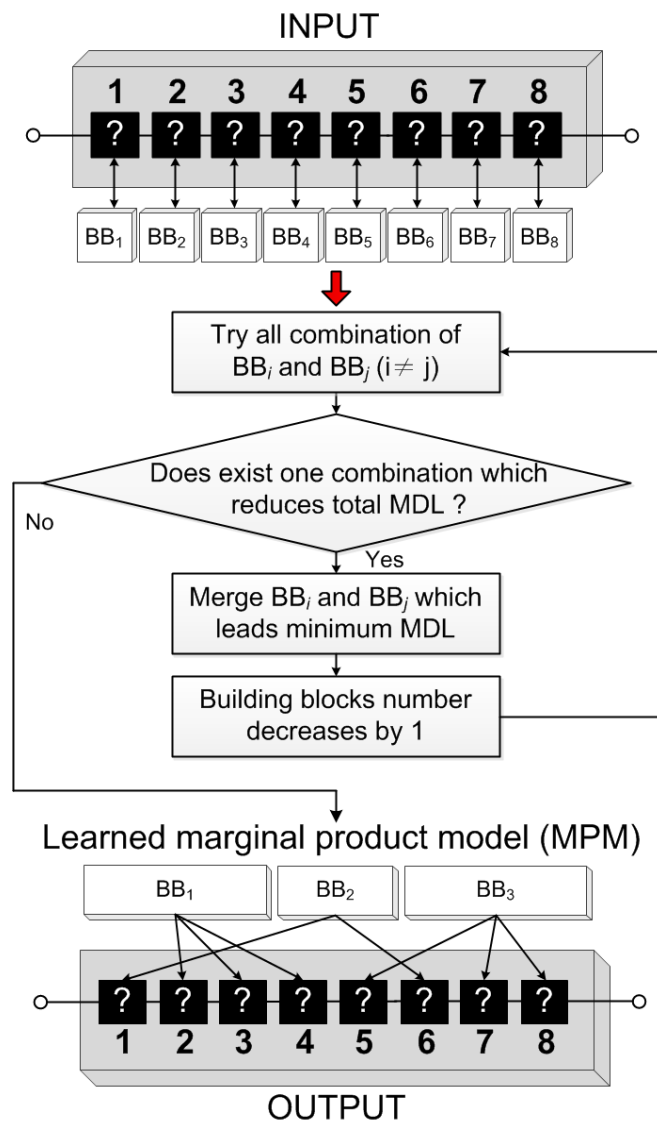


Fig. 2.17. Example of circuit linkage learning based on MDL model and greedy algorithm. This matching network is transformed to 8 building blocks of GA at first, and it is transformed to 3 building blocks after greedy MPM search.

The MPM points out which component should be linked with other components, because good solutions in population have the same combination pattern according to the MDL model. Fig. 2.18 illustrates a typical method that using MPM to perform building-block-wise crossover to generate new solutions; it uses roulette-wheel sampling to sample circuit patterns and combines them, the statistic probability is counted when computing MDL. However, we employ MPM to perform another version of building-block-wise crossover shown in Fig. 2.19, which exchanges sub-circuits represented by building blocks with a probability of $p_c$. The reason of not using roulette-wheel sampling to generate circuit patterns is that the method may destroy original real-valued parameters due to sampling noise.
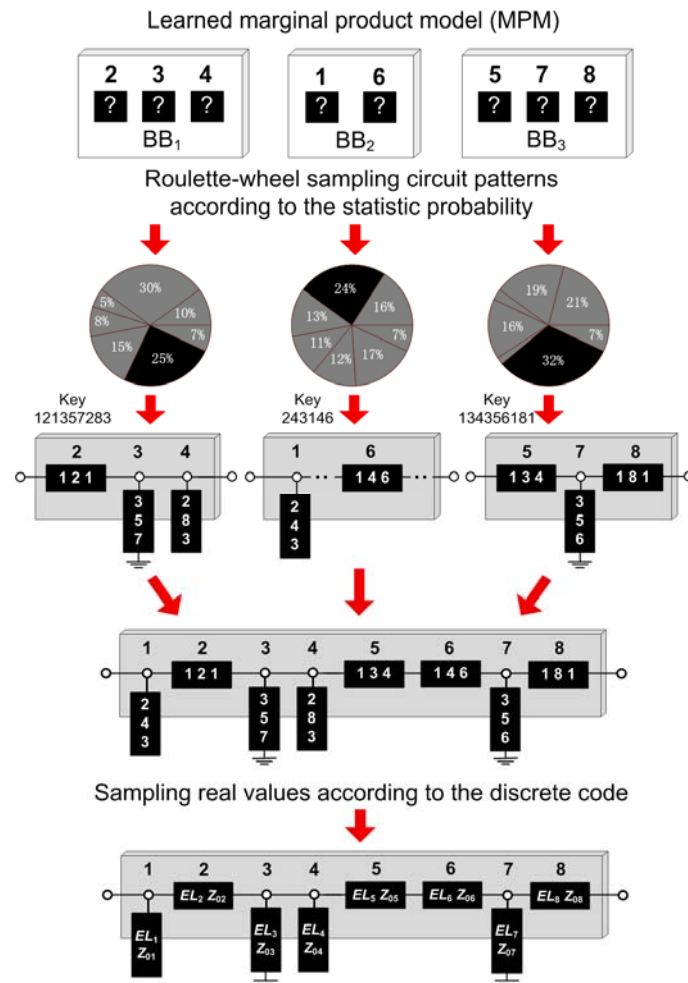


Fig. 2.18. A flow of typical building-block-wise crossover. It uses the learned MPM and roulette-wheel to sample circuit patterns and combines them to a new solution.
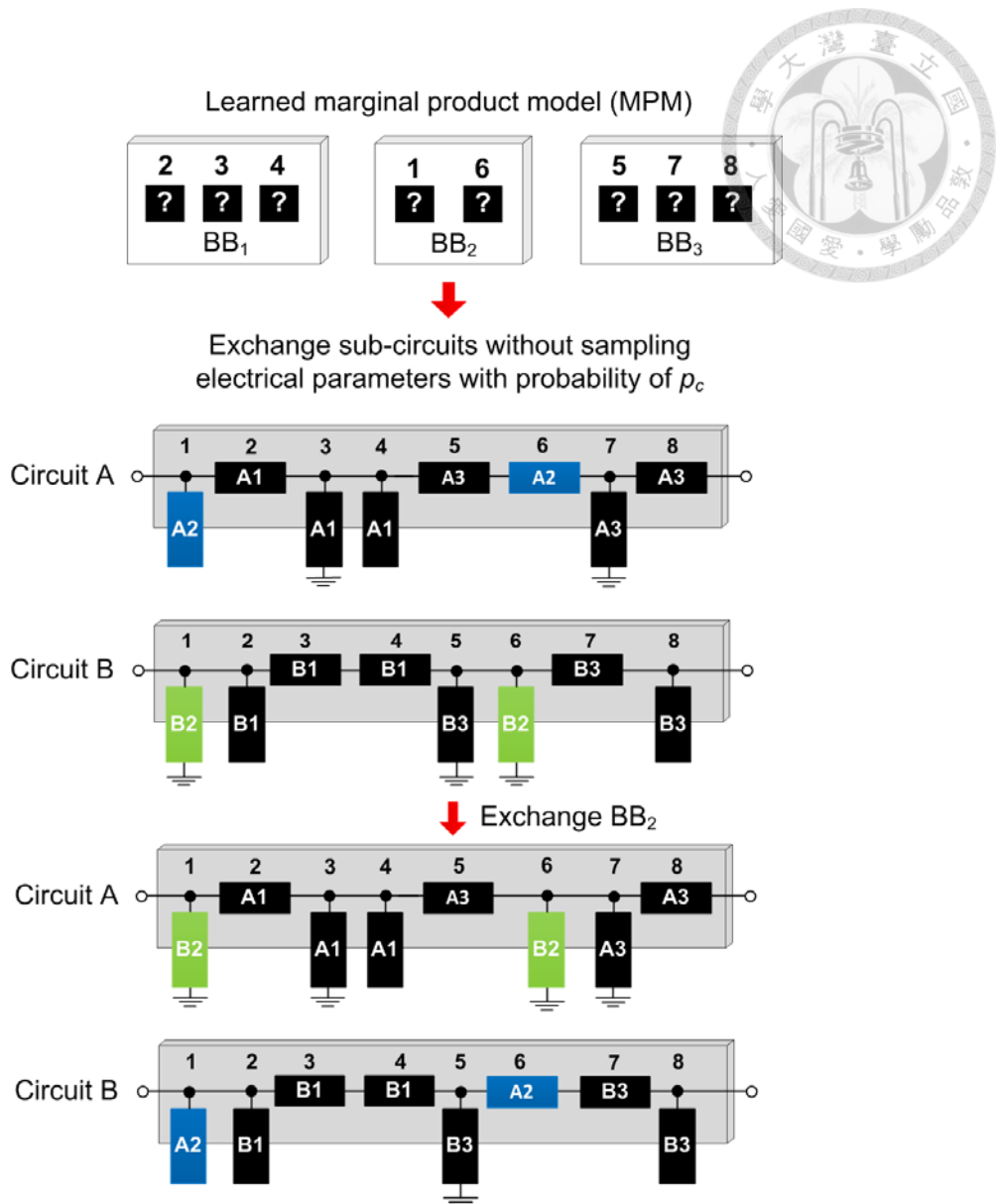
Fig. 2.19. A modified version of building-block-wise crossover we apply. It uses the learned MPM to exchange sub-circuits without changing the electrical parameters.

After rECGA operation, the fitness of each chromosome is evaluated again, and the ES is used to exploit the solutions with near optimal or optimal circuit topologies found by rECGA. The first step is to sieve $m$ solutions with different circuit architectures from the best solution to the worst solution, followed by performing (1+1) ES [16] on these $m$ solutions independently, and each sub-routine consumes $n / m$ function evaluations. ES only perturbs the electrical parameters of these $m$ circuits without changing circuit topology. Fig. 2.20 shows this procedure. When $m$ is small, each selected circuits can be

tuned further more; while *m* is large, each selected circuits can only change slightly, but it can avoid immature convergence due to pay lot of attentions on several different circuits. The reason to perform ES on individuals instead of $(\mu+\lambda)$ ES is to avoid interacting between immature circuit architectures with each other. This procedure plays a key role in the proposed algorithm as it promotes quality of those good circuits and helps the optimal circuit to outstand in lots of local optima.

(1+1) ES generates solution candidate according to following equation

$$y = x + \sigma N(0,1) \tag{2.22}$$

where *x* denotes the parent, *y* denotes the offspring and $\sigma$ is the step size to adjust standard Gaussian distribution *N*. The solution of next generation is selected in the parent *x* and the offspring *y* according to the fitness. The standard Gaussian random number pair can be generated from the Box-Muller transform method [40] as

$$\begin{aligned} &\textbf{repeat} \\ &\quad x \leftarrow \text{unifrom\_rand()} \\ &\quad y \leftarrow \text{unifrom\_rand()} \\ &\quad d \leftarrow x^2 + y^2 \\ &\textbf{until } 0 < d < 1 \\ &f \leftarrow \sqrt{-2(\ln d)/d} \\ &\textbf{return } (f \times x, f \times y) \end{aligned} \tag{2.23}$$

We use the 1/5 success rule to update the step size [17]. The following equation shows a method to update step size

$$\sigma \leftarrow \sigma \times \exp(\frac{1}{3} \times \frac{p_s - p_{target}}{1 - p_{target}}) \tag{2.24}$$

where the $p_s$ denotes probability of successful offspring in the recent *d* generation and $p_{\text{target}}$ is 1/5. If there are 1/5 offspring are better than parent, the step size $\sigma$ will be increased, or be decreased otherwise. Both the initial step size $\sigma$ and *d* which we apply are 5.0 for local search.
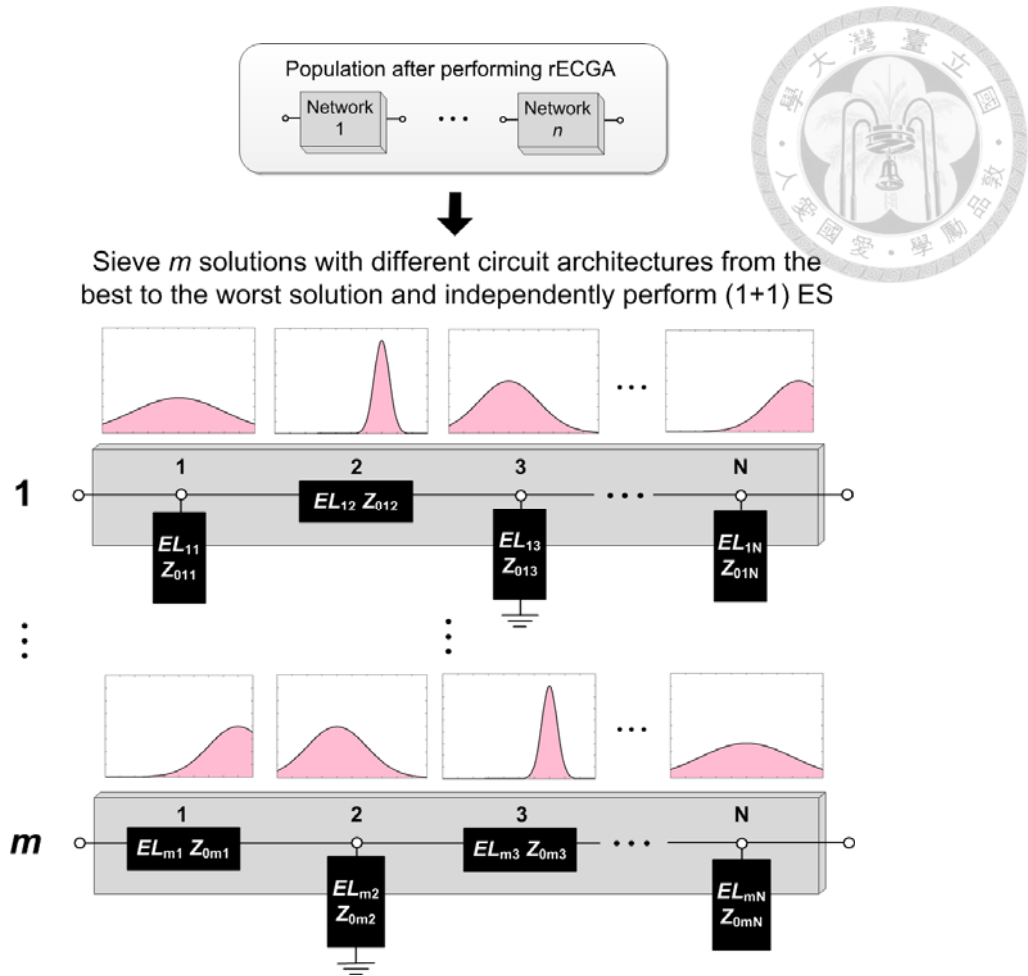
Fig. 2.20. The ES procedure in the proposed algorithm. The first step is to sieve *m* solutions with different circuit architectures from the best solution to the worst solution, followed by independently performing (1+1) ES with Gaussian distribution perturbation and self-adaption on these *m* solutions.

It is beneficial to use niching technique to handle multimodal characteristic of circuit synthesis problem. The function of niching method is to keep diversity of population by maintaining a few individuals to represent the information for each local optimum, and use localized competition to exploit solutions. Our algorithm employs the restricted tournament replacement (RTR) [26]. RTR restricts competitions between new solutions and the most similar solution in randomly selected *w* original solutions before performing rECGA and ES. A method to select appropriate *w* in circuit synthesis problem is to count the number of all possible topology; for instance, the topology count for a *N*-order matching network (component are transmission line, open stub, short stub)

with the only constraint that a node cannot join $\geq 3$ shunt-connected components can be

derived as follow

$$\text{Topology count} = (\text{total number of possible topologies}) -$$
$$(\text{count of infeasible topologies}) \qquad (2.25)$$
$$= 3^N - \sum_{i=3}^{N} 2^i (N - i + 1)$$

For example, when $N$ is 4, the feasible topology count is

$$\begin{aligned} \text{Topology count} &= 3^4 - 2^3(4-3+1) - 2^4(4-4+1) \\ &= 81 - 16 - 16 \qquad\qquad\qquad\qquad (2.26) \\ &= 49 \end{aligned}$$

Therefore, the $w$ can be selected as slight larger than 49 to keep all possible feasible

topology in solution space, and perform local competition between the similar topology.

Table 2.2 shows the feasible topology count with different matching network order,

which can be used as reference value of window size $w$.

The similarity is determined by hamming distance [41] between electrical

parameters of two chromosomes, and there is a penalty added to the calculated distance

if the circuit architecture is not the same to ensure the diversity of circuit topologies in

the solution space. The more similar the circuits are, the shorter hamming distance is. In

our case, the penalty of 100 is added to hamming distance if a component type in

specified position is not the same between two chromosomes. Fig. 2.21 shows an

example to calculate the similarity between chromosomes and illustrate flow of RTR.

Table 2.2.  Feasible circuit topology count with different network order, which can be used as reference value of window size $w$.

| Network Order | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|
| Count | 19 | 49 | 155 | 521 | 1731 | 5601 | 17701 | 55033 |
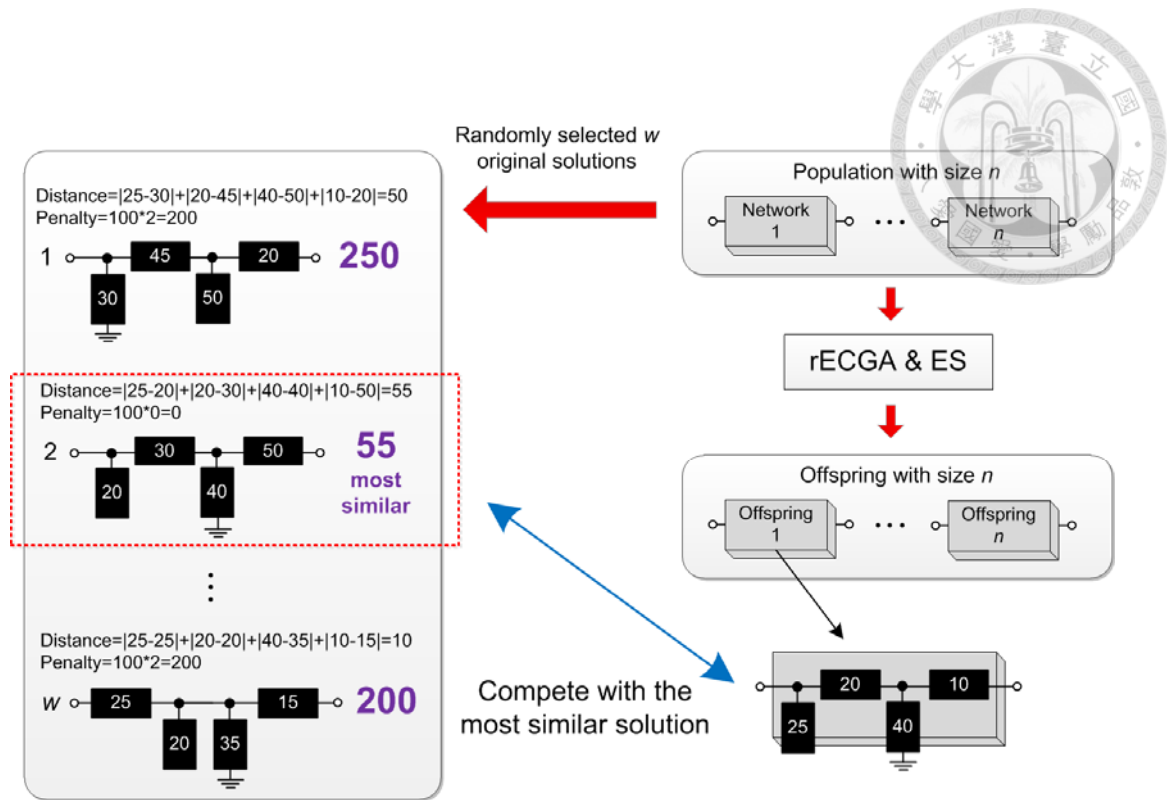
Fig. 2.21. An example showing how to use RTR for generating offspring in next generation. For each solution produced by ES, the first step is to randomly select *w* solutions before performing rECGA, followed by finding the most similar original solution and compete with it, if the offspring is the better, replace the original solution, otherwise discard the offspring.

After one generation of evolution process, our algorithm repairs infeasible solutions again to ensure realistic solutions. Above procedures iterate until the termination condition is satisfied, function evaluation count reaches the limit, the program hit time constraint, or the population is converged.

Table 2.3 shows the time complexity of each procedure in the proposed algorithm. The most time-consuming steps are evaluating the fitness in rECGA and ES operations, MPM learning or RTR. However, this time complexity is efficient enough to perform circuit synthesis under real-time constraint.

Table 2.3. Time complexity of each procedure. Where $n$ denotes the population size, $l$ denotes the chromosome length, $C$ denotes the time to compute circuit response, $s$ denotes selection pressure and $w$ denotes windows size of RTR.

| Procedure | Time Complexity |
|---|---|
| Evaluate fitness | $\Theta(nlC)$ |
| Tournament selection | $\Theta(ns)$ |
| SoD | $\Theta(nl \log n)$ |
| MPM learning | $\Theta(nl^2 \log l)$ |
| Building-block-wise crossover | $\Theta(nl)$ |
| (1+1) ES | $\Theta(nlC)$ |
| RTR | $\Theta(nwl)$ |
| Repair infeasible solutions | $\Theta(nl)$ |
| Total | $T(n) \approx \max\{\Theta(nlC),\ \Theta(nl^2 \log l),\ \Theta(nlw)\}$ |

## 2.3.5 Speedup with Parallel Computation

Evaluate the fitness for each feasible solution may be the most time-consuming step than other procedures. Fortunately, it performs good data parallelism to be evaluated in parallel. The primitive task is to compute response of a circuit. It is recommended to use distributed block-data decomposition scheme [27] to schedule tasks. There are $p$ processors and relative id are $0, 1, \cdots, p-1$, the computation tasks scheduled to each processor as

$$\text{Computation range} = C + [\left\lfloor \frac{id * n}{p} \right\rfloor, \left\lfloor \frac{(id+1)*n}{p} \right\rfloor - 1] \tag{2.27}$$

$$\text{Task } j \text{ is belong to processor } \left\lfloor \frac{p(j+1)-1}{n} \right\rfloor \tag{2.28}$$

where $C$ denotes a constant used to shift index, which is problem dependently. Each process takes charge in computing the fitness of chromosome of the index range in (2.27) independently.

In our case, we use multi-thread programming on a symmetric-multiprocessing (SMP) machine to handle this bottleneck of time consumption. Fig. 2.22 shows an example of parallel computing circuit response which uses distributed block-data decomposition scheme. If the problem size is too large to use single computer, the message passing between multi-computers can be used to speed up further more; the general purpose graphic compute unit (GPGPU) can also be employed to accelerate computing the circuit response.

Although the operations of rECGA and ES are efficient, they can be applied with parallel computation for further speedup due to inherit data parallelism. For instance, the tournament selection can be divided into many independent parts; each part is a tournament of population. Therefore, many threads can be created to handle individual
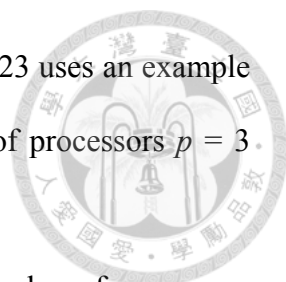
tournament according to (2.27) and synchronize each result. Fig. 2.23 uses an example to illustrate parallel version of tournament selection with number of processors $p = 3$ and $s = 20$.

Fig. 2.24 shows the speedup of our algorithm with different number of processors under SMP architecture; the speedup is slightly lower than processors number due to inherent sequential procedure in algorithm.
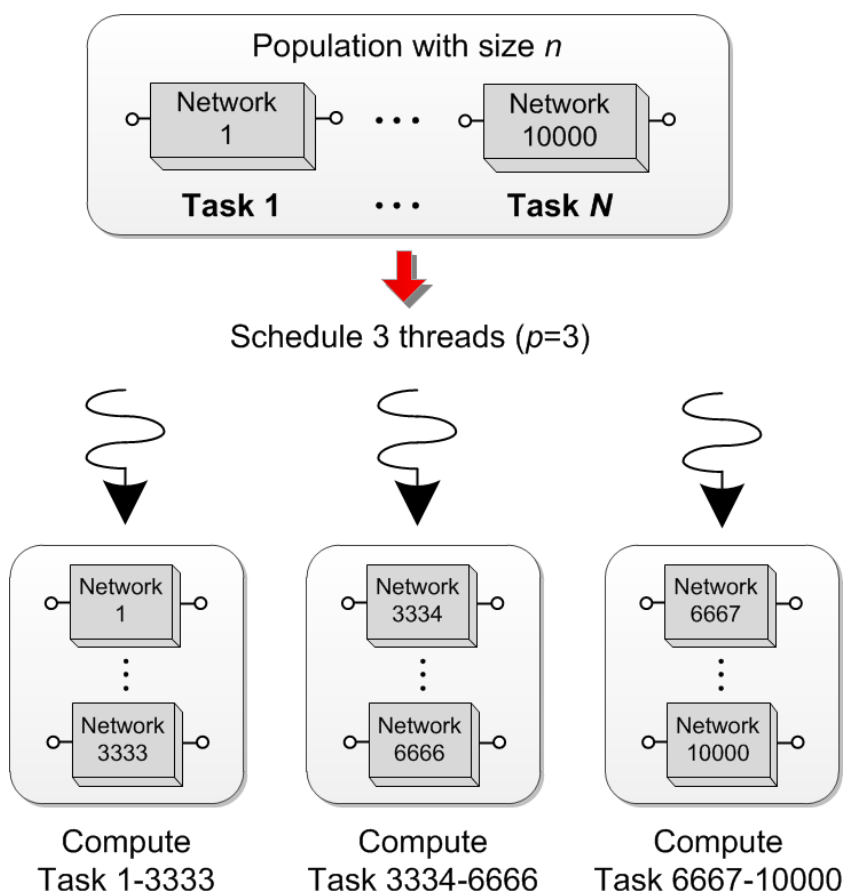


Fig. 2.22. An example of parallel computing circuit response by using distributed block-data decomposition scheme.
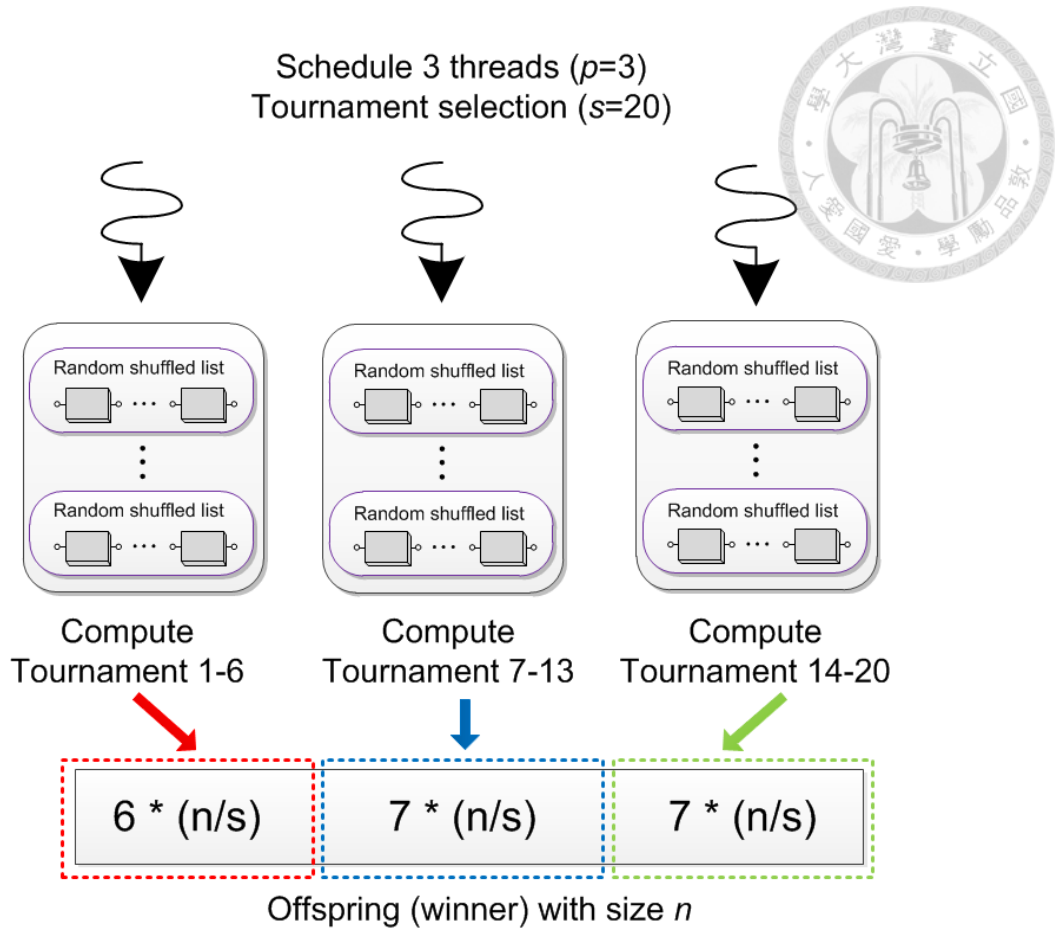
Fig. 2.23. Parallel version of tournament selection (without replacement).
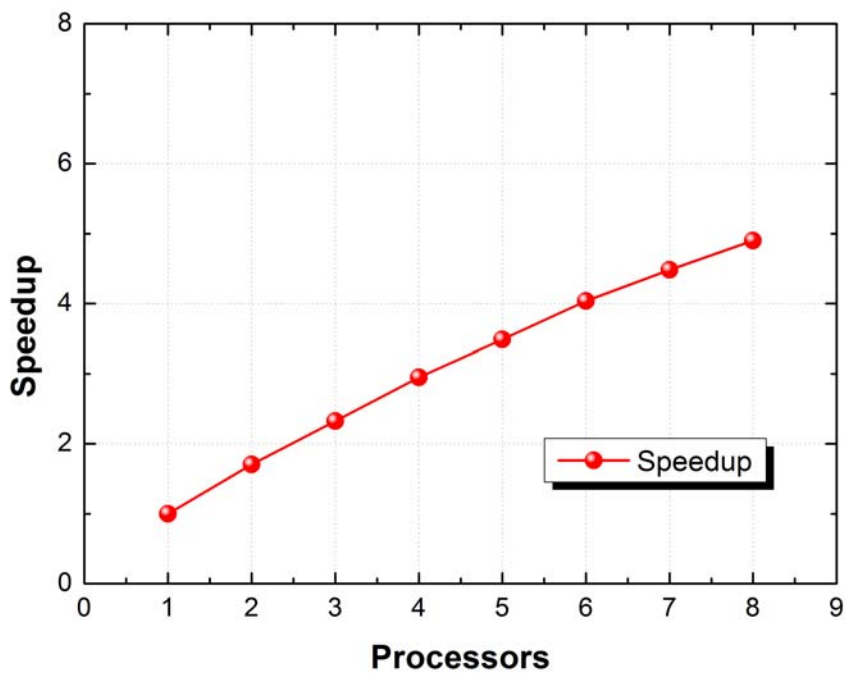


Fig. 2.24. Speedup of our algorithm with different number of processors under SMP architecture.

## 2.4    Performance Measurements

In order to evaluate the performance measurement of proposed algorithm, several computer experiments have been performed to compare with previously published work. These experiments are matching networks synthesis on transistors and a dual-band filter design [9], respectively.

### 2.4.1    Experiments of Microwave Circuit Design I: Matching Γ Curve

Without loss of generality, these experiments are performing conjugate matching syntheses on two commercial packaged transistors; they are NE-42484 [42] and ATF-58143 [43] respectively. The synthesis results are used to compare with previous published algorithm [9], such as function evaluations and average optimal solution quality.

For simplicity, the matching targets of both transistors are to match simultaneous-conjugate-matching frequency responses (i.e. $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$), and the $Z_0$ of transmission line is fixed at 50 Ω. The metric of a good matching network is based on Eqn.  (2.6).

Fig. 2.25 illustrates the first synthesis target to match NE-42484. The device is biased at recommend value in the data sheet that $V_D = 3V$ and $I_D = 20$ mA. The gate of the transistor is connected with small resistor to ensure the existence of simultaneous-conjugate-matching condition, and the effective series inductance (ESL) is also considered. The global optimum of this matching target has been obtained by enumeration with large computation efforts. Fig. 2.26 shows the curves of $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ of stabilized NE-42484 in the Smith chart, and the correspondent optimal circuits and responses found by enumeration are also illustrated. Fig. 2.27 shows the optimal matching result with ideal passive components, and the in-band region is highlighted in

the figure (2.2-2.8 GHz). The matched small signal gain almost fit to the maximum available gain (MAG) of the stabilized device due to optimized matching.

We employ the proposed algorithm and [9] to synthesize these optimal matching networks and perform statistic for comparison, such as function evaluations and average optimal fitness. The criterion of finding the optimal solution is that the final population contains at least one solution, whose circuit architecture is the same as the optimal solution and fitness is larger than 95 % that in optimal circuit. Table 2.4 shows the parameter settings of our algorithm in these experiments, while the parameters of [9] remain the same in the most part and be slightly optimized for this experiment. Since the number of feasible topology of 4th-order circuits is about 50 according to Eqn. (2.25), the restricted windows size $w$ of RTR is selected as 50, and the sieving number $m$ used in exploration procedure is chosen as 20 for exploiting only on about top 1/2 circuit topology. The initial step size $\sigma$ in ES is selected as 5.0 because this range of perturbation for electrical parameter can be regarded as fine tuning in general.

Figs. 2.28 and 2.29 show the statistic results of conjugate matching synthesis of NE-42484. Each experiment executes 30 independent runs, the $x$-axis represents population size $n$ from 100 to 10000 ($n = 100, 300, 500, 1000, \cdots, 9500, 10000$) with totally $100n$ function evaluations, and the y-axis denotes times of finding the global optimum for conjugate matching and average optimal fitness, respectively.

Since the matching problem for $\Gamma_{MS}(f)$ is too hard for SGA to solve, the optimal solution of [9] is always trapped into local optima instead of global optimum, thus the times of finding global optimum of SGA is almost zero; however, our algorithm can find global optimum with 100% statistic probability when $n$ is larger than only 500, that is, we can use smaller computation effort to find the global optimum. Besides, the average fitness of the found best solution is always larger than that in [9] apparently.

Table 2.4. Parameter settings of our algorithm.

| Parameter | Value |
|---|---|
| Selection pressure ($s$) | 20 |
| Probability of crossover ($p_c$) | 0.5 |
| Split rate of SoD ($\gamma$) | 0.5 |
| Restricted window size of RTR ($w$) | 50 |
| Initial step size of ES ($\sigma$) | 5.0 |
| Updating generation of ES ($d$) | 5 |
| Sieving number for ES ($m$) | 20 |
| Total function evaluations | $100n$ |
| Reference frequency ($f_0$) | 2.5 GHz |
| Sampling points | 30 |
| Electrical length (°) | [0, 100] |
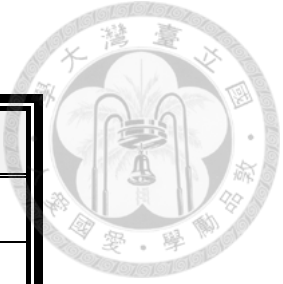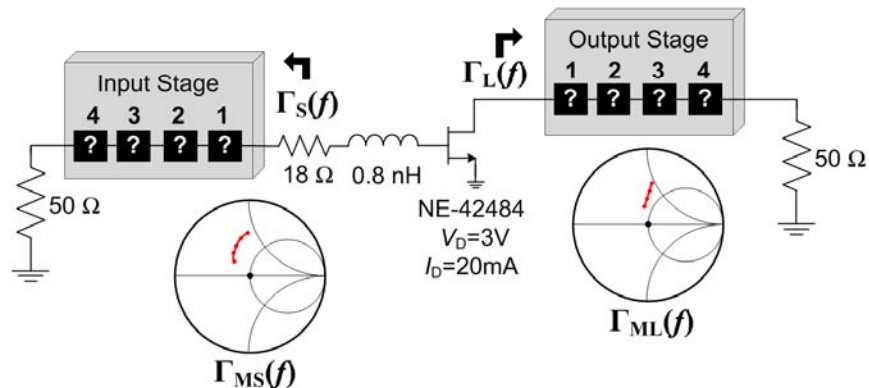


Fig. 2.25. The first example for performance measurement of our proposed algorithm. The synthesis target is to employ 4th-order network to synthesize broadband simultaneous-conjugate-matching from 2.2 to 2.8 GHz. The data sheet of NE-42484 can be obtained from the website [42].
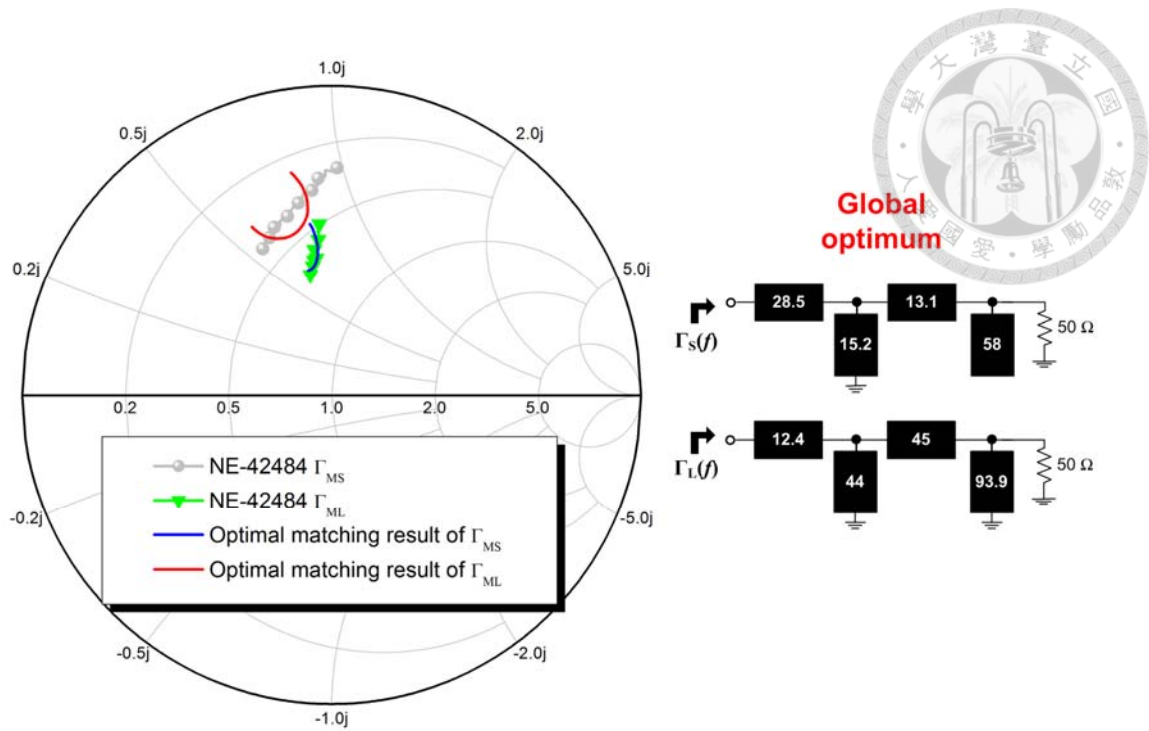
Fig. 2.26. The curves of $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ of stabilized NE-42484 transistor, and the correspondent frequency response (solid line without symbol) and circuit topology of optimal matching network are also illustrated.
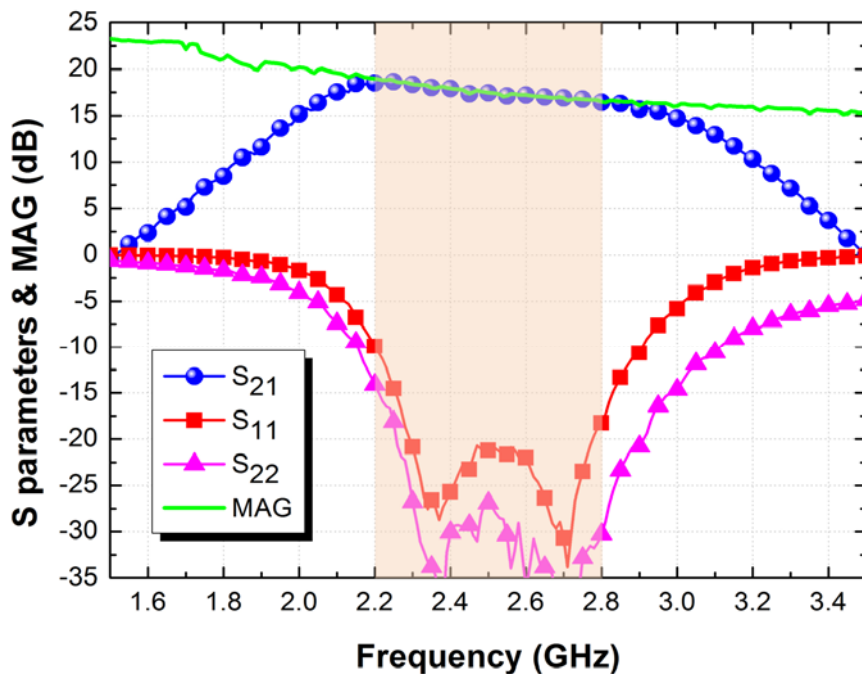


Fig. 2.27. The S-parameters of matched NE-42484, which is matched with global optima of $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$. It is obvious that the $|S_{21}|_{dB}$ is almost the same of MAG in 2.2-2.8 GHz (highlight) due to optimized matching.
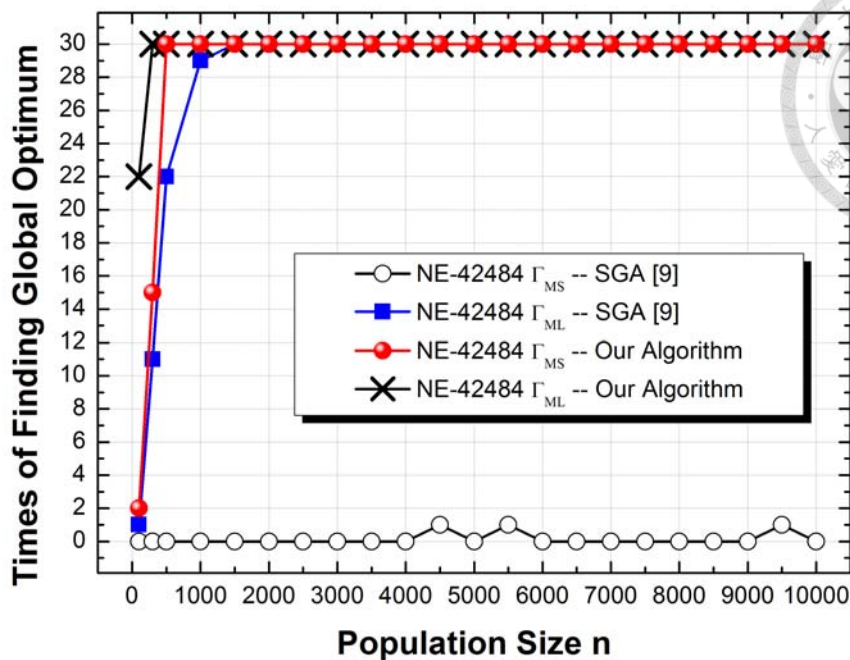
Fig. 2.28. The times of finding global optimum in matching $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ of NE-42484, and compare our work to the published algorithm [9]. The function evaluations of algorithms are $100n$. Since the $\Gamma_{MS}(f)$ is too difficult for SGA to solve in this case, [9] always takeover by other local optima; while our algorithm can find global optimum with 100% statistic probability when $n$ is larger than 500.
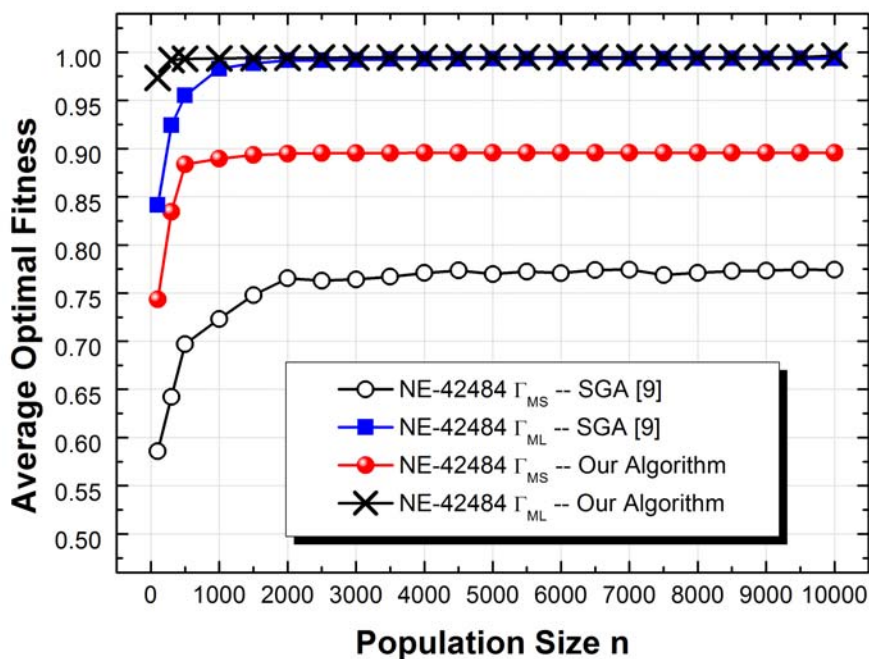


Fig. 2.29. Average optimal fitness of 30 independent runs for $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ with total function evaluations of each algorithm $100n$. The averages of our algorithm are always larger than that in [9] apparently.
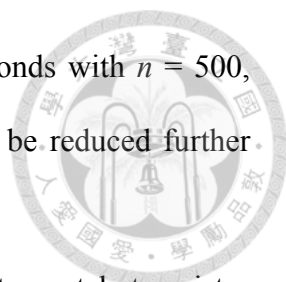
The program executing time of our algorithm is about 20 seconds with $n = 500$, function evaluations $100n$ and using a 2.13 GHz processor; it can be reduced further more by adopting parallel computation.

Fig. 2.30 illustrates second matching experiment, which is to match transistor ATF-58143, and Fig. 2.31 shows the frequency response of $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ of stabilized device and the optimal matching results. Fig. 2.32 shows the correspondent $S$-parameters of ATF-58143 with the optimal matching. The statistic results are shown in Figs. 2.33 and 2.34, which are consistent with the first experiment that our algorithm can find global optimum with a high statistic probability with small computation efforts.

According to the results of these two experiments, it can be observed that the proposed algorithm outperforms published work [9], since our algorithm can fast and accurately find global optimum with a few computation efforts.
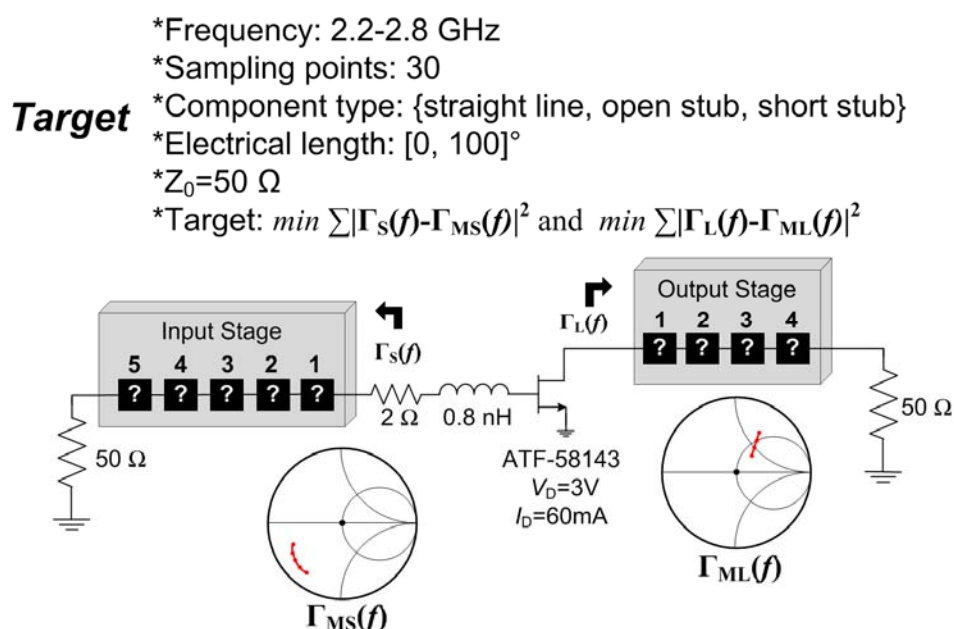


Fig. 2.30. The second example for performance measurement of our proposed algorithm. The synthesis target is to employ 4th-order output network and 5th-order input network to synthesize broadband simultaneous-conjugate-matching from 2.2 to 2.8 GHz. The data sheet and design kits of ATF-58143 can be obtained from the website [43].
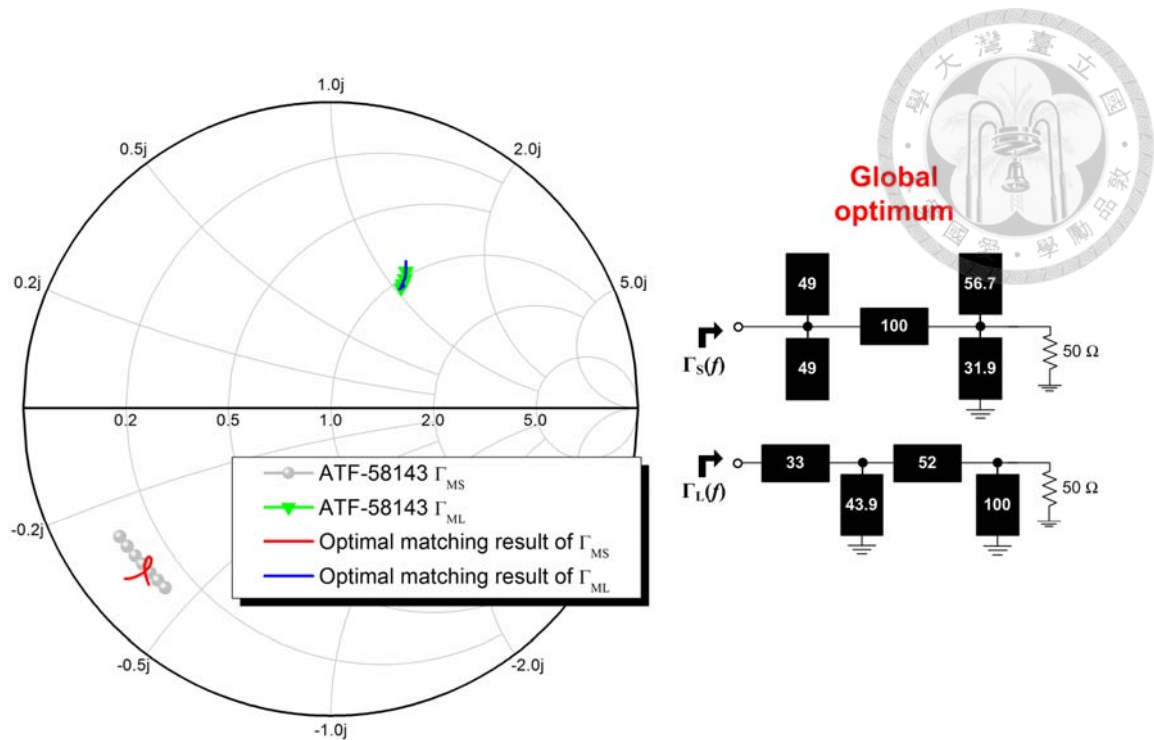
Fig. 2.31. The curves of $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ of stabilized ATF-58143 transistor, and the correspondent frequency response (solid line without symbol) and circuit topology of optimal matching network are also illustrated.
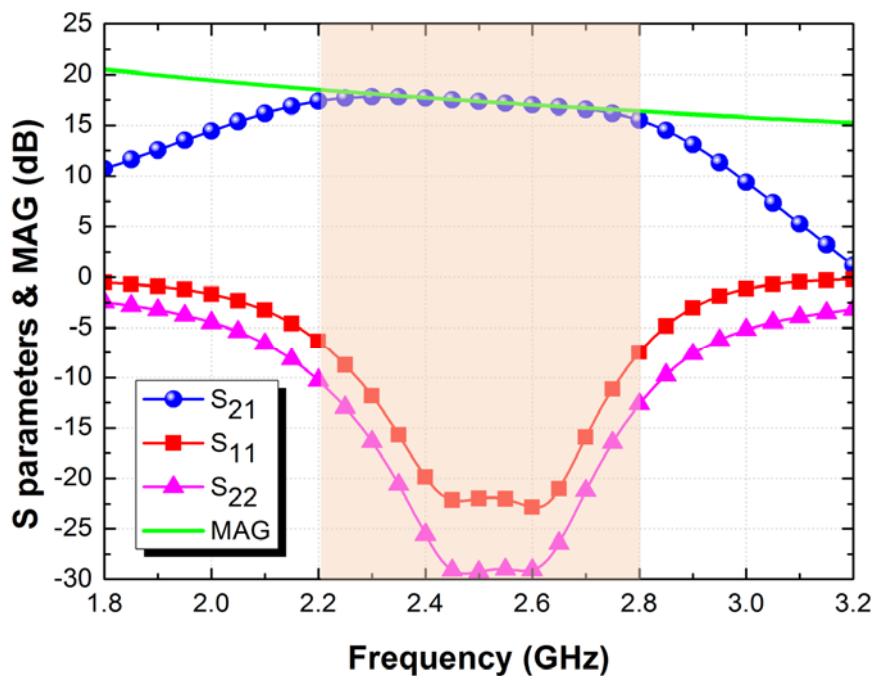


Fig. 2.32. The $S$-parameters of matched ATF-58143 which is matched with global optima of $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$. It is obvious that the $|S_{21}|_{dB}$ is almost the same of MAG in 2.2-2.8 GHz (highlight) due to optimized matching.
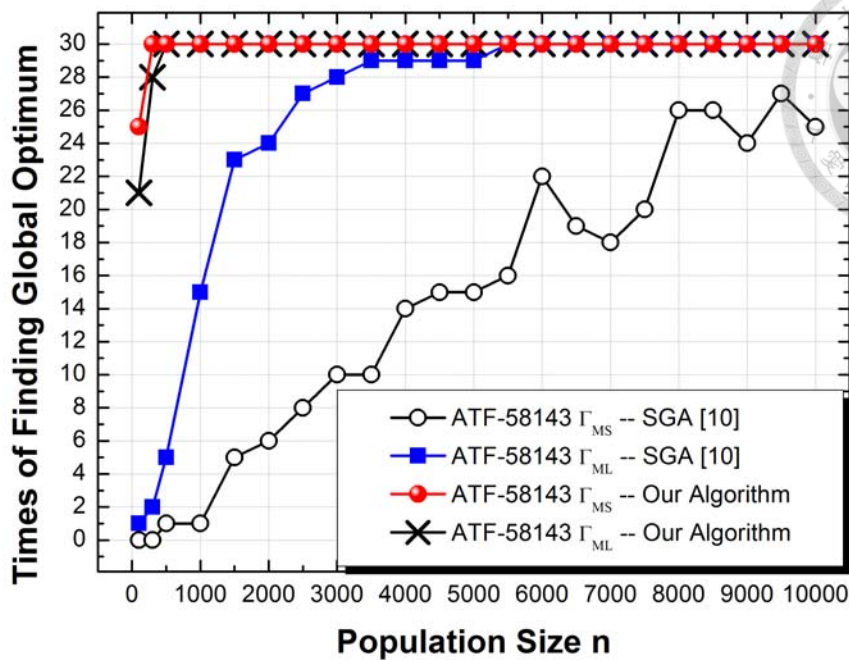
Fig. 2.33. The times of finding global optimum in matching $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ of ATF-58143, and compare our work to the published algorithm [9]. The function evaluations of algorithms are $100n$. Since the $\Gamma_{MS}(f)$ is too difficult for SGA to solve in this case, [9] easily takeover by other local optima; while our algorithm can find global optimum with 100% statistic probability when $n$ is larger than 500.
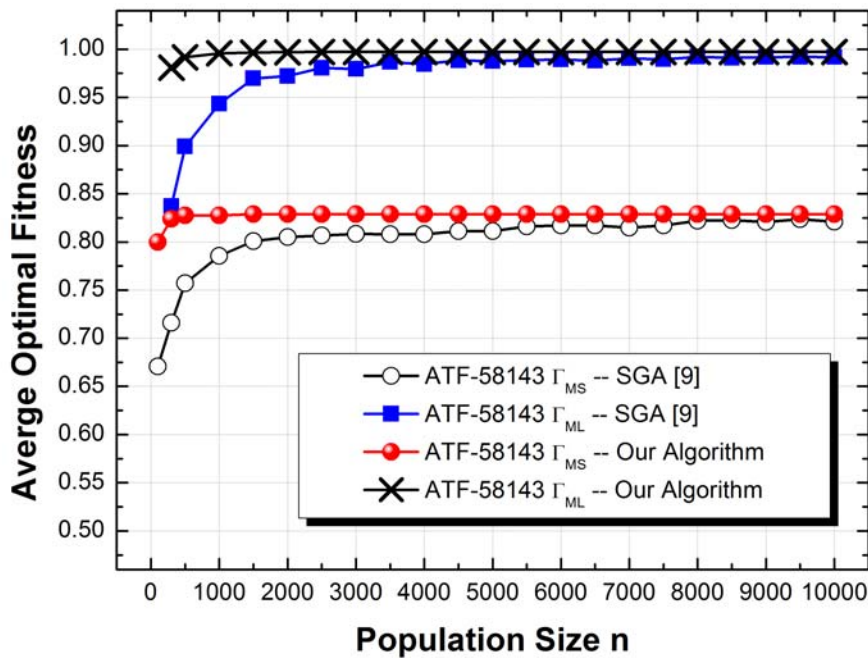


Fig. 2.34. Average optimal fitness of 30 independent runs for $\Gamma_{MS}(f)$ and $\Gamma_{ML}(f)$ with total function evaluations of each algorithm $100n$. The averages of our algorithm are always larger than that in [9] apparently.

Due to the appropriate niching method and efficient local search, the proposed algorithm can simultaneously find a lot of sub-optimal circuits, that is, it can find optimal electrical parameters with different circuit architectures for flexible choices. This makes our circuit synthesis algorithm more flexible and applicable for general applications. Fig. 2.35 illustrates the frequency response of sub-optimal circuits found by the proposed algorithm ($n = 1000$ and $100n$ evaluations) in problem synthesizing $\Gamma_{MS}(f)$ of NE-42484. The fitness of these solutions is higher than 95% that in correspondent sub-optima.
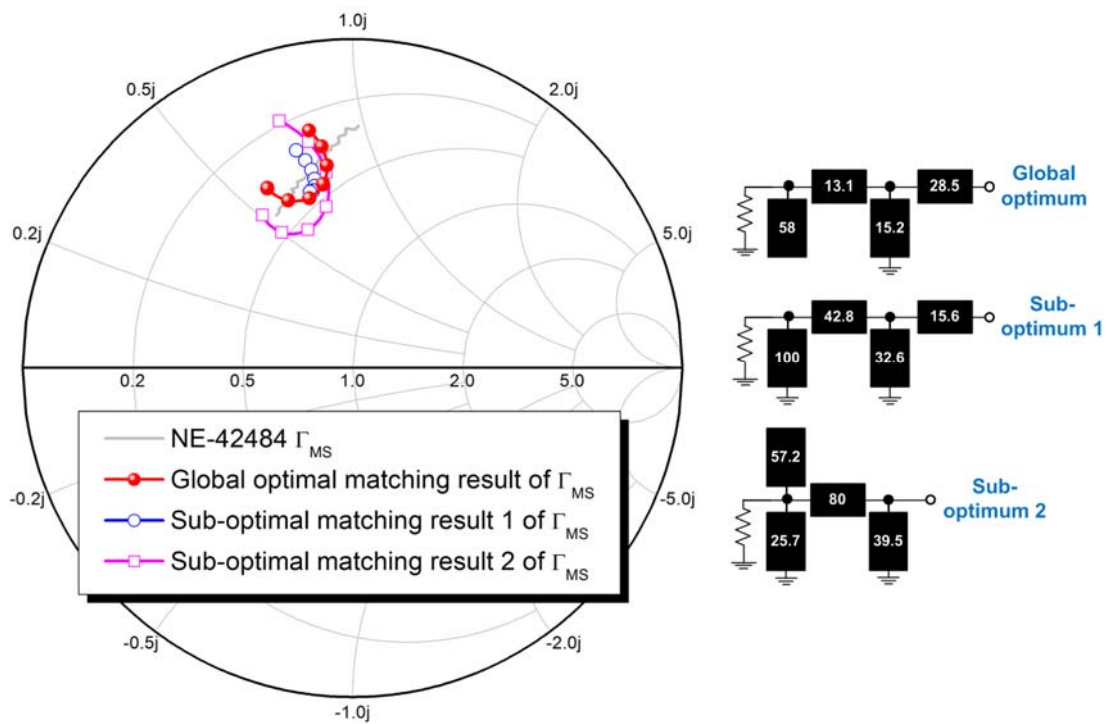


Fig. 2.35. The frequency responses and correspondent circuit topology of another two sub-optimal circuits for matching $\Gamma_{MS}(f)$ of NE-42484.

46

## 2.4.2 Experiments of Microwave Circuit Design II: Dual-Band Filter

If the algorithm can synthesize the network with arbitrary frequency response of *S*-parameters, it is practical to synthesize the matching network in circuit design. The filter synthesis is very suitable for the performance measurement.

For comparison, we repeat the dual-band filter synthesis reported in [9] with slightly different design target. Fig. 2.36 illustrates the synthesis specifications of a dual-band filter with 25 components in matching network, which conform the IEEE 802.11 a/b/g WLAN systems. The total square error is defined as

$$\text{Square error} = \sum_i E_{in-band} + \sum_j E_{out-band} \tag{2.29}$$

$$E_{in-band} = w_{S11\_in}E(|S_{11}|_{dB}) + w_{S21\_in}E(|S_{21}|_{dB}) \tag{2.30}$$

$$E_{out-band} = w_{S21\_out}E(|S_{21}|_{dB}) \tag{2.31}$$

where the $E(|S|)$ denotes the function to calculate the square error from the *S*-parameter; while the $w_{S11\_in}$ and $w_{S21\_in}$ denote the weight of square error of pass-band $S_{21}$ and $S_{11}$ respectively, and the $w_{S21\_out}$ is for rejection-band.

The synthesized components are transmission lines, open stubs, short stubs, where the electrical length of transmission lines are restricted between 0° and 120° at 2 GHz and characteristic impedance are from 40 to 110 Ω. For simplicity, there is only one constraint that no node can join ≥ 3 shunt-connected components. Table 2.5 shows the parameter settings of our algorithm, while the parameters of algorithm in [9] remain the same because they have been optimized for this filter design.

Fig. 2.37 shows the statistic results, which is the average of optimal fitness in 30 independent runs, and the *x*-axis represents function evaluations with population size *n* = 1000. It is obvious that the proposed algorithm can find the better solution under the same computation effort, and this result is consistent with previous experiments that our

algorithm is robust. Fig. 2.38 shows *S*-parameters of a synthesized dual-band filter (fitness = 0.035) and Table 2.6 illustrates the correspondent topology with ideal passive components, the computation time is about 4 minutes with $200n$ function evaluations.
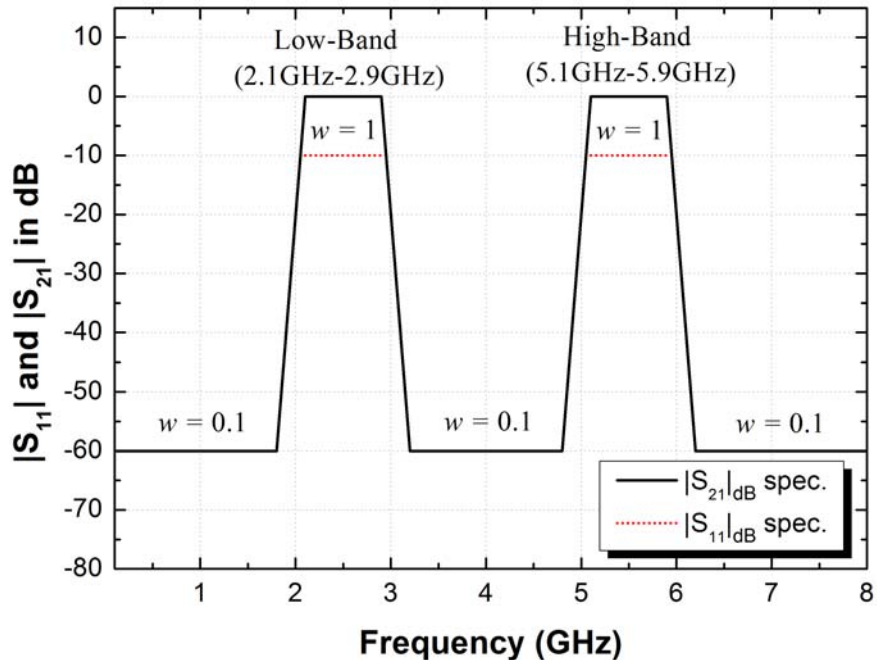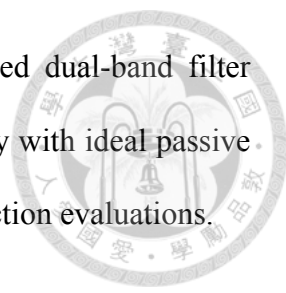


(a)
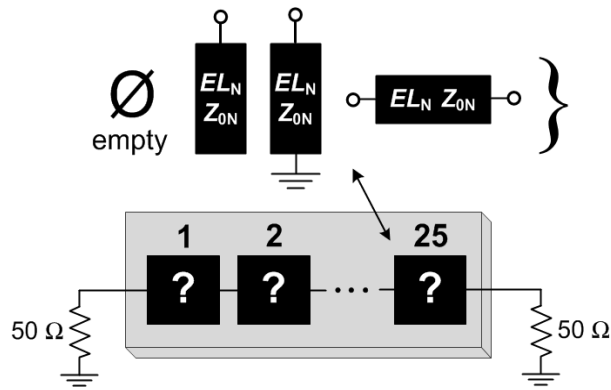


(b)

Fig. 2.36. (a) Specification of the dual-band filter with specified weights between in-band and out-band. (b) Using 25 distributed components for circuit synthesis.

Table 2.5. Parameter settings of our algorithm in filter synthesis.

| Parameter | Value |
|---|---|
| Population size ($n$) | 1000 |
| Selection pressure ($s$) | 30 |
| Probability of crossover ($p_c$) | 0.5 |
| Split rate of SoD ($\gamma$) | 0.5 |
| Restricted window size of RTR ($w$) | 100 |
| Initial step size of ES ($\sigma$) | 5.0 |
| Updating generation of ES ($d$) | 5 |
| Sieving number for ES ($m$) | 30 |
| Reference frequency ($f_0$) | 2 GHz |
| Sampling step (GHz) | 0.1 |
| Electrical length (°) | [0, 120] |
| Characteristic Impedance ($\Omega$) | [40, 110] |
| ($w_{S11\_in}$, $w_{S21\_in}$, $w_{S21\_out}$) | (1, 1.5, 0.1) |
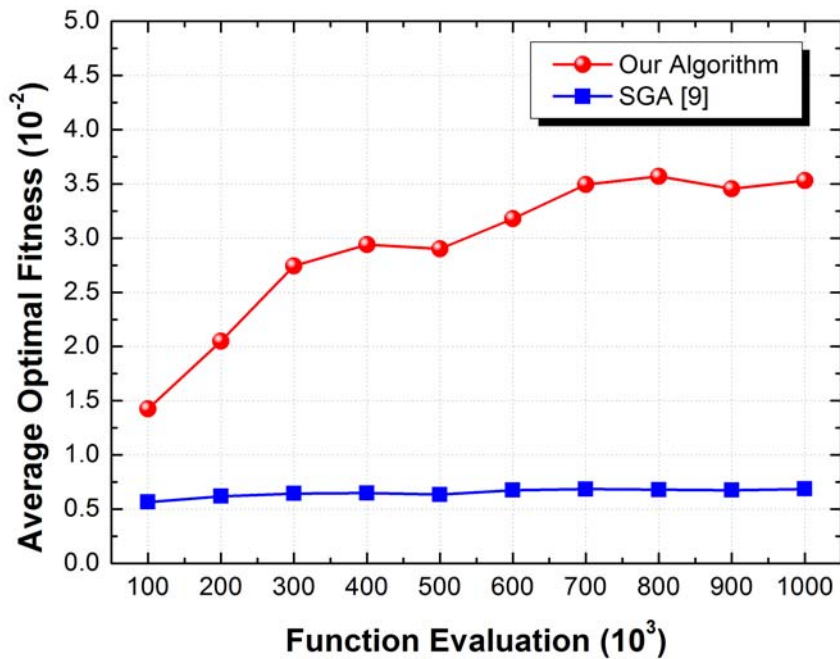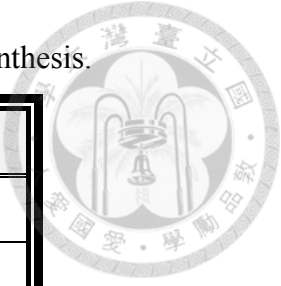


Fig. 2.37. Average optimal fitness of 30 independent runs in dual-band filter synthesis. The $x$-axis is function evaluations with $n = 1000$.
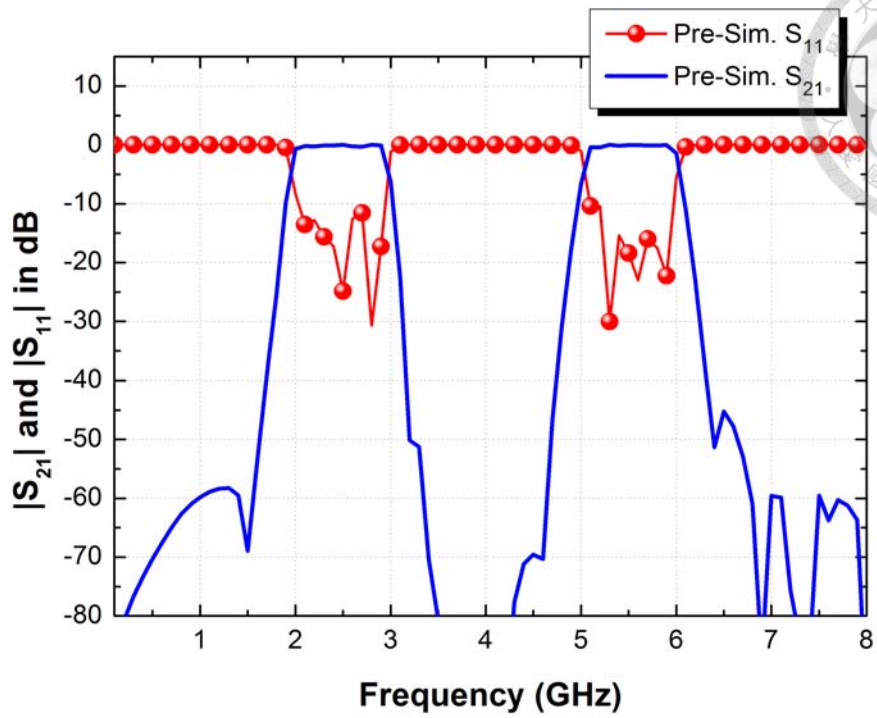
Fig. 2.38. *S*-parameters of the synthesized dual-band filter (fitness = 0.035) with ideal passive components for statistic comparison

Table 2.6. Components from number 1 to 25 in the optimized dual-band filter network with reference frequency 2 GHz.

| Num. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Type | $\phi$ | ▮ | ▬ | $\phi$ | ▬ | $\phi$ | ▮ | ▬ |
| **EL** (°) **Z₀** (Ω) | | 118 82 | 82 58 | | 5 70 | | 28 86 | 100 87 |

| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|
| ▬ | ▮ | ▮ | ▬ | ▮ | ▮ | ▬ | ▮ | ▬ |
| 7 81 | 90 55 | 29 62 | 111 96 | 43 94 | 43 40 | 36 105 | 97 43 | 33 103 |

| 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | |
|---|---|---|---|---|---|---|---|---|
| ▮ | ▬ | ▮ | ▮ | ▬ | $\phi$ | $\phi$ | ▮ | |
| 104 40 | 21 76 | 113 62 | 27 71 | 16 57 | | | 99 61 | |

50

According to these two experiments, it is concluded that our algorithm can fast and accurately synthesize desired matching network and outperform the previously published algorithm [9]. The main reason to cause the difference is that we consider several characteristics of solution space and specifically design for circuit synthesis, rather than only use the simple genetic algorithm operators.

### 2.4.3 Discussions of Performance with Different Procedures

In order to illustrate the effect of each procedure in our proposed algorithm, we perform several experiments with different operation combinations and compare each statistic results. The matching problem is to match $\Gamma_{MS}(f)$ of NE-42484 as shown in Fig. 2.25, and the parameters setting is the same in Table 2.4.

The first comparison is between two different versions of building-block-wise crossover, which are illustrated in Fig. 2.18 (using roulette-wheel to sample circuit patterns) and Fig. 2.19 (exchanging sub-circuit represented by MPM). Fig. 2.39 shows the statistic results of employing these two operators, while the other procedures in Fig. 2.9 remain the same. The $x$-axis represents population size $n$ from 100 to 10000 ($n$ = 100, 300, 500, 1000, ···, 9500, 10000) with a total of $100n$ function evaluations. Using roulette-wheel sampling method to generate new solutions may destroy good real-valued (non-integer) parameters due to sampling noise, thus the average performance is lower than the proposed modified version of building-block-wise crossover shown in Fig. 2.19

The second comparison is between four different versions of implementation of ES, which are performing ($\mu+\lambda$) ES on top 10% solutions with and without perturbation of circuit topology and applying the proposed (1+1) ES (as shown in Fig. 2.20) on top $m$ different-topology circuit independent with and without perturbation of circuit topology.
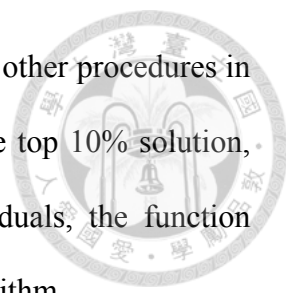
Fig. 2.40 shows the statistic results of these two operators, while the other procedures in Fig. 2.9 remain the same. The $(\mu+\lambda)$ ES runs 10 generations on the top 10% solution, while the (1+1) ES runs $n/m$ generations on each sieved individuals, the function evaluation of both procedures is $n$ per iteration of our proposed algorithm.

Since the standard deviation of Gaussian random variables of $(\mu+\lambda)$ ES is calculated according to a group of circuits, the local optima will affect the accuracy of standard deviation and easily cause immature convergence, thus the average performance of $(\mu+\lambda)$ ES is lower than the proposed (1+1) ES in this case. Fig. 2.40 also shows the effect of perturbing the circuit topology; the one without changing topology is better since it can avoid the solution with circuit topology of global optimum changing to other sub-optimal topology before convergence.

The third comparison is the algorithm architecture. Fig. 2.41 shows the statistic results of naïve rECGA, combining rECGA and (1+1) ES with and without RTR niching method (as shown in Fig. 2.21). The naïve rECGA cannot be used to solve this problem like [9] due to the no-free-lunch theorem, that is, the operator is too simple to solve circuit synthesis. Although ES procedure can help in exploiting solution candidates, it is beneficial to employ niching method to enhance the probability of finding global optimum because of the multimodal characteristic of solution space of circuit synthesis.

Based on these comparisons, it can be concluded that the proposed algorithm is designed particularly for circuit synthesis problems, and each procedure has the specific function that is irreplaceable. This is the main reason that the proposed algorithm outperforms previously published work [9], which only employs SGA.
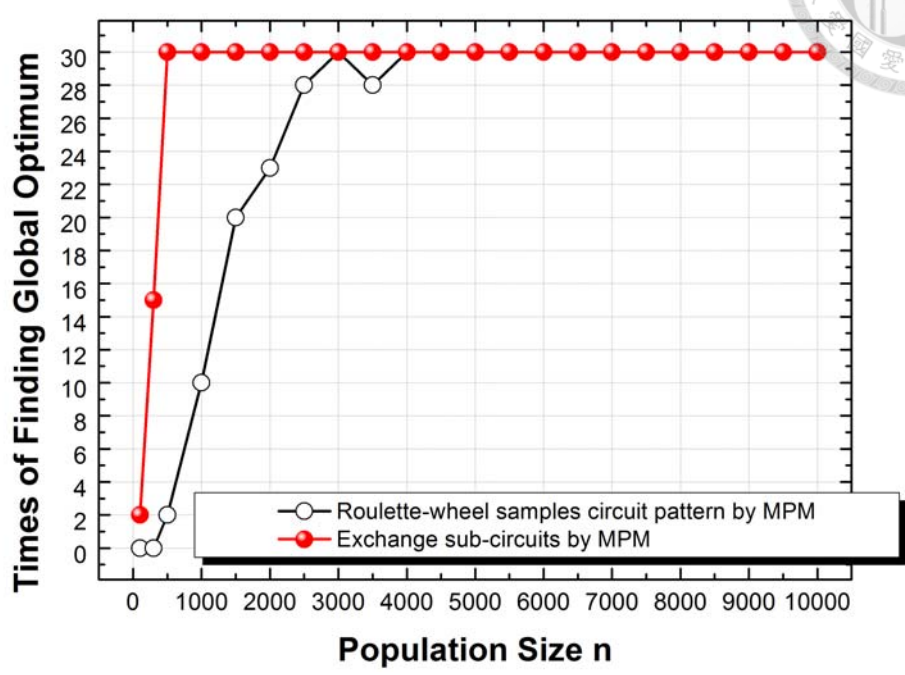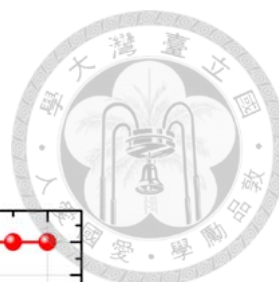
(a)
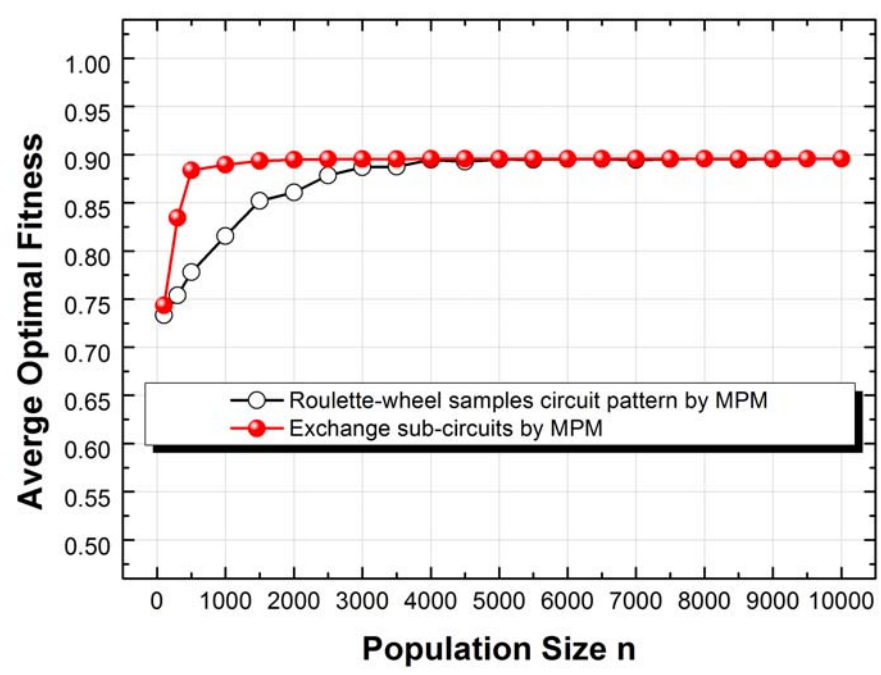


(b)

Fig. 2.39. (a) shows respect times of finding global optimum with two different version of building-block-wise crossover. (b) shows the average fitness of optimal solution in 30 independent runs.
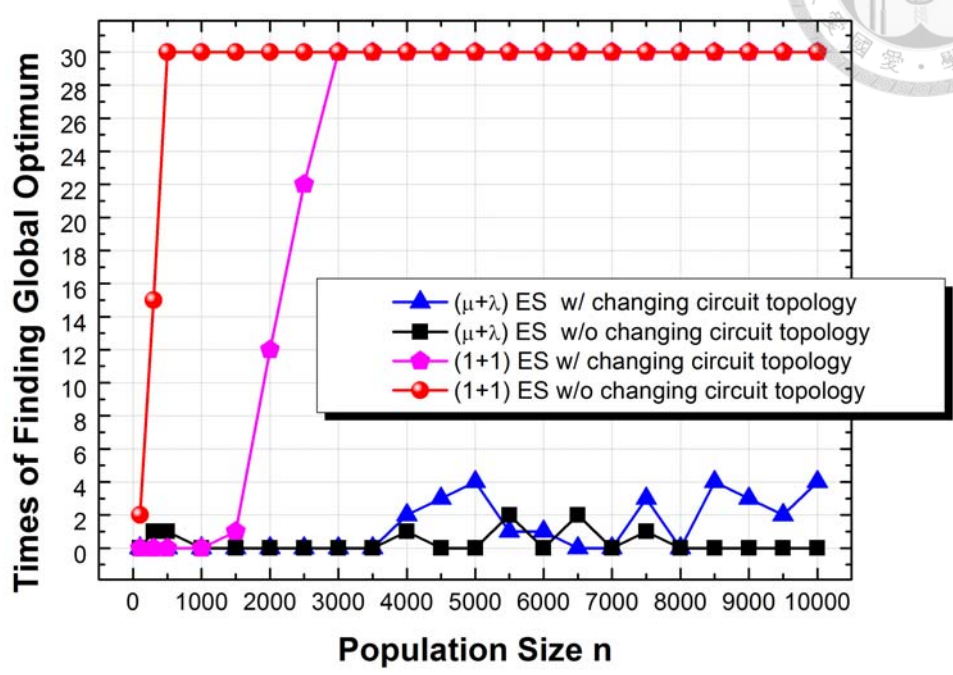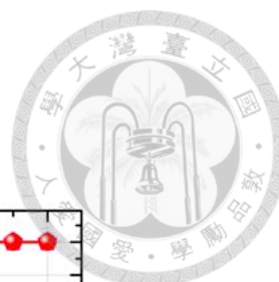
(a)
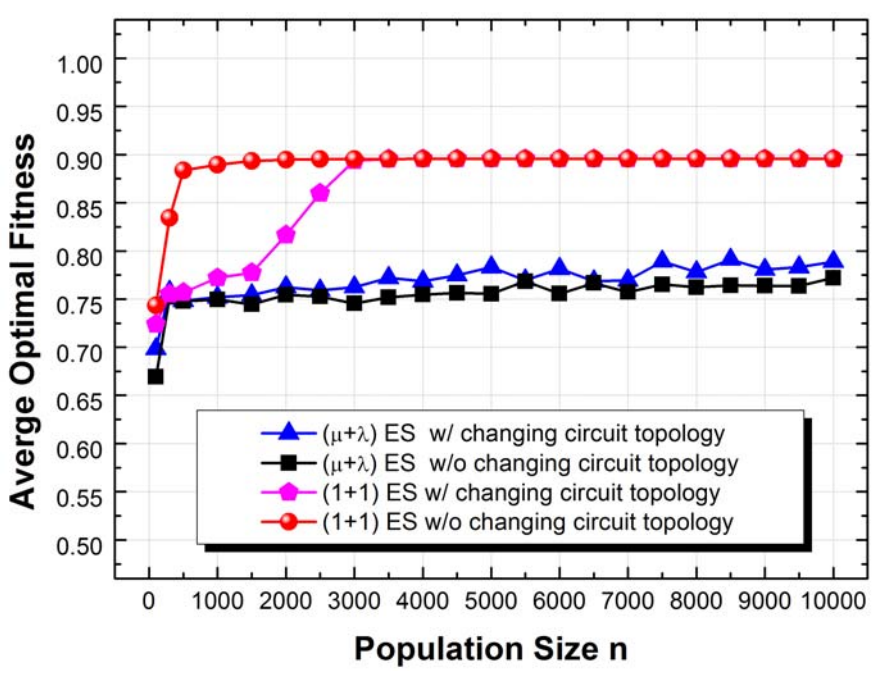


(b)

Fig. 2.40. (a) shows respect times of finding global optimum of different implementation of ES. (b) shows the average fitness of optimal solution in 30 independent runs.
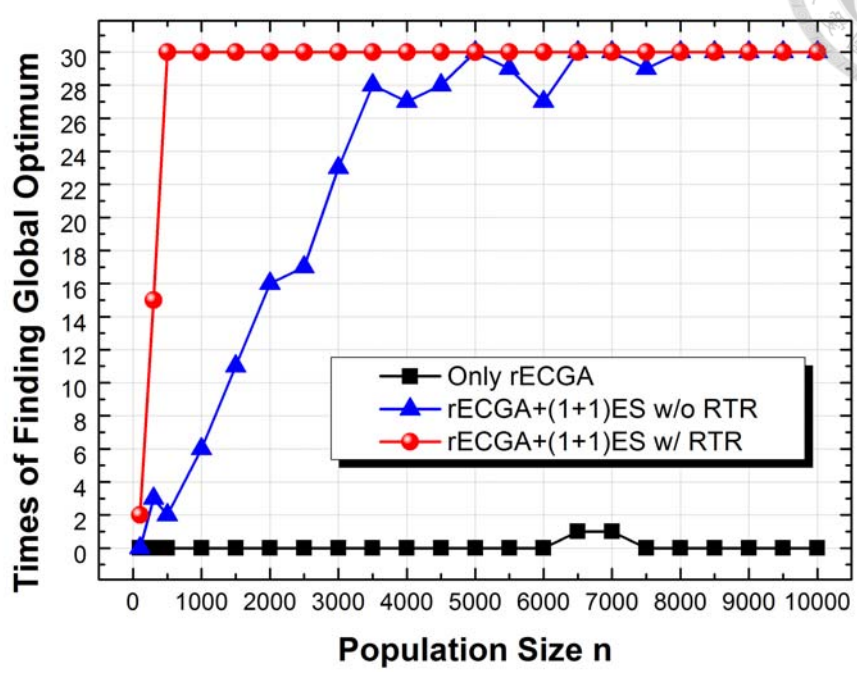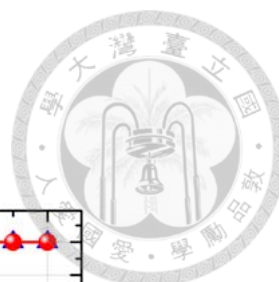
(a)
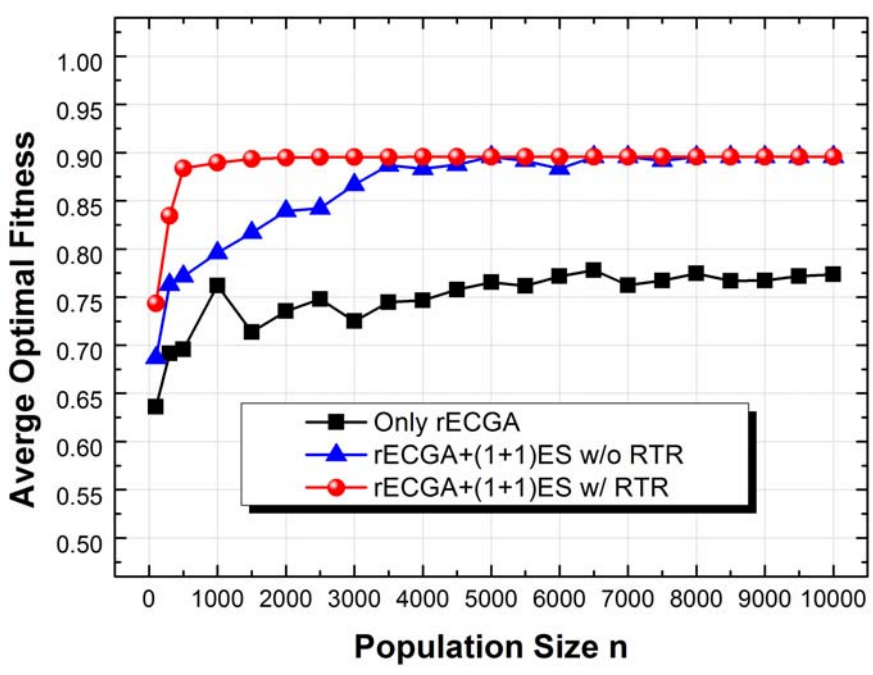


(b)

Fig. 2.41. (a) shows times of finding global optimum with different algorithm architecture. (b) shows the average fitness of optimal solution in 30 independent runs.

# Chapter 3　Design of Microwave and Millimeter Wave Broadband Amplifiers via Circuit Synthesis Algorithm

Two broadband amplifiers have been designed and implemented to demonstrate the proposed circuit-synthesis algorithm. They are a 110-180 GHz amplifier in CMOS 65-nm process and a Ka-band high efficiency power amplifier in 0.1-μm pHEMT process respectively. The small signal gain of 110-180 GHz amplifier is the first circuit covering full D-band in CMOS process; we also develop an optimization method to determine the minimal number of cascade stages and propose an impedance transformation technique to achieve desirable gain performance and wide bandwidth. The Ka-band medium power amplifier performs broadband output power and power-add efficiency (PAE) due to the synthesis matching networks.

## 3.1　A 110-180 GHz Broadband Amplifier in 65-nm CMOS Process

To increase the gain performance beyond 100 GHz in CMOS process, it is inevitable to use several gain stages; the gain limitation of the transistor ($f_{max}$) and losses of on-chip passive component are the bottleneck in circuit design. Since our design target is to simultaneously achieve the full D-band bandwidth and desirable gain performance, it is suitable to develop particular design method to ensure the optimal usage of each cascade stage, and use proposed algorithm to synthesize the optimal broadband matching.

### 3.1.1 Optimization Method for Inter-stage Matching

In order to ensure optimal usage of each cascade stage, an optimization method is developed. Each cascade stage can be view as a two-port network like a black-box, which can provide the maximum available gain (MAG) [38] from its two-port S-parameters, as shown in Fig. 3.1. Using search algorithm to synthesize the inter-stage matching is beneficial for raising and flatting the frequency response of MAG. In our case, we apply the proposed algorithm for this inter-stage matching synthesis. If the optimized MAG cannot achieve the specification under current number of applied stage, it is necessary to cascade another gain-cell until gain performance excesses the design target. Fig. 3.2 illustrates this bottom-up optimization approach, which ensures the optimal usage of each stage and leads the minimal number of cascade stage. Without the redundant cascade stages, the dc-power consumption and chip size can be reduced.
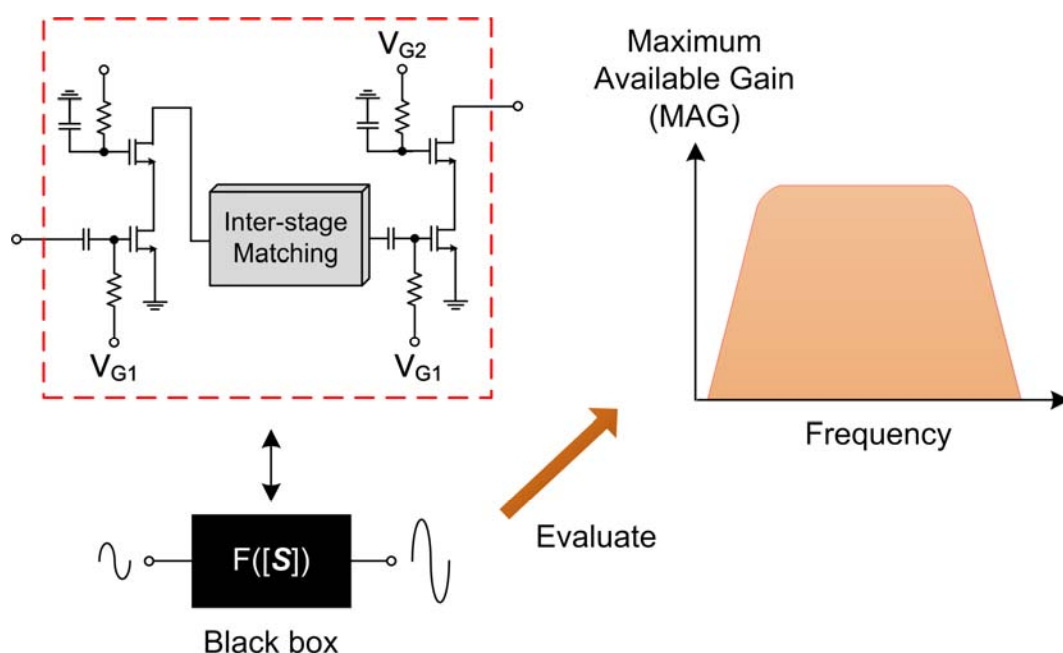


Fig. 3.1. Black-box view of a cascade stage. The maximum available gain (MAG) can be obtained by evaluating the S-parameters provided by the two-port network. Using optimization algorithm to synthesize the inter-stage matching is beneficial for raising and flatting the frequency response of MAG.
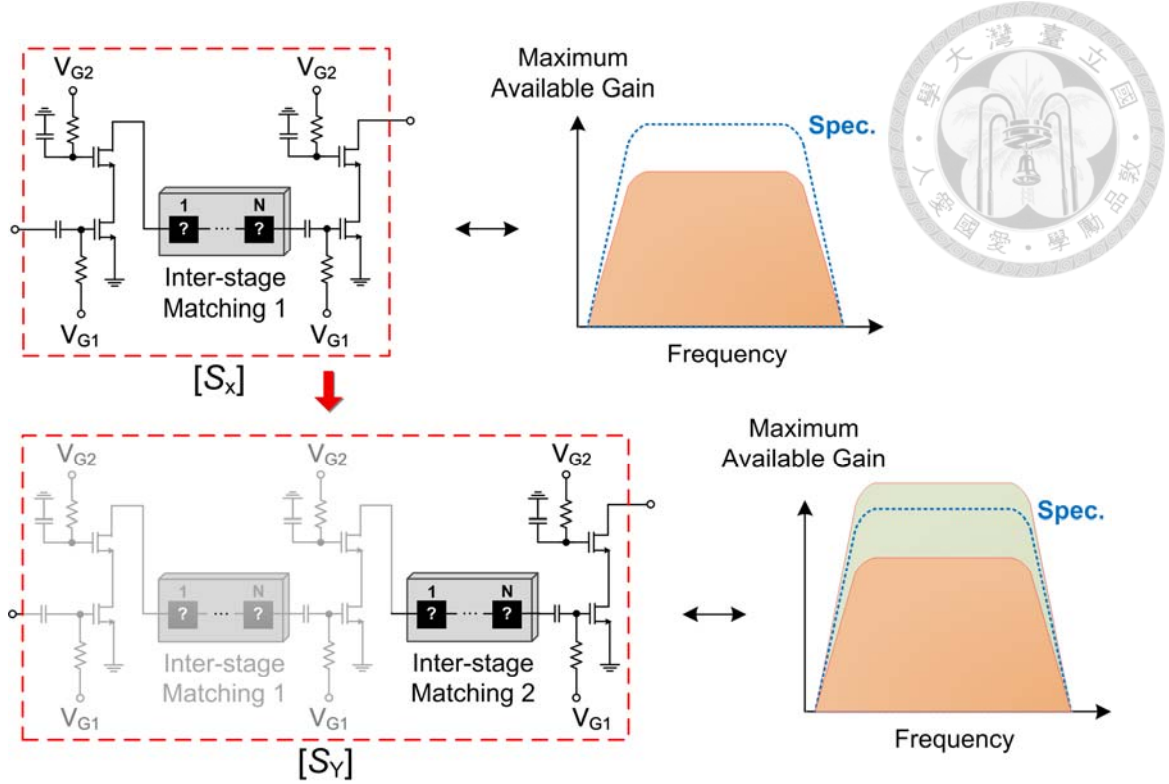
Fig. 3.2. A bottom-up approach to cascade gain stage. If the MAG of the cascade stage cannot achieve the design target after synthesis of inter-stage matching, it is necessary to cascade another gain cell for raising the performance until the MAG excesses the target response.

In order to guarantee optimized MAG is desirable (broadband, high gain and flat frequency response), several rules are adopted to guide the search algorithms to find acceptable solutions. The response of optimized circuit is totally dependent on the rules, which are set according to the prior knowledge of engineers. Therefore, the following square errors between target frequency response and solution candidates are adopted, which can control the frequency response and make optimized result desirable

$$\Delta_{all} = \Delta_{in-band} + \Delta_{out-band} \qquad (3.1)$$

$$\Delta_{in-band} = \sum_{i=1}^{n} \left\{ E_{gain}([S(f_i)]) + E_{flat}([S(f_i)]) + E_{stable}([S(f_i)]) \right\} \qquad (3.2)$$

$$E_{gain}([S(f_i)]) = \left| \text{Gain\_spec}_{in-band} - \text{MAG}([S(f_i)]) \right|^2, \\ \text{if MAG}([S(f_i)]) < \text{Gain\_spec}_{in-band} \qquad (3.3)$$

58

$$E_{flat}([S(f_i)]) = \left|\text{MAG}([S(f_i)]) - \text{MAG}([S(f_{i-1})])\right|^2,$$
$$\text{if } \left|\text{MAG}([S(f_i)]) - \text{MAG}([S(f_{i-1})])\right| > \alpha_{ripple} \tag{3.4}$$

$$E_{stable}([S(f_i)]) = penalty \times \left|\text{Gain\_spec}_{in-band} - \text{MAG}([S(f_i)])\right|^2,$$
$$\text{if K-factor}([S(f_i)]) < 1 \tag{3.5}$$

where $\Delta_{all}$ denotes the total square error, $\Delta_{in\text{-}band}$ and $\Delta_{out\text{-}band}$ denotes the in-band and out-band part of $\Delta_{all}$ respectively, $n$ denotes the number of sampling points in out-band frequency, MAG is function to evaluate maximum available gain, $\alpha_{ripple}$ denotes the maximum allowable gain ripple, *penalty* is used to control circuit stability and $w_{out\text{-}band}$ denotes the weight between in-band and out-band. The $E_{gain}$ ensures the in-band high gain response, while the $E_{flat}$ reduces gain ripple by comparing the adjacent points and the $E_{stable}$ guarantees the unconditional stability of the two-port network by monitoring the k-factor [38]. For a solution candidate, when its maximum available gain excesses the design, $E_{gain}$ stops to increase and the $E_{flat}$ will dominate the $\Delta_{in\text{-}band}$ and inform the search algorithm to exploit solutions with flat frequency response. Therefore, the synthesized results will have both high and flat frequency response of maximum available gain.

Equations (3.2)-(3.5) is search rules for in-band response, but it is suitable to consider the out-band frequency response to ensure reasonable response, such as avoiding the occurrence of return gain. The out-band rules can be set as follow

$$\Delta_{out-band} = w_{out-band} \times \sum_{i=1}^{m} \left\{ E_{rejection}([S(f_i)]) + E_{stable}([S(f_i)]) \right\} \tag{3.6}$$

$$E_{rejection}([S(f_i)]) = \left|\text{Gain\_spec}_{out-band} - \text{MAG}([S(f_i)])\right|^2,$$
$$\text{if MAG}([S(f_i)]) > \text{Gain\_spec}_{out-band} \tag{3.7}$$

$$E_{stable}([S(f_i)]) = penalty \times \left|\text{Gain\_spec}_{out-band} - \text{MAG}([S(f_i)])\right|^2,$$
$$\text{if K-factor}([S(f_i)]) < 1 \tag{3.8}$$

where $m$ denotes the number of sampling points in out-band frequency, $w_{out\text{-}band}$ is used to control the ratio of square error between in-band and out-band, the $E_{\text{rejection}}$ can control the out-band gain rejection. The value of $w_{out\text{-}band}$ should set small, otherwise the optimized result of in-band circuit response will be unreasonable. Since the main consideration of out-band is high rejection in general, therefore the $E_{\text{flat}}$ can be ignore in out-band consideration.

Fig. 3.3 shows an example of target response for search algorithm to optimize. It is beneficial to keep few space of frequency between in-band and out-band interval, which controls the slope of response and guide the algorithm to search efficiently. The $w_{out\text{-}band}$, $\alpha_{\text{ripple}}$ and *penalty* should be selected carefully to ensure desirable response.
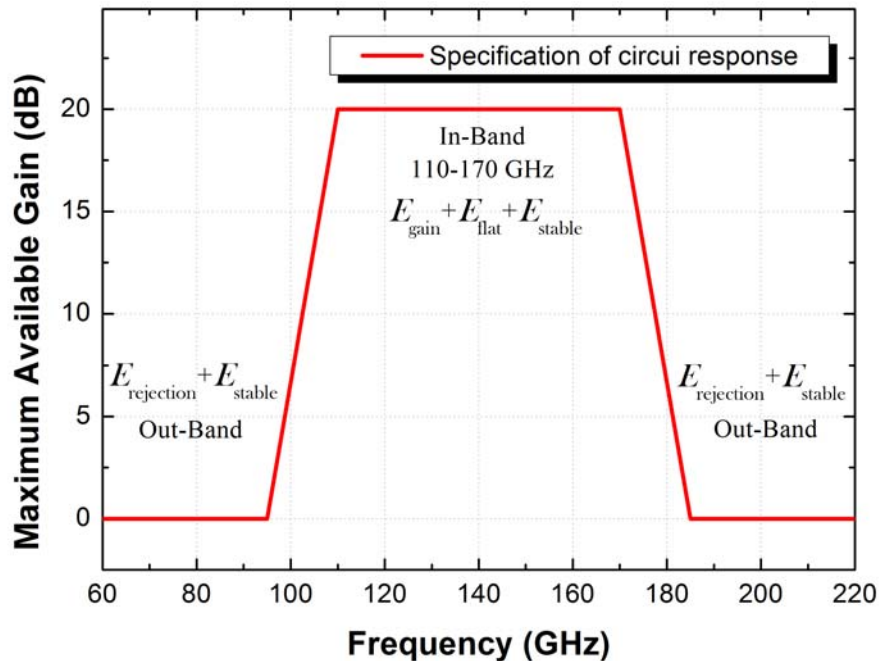


Fig. 3.3. Example of specification for search algorithm to optimize. The $E_{\text{gain}}$ ensures the in-band high gain response, while the $E_{\text{flat}}$ reduces gain ripple by comparing the adjacent points, the $E_{\text{stable}}$ guarantees the unconditional stability of the two-port network by monitoring the k-factor and the $E_{\text{rejection}}$ can control the out-band gain rejection.

## 3.1.2 Selecting Gain-Cell

This proposed MMW amplifier is fabricated in TSMC 65-nm GP 1P9M CMOS process. The target is to simultaneously achieve the full D-band bandwidth and desired small signal gain.

The maximum gain (MSG/MAG) of small-size common source (CS) and cascode cell are shown in Fig. 3.4. Both transistors are biased near peak of the maximum trans-conductance ($g_m$) under standard supply voltage. Although the dc consumption of cascode cell is larger than that of CS, the gain is about twice of CS around 170 GHz. For the sake of obtaining the high gain performance under the minimum amount of cascade stage, we select cascode topology as the primary gain cell instead of common-source topology, and the chip size can be reduced.
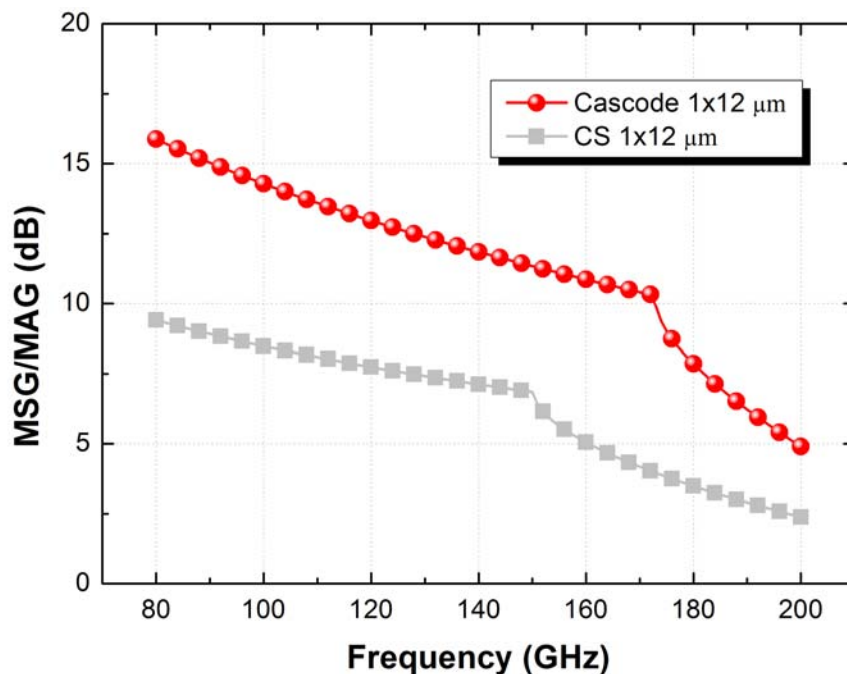


Fig. 3.4. Maximum gain (MSG/MAG) of small-size CS and cascode cell. Both transistors are bias near the $g_m$ peak under standard supply voltage. We employ cascode topology as the primary gain cell instead of common-source topology for reducing chip size.

Fig. 3.5 is a contour plot that shows the gate-finger number and width of transistor of cascode pair versus the turning point of MSG/MAG, where the device sizes of the cascode pair are the same and the bias condition of each transistor is near peak of the $g_m$ peak with $V_{DD}$ = 2.4 V. Although the smallest size has the highest frequency of turning point, the saturated power is too small to be applicable; therefore, the gate width and finger number we select are 1 μm and 12 respectively, whose turning-point frequency is slightly higher than 170 GHz and is higher than other combinations with the same total gate width.
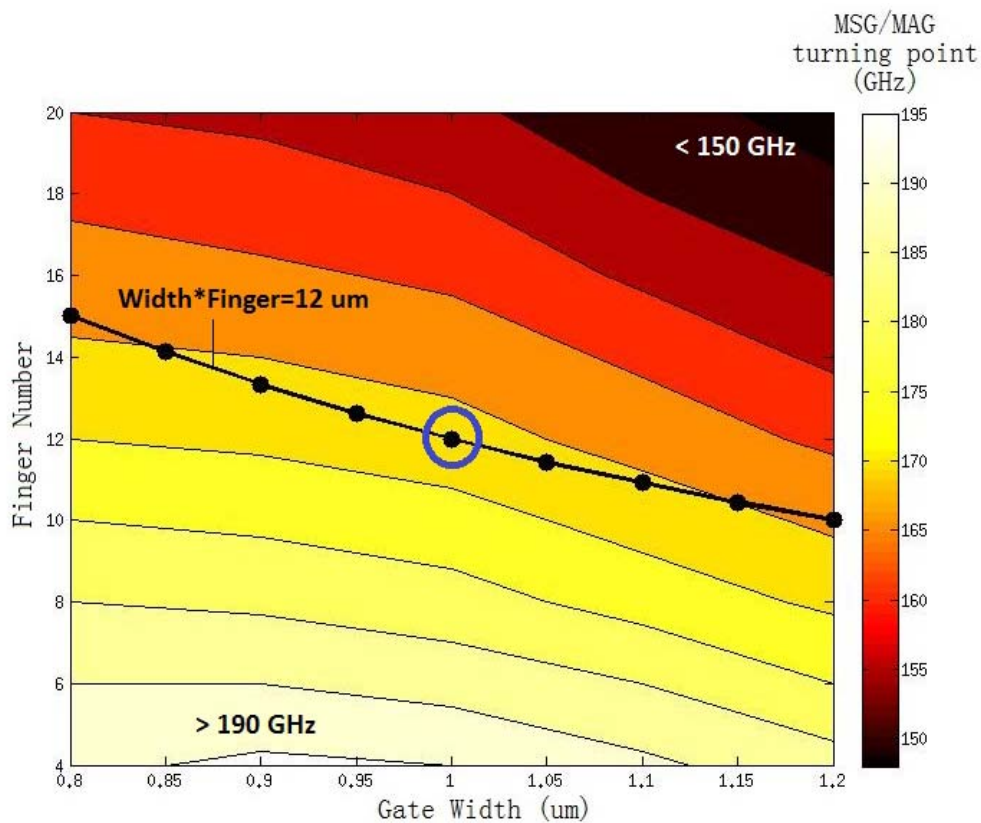


Fig. 3.5. Sweep finger number and gate width of transistors of cascode pair. The black curve representes $xy$ = 12 μm, which $x$ denotes gate width and $y$ denotes finger number.

### 3.1.3 Synthesis of Broadband D-band Amplifier

First, we use the proposed bottom-up approach (as shown in Fig. 3.2) to synthesize each inter-stage matching until the MAG of cascade stage achieves the target response, followed by synthesizing the input and output matching to obtain the available gain.

The in-band and out-band parameter settings to guide search algorithm (Eqns. (3.2) -(3.8)) are shown in Table 3.1 and Table 3.2, respectively. The specifications of in-band MAG are lower than the upper bound (summation of MAG of each gain-cell) because the upper bound is too difficult to obtain. If the Gain_spec$_{in\text{-}band}$ is set as high as the upper bound, the $E_{gain}$ will be too high and dominate $\Delta_{all}$, which makes the $E_{flat}$, $E_{stable}$ and $\Delta_{out\text{-}band}$ useless (i.e. the circuit will be unstable or the gain ripple will be large) because they are much smaller than $E_{gain}$. Although there is no general rule for setting the Gain_spec$_{in\text{-}band}$, the performance of previously published D-band CMOS amplifier can be referred for setting the initial value of Gain_spec$_{in\text{-}band}$. If the optimized frequency response easily achieves the target, the Gain_spec$_{in\text{-}band}$ can slightly increase toward the upper bound; else if the optimized result has unacceptable gain ripple, the Gain_spec$_{in\text{-}band}$ should decrease or stop to increase to ensure the flat frequency response.

Fig. 3.6 shows specification of frequency response with different cascade stage. The ideal passive components are employed in optimization first to observe how large the gain response the circuit can perform, and the parameters settings of our algorithm are the same in Table 2.4. Although the passive components are lossless, the k-factor of the cascade black-box may be different from that of single gain-cell due to including other transistors. The k-factor of the synthesized result using lossless components is indeed greater than 1 (unconditionally stable). The higher k-factor in post-simulation can be obtained due to the losses of passive components, which makes the circuit more

stable. Figs. 3.7 and 3.8 show the synthesis results with ideal passive components and network order $N = 3, 4$ respectively. Figs. 3.9 and 3.10 illustrate the correspondent synthesized circuit topology. The synthesized frequency response of 4-stage cascode is desirable with high gain and flat frequency response in pass-band, and the out-band rejection is high enough to ensure the circuit stabilities. Although the synthesized results with $4^{th}$-order inter-stage matching are slightly closer to the specifications than results with $3^{rd}$-order matching, the complicated circuit topology will increase the chip size. Therefore, we apply the $3^{rd}$-order networks in our circuit. The T-network of synthesis result in $3^{rd}$-order networks can transform the impedance between cascode stages as illustrated in Fig. 3.11. The right-side transmission line moves the impedance to the high admittance circle, followed by a short stub to pull the impedance toward the location near the edge of Smith chart, and finally the left-side transmission line transforms the impedance to the conjugate of output impedance of cascode stage.
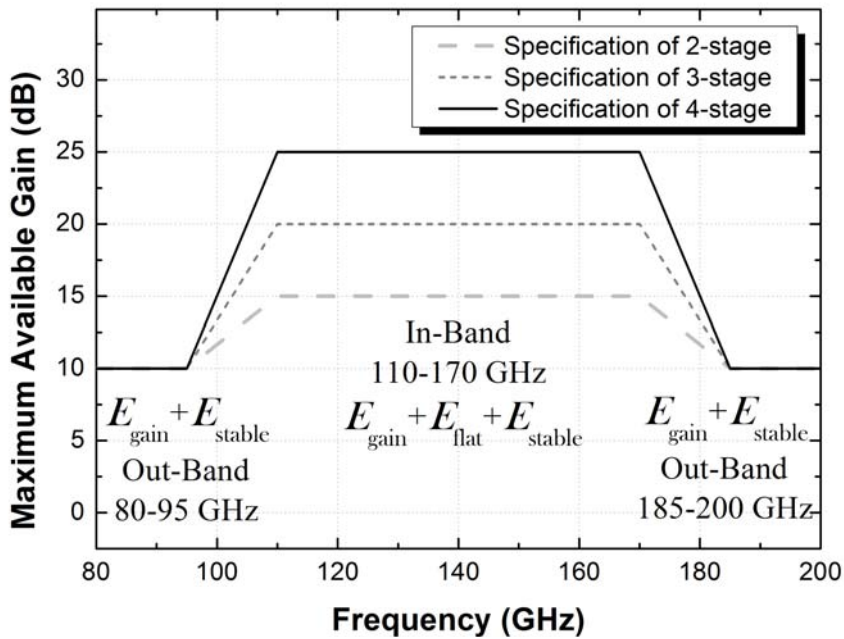


Fig. 3.6. Specification for search algorithm to optimize with different cascade stage in ideal simulation.
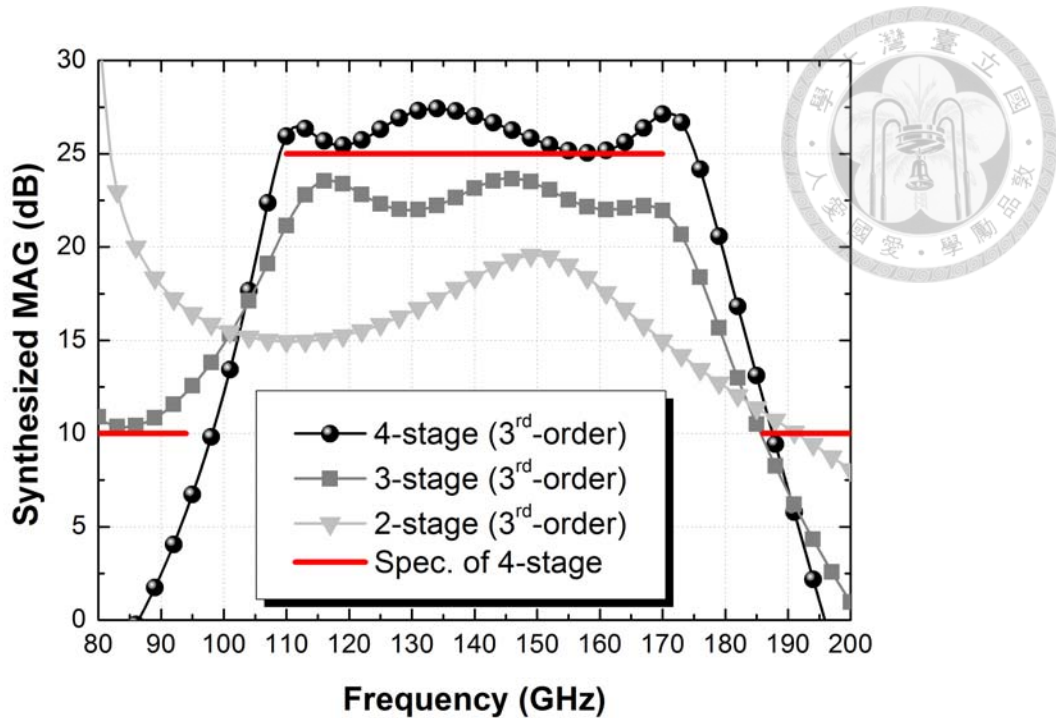
Fig. 3.7. Result of synthesized MAG of two-port network with different number of cascade stage, where the inter-stage matching employ 3rd-order matching network. The frequency response of 4-stage cascode is desirable with high gain and flat frequency response.
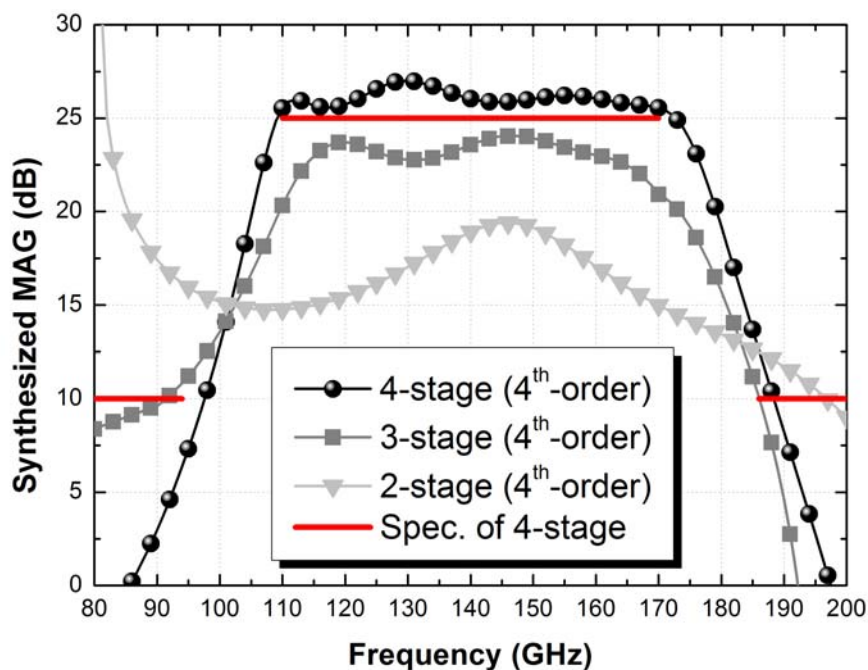


Fig. 3.8. Result of synthesized MAG of two-port network with different number of cascade stage, where the inter-stage matching employ 4th-order matching network. These solutions are more close to the specification than the results of using 3rd-order matching network, but the correspondent circuit topology is more complex for layout.
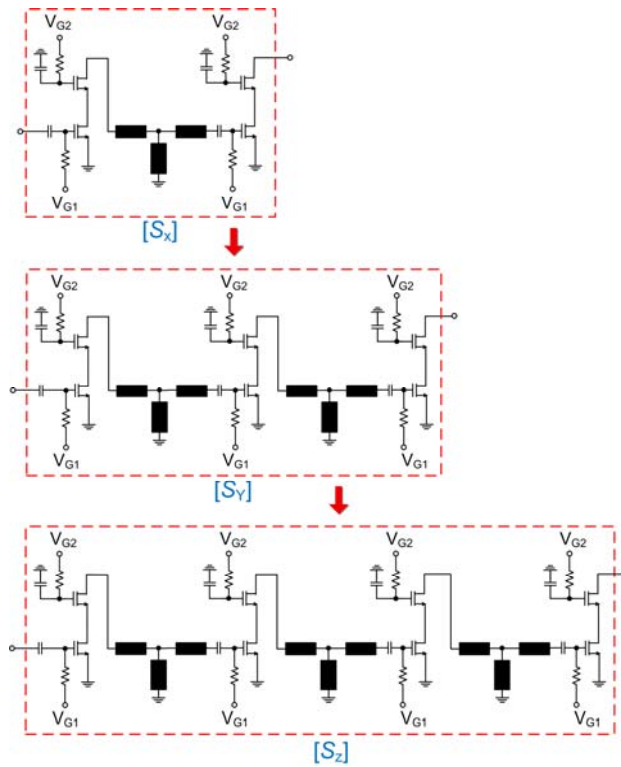
Fig. 3.9. Synthesis results with different number of cascade stage in 3rd-order inter-stage matching network. All the synthesized inter-stage matching result are the simplex T-network.
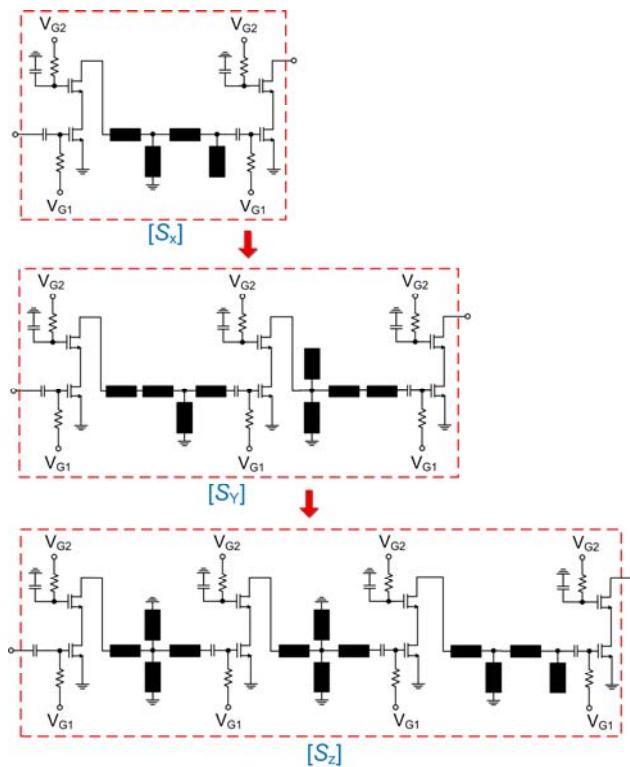


Fig. 3.10. Synthesis results with different number of cascade stage in 4th-order inter-stage matching network. The result of employing 4th-order network is more complicated for layout.
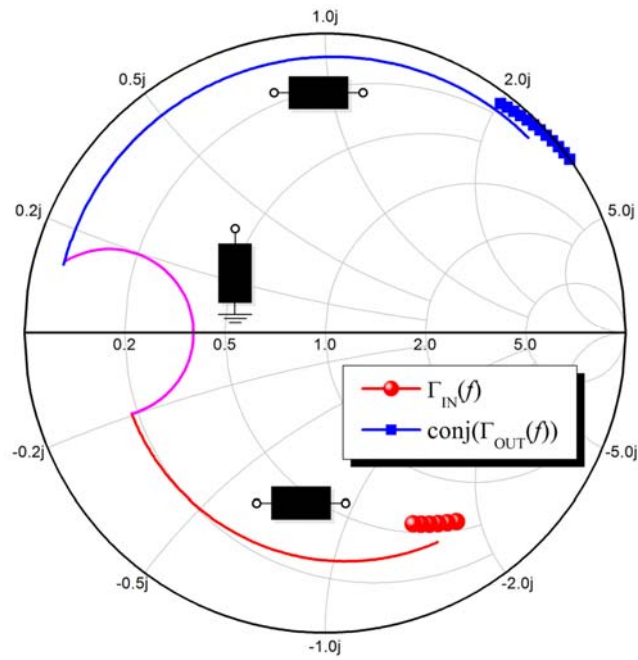
Fig. 3.11. Illustration of impedance transformation with the synthesized T-network, which is the optimal synthesized result of 3$^{rd}$-order inter-stage matching network.


Table 3.1.  Parameter settings of search rules for in-band response.

| Parameter | Value |
|---|---|
| Gain_spec$_{in\text{-}band}$ | 15, 20, 25 for 2-4 stage |
| $\alpha_{ripple}$ | 0.2 |
| Frequency Range | 110-170 GHz |
| Sampling Step | 1 GHz |


Table 3.2.  Parameter settings of search rules for out-band response.

| Parameter | Value |
|---|---|
| Gain_spec$_{out\text{-}band}$ | < 10 dB |
| $w_{out\text{-}band}$ | 0.1 |
| *Penalty* | 10 |
| Frequency Range | 80-95 and 185-200 GHz |
| Sampling Step | 1 GHz |

After the synthesis of the inter-stage matching, the optimized MAG of cascade stage can be obtained by synthesizing the simultaneously-conjugate-matching as shown in Fig. 3.12. Due to the parasitic capacitance effect in large size transistors beyond 100 GHz, it is necessary to decrease total gate width to optimize the bandwidth and high gain frequency response. However, the small device size makes it difficult to transform output impedance of the cascode stage to the system impedance 50 Ω, which leads to the poor output return loss and low gain. The in-band $\Gamma_{ML}(f)$ of each number of cascade stage are shown in Fig. 3.13, and the synthesized $4^{th}$-order matching result $\Gamma_L(f)$ is also in the figure. Fig. 3.14 shows the $S$-parameters of the 4-stage cascode with optimized simultaneous-conjugate matching of $4^{th}$-order network. Although the optimized MAG provided by the 4-stage cascode is high, the gain of matched circuits cannot achieve the optimized MAG and the return loss is undesirable, because the output impedance of cascode stage is too high to perfectly match. Therefore, it is suitable to apply other techniques to achieve broadband response rather than increasing the network order.
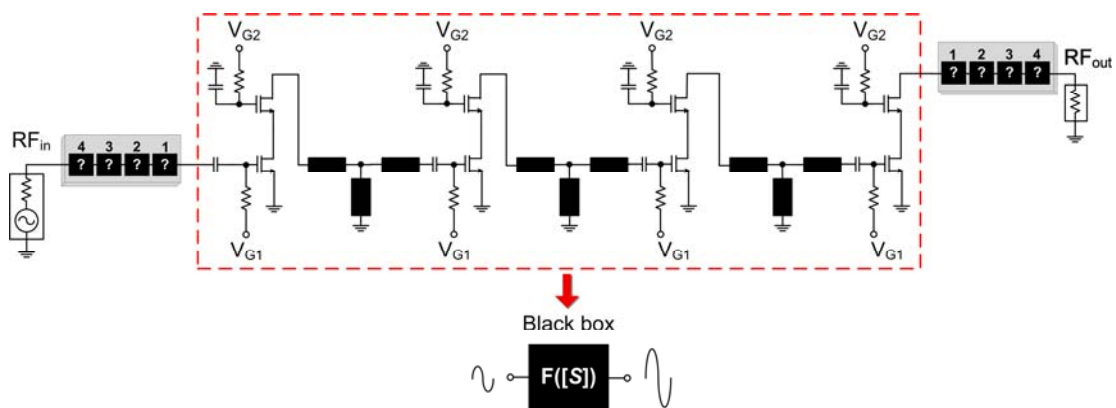


Fig. 3.12. Synthesis of simultaneously-conjugate-matching with the optimized inter-stage matching to obtain the MAG provided by the 4-stage cascode.
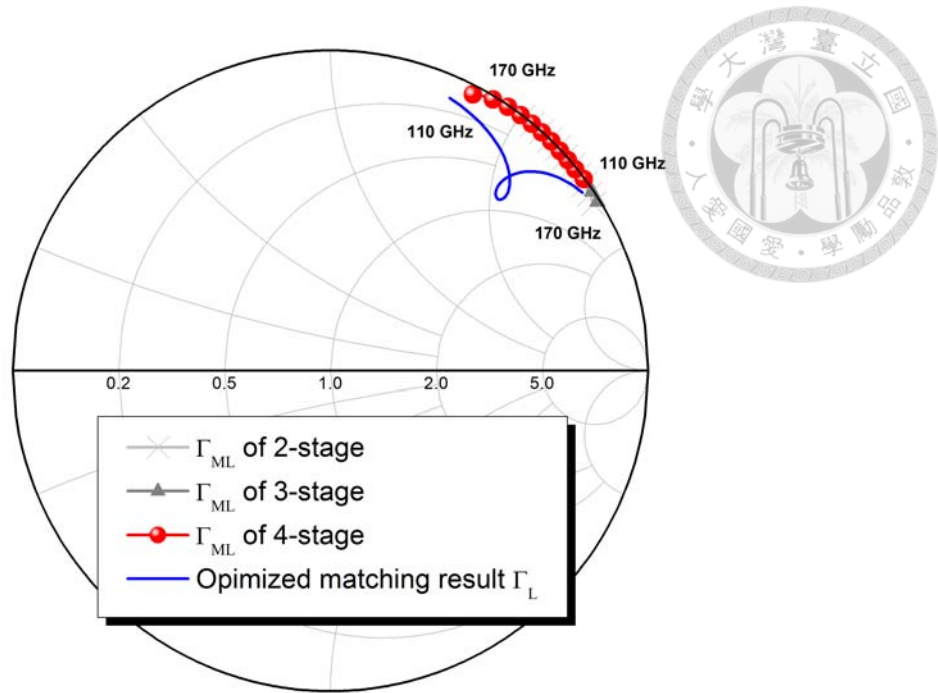
Fig. 3.13. The in-band $\Gamma_{ML}(f)$ of each cascade stages after optimization of inter-stage matching. The synthesized 4th-order matching result $\Gamma_L(f)$ cannot achieve the target response under reasonable network order.
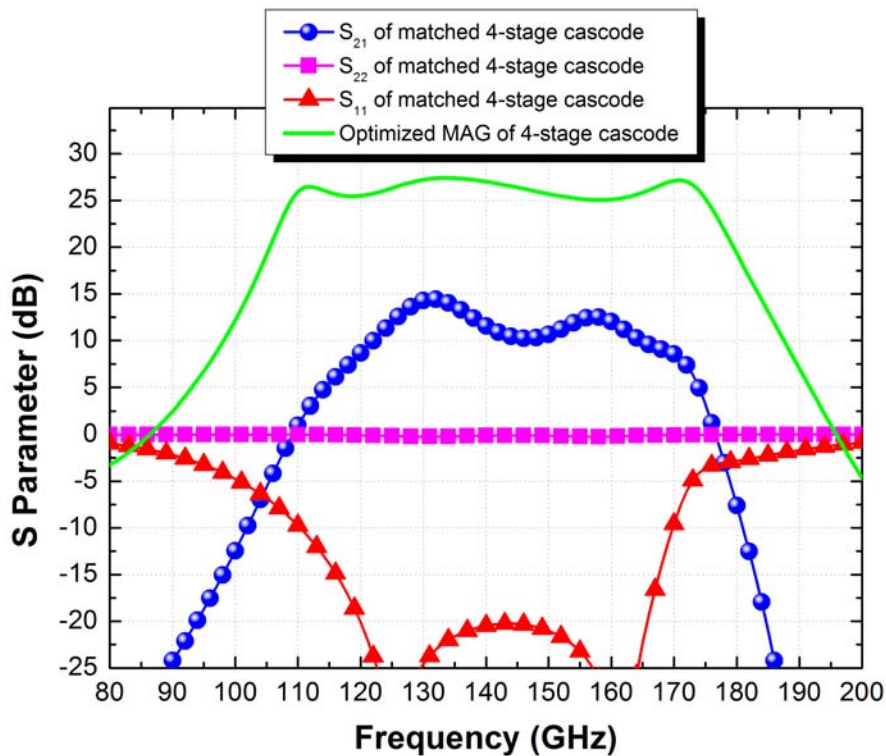


Fig. 3.14. The *S*-parameters of 4-stage cascode that are matched with synthesized input and output 4th-order network. Due to the high output impedance of cascode stage is difficult to match, the gain cannot achieve the optimized MAG and the output return loss is undesirable.

Fortunately, this problem can be solved by adding a CS stage for trans-impedance as Fig. 3.15, because of the high input and lower output impedance characteristic of CS topology. Fig. 3.16 illustrates this output impedance transformation result, it can be observed that the frequency response of $\Gamma_{ML}(f)$ is closer to the system impedance 50 $\Omega$ than 4-stage cascode without trans-impedance, which is easier to match than the impedance near the periphery of Smith chart.

After connecting a CS stage for trans-impedance, the synthesis procedure is performed again with $3^{rd}$-order inter-stage matching, and the synthesized inter-stage matching topology is as the same in Fig. 3.9 (all network is the T-network). Fig. 3.17 shows $S$-parameters of 4-stage cascode with 1-stage trans-impedance stage, which are matched with synthesized input and output $4^{th}$-order network. Based on the trans-impedance procedure, the matched circuit response can achieve higher gain and wider bandwidth performance than the 4-stage cascode without trans-impedance. Although adding a trans-impedance stage increases the chip size and dc-power consumption, the circuit performance can be significantly enhanced.
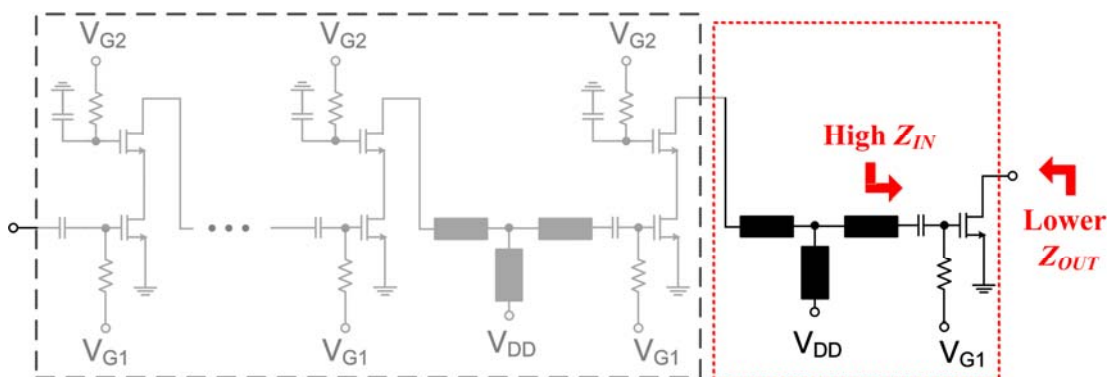


Fig. 3.15. Procedure of adding a trans-impedance stage following cascode stage.
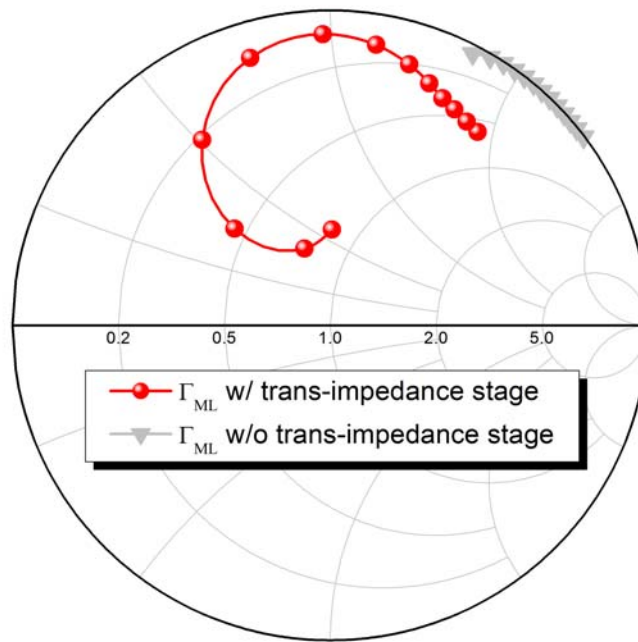
Fig. 3.16. Output impedance transformation of 4-stage cascode by using trans-impedance stage with ideal passive components.
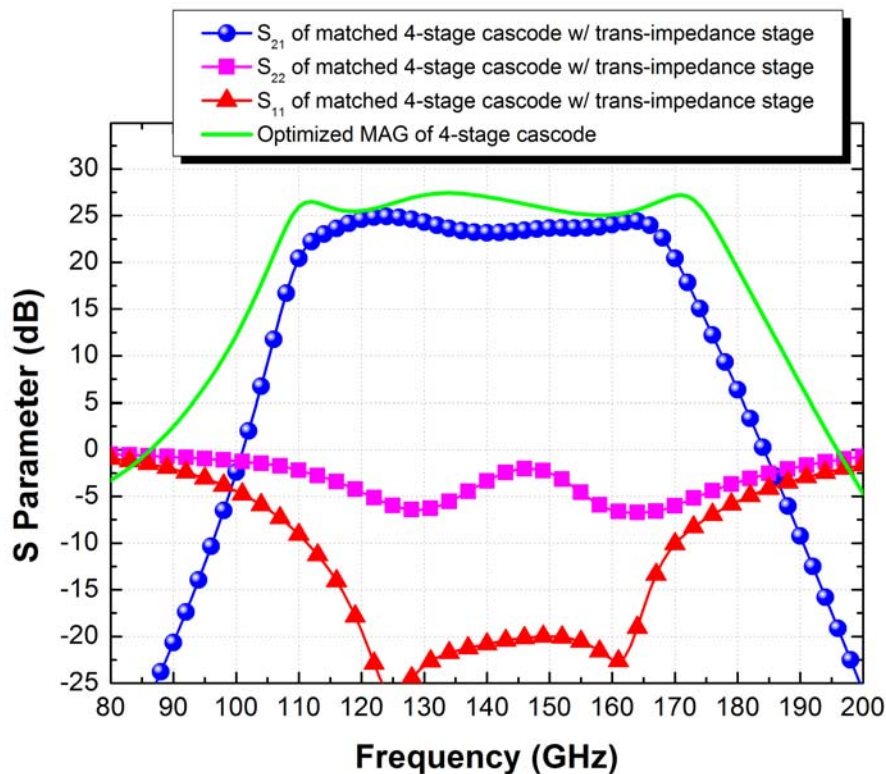


Fig. 3.17. The *S*-parameters of 4-stage cascode with 1-stage trans-impedance stage that are matched with optimized input and output 4th-order network. Due to the trans-impedance procedure, the matched circuit response can achieve higher gain and wider bandwidth performance than the 4-stage cascode without trans-impedance (Fig. 3.14).

## 3.1.4  Post-Simulation Results

Fig. 3.18 is optimization result which has considered the non-ideal effect of on-chip passive components with full-wave EM simulations (Sonnet [44]). The long transmission lines are replaced with inductor for reducing chip size. Due to the loss of on-chip passive components and the non-ideal effect of high-order mode EM wave, the optimized maximum available gain is lower than ideal optimized results.
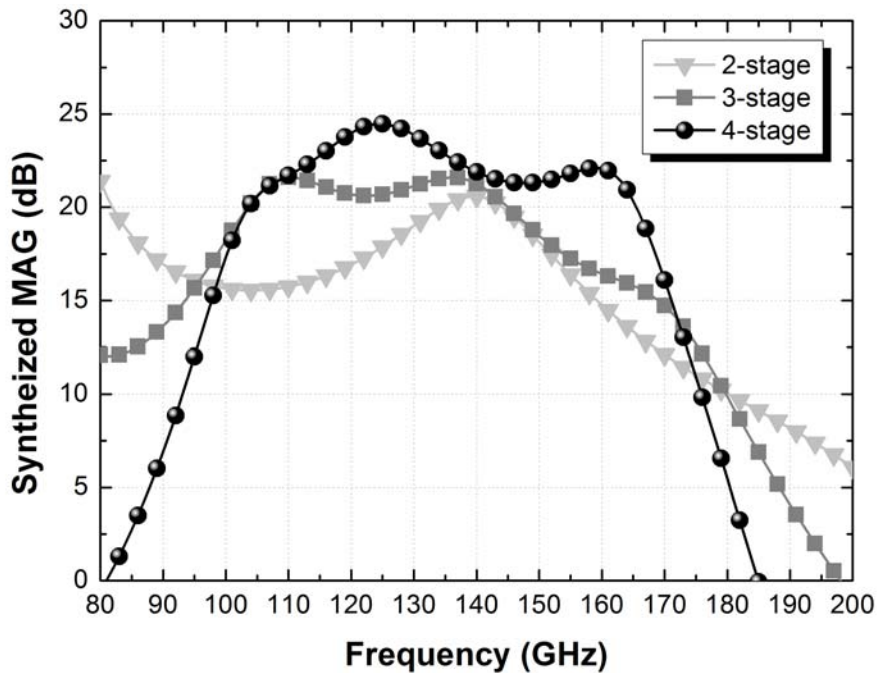


Fig. 3.18. Result of synthesized MAG after considering the non-ideal effect of passive components and full-wave EM simulations

Fig. 3.19 illustrates this output impedance transformation provided by the trans-impedance stage and correspondent synthesized matching result. It is very difficult to match the output impedance without the suitable trans-impedance stage, because the $\Gamma_{ML}(f)$ is near to the edge of Smith chart, which causes the difficulty of matching and degrade the gain and bandwidth performance. Since the trace of $\Gamma_{ML}(f)$ becomes circular shape after impedance transformation, it is simple to perform broadband matching as

long as the frequency response of output matching network to the center of the circle can be designed. Using this approach, the distance between the $\Gamma_{ML}(f)$ and synthesized $\Gamma_L(f)$ will be minimized (i.e. the response is broadband). The square error can be reduced about 80% by performing the trans-impedance, which makes the frequency response broadband and desirable. Both the post-simulation results of $S$-parameters are shown in Fig. 3.20. Due to the trans-impedance effect, the matched circuit response can achieve higher gain and bandwidth performance than the response of without trans-impedance 4-stage cascode.

Fig. 3.21 shows the overall circuit schematic, and the chip photograph is shown in Fig. 3.22 with a chip size 0.57×0.65 mm$^2$. It is 4-stage cascode with a trans-impedance stage. The four cascode stages provide the primary gain performance and the trans-impedance stage can improve output matching. There are slight differences between each cascode stage in synthesis procedures to ensure reasonable results, such as source degeneration at the first stage to improve the input return loss, and the larger device size of output stage for better power handling. Although the inter-stage stability was not considered in the proposed algorithm, it has been checked to avoid the oscillation after the post-simulation of the entire circuit, as shown in Fig. 3.23.
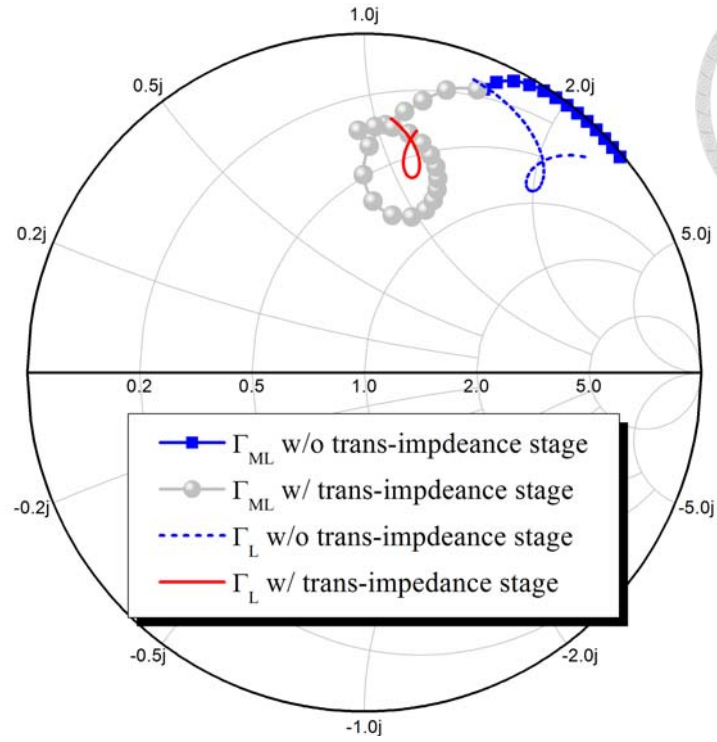
Fig. 3.19. Output impedance transformation of four stage cascode by using trans-impedance stage. The 4$^{th}$-order matching result $\Gamma_{L.}(f)$ is synthesized by the proposed algorithm.
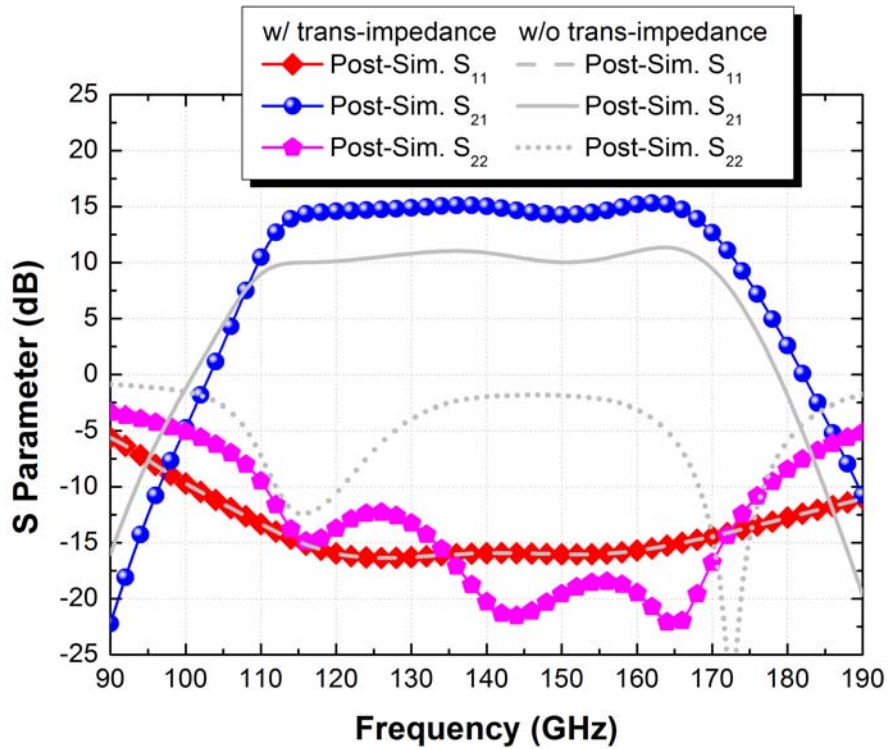


Fig. 3.20. Post-simulation result of *S*-parameters with (symbol line) and without trans-impedance. Due to the trans-impedance effect and synthesized matching, the small signal gain, return loss and bandwidth are desirable.
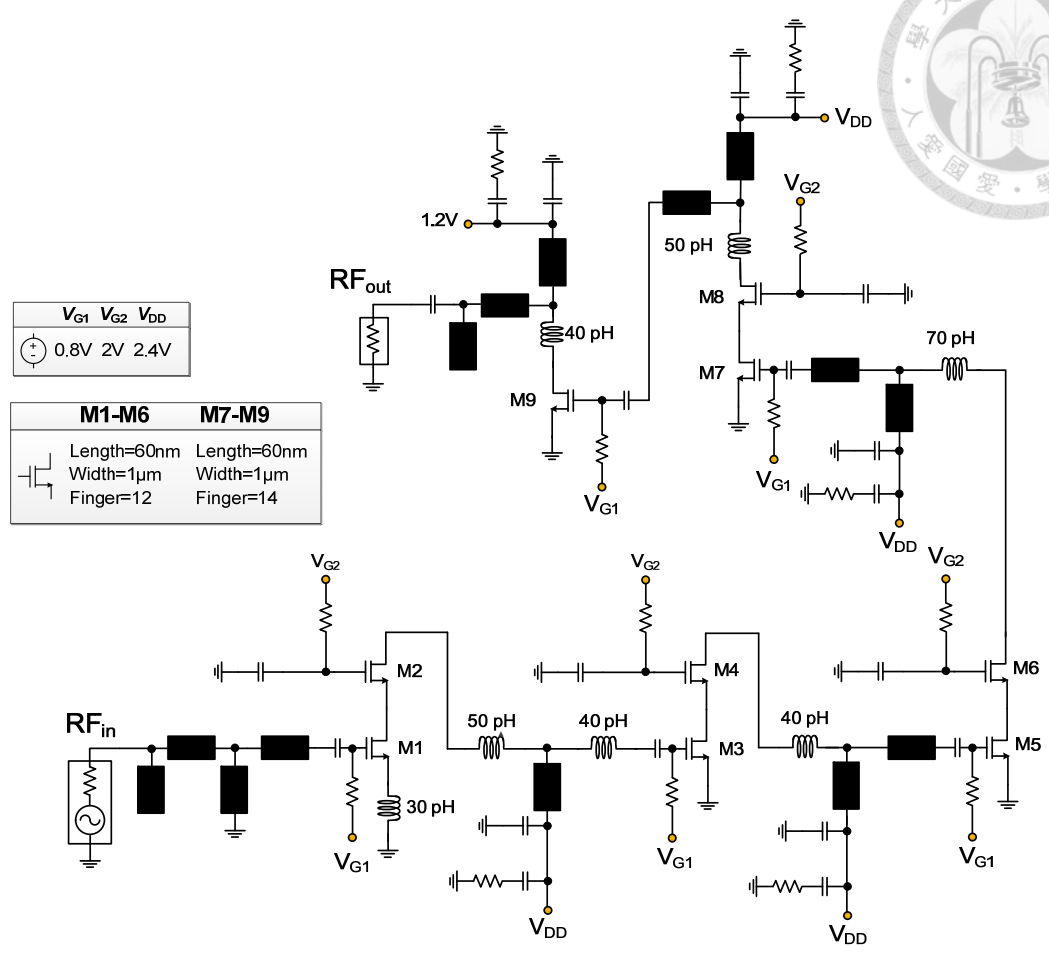
Fig. 3.21. Circuit schematic of the four cascode stage and one trans-impedance stage. The four cascode stages provide the primary gain performance and the trans-impedance stage can improve output matching.
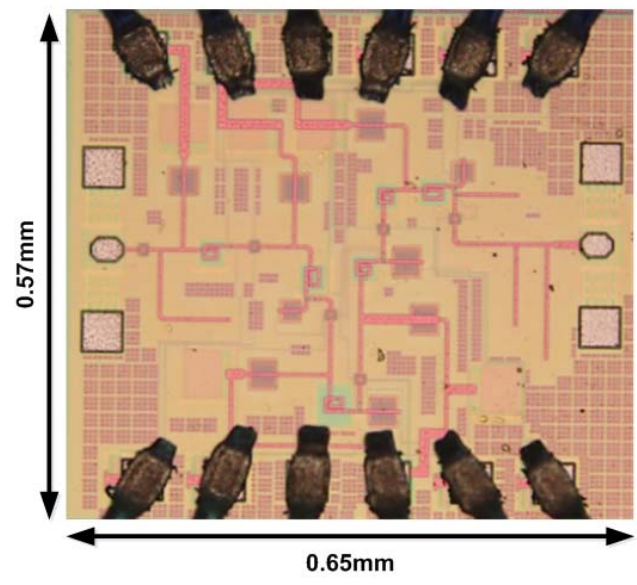


Fig. 3.22. Chip photograph with chip size 0.37 mm$^2$.
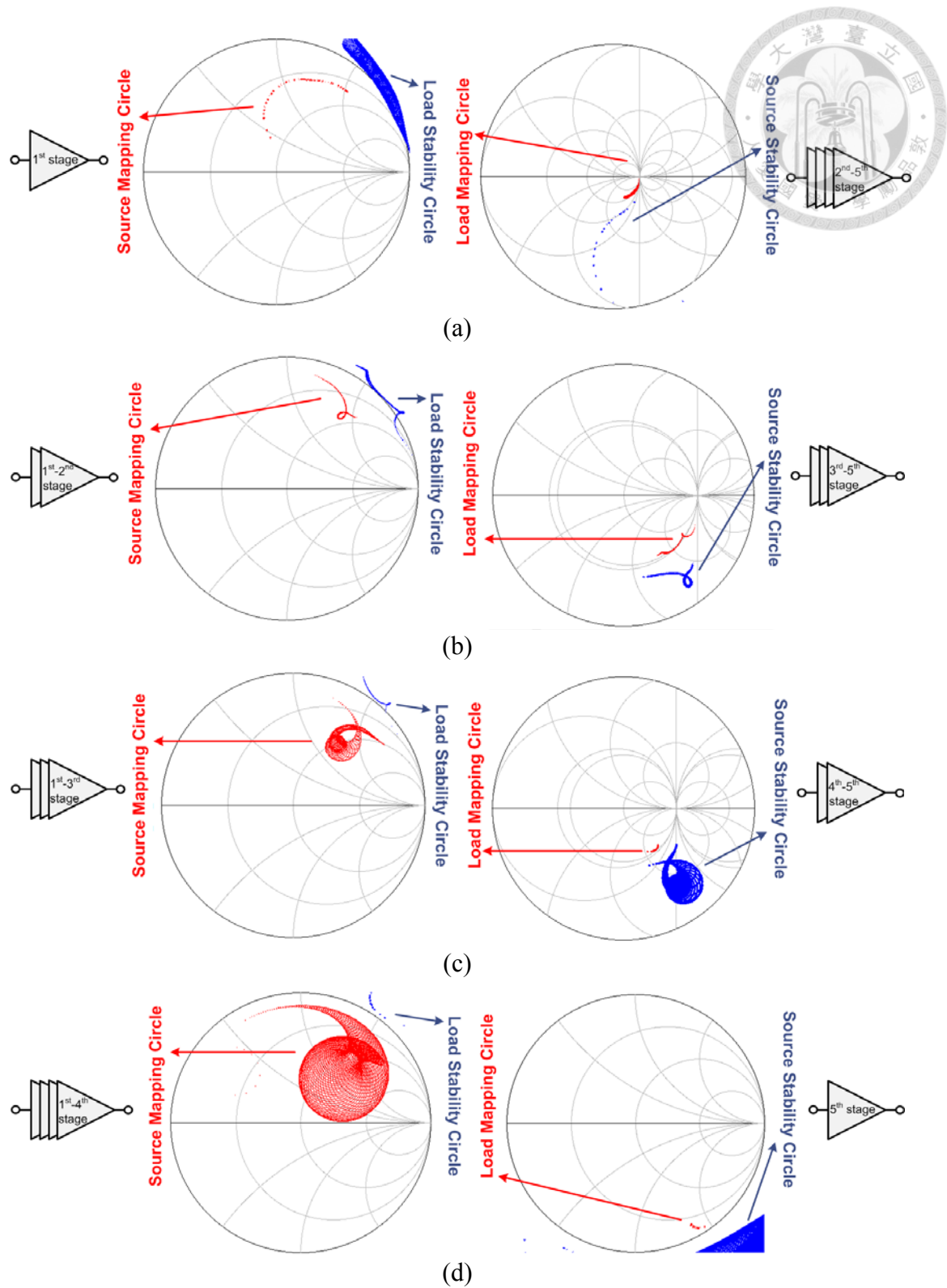
Fig. 3.23. (a)-(d) inter-stage stability check from 1 to 200 GHz. The mapping circles do not intersect with the correspondent stability circles, that is, the oscillation is avoided in the inter-stage.

### 3.1.5   Measurement Results

The *S*-parameters of this broadband amplifier are measured from 110 GHz to 220 GHz by employing Agilent VNA. The measurement and simulation result is illustrated in Fig. 3.24. Due to the imperfect transistors characteristic modeling at this frequency, there are some mismatch between the simulation and measurement. It shows a broadband amplification characteristic of the small signal gain higher than 10 dB from 110 GHz to 180 GHz, and 19-dB maximum small signal gain is at 170 GHz. Since we use the 1.2-V MOS-varactors in bypass circuits rather than 3.3-V MOS-varactors by mistake, it causes an unnecessary 43-mW extra dc power consumption in the bias network, and thus the total dc power is 109 mW at $V_{DD} = 2.4$ V, but the actual dc power dissipated in transistors is only 66 mW.
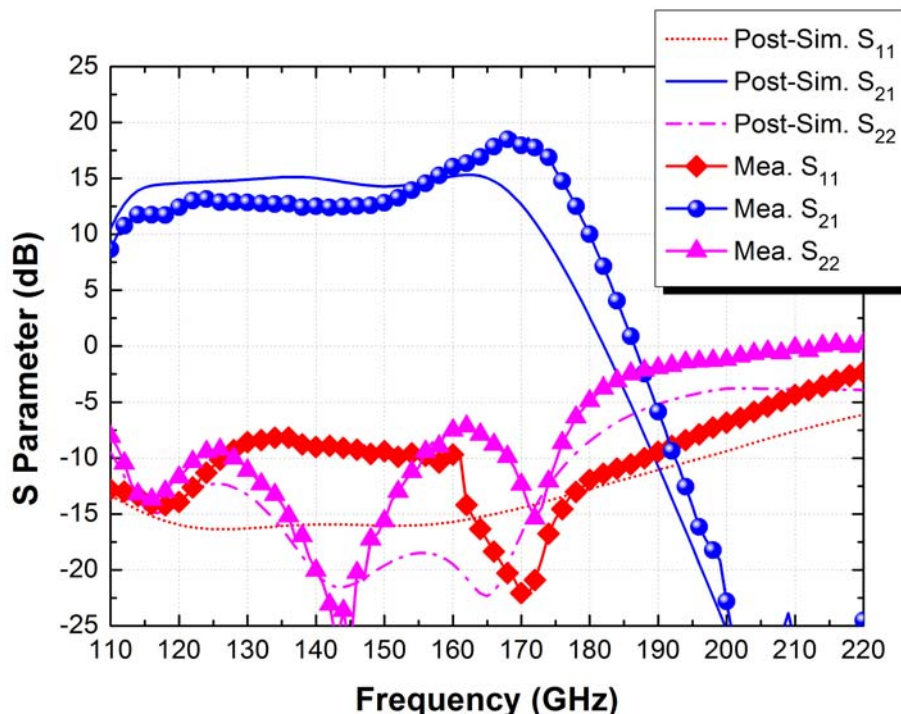
Fig. 3.24. Measurement and simulation results of *S*-parameters from 110 GHz to 220 GHz. It shows a broadband amplification characteristic of the small signal gain higher than 10 dB from 110 to 180 GHz, and 19-dB maximum small signal gain at 170 GHz.

Table 3.3 summarizes the amplifier performance of this work and compares with previously published D-band amplifiers in silicon. It can be observed that the gain performance in this work is comparable to the published amplifiers, but our amplifier has the widest bandwidth (The amplification bandwidth with gain > 10 dB from 110-180 GHz) with the reasonable dc-power consumption and chip size, and indeed covers the entire D-band.

In this work, we use the proposed design method and matching network synthesis algorithm to successfully achieve the broadband bandwidth and high gain performance for D-band amplifier. This is the first amplifier covering full D-band in CMOS process.

Table 3.3. Summary of the previously published D-band Si-based amplifier.

| Reference | Ref.[28] 2008 RFIC | Ref.[29] 2010 ASSCC | Ref.[30] 2009 ISSCC | Ref.[31] 2012 IMS | Ref.[32] 2008 RFIC | This Work |
|---|---|---|---|---|---|---|
| Process | 65-nm CMOS | 65-nm CMOS | 65-nm CMOS | 65-nm CMOS | 0.13-μm SiGe | 65-nm CMOS |
| Topology | 6-Stage Single-Ended LNA | 3-Stage Differential Amplifier | 3-Stage Single-Ended Amplifier | 4-Stage Single-Ended PA | 5-Stage Single-Ended Amplifier | 5-Stage Single Ended Amplifier |
| 3-dB Center Frequency (GHz) | 140 | 144 | 150 | 140 | 165 | 170 |
| Gain > 10 dB Frequency Range (GHz) | N/A* | 127-157 | N/A* | N/A* | 159-179 | 110-180 |
| 3-dB Bandwidth (GHz) | 10 | < 5 | 27 | > 30 | 15 | 15 |
| Gain > 10 dB Bandwidth (GHz) | 0 | 30 | 0 | > 30 | 20 | 70 |
| Peak Gain (dB) | 8 | 20.6 | 8.2 | 15 | 14 | 19 |
| dc Power (mW) | 63 | 102 | 25.5 | 115.2 | 135 | 66** |
| Chip Size (mm$^2$) | 0.06 *** | 0.21 | 0.41 | 0.38 | 0.14 | 0.37 |

**\*:Not available.**
**\*\*:The actual dc power dissipated in transistors.**
**\*\*\*: Estimate from chip micrographs (not including test pads).**

## 3.2 Synthesis of Ka-band Broadband Medium Power Amplifier in 0.1-µm GaAs pHEMT Process
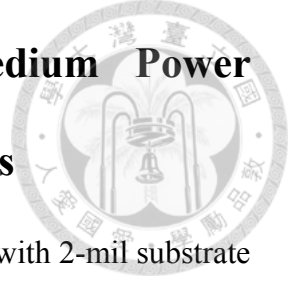
An integral circuit fabricated in 0.1-µm GaAs pHEMT process with 2-mil substrate is designed and implemented for demonstration. This circuit synthesized by our algorithm is a full-Ka band (26-40 GHz) medium power amplifier.

### 3.2.1 Problem Decomposition and Synthesis

The matching network designs can be decomposed into three parts as shown in Fig. 3.25 and it can be solved sequentially from the output stage to the input. The correspondent synthesized results are also shown in the figure. The first step is to synthesize the output network such as broadband load-pull matching; followed by transforming the input impedance of second stage to the output impedance of driver-stage like $\Gamma_{ML}(f)$ or load-pull response; finally, transform input impedance of driver-stage to the system impedance 50 Ω. The total device size ratio of output stage to driver-stage is selected as 3:1 for the higher power-added efficiency (PAE) consideration; however, a smaller device size of driver-stage makes input return loss poorer. Therefore, an RC feedback is inserted in driver-stage to improve the input return loss, and both driver and output stage are treated as black-box in optimization. If the synthesized result of input return loss is lower than 10 dB, the feedback is slightly modified and synthesis is performed again until the desirable return loss is obtained.

The parameter settings of our algorithm are as same as Table 2.4. Although the synthesis of the characteristic impedance of transmission line can help to enhance solution quality, we fix it as 72.8 Ω in 2-mil substrate for layout consideration. In order to achieve broadband response of maximum output power, it is beneficial to record the load-pull results of each frequency in 26-40 GHz as target frequency response to be
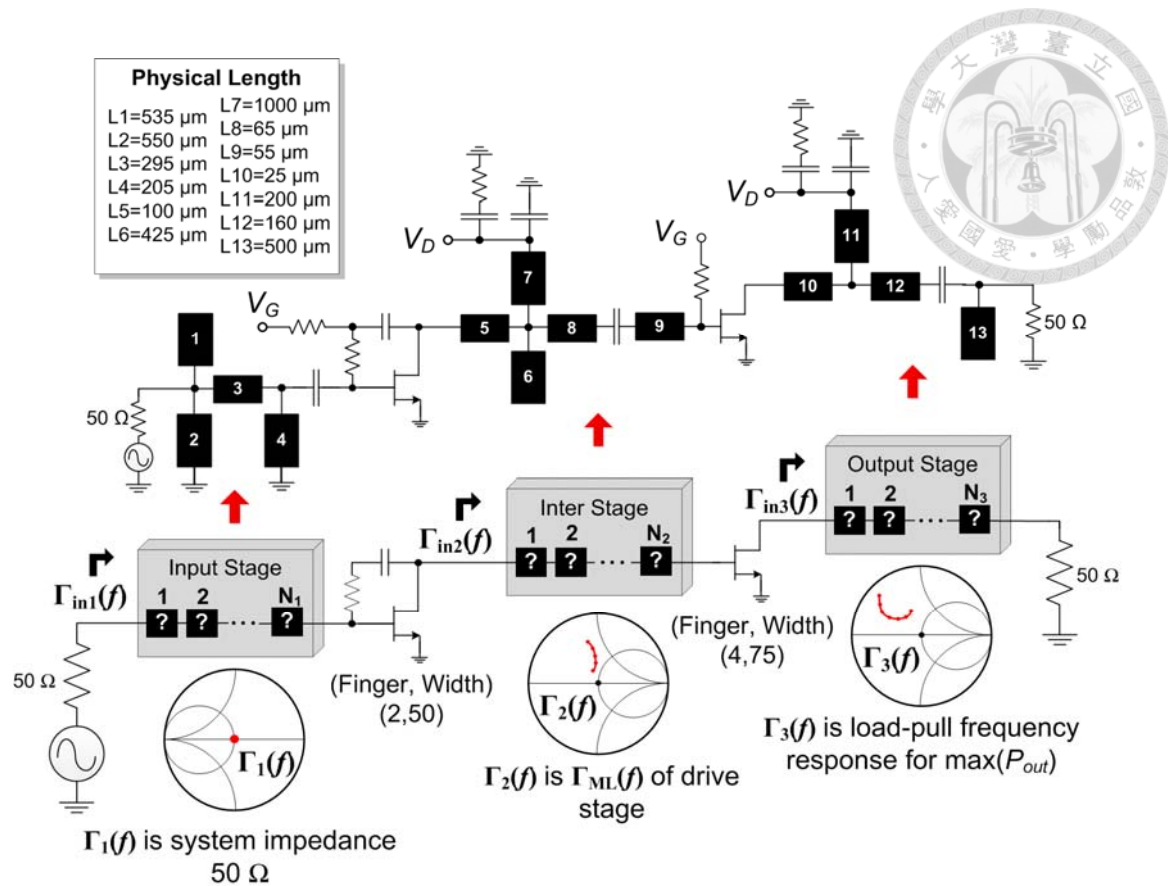
Fig. 3.25. The decomposition of synthesis two-stage single-ended power amplifier. It can be solved sequentially from output stage to circuit input. The first step is to synthesize the broadband load-pull matching, followed by transforming the input impedance of second stage to the output impedance of driver-stage; finally, transform input impedance of driver-stage to the system impedance 50 Ω.
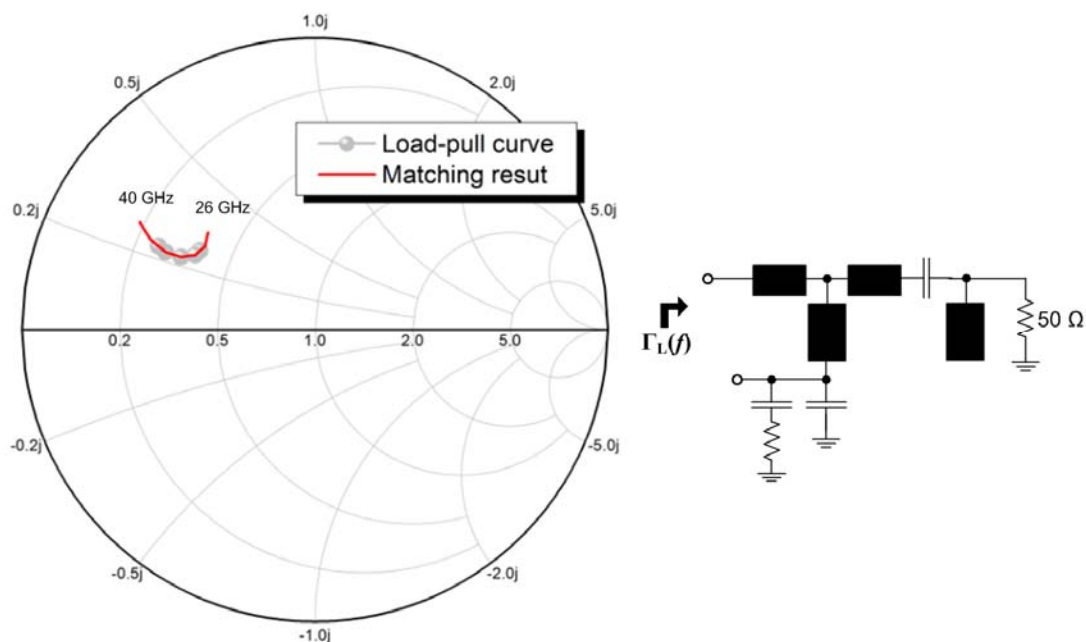


Fig. 3.26. Crcuit synthesis result of broadband load-pull matching and correspondent synthesized matching network.

synthesized, and the sample step is 1 GHz in our case. Fig. 3.26 shows the results of matching load-pull curve and correspondent synthesized matching network. It is obvious that the synthesized result can fit the target response in most part, which makes the frequency response of output power broadband.

The chip photograph is shown in Fig. 3.27 with a compact chip size of $1 \times 1.5$ mm$^2$. Since the coupling effect should be taken into account, several physical parameters are slightly adjusted with the help of commercial full-wave EM software (Sonnet [44]). Due to the synthesized matching network, the circuit performs the good broadband responses. Fig. 3.28 shows the simulation results of $S$-parameters, both the small signal gain and return loss are desirable. The large signal result is shows in Fig. 3.29, the $P_{sat}$ is 21-22 dBm around the Ka-band with the high PAE. The diagram of simulated input power versus output power is illustrated in Fig. 3.30. The long short stub in the inter-stage matching is replaced with a 1-turn spiral inductor to compact the chip size, and there is only negligible difference in the frequency response as shown in Fig. 3.31.
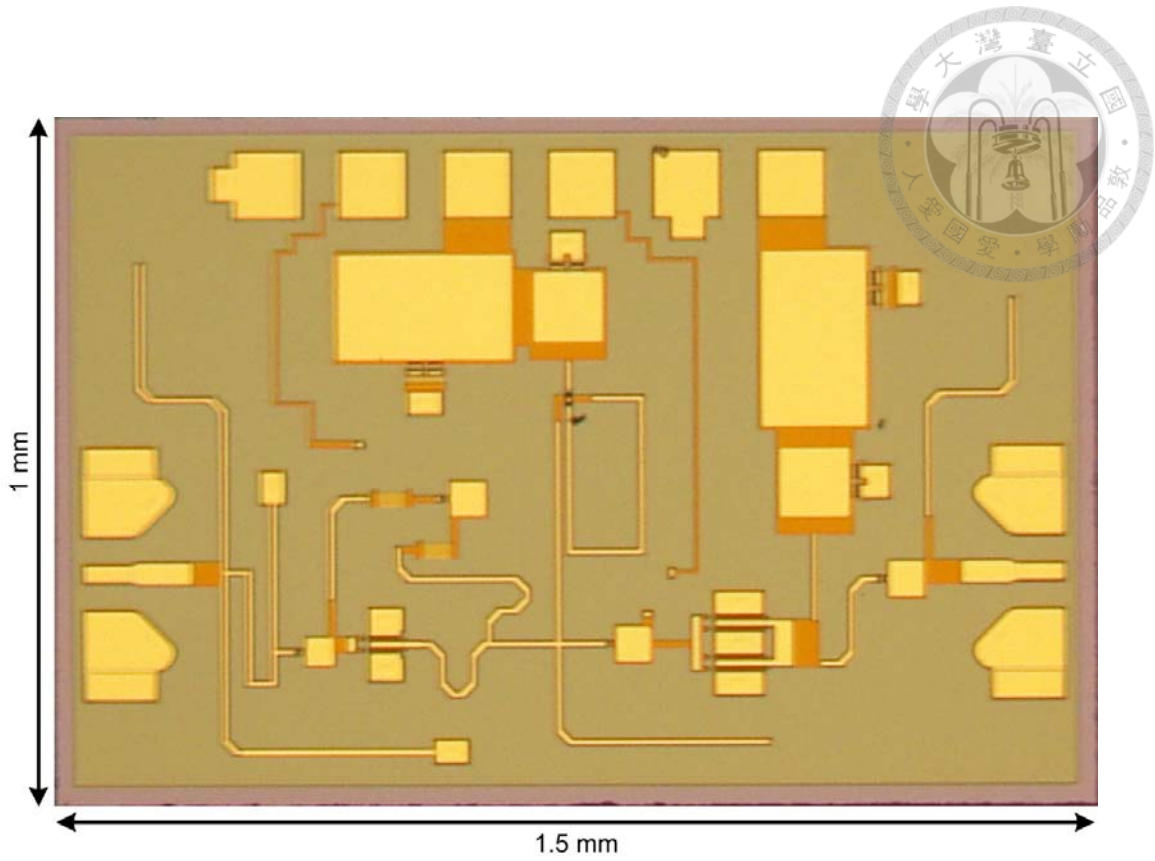
Fig. 3.27. Chip photograph with a compact chip size $1 \times 1.5$ mm$^2$.
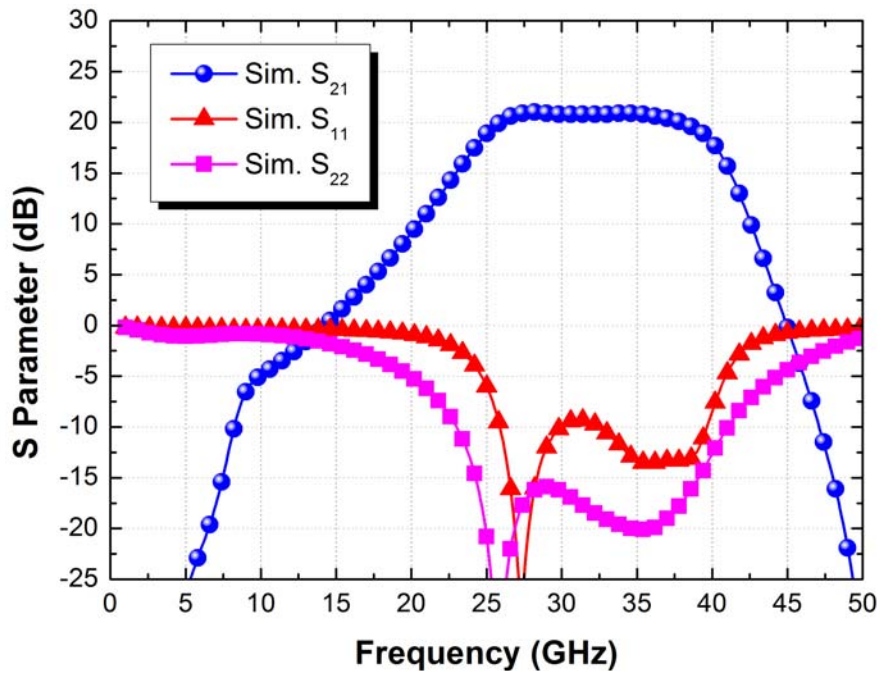


Fig. 3.28. Simulation results of *S*-parameters. The frequency response of gain and return loss is desirable due to synthesized matching.
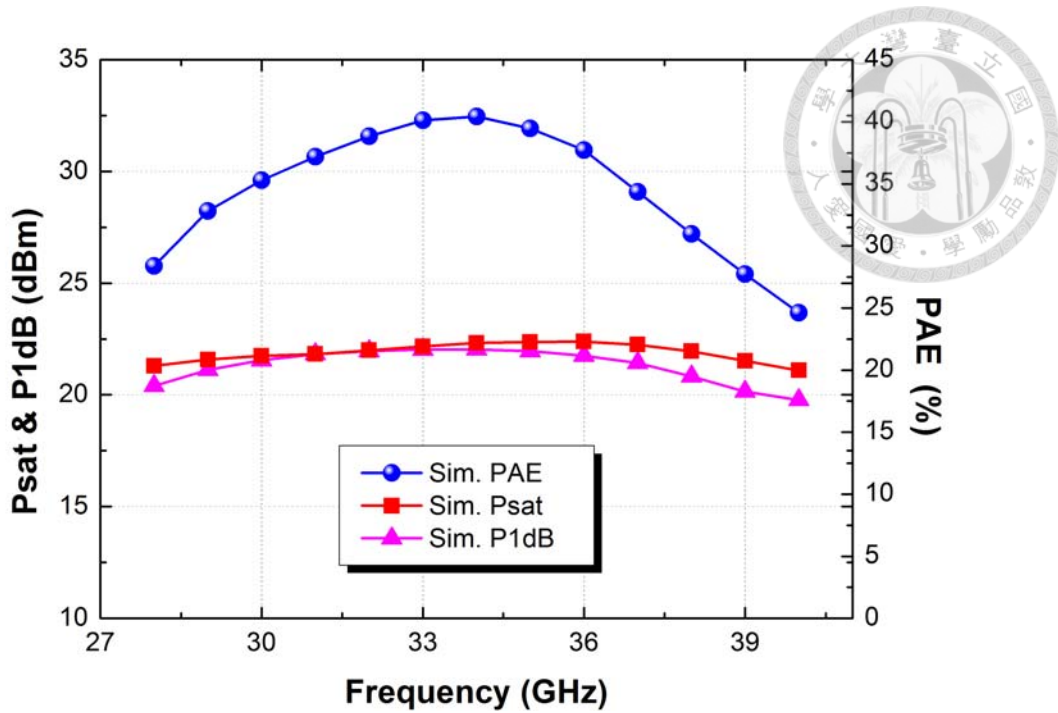
Fig. 3.29. Simulation results of large signal. This power amplifier performs the broadband responses and covers the full Ka-band (26-40 GHz).
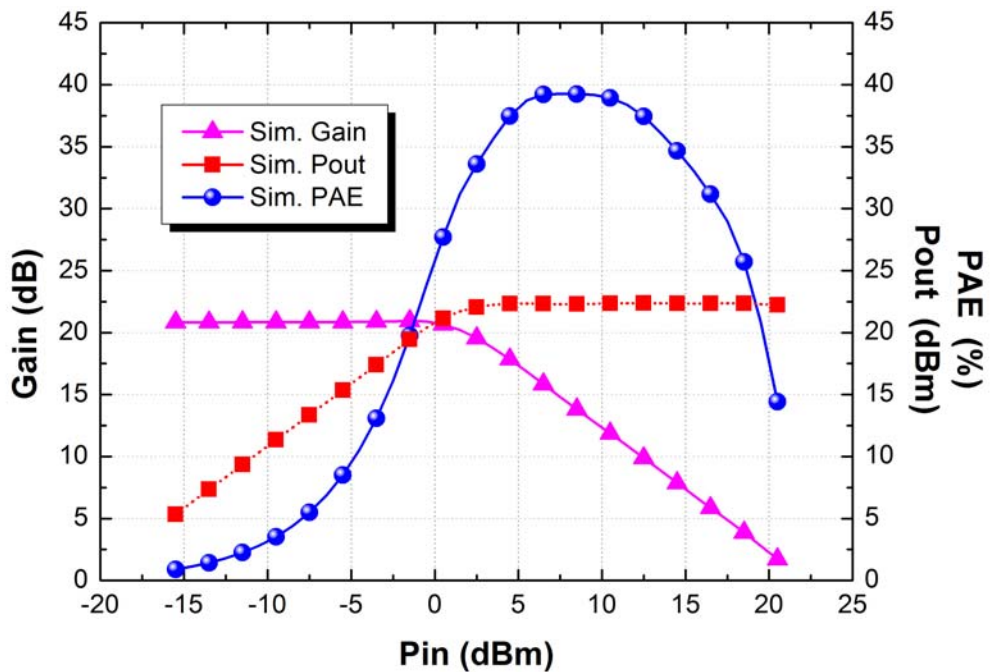


Fig. 3.30. Simulation results of input power versus output power at center frequency 34 GHz. The $P_{sat}$ is 22 dBm with a 40% peak PAE.
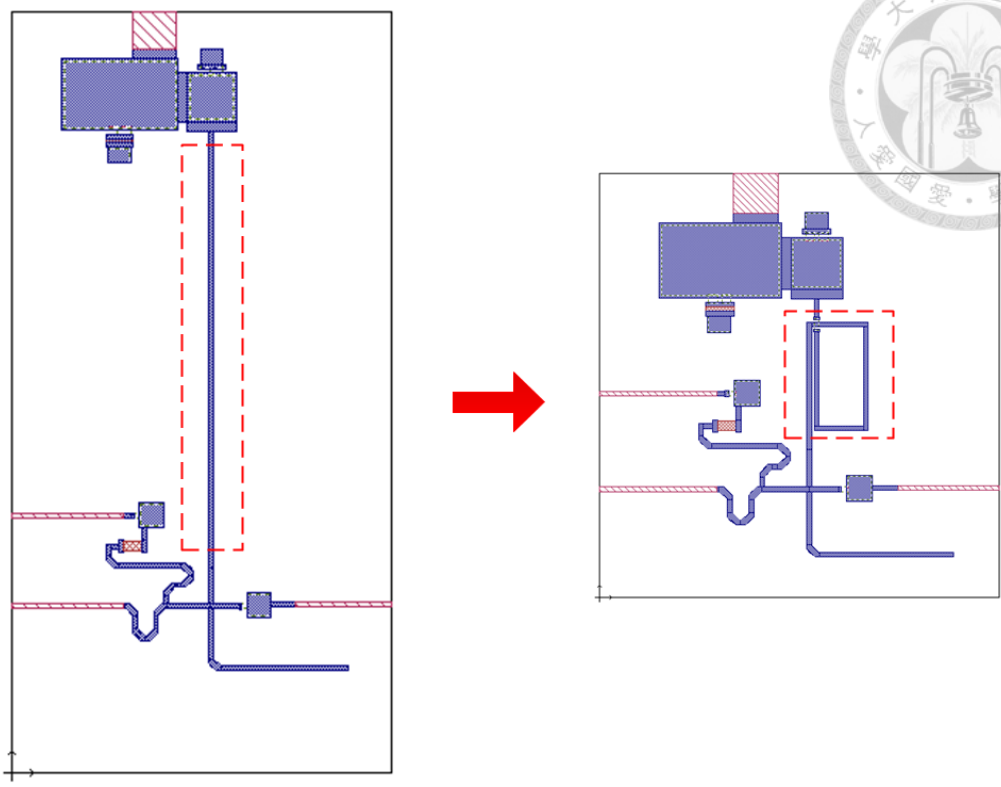
(a)
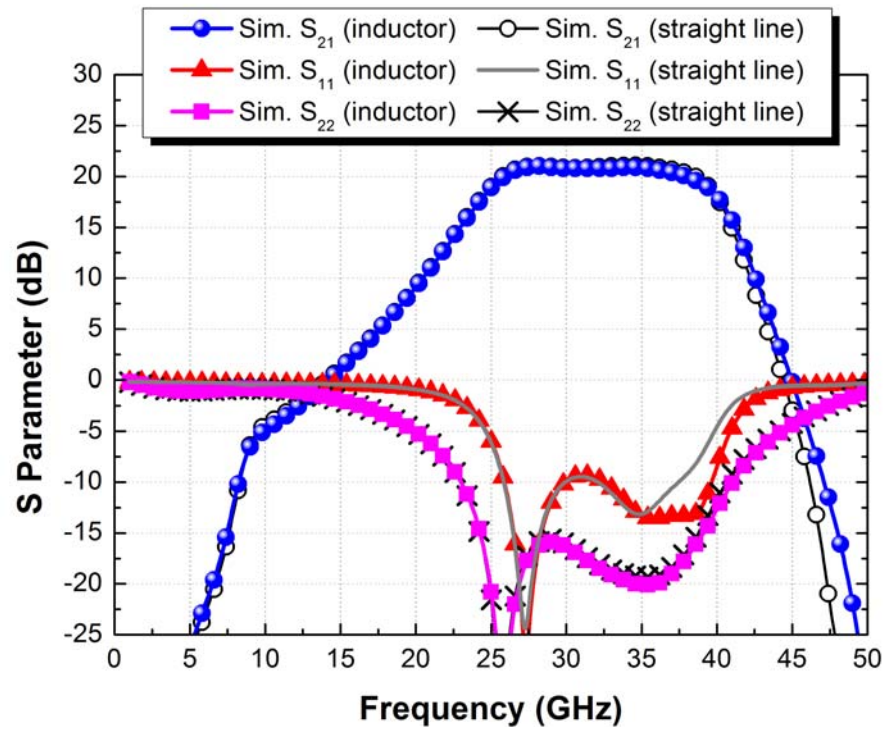


(b)

Fig. 3.31. (a) Illustration of compacting Inter-stage matching (layout view in Sonnet).
(b) The post-simulation results after replacing the long shot stub to a 1-turn spiral
inductor. The difference of frequency response is few enough to neglect.

## 3.2.2　Measurement Results

Fig. 3.32 shows measurement results of the *S*-parameters from 1 GHz to 50 GHz by employing Agilent VNA. Fig. 3.33 shows measurement results of large signal in Ka-band. The measured output power $P_{sat}$ at peak PAE is 19-22 dBm, the correspondent PAE is 22-35%. Fig. 3.34 shows the diagram of input power versus output power at the central 34 GHz. Due to good synthesis of matching network via proposed algorithm, the frequency response of output power and PAE are broadband, and both the small signal gain and return loss are desirable.

Due to the imperfect transistors modeling of both large and small signal at this frequency, there are some mismatch between the simulation and measurement; the dc-IV curve of output stage is shown in Fig. 3.35, and the $V_G$ is slightly changed (shift 0.05 V) in measurement to maintain the same drain current at simulation. The mismatch of the transistor behavior makes the simulation and measurement inconsistent. Therefore, constructing a better transistor model [45]-[46] in this 0.1-μm GaAs pHEMT process is beneficial for more accurate design.

Table 3.4 summarizes the performance of this work and compare with previously published HEMT Ka-band power amplifier. This experiment illustrates a simple design methodology of medium power amplifier under real-time constraint by circuit synthesis algorithm, and this synthesis can be extended to the architecture with transistors combining for a higher output power.
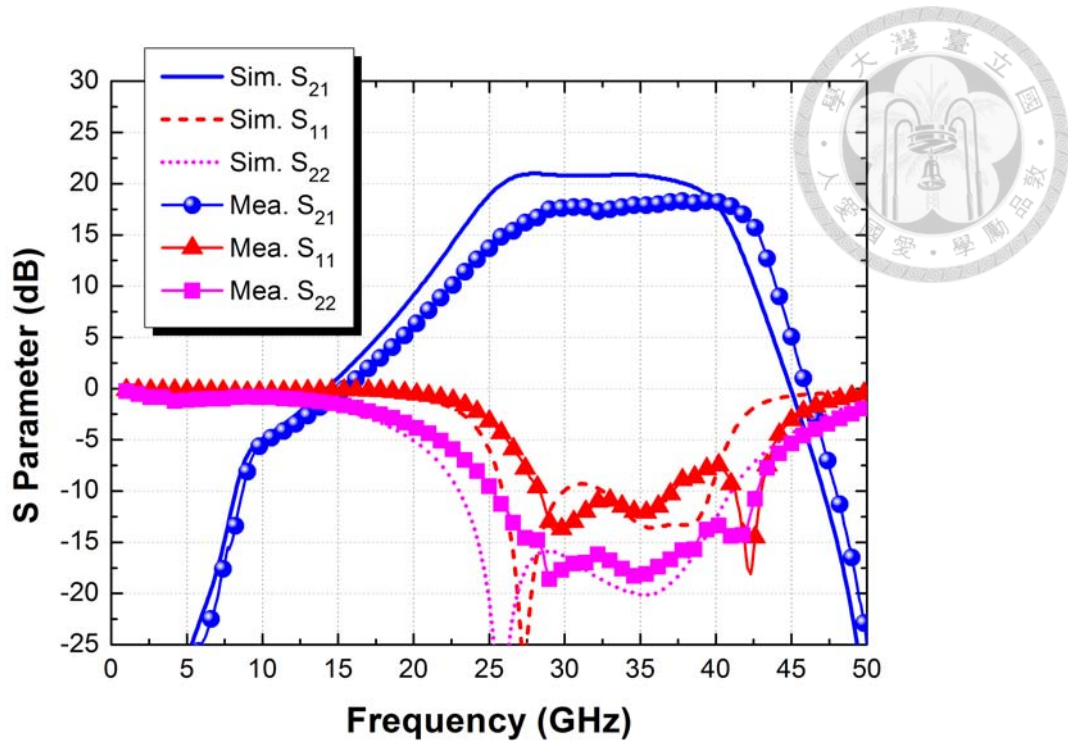
Fig. 3.32. Measurement and Simulation results of *S*-parameter from 1 GHz to 50 GHz. Both the in-band gain and return loss are desirable due to good synthesized matching.
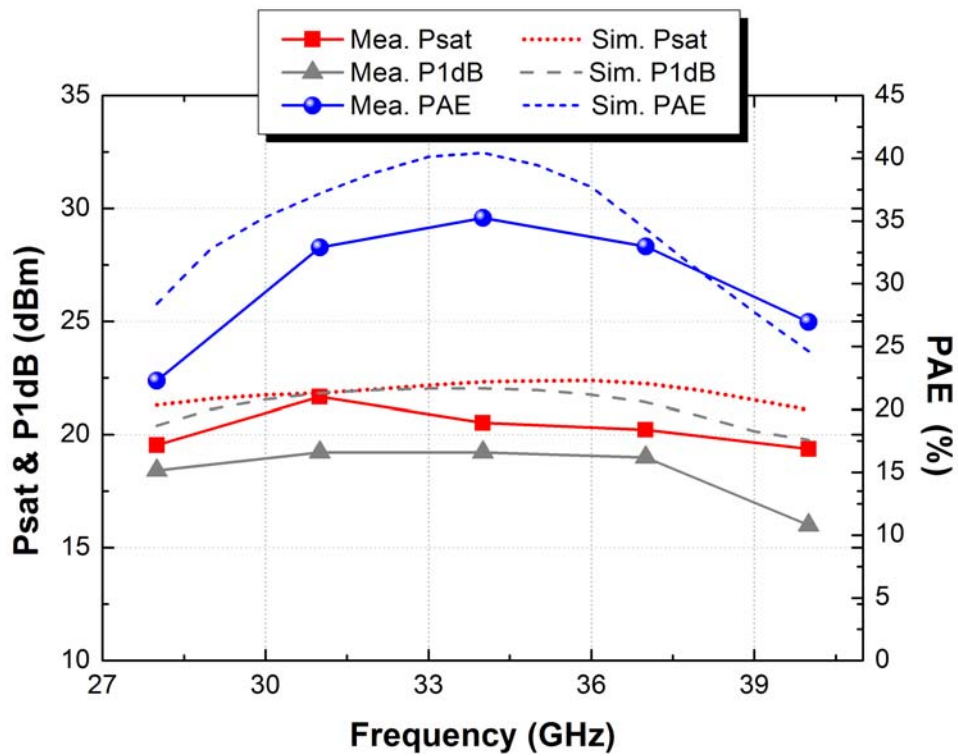


Fig. 3.33. Frequency response of large signal. The $P_{sat}$ is 22 dBm with 33% peak PAE at 31 GHz.
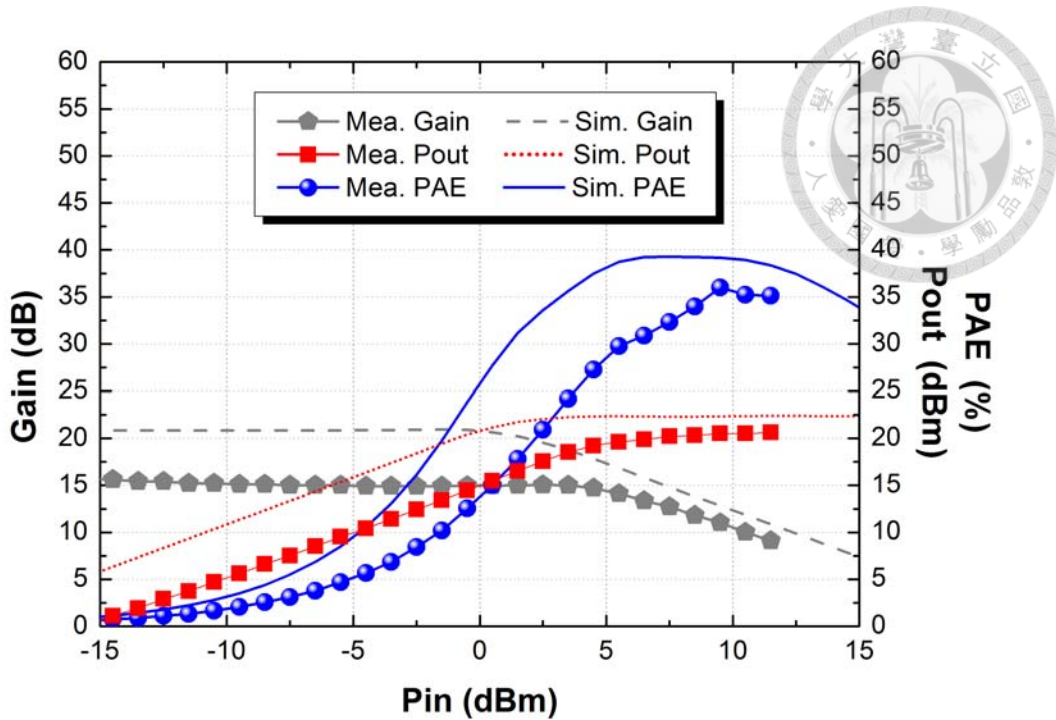
Fig. 3.34. Measurement result of input power versus output power. The $P_{sat}$ is 20.6 dBm with 35% peak PAE at 34-GHz central frequency.
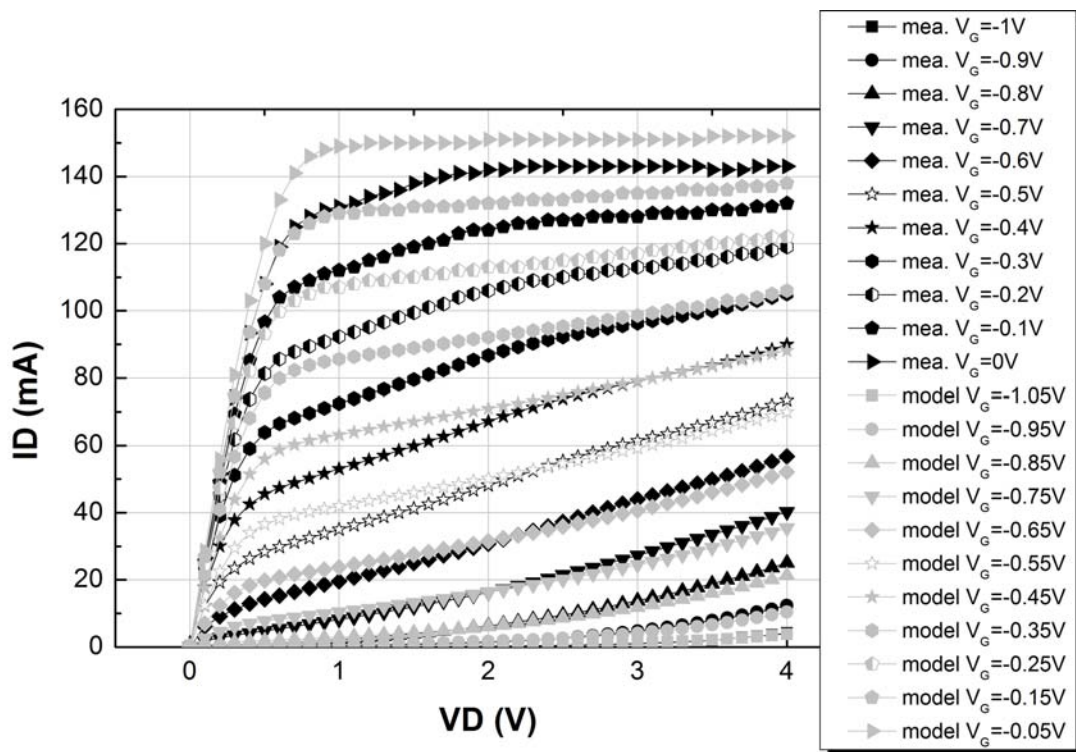


Fig. 3.35. dc-IV of output stage of simulation and measurement. The $V_G$ is slightly changed (shift 0.05 V) in measurement to maintain the same drain current at simulation.
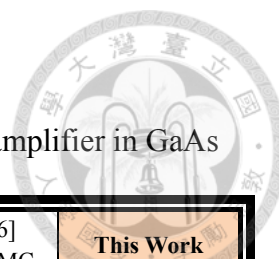
Table 3.4. Summary of the previously published Ka-band power amplifier in GaAs HEMT process.

| Reference | Ref.[33] 2001 SSICT | Ref.[34] 2005 APMC | Ref.[35] 2012 MTT | Ref. [36] 2006 EUMC | **This Work** |
|---|---|---|---|---|---|
| Process | 0.25-μm GaAs HEMT | 0.15-μm GaAs HEMT | 0.15-μm GaAs HEMT | 0.15-μm GaAs HEMT | 0.1-μm GaAs HEMT |
| Topology | 2-Stage Balanced | Distributed | 1-Stage Single-ended | 4-Stage Single-Ended | 2-Stage Single-Ended |
| 3-dB Frequency (GHz) | 17-36 | 4-37 | 17-35 | 35-42 | **28-42** |
| Gain (dB) | 11-14 | 12-15 | 9-12 | 26-28 | **16-18*** |
| dc Supply (V) | 5 | 5 | 4 | 5 | **4** |
| $P_{sat}$ (dBm) | 23-24 | 20-23 | 22.5-23.5 | 26 | **19-22*** |
| $OP_{1dB}$ (dBm) | 21.5-23 | 19-21 | 21-22 | 24 | **16-18*** |
| $PAE_{@Peak}$ (%) | 22-35 | 15 | 30-40 | 14 | **22-35*** |
| Chip Size ($mm^2$) | 4.32 | 3 | 1.5 | 3.5 | **1.5** |

**\*Measurement results in the 3-dB frequency range.**

# Chapter 4    Conclusion and Future Work
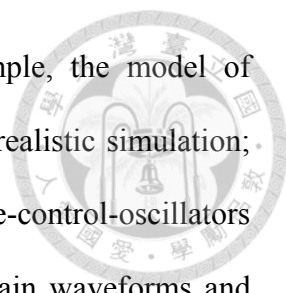
A novel evolutionary algorithm for microwave circuit synthesis has been presented, which combines advantages of linkage learning from GA and exploration ability from ES with special niching method. Due to the specific design of proposed algorithm for matching network synthesis, the possibility of immature convergence is reduced and the optimum matching network can be found rapidly.

Several experiments have validated the performance of the proposed algorithm. The statistic results show that our algorithm outperforms the previously published works, since it can fast find the optimum solution with fewer computation efforts. Besides, the proposed algorithm can simultaneously find many sub-optimal circuits with different topology, which makes the design flexible in choosing the desired circuit architecture.

Two broadband amplifiers have been designed and implemented to demonstrate the proposed algorithm. The 110-180 GHz amplifier is the first amplifier covering full D-band in 65-nm CMOS process, and an optimization method to determine the required number of cascade stage for specific target has been presented. The Ka-band 0.1-μm GaAs pHEMT power amplifier performs broadband gain performance, return loss, output power and PAE, due to the synthesized broadband matching networks. It is concluded that the proposed algorithm indeed help microwave-circuit engineer to fast design the optimal matching networks in a few seconds, and indeed makes the circuit performance better and desirable.

In future work, the design engineers can synthesize the specific circuit frequency response by integrating the correspondent numerical simulation methods in the
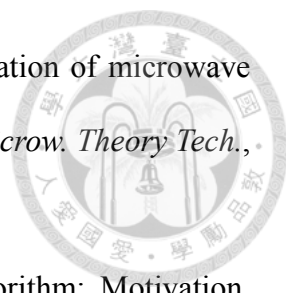
proposed algorithm or modifying the fitness function. For example, the model of passive components can include metal and substrate loss for more realistic simulation; the optimizations of large-signal frequency response for voltage-control-oscillators (VCOs) and mixers can be conducted by evaluating the time-domain waveforms and frequency-domain spectrum. Further, the programmer can transform the parameters of transistor, such as device size and bias conditions, to the genes of chromosome for extending the function of circuit synthesis. The layout consideration such as chip size and coupling effect, or the minimally required order of matching network can be implemented in the fitness function with the engineer experiences.
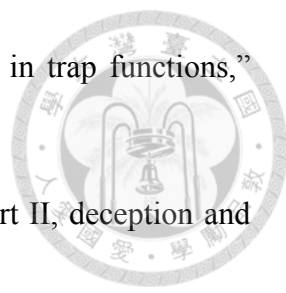
Therefore, the proposed algorithm can be the kernel of circuit synthesis, and the design engineers can slightly modify some procedure for general-purpose applications.

# REFERENCE

[1] J. Franco, "The brick wall: NP-completeness," *IEEE Potential*, vol. 14, no. 4, pp. 37-40, Oct. 1997.

[2] S. S. Rao, *Engineering Optimization Theory and Practice*, 2nd ed., Hoboken, NJ: John Wiley & Sons, Inc., 2009.

[3] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.

[4] D. E. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[5] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization*. New York: Wiley, 2000.

[6] A. John and R. H. Jansen, "Evolutionary generation of (M)MIC component shapes using 2.5D EM simulation and discrete genetic optimization," *IEEE MTT-S Int. Microwave Symp. Dig.*, 1996, pp. 745-748.

[7] T. Nishino and T. Itoh, "Evolutionary generation of microwave line segment circuits by genetic algorithms," *IEEE Trans. Microw. Theory Tech.*, vol. 50, no. 9, pp. 2048-2055, Sep. 2002.

[8] T. Nishino and T. Itoh, "Evolutionary generation of 3-D line-segment circuits with a broadside-coupled multiconductor transmission-line model," *IEEE Trans. Microw. Theory Tech.*, vol. 51, no. 10, pp. 2045-2054, Oct. 2003.

[9] M.-I. Lai and S.-K. Jeng, "Compact microstrip dual-band bandpass filters design using genetic-algorithm techniques," *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 1, pp. 160-168, Jan. 2006.

[10] Y. A. Hussein and S. M. El-Ghazaly, "Modeling and optimization of microwave devices and circuits using genetic algorithms," *IEEE Trans. Microw. Theory Tech.*, vol. 52, no. 1, pp. 329-336, Jan. 2004.

[11] D. E. Goldberg, B. Korb and K. Deb, "Messy genetic algorithm: Motivation, analysis, and first results," *Complex Systems*, vol. 3, pp. 493-530, 1989.

[12] G. R. Harik, F. G. Lobo and K. Sastry, "Linkage learning via probabilistic modeling in the extended compact genetic algorithm (ECGA)," *Studies in Computational Intell.*, vol. 33, pp. 39-61, 2006

[13] T.-L. Yu, D. E. Goldberg *et al*, "Dependency structure matrix, genetic algorithms, and effective recombination," *IEEE Evol. Computation Congr.*, vol. 17, no. 4, pp. 595-626, Dec. 2009.

[14] M. Pelikan, D. E. Goldberg *et al*, "BOA: the Bayesian optimization algorithm," *Proc. of the Genetic and Evol. Computation Conf.*, I:525-532, 2009.

[15] L. Fossati, P. L. Lanzi, K. Sastry, D. E. Goldberg and O. Gomez, "A simple real-coded extended compact genetic algorithm," *IEEE Evol. Computation Congr.*, pp. 342-248, Set. 2007.

[16] I. Rechenberg, *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Formmann-Holzboog, .Stuttgart, 1973.

[17] M. Dianati, I. S. and M. T., "An introduction to genetic algorithm and evolution strategies," Computer engineering, Univ. of Waterloo, Ontario, Canada.

[18] D. H. Wolper, W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Computation*, vol. 1, no. 1, Apr. 1997.

[19] P.-C. Hung and Y.-P. Chen, "iECGA: Integer extended compact genetic algorithm," Dept. Elect. Eng., National Chiao Tung Univ, Taiwan, no. NCL-TR-2006005, Feb. 2006.

[20] D. E. Goldberg, B. Korb and K. Deb, "Analyzing deception in trap functions," *Found. of Genetic Algorithms 2*, pp. 93-108, 1993.

[21] D. E. Goldberg, "Genetic algorithms and Walsh functions: Part II, deception and its analysis," *Complex System*, vol. 3, pp. 153-171, 1989.

[22] Mitchell, T. *Machine Learning*. McGraw Hill, New York, 1997.

[23] Y.-P. Chen and C.-H. Chen, "Enabling the extended compact genetic algorithm for real-parameter optimization by using adaptive discretization," *IEEE Evol. Computation Congr.*, vol. 18, no. 2, pp. 199-228, Apr. 2010.

[24] S. Tsutsui, M. Pelikan and D. E. Goldberg, "Evolutionary algorithm using marginal histogram models in continuous domain," Illinois Genetic Algorithms Laboratory, Univ. of Illinois at Urbana-Champaign, Report No. 2001019.

[25] C. E. Shannon, "A mathematical theory of communication," *The Bell System Tech. J.*, vol. 27, pp. 379-423, Jul. 1948.

[26] Harik G., "Finding multimodal solutions using restricted tournament selection," Illinois Genetic Algorithms Laboratory, Univ. of Illinois at Urbana-Champaign.

[27] M. J. Quinn, *Parallel Programming in C with MPI and OpenMP*. New York: McGraw-Hill, 2003, ch. 5.

[28] S. T. Nicolson *et al*., "A 1.2V, 140GHz receiver with on-die antenna in 65nm CMOS," in *IEEE Radio Frequency Integrated Circuits Symp.*, pp. 229-232, July 2008.

[29] Z. Xu *et al*., "A compact, fully differential D-band CMOS amplifier in 65nm CMOS, " in *IEEE Asian Solid-State Circuits Conference*, Nov. 2010.

[30] M. Seo *et al*., "A 1.1V 150 GHz amplifier with 8dB gain and +6 dBm saturated output power in standard digital 65nm CMOS using dummy-prefilled microstrip lines," in *Int. Solid-State Circuits Conference Dig. Tech. Paper*, pp.484-485, Feb.

2009.

[31]  Z.-M. Tsai *et al.*, "A 1.2V broadband D-band power amplifier with 13.2-dBm output power in standard RF 65-nm CMOS," *IEEE MTT-S Int. Microw. Symp.*, Jun. 2012.

[32]  E. Laskin *et al.*, "170-GHz transceiver with on-chip antennas in SiGe technology," in *IEEE Radio Frequency Integrated Circuits Symp.*, pp 637-640, Jul. 2008.

[33]  Y. C. Lee and C. S. Pard, "17–36 GHz broadband PHEMT MMIC power amplifier for point-to-multipoint applications," in *Pro. Int. Conf. Solid-State and Integrated Circuits Tech.*, vol. 2, pp. 1320–1323, 2001.

[34]  M.-C. Chuang, M.-F. Lei, and H. Wang, "A broadband medium power amplifier for millimeter-wave applications," in *IEEE Asia-Pacific Microw. Conf., Proc.*, vol. 3, pp. 1593–1595, 2005.

[35]  P.-C. Huang, Z.-M. Tsai, K.-Y. Lin and H. Wang, "A 17–35 GHz broadband, high efficiency PHEMT power amplifier using synthesized transformer matching technique" in *IEEE Trans. Microw. Theory Tech.*, vol. 60, no. 1, pp. 112-119, Jan. 2012.

[36]  A. Bessemoulin, S. J. Mahon, J. T. Harvey, and D. Richardson, "Compact K-band watt-level GaAs PHEMT power amplifier MMIC with integrated ESD protection," in *IEEE Eur. Microw. Conf. Dig.*, pp. 1743–1746, Jun. 2006.

[37]  T. H. Cormen *et al.*, *Introduction to Algorithms*, 3rd ed., Cambridge MA: MIT Press, 2009.

[38]  G. Gonzales, *Microwave Transistor Amplifier Analysis and Design*, 2nd ed., Taipei Taiwan: Pearson Education Taiwan Ltd., 2010.

[39]  D. E. Goldberg and K. Deb, "A comparative analysis of selection schemes used in genetic algorithms," Illinois Genetic Algorithms Laboratory, Univ. of Illinois at

Urbana-Champaign.

[40] D. B. Thomas and W. Luk, "Gaussian random number generators," *ACM Computing Surveys*, vol. 39, no. 4, article 11, Oct. 2007.

[41] R. W. Hamming, "Error detecting and error correcting codes," *Bell System Tech. J.*, vol. 29, no. 2, pp. 147–160, 1950.

[42] Data sheet of packed transistor NE-42484C. [Online]. Available: http://www.allpdf.net/datasheet/fet/fet_rf_l/NE4/NE42484C_1.pdf.

[43] Data sheet and design kits of packed transistor ATF-58143. [Online]. Available: http://www.avagotech.com/pages/en/rf_microwave/transistors/fet/atf-58143/.

[44] "*Sonnet Users Guide, Release 12,*" Sonnet Software Inc., North Syracus, NY, 2009.

[45] I. Angelov, H. Zirath, N. Rorsmann, "A new empirical nonlinear model for HEMT and MESFET devices," in *IEEE Trans. Microw. Theory Tech.*, vol. 12, Dec. 1992.

[46] I. Angelov, L. Bengtsson, M. Garcia, "Extensions of the Chalmers nonlinear HEMT and MESFET model," in *IEEE Trans. Microw. Theory Tech.*, vol. 44, No. 10, Oct. 1996.