Genet Program Evolvable Mach (2010) 11:35Ð59
DOI 10.1007/s10710-009-9085-2

ORIGINAL PAPER

Automated synthesis of resilient and tamper-evident analog circuits without a single point of failure

Kyung-Joong Kim Æ Adrian Wong Æ Hod Lipson

Received: 13 August 2008 / Revised: 10 March 2009 / Published online: 26 April 2009

Abstract This study focuses on the use of genetic programming to automate the design of robust analog circuits. We deÞne two complementary types of failure modes: partial short-circuit and partial disconnect, and demonstrated novel circuits that are resilient across a spectrum of fault levels. In particular, we focus on designs that are uniformly robust, and unlike designs based on redundancy, do not have any single point of failure. We also explore the complementary problem of designing tamper-proof circuits that are highly sensitive to any change or variation in their operating conditions. We Þnd that the number of components remains similar both for robust and standard circuits, suggesting that the robustness does not necessarily come at signiÞcant increased circuit complexity. A number of Þtness criteria,

K.-J. Kim
Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853, USA
e-mail: KK499@cornell.edu

Present Address:
K.-J. Kim
Department of Computer Engineering, Sejong University, 98 Gunja-Dong, Gwangjin-Gu, Seoul 143-747, Republic of Korea
e-mail: kimkj@sejong.ac.kr

A. Wong
Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853, USA

Present Address:
A. Wong
Sandia National Laboratories, Livermore, CA 94550, USA
e-mail: awong@sandia.gov

H. Lipson (✉)
Computing and Information Science, Cornell University, 216 Upson Hall, Ithaca, NY 14853-7501, USA
e-mail: Hod.lipson@cornell.edu; HL274@cornell.edu

including surrogate models and co-evolution were used to accelerate the evolutionary process. A variety of circuit types were tested, and the practicality of the generated solutions was verified by physically constructing the circuits and testing their physical robustness.

Keywords Analog circuit · Robustness · Evolutionary strategies · Low-pass filter · Hardware implementation · Tamper-evident circuits

1 Introduction

The design of analog circuits is known to be a challenging problem, as the continuous transient nature and frequency response make logical deduction unintuitive. It is not surprising that evolutionary algorithms have been particularly useful for this task, resulting in numerous successful implementations (Table 1). Evolutionary algorithms search for an appropriate topology, component types, and the value of components starting with random initial candidates and progressing through a series of genetic variations using a Darwinian selection process.

A particularly challenging task in analog circuit synthesis is the design of fault tolerant circuits. Traditionally, fault tolerance is considered as an afterthought either by externally protecting the circuit or by duplicating circuit modules to form redundant subsystems that are combined through a voting mechanism. That approach, however, makes the demultiplexing point itself a single point of failure. Another approach to resilience is making adaptive circuits whose parameters can adjust (often evolve) in situ to compensate for failure in real time. In that case, the adaptation mechanism itself becomes a single point of failure, since a fault in that circuit might modify the original circuit arbitrarily. An alternative approach is to design circuits that are inherently robust.

In this paper we consider the problem of synthesizing circuits that are designed a-priori to be robust so that a failure in *any* component would lead to minimal performance degradation. Such circuits have no single point of failure, yet present an even more challenging design task. A number of studies exploring this approach focus on different types of defects in analog circuits such as component removal [1], parameter variations [2], and external environment changes [3].

In this paper we focus on a spectrum of faults that are equivalent to adding a fault-emulating resistor in series or in parallel with any component in the circuit. The degree of damage can then be adjusted by increasing or decreasing that resistance. The fitness criterion for evolving such circuits is to maximize the worst-case performance among all possible placements and values of the fault-emulating resistor. We examine how the topology of the circuit evolves to accommodate these types of failure possibilities.

We also examine the complementary problem of designing tamper-evident circuits that maximize performance degradation subject to the smallest perturbation, such as connecting a voltmeter across one of their components. The criterion for evolving such circuits is to minimize the best-case performance among all possible circuit changes, while maintaining good performance of the intact circuit.

Table 1 Summary of evolving analog circuit research

Authors	Type	Tasks
Koza et al. [9]	GP	Low-pass Þlter, crossover Þlter, source identiÞcation, ampliÞer, computational circuit, time-optimal controller circuit, temperature-sensing circuit, and voltage reference circuit
Koza et al. [13]	GP	Balun circuit, voltage-current conversion circuit, cubic signal generator, register-controlled variable capacitor, and high-current load circuit
Hu et al. [2]	GP	Low-pass Þlter, and high-pass Þlter
Wang et al. [14]	GP	Voltage ampliÞer and low-pass Þlter
Sripramong et al. [15]	GP	CMOS ampliÞer
Ciccazzo et al. [6]	IP	Low-pass Þlter
Goh et al. [10]	GA	Low-pass Þlter
Hollinger et al [1]	GA	Robot controller
Zebulum et al. [7]	GA	Control systems
Keymeulen et al. [16]	GA	Multiplier
Zebulum et al. [3]	GA	Half-wave rectiÞer, NOR gate, and oscillator
Lohn et al. [17]	GA	Stethoscope circuit, and butterworth low-pass Þlter
Layzell et al. [18]	GA	Inverter, ampliÞer, and oscillator
Natsui et al. [19]	GA	nMOS current mirror
Dastidar et al. [20]	GA	Comparator, oscillator, and XOR gate
Ando et al. [21]	GA	Band elimination Þlter, asymmetric bandpass Þlter, and low-pass Þlter
Mattiussi et al. [22]	GA	Voltage reference, temperature sensor, and gaussian function generator
Xia et al. [23]	GA	Voltage ampliÞer, and low-pass Þlter
Grimbleby [24]	GA	Low-pass Þlter, and asymmetric bandpass Þlter
Berenson et al. [25]	GA	Neural network controller
Sapargaliyev et al. [26]	ES	Low-pass Þlter
Biondi et al. [27]	MOEA	Operational transconductance ampliÞer, and Þfth-order leapfrog Þlter
Nicosia et al. [28]	MOEA	Leapfrog Þlter for W-LAN, low noise ampliÞer for DVBS, and low noise ampliÞer for W-LAN
Zinchenko et al. [29]	EDA	Low-pass Þlter

IP = immune programming

Fitness evaluation for robustness is especially expensive as many variations of the circuit need to be simulated to determine its worst-case or average performance under a range of faults. In this work we consider faults as a distribution rather a discrete event (such as a removal of a component). We then reduce the computational cost by creating a surrogate Þtness model that samples the distribution [4]. A number of static, dynamic and co-evolving sampling schemes were considered and compared, showing a co-evolutionary approach may prove to be the most efÞcient. Finally, like alpha-beta pruning in a game tree, if there is no possibility that an individual could survive in the next generation, its evaluations are skipped.

We tested the proposed method on the design robust low-pass Þlters and the Þnal results were compared to circuits evolved without robustness consideration, standard low-pass Þlters manually designed and other circuits evolved by genetic programming. We also veriÞed the practicality of the resulting circuits by physically constructing them and testing their performance under fault. It is interesting to note that though many circuits have been evolved in the literature, none of the simulation-based studies have actually built and tested the resulting circuits in practice.

This paper is organized as follows: the background section describes the current status of evolving analog circuits. The problem statement section provides a deÞnition of robustness in the presence of partial short and disconnection damages. The method section provides details of the algorithms and heuristics are explained. The results and discussion sections show a variety of results on the low-pass Þlter evolution in terms of robustness and computational cost and compare their performance to other methods. The paper concludes by testing the evolved circuits in reality.

2 Background

2.1 Evolving analog circuits

There are a number studies examining the evolution of analog circuits, and some deal with robustness issues. Table 1 summarizes previous works on the topic with the type of evolution and the tasks evolved. GP and GA are dominant methodologies in this area, though a number of other methodologies have been used. The wide range of evolved circuits shows the promising aspect of the evolutionary electronics. Some of these results are human-competitive [5]. It includes Þlter, computational circuit, robot controller and digital component which can be used for further complex digital circuit evolution [4].

2.2 Evolving fault-tolerant circuits

There are a relatively small number of attempts to evolve robust or fault tolerant analog circuits (Table 2). Faults considered are both internal (manufacturing error, aging, short and disconnect) or external, (environment temperature, actuation error, and environmental noise). The internal failure is simulated by deleting one component at each time, changing the parameter values of component, and switching connections. The computational cost of evaluation increases signiÞcantly because multiple simulations are required, proportional to the number of components in the circuit multiplied by the number of failure modes per component. SpeciÞc application areas that have been targeted are robots with a noisy environment or actuation error [1] and analog circuit working in extreme environment like space [3].

Table 2 Summary of evolving robust analog circuit research

Authors	Internal failure	External failure
Hollinger et al [1]	One component removal	ModiÞcation of plant transfer function
Zebulum et al. [7]	One component removal	Power dissipation, intrinsic noise
Hu et al. [2]	Component parameter variation	
Nicosia et al. [28]	Component parameter variation	
Keymeulen et al. [16]	Open/close switches	
Zebulum et al. [3]		Extreme low temperature
Layzell et al. [18]	One transistor removal	
Ando et al. [21]	Component parameter variation	
This paper	One component partial short one component partial disconnection	

3 Problem statement

3.1 DeÞnition of robustness

Analog circuit design is composed of three steps [6]: (a) structure (b) sizing (c) layout. In the structure stage, an experienced designer chooses appropriate topology for a speciÞed functionality. The goal of the sizing step is to Þnd parameter values of components in the topology. Initially ideal parameters are considered, while ignoring tolerance considerations. After the ideal design, yield optimization is performed to Þnd circuits that are robust to manufacturing and operational variations. In the Þnal layout step, circuit board embedding is planned in consideration of manufacturability and yield. Worst case analysis, yield analysis, statistical yield analysis (Monte Carlo method), and geometric yield analysis are typically used in these processes.

In previous studies (e.g. [1], [7]), robustness is deÞned as the average or worst case performance of a circuit after deleting one component at a time from the circuit. However, circuit failures are often more subtle in nature. Here we consider partial short and disconnection damages to each component. The two damages are simulated with a resistor and the degree of damages is controlled by changing the value of the resistor (Fig. 1). At an extreme, these failures correspond to removal of a component, but allow for more realistic partial degradation as well. While this damage representation is not universal it does cover a large range of faults. More elaborate, component-speciÞc failure modes could be considered in the future.

Figure 1 shows an example of the damages controlled by a resistor. In partial short damage, the resistor is attached to the component in parallel. If the resistorÕs value is zero, the component is completely short. On the other hand, there is no damage if the resistorÕs value is inÞnite. By adjusting the value of the resistor from zero to inÞnite, the degree of damage is controllable. In partial disconnection case, the resistor is attached to the component in serial manner. Inversely, it is completely disconnected if the resistor has inÞnite value because the current cannot go through the component. If the value is zero, there is no effect on the component. Similarly,

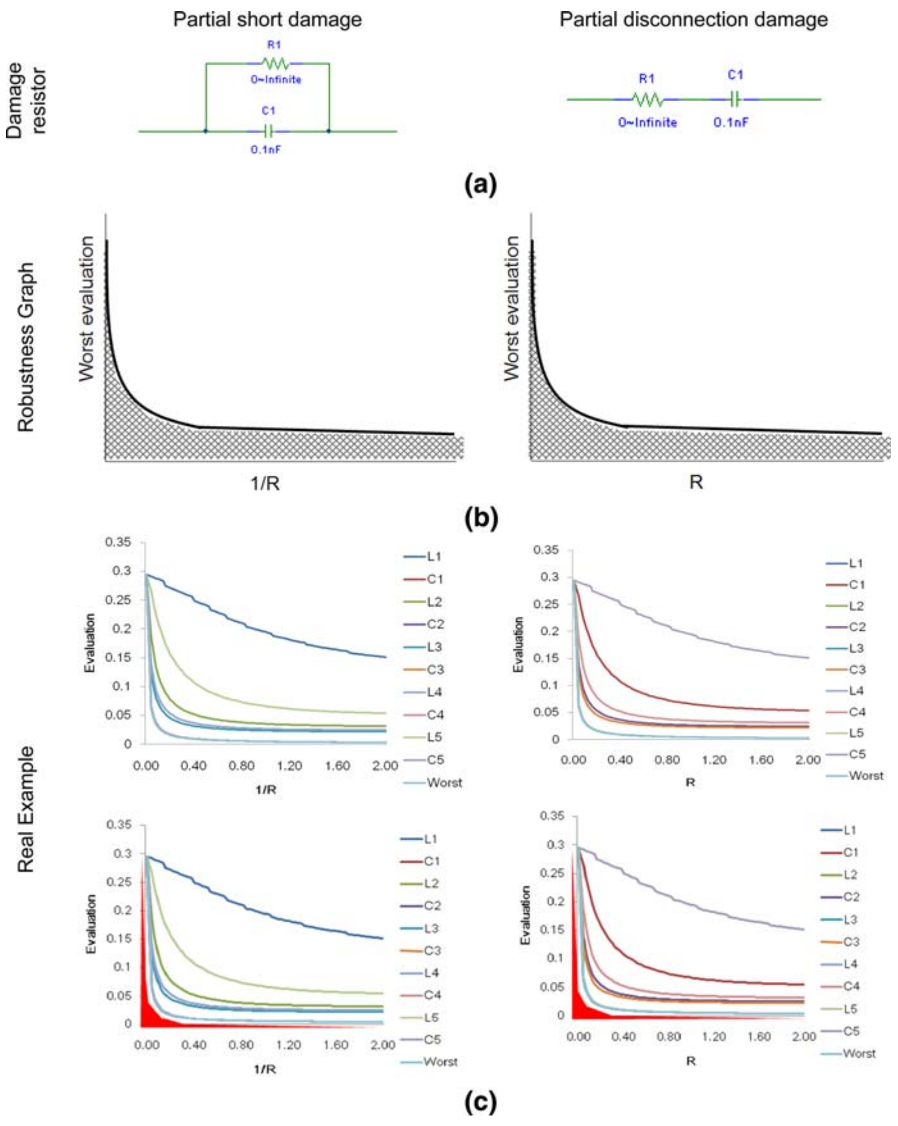


Fig. 1 Two types of damages and corresponding robustness graphs. a Parallel resistor emulates a shorted component, serial resistor emulates a disconnection, b robustness is deÞned as the shaded area below the worst case graph. c Examples for 10th order Butterworth low-pass Þlter

the effect of the disconnection damage to the component can be adjustable by the value of resistor.

We estimate the performance of a damaged circuit by adding damage-resistors in parallel and series to each component. For any given damage resistor value, we scan across all components and determine the worst case performance for that value. Note that an increase of the damage-resistor value results in emulating a more severe disconnects damage or a less-severe short-circuits damage. We can then plot the worst case performance of the circuit across the full range of the damage-resistor. The robustness is deÞned as the area below the curve (the cross-hatched area in Fig. 1).

3.2 Evolving robust analog circuits

The circuit topology and parameters were evolved using only mutations. Initially, P parents are generated using an embryonic circuit and each produces one offspring by one of the mutations. The next step is circuit simpliÞcation that combines identical components in a serial or parallel conÞguration into a single component. This prevents the circuit from gaining robustness simply by replacing one component with multiple components in series or parallel conÞguration, although that is a valid but trivial approach to making more robust circuits. We then use a circuit simulator [3] to evaluate each circuitÕs output response.

The robustness evaluation procedure is the most computationally expensive and some techniques are proposed to increase efÞciency. Finally, the P best individuals are selected from $2 \times P$ circuits (parents plus offspring). The algorithm terminates when the number of generations is larger than the maximum predeÞned. Figure 2 summarizes this algorithm.

3.2.1 Initialization

An embryonic circuit is a template to generate initial random circuits. It deÞnes the voltage source, load resistor, source resistor, ground, and a probing point. Figure 3a show an example of the embryonic circuit for low-pass Þlter evolution. It contains a 2 V AC voltage source, 1-K source resistor ($R1$), 1-K load resistor ($R2$), ground and a probing point. Initially, the dotted empty box is replaced with one new component whose type and values are randomly chosen.

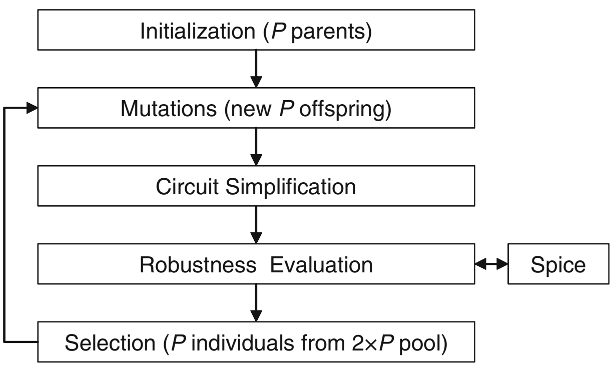


Fig. 2 Overview of the algorithm

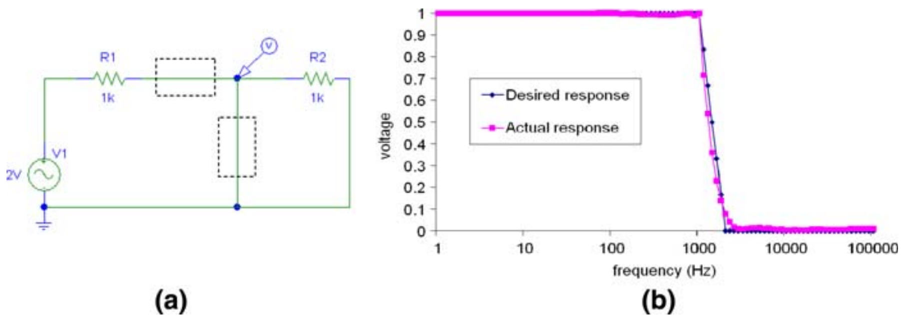


Fig. 3 Evolution starting point and goal. a The embryonic circuit and b the desired response for a low-pass Þlter

3.2.2 Mutations

For each circuit, one component is randomly selected except source resistor, load resistor and voltage source. Subsequently, one of eight different mutations is randomly chosen and applied to the component.

1. Parameter change: the componentÕs value is assigned as a new randomly chosen value.
2. Type change: the component type is swapped to a different one randomly.
3. Parallel addition of a different type component: a new component (with a different type) is added in parallel conÞguration to the component. The type and value of the new component is randomly chosen.
4. Serial addition of a different type component: same as above accept the addition in serial conÞguration.
5. Component deletion: the component is removed from the circuit.
6. Ground setting: the component is connected to the ground.
7. Replacement: the component is replaced with a new component (possibly of the same type).
8. Adding a component: a new component bridges between two randomly chosen wires (not identical wire).

3.2.3 Robustness evaluation

A circuit is evaluated by the difference between actual and desired responses. Robustness is deÞned as the integral of the worst case by damage over all resistor values. N is the number of components in a circuit excluding source and load resistors.[1]

[1] The source and load resistors in the embryonic circuit are 1-K and the incoming 2 V signal is divided in half. From this, it is possible to assume that the optimal output response in low frequency ($f < 1$ kHz) is 1 V. If their values are changing from damage, the optimal output response has to be changed. This results in the change of the ÞlterÕs original speciÞcation. We assume that the two resistors are tamper-proofed. Also the change of the input voltage source is not considered.

$f(c, R)$ returns an evaluation value when the component is damaged by a resistor R. Φ returns a worst case.

$$\text{Robustness} = \int_{R=0}^{\infty} \Phi(f(c_1, R), \ldots, f(c_N, R))$$

It is impractical to calculate a worst case across all damage-resistor values due to the computational cost associated with simulated each candidate circuit $O(Ns)$ times where N represents the number of components and s represents the number of samplings of damage values. Robustness is approximated based on a worst case for a small value of s, with the location of the samples selected strategically. Table 3 summarizes a variety of proposed strategies to approximate the robustness graph and their Þtness function used in evolution. In the randomized strategy, the resistor values used change randomly across generations.

We used an evolutionary strategy for evaluations, where at each generation the entire population of size M is used to generate new set of M offspring, then the resulting $2M$ set is ranked and the top M selected as the new population. It is not necessary to re-evaluate parents from previous generation if the Þtness is deterministic (unchanging across generation). Similarly, if one of the Ns evaluations of a candidate circuit falls below the worst parent, the remaining evaluations for that circuit can be aborted. This assumption cannot be guaranteed for stochastic sampling.

We also studied the use of coevolution to dynamically determine the two sampling points for assessing partial short damage. The initial two sampling points were chosen randomly. After every 500 generation, an evolutionary algorithm searched for new two sampling points based on their accuracy in prediction of robustness [4]. If R_i is a ranking of ith circuit sorted by a full robustness calculation (sum of worst Þtness over 101 points from 0.0 to 2.0) and $E(R_i)$ a ranking of ith circuit sorted by estimated robustness with two sampling points, then the Þtness of two sampling points was their predictive ability, i.e., the correlation between the true ranking and the predicted ranking. The predictive ability was estimated as the sum over the population of $[R_i - E(R_i)]^2$. The co-evolutionary approach outperformed the best robust circuit with two sampling points (0.0, 2.0). It took only 1,000 generations to get the same robustness with the best one evolved for 10,000 generations (Fig. 4).

4 Experimental results

A low-pass Þltering is a widely used test task in evolutionary analog circuit research. The merit of a Þlter circuit is evaluated based on the difference between actual and desired frequency responses. The difference is summed over 101 sampling points ranged from 1 Hz to 100 kHz. A ÔÔDonÕt careÕÕ band (from 1 to 2 kHz) is ignored in the calculation. The evaluation f is deÞned as follows.

$$f = \frac{1.0}{\sum_{i=1}^{101} |\text{Error}_i| \times C} \quad C = \begin{cases} 1, & |\text{Error}| \leq 0.01 \\ 10, & |\text{Error}| > 0.01 \end{cases}$$

Table 3 Sampling-based Þtness approximations

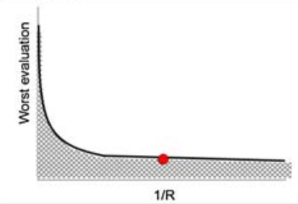
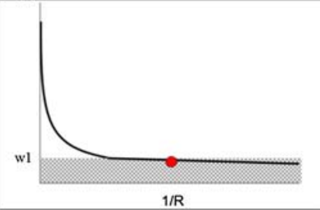
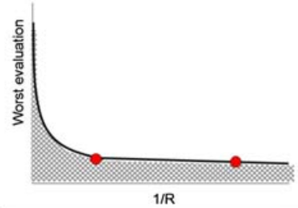
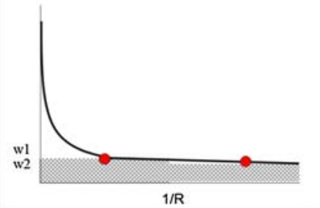
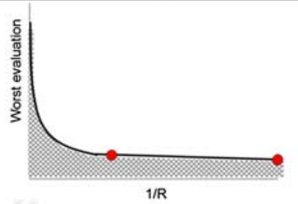
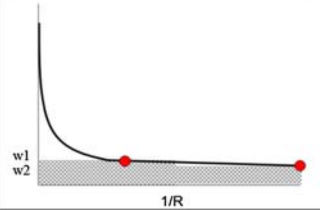
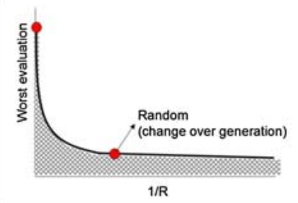
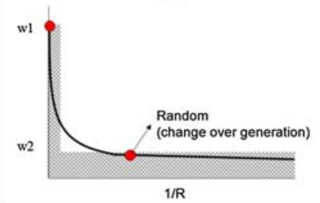
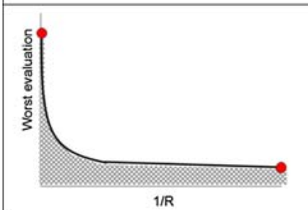
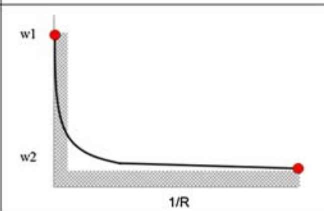
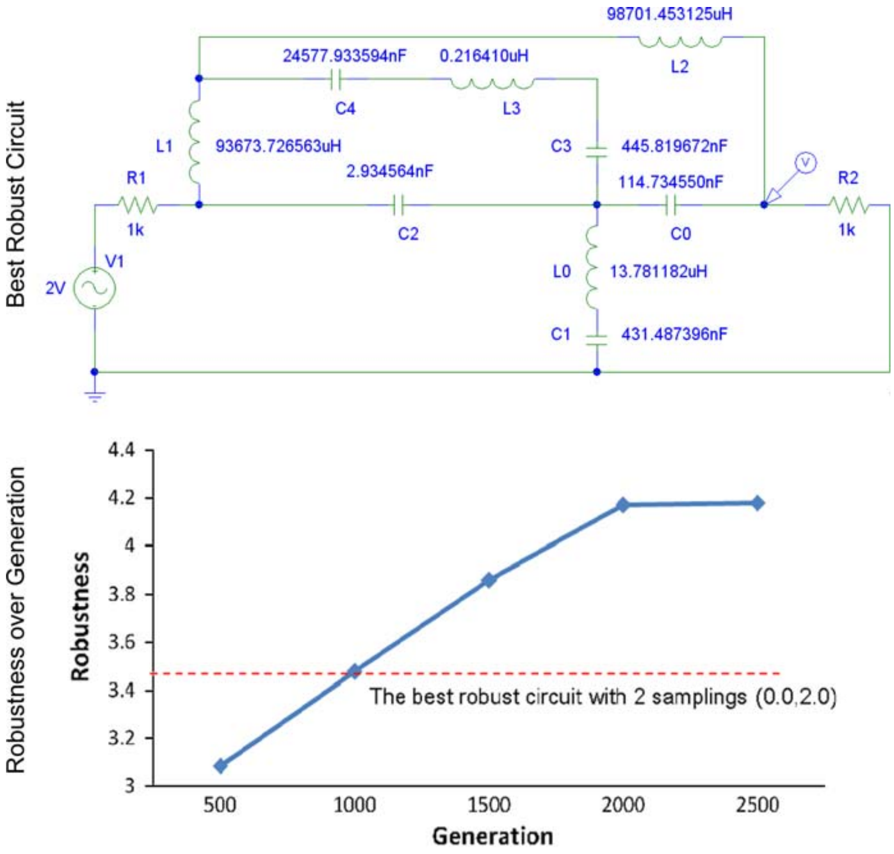
Samplings	Approximation	Fitness Function	Points
		*w*1	½
		*w*1+*w*2	¼, ¾
		*w*1+*w*2	⅓, Max
		*w*1×*w*2	Min, Random
		*w*1×*w*2	Min, Max

Table 4 summarizes parameters used in this experiment. Node is deÞned as a point on a circuit where two or more components meet. The number of node is limited to prevent circuit from being complex. The experiments run Þve times.

Figure 5 shows the robustness graph for normal and robustness evolution with various sampling approaches. For partial short damage, two samplings at 0.0 ($R = \infty$) and 2.0 ($R = 0.5$-K) showed the best robustness. Circuits from the normal evolution performed well when there is no damage but its performance radically



Sampling Points

Generation	Sampling Points	Corresponding Resisotrs
0	(0.834010, 0.556780)	(1.2K, 1.8K)
500	(0.834010, 0.556780)	(1.2K, 1.8K)
1000	(1.969115, 0.101932)	(0.5K, 9.8K)
1500	(0.919095, 0.086062)	(1.1K, 11.6K)
2000	(0.011231, 0.579180)	(89.0K, 1.7K)
2500	(0.533586, 0.375378)	(1.8K, 2.6K)

Fig. 4 Co-evolving faults. Co-evolution was used to determine the two sampling points for assessing partial short damage. The initial two sampling points were chosen randomly. After every 500 generation, evolutionary algorithm searches for new two sampling points based on their accuracy in prediction of robustness [30]. The co-evolutionary approach outperformed the best robust circuit obtained using the best fixed fitness criterion of two sampling points (0.0, 2.0)

degraded after the increase of damage. In partial disconnection damage, the best robustness was achieved from two samplings at 0.5 ($R = 0.5$-K) and 1.5 ($R = 1.5$-K). Although the circuits from normal evolution performed better than the two samplings in small damages, it changed around at 0.2-K. It shows that the best sampling strategy for different type of damage is varying. Table 5 summarizes statistics of results for the different strategies for five independent runs.

Table 4 Parameters

Evolution	Simulator	WinSpice
	Population size	20
	Mutation rate	100%
	Maximum generation	10,000
Circuit	Component type	Capacitor (C), inductor (L), resistor (R)
	Maximum node number	10
	Capacitor value range	1Ð10^5 nF
	Inductor value range	0.1Ð10^5 μH
Robustness	*R* range for partial short	0.5-KΩÐ
	R range for partial disconnection	0Ð2-K

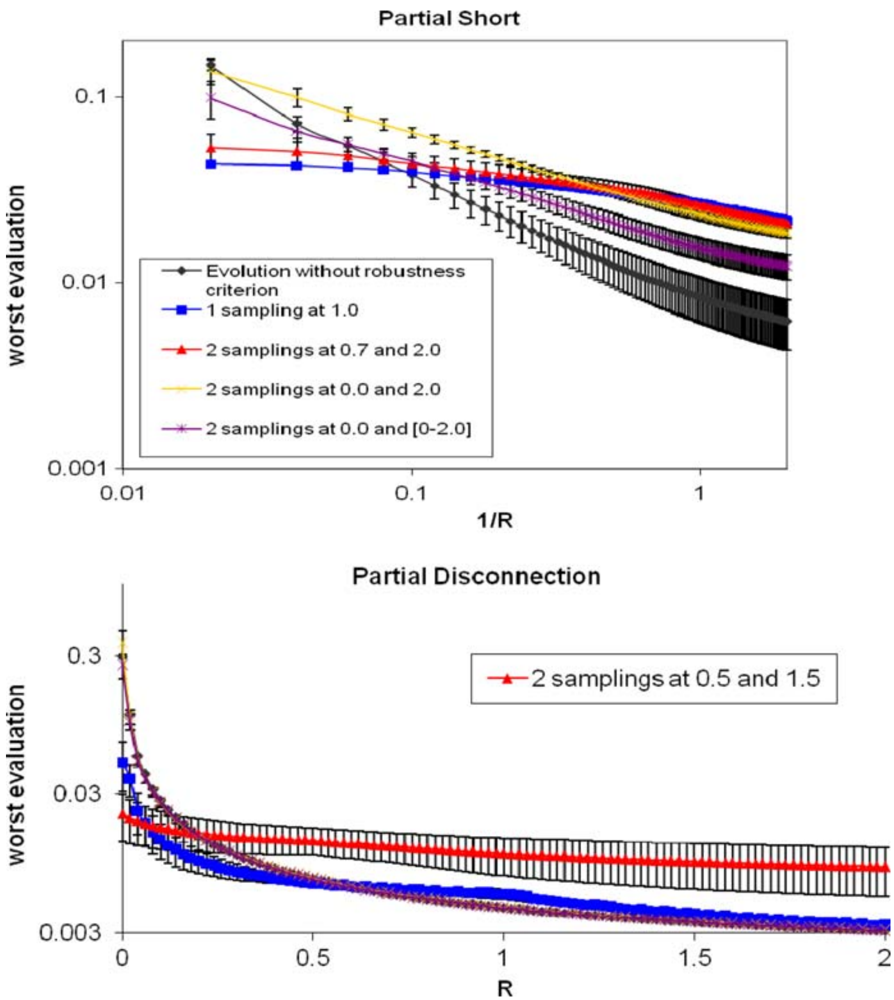


Fig. 5 Robustness graphs for two types of damages for Þve independent runs

Table 5 Results statistics

		Robustness[a]	Value range (minÐmax)	MaxÐmin
Short	Normal	1.63 ± 0.26	0.00 ± 0.00Ð0.30 ± 0.02	0.30
	1 sampling (1.0)	2.81 ± 0.11	0.02 ± 0.00Ð0.05 ± 0.00	0.03
	2 samplings (0.7, 2.0)	2.88 ± 0.22	0.02 ± 0.00Ð0.06 ± 0.01	0.04
	2 samplings (0.0, 2.0)	3.11 ± 0.13	0.02 ± 0.00Ð0.18 ± 0.04	0.16
	2 samplings [0.0, (0Ð2)]	2.14 ± 0.17	0.01 ± 0.00Ð0.16 ± 0.07	0.15
Disconnect	Normal	1.14 ± 0.02	0.00 ± 0.00Ð0.30 ± 0.02	0.30
	1 sampling (1.0)	0.71 ± 0.09	0.00 ± 0.00Ð0.05 ± 0.03	0.05
	2 samplings (0.5, 1.5)	1.23 ± 0.37	0.01 ± 0.00Ð0.02 ± 0.01	0.01
	2 samplings (0.0, 2.0)	1.20 ± 0.05	0.00 ± 0.00Ð0.37 ± 0.08	0.37
	2 samplings [0.0, (0Ð2)]	1.07 ± 0.08	0.00 ± 0.00Ð0.26 ± 0.05	0.26

[a] Robustness is summed over 101 points from 0.0 to 2.0 for Þve independent runs

4.1 Circuit analysis

Figure 6 shows circuit diagrams and output responses of the best circuits using standard and robust evolution. The number of components remains around 9 or 10 both for robust and standard circuits, suggesting that the robustness does not necessarily come at increased circuit complexity. Circuits are compact due to the simpliÞcation and the limit of node number. When there is no damage, both circuits evolved using standard evolution and robust evolution for partial short performed well and with the desired response. The circuit for disconnection damage showed imperfect output response.

When partial short damage was applied, the robust circuit maintained its original curve shape exhibiting a degradation of only 6.27% in area under the frequency response curve[2], but the circuit evolved using standard criteria lost its original function. When partial disconnection damage was applied, the circuit evolved using standard criteria showed severe degradation. However, the robust circuit maintained its original function well, exhibiting a degradation of only 3.52% in area under the frequency response curve. Although the robust circuit showed relatively low performance at the no damage situation, its degradation was relatively small.

A sensitivity analysis shows that the worst component changes over different damage-resistor values and sensitivity to damage of components are varying. In partial short damage, the worst component at R = 0.02 (50-K) is L0 but it changes to C3 between 0.04 (25-K) and 0.92 (1.0-K). After then, it returns to L0 again after 0.94 (1.1-K). In disconnection damage, the worst component from 0.2 to 0.42-K is L0 but it changes to C3 after 0.42-K. From the circuit diagram, the worst components (L0 and C3) are very important one to bridge between the source resistor (R1) and remaining circuits.

Figure 7 shows the comparison of robustness graph with standard low-pass Þlters (10th order Butterworth and Chebychev circuits) and evolved one from Koza et al [9]. It shows that the robust circuit showed better robustness than other known low-pass Þlters.

[2] The degradation ratio is deÞned as Current degradation / Maximum degradation.

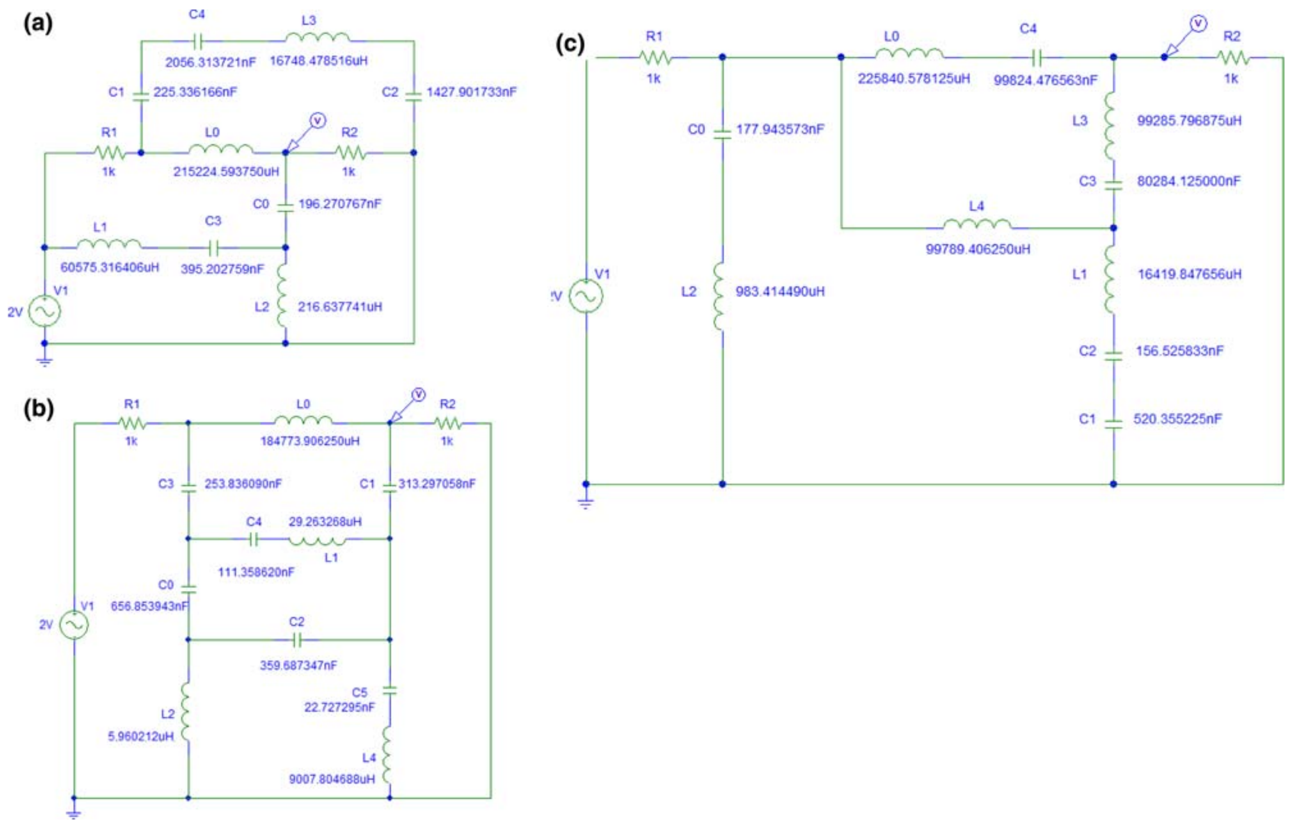


Fig. 6 Evolved circuits and their output responses. $R = 0.5$-K for partial short damage, $R = 2.0$-K for partial disconnection damage; a Evolution without robustness, b robust to partial short, c robust to partial disconnection

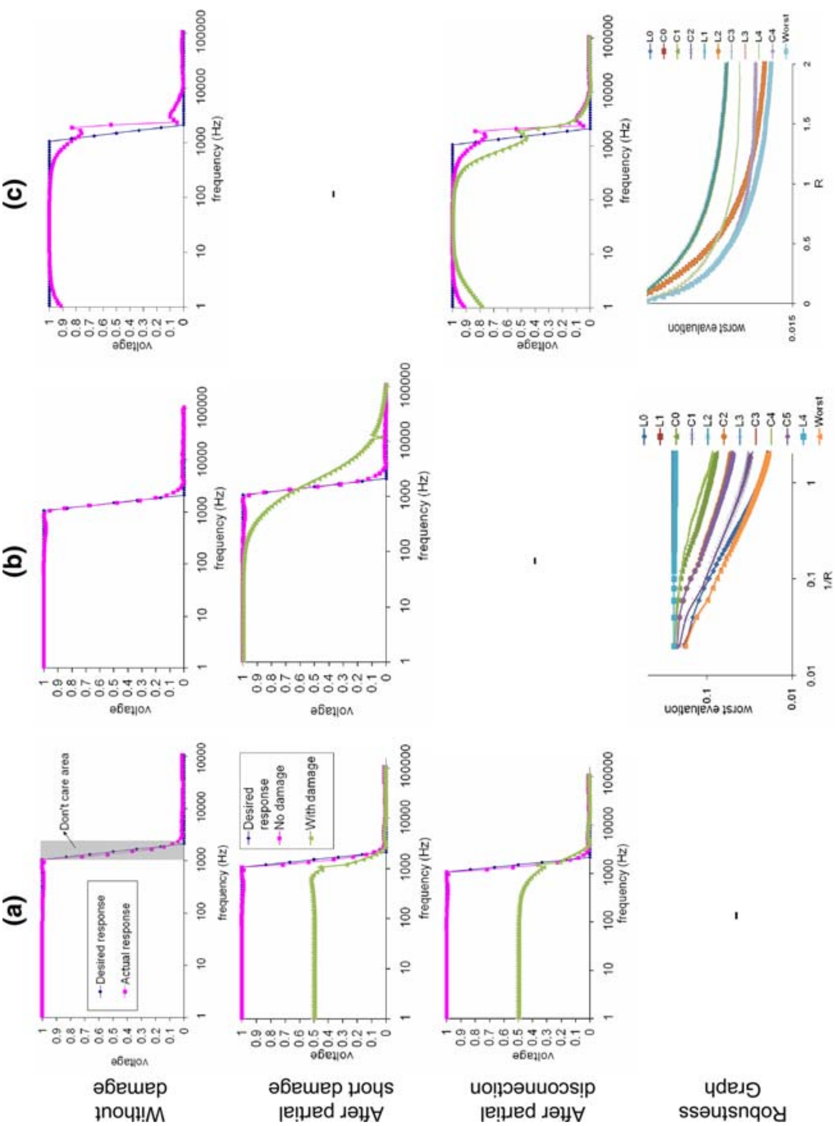


Fig. 6 continued

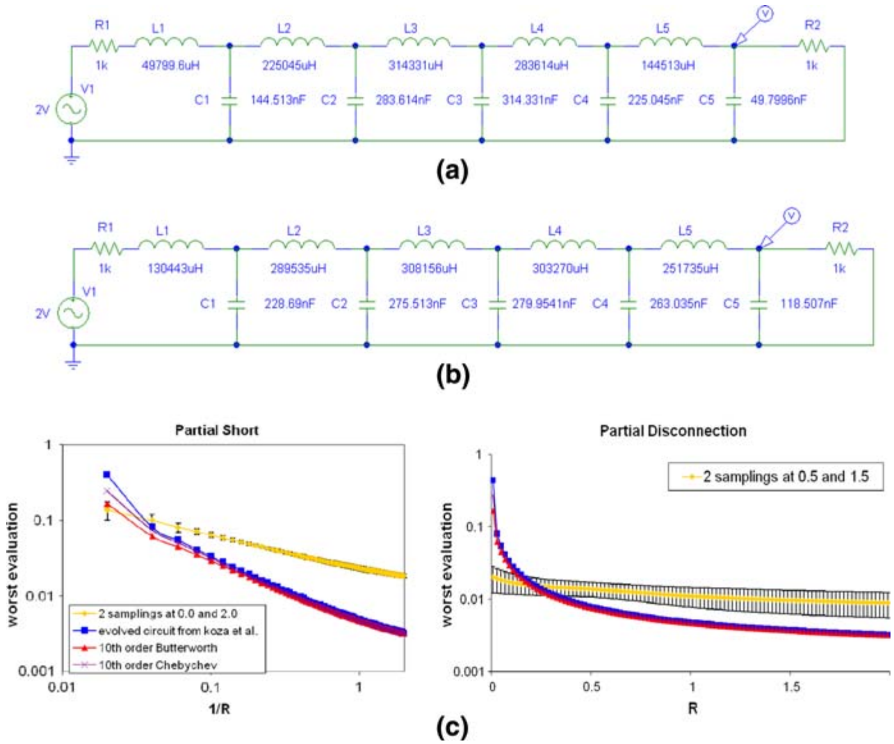


Fig. 7 Comparison with KozaÕs evolved circuit [9], 10th order Butterworth and 10th Order Chebychev low-pass Þlter [31]. a 10th order Butterworth circuit diagram, b 10th order Chebychev circuit diagram, c robustness graph

4.2 Computational cost analysis

Table 6 summarizes the number of evaluations carried out during the evolution process. It shows that the pruning unnecessary evaluation improves the efÞciency Þve times better than the one without pruning. Compared to the normal evolution,

Table 6 The number of evaluations required in the evolution (partial short damage)

	No. of evaluations without pruning (A)[a]	No. of evaluations with pruning (B)	EfÞciency (A/B)
Normal	2,00000 ± 0	2,00000 ± 0	1
1 sampling (1.0)	2,095740 ± 2,06470	4,15565 ± 3,9189	5.04
2 samplings (0.7, 2.0)	3,512199 ± 2,86618	6,46455 ± 6,5038	5.43
2 samplings (0.0, 2.0)	2,053189 ± 9,6299	5,19400 ± 3,0682	3.95
2 samplings [0.0, (0Ð2)]	3,530166 ± 3,11264	5,50790 ± 4,4639	6.40

[a] The time saving by ignoring the parent reevaluation is considered

the robustness evolution required two or three times more computational cost. In partial disconnection damage, the similar computational cost efÞciency was achieved.

4.3 Evolution of tamper-evident circuits

The inverse deÞnition of robustness can be used to evolve tamper-evident circuits which are super-sensitive to any modiÞcation or damage. If there is no damage, the circuit works well but its performance degrades with the introduction of any modiÞcation or inspection tools. This property is useful to design secure circuits performing an important task while avoiding reverse engineering or modiÞcation by tampers. A tamper-evident circuit satisÞes two conditions: (1) it shows acceptable performance in case of no modiÞcation; (2) its performance degrades signiÞcantly in the presence of a modiÞcation to any component.

The Þtness of a tamper evident circuit is the difference between its intact performance and the best performance under modiÞcation. Figure 8 shows the deÞnition of the tamper-evident property. Unlike a robustness graph, it depicts the upper performance of the circuit subject to damage or modiÞcation. The circuit with acceptable original performance is a better tamper-evident circuit than others if the lower area (shaded area) is small.

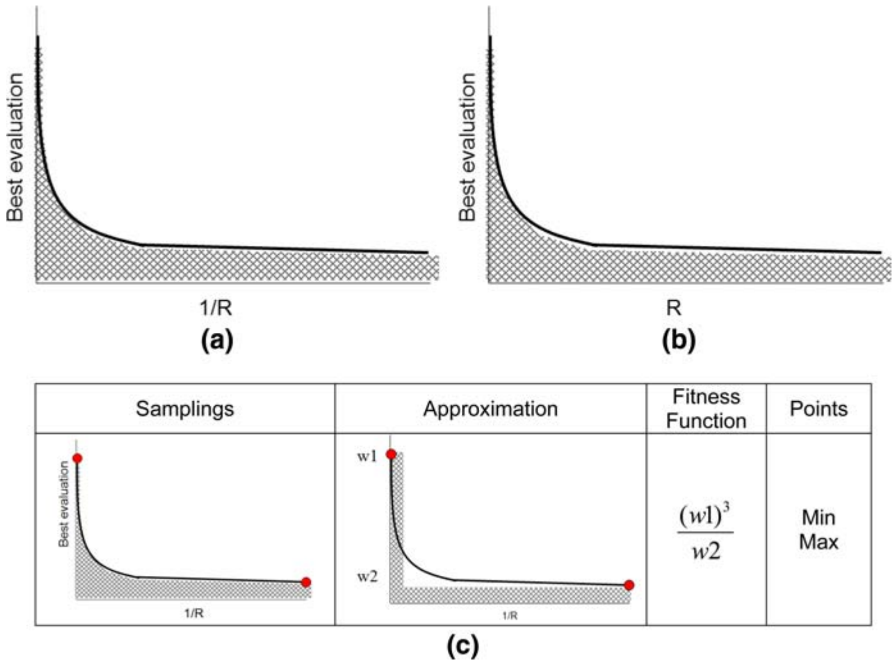


Fig. 8 The deÞnition of tamper-evident property and a sampling approach. a Partial short damage, b partial disconnection damage, c a sampling approach for tamper-evident circuits

Figure 9 shows circuit diagrams, output responses and robustness graphs of tamper-evident circuits evolved. In partial short damage, the Þnal circuit had very small number of components. It lost its performance because there is very small number of additional components to complement the broken one. In partial disconnection damage, the circuit had a linear connection of multiple components and they were fragile to the disconnections. In robustness graph, tamper-evident circuits always had lower performance when there is damage.

Figure 10 shows the comparison of the output response change when the two damages are introduced. In any case, the normal circuits didnÕt lose their original

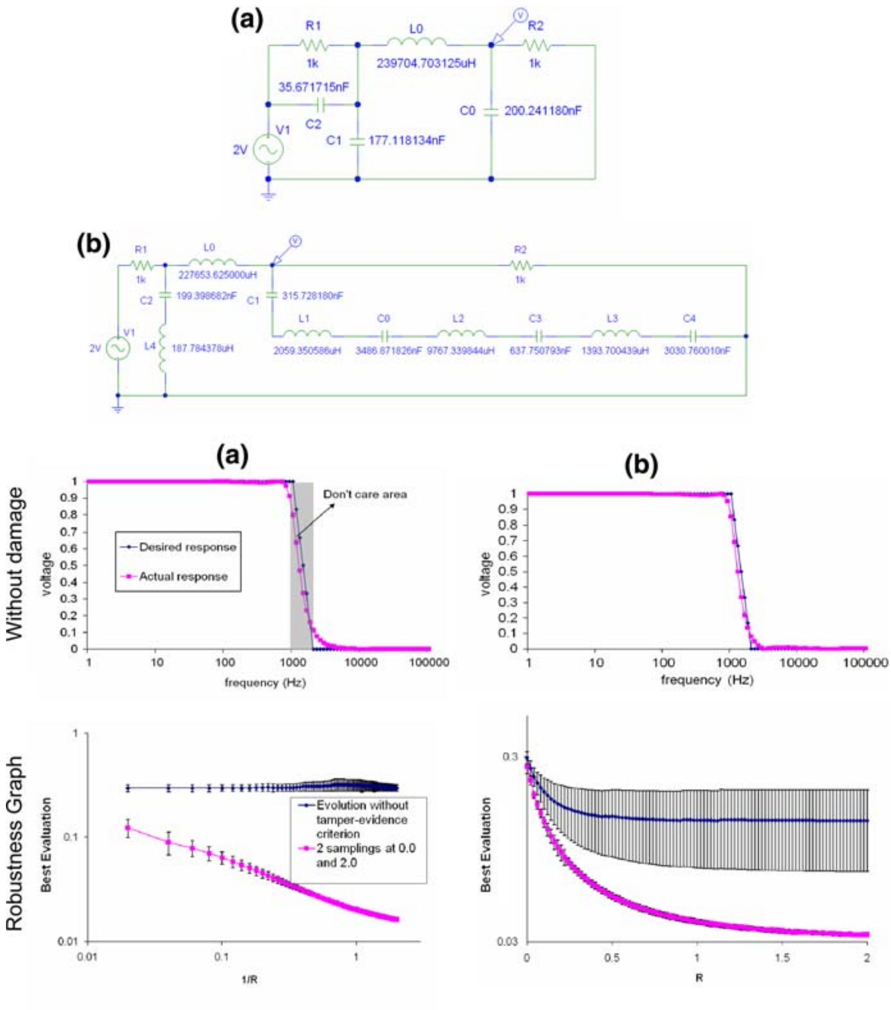


Fig. 9 Circuit diagrams, output responses and robustness graph of tamper-evident circuits evolved. a The best tamper-evident circuit for partial short, b the best tamper-evident circuit for partial disconnection

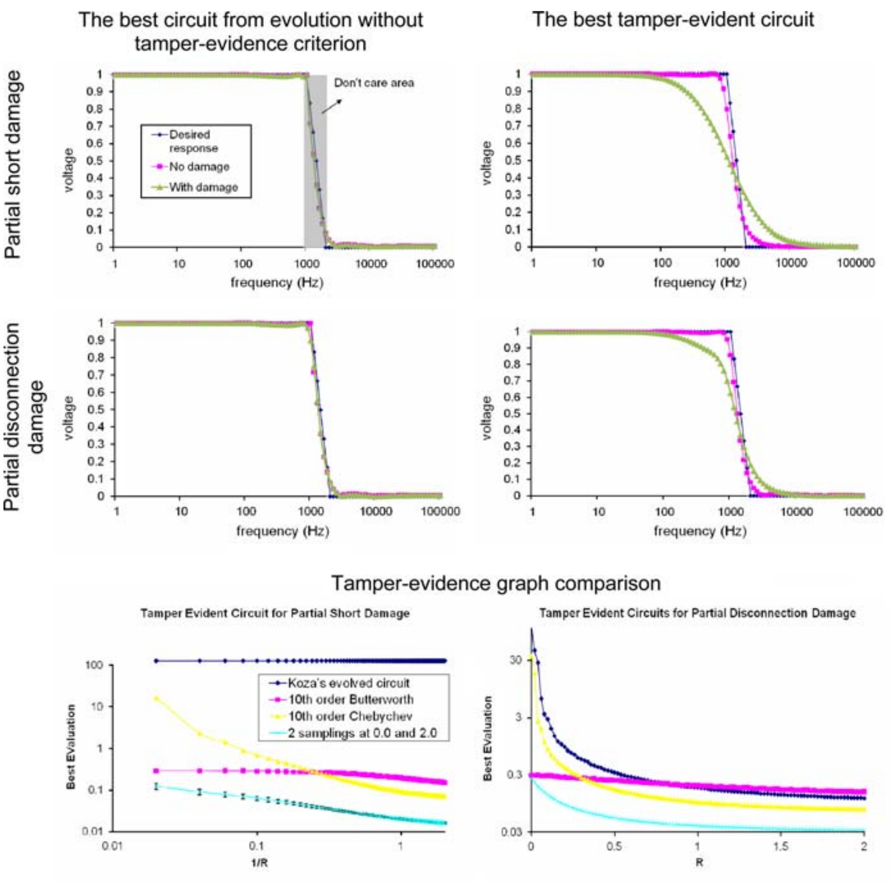


Fig. 10 The effect of damages to the output response and the comparison of tamper-evidence graph with other circuits. $R = 0.5$-K for partial short damage, $R = 2.0$-K for partial disconnection damage

performance by damage but the tamper-evident circuits lost its original output response pattern. In tamper-evidence graph comparison, the evolved circuits always showed the lower performance (most sensitive) to damage.

4.4 Other circuits

We tested the performance of the algorithm on other circuit tasks to demonstrate more general applicability. We evolved a low pass, band-pass, notch-pass and high-pass Þlters. The best results are shown in Fig. 11. These circuits were not studied in depth.

4.5 Physical implementation

We tested the validity of the evolved circuits by building them in reality. The values of each component in the evolved circuits are real values that are not generally

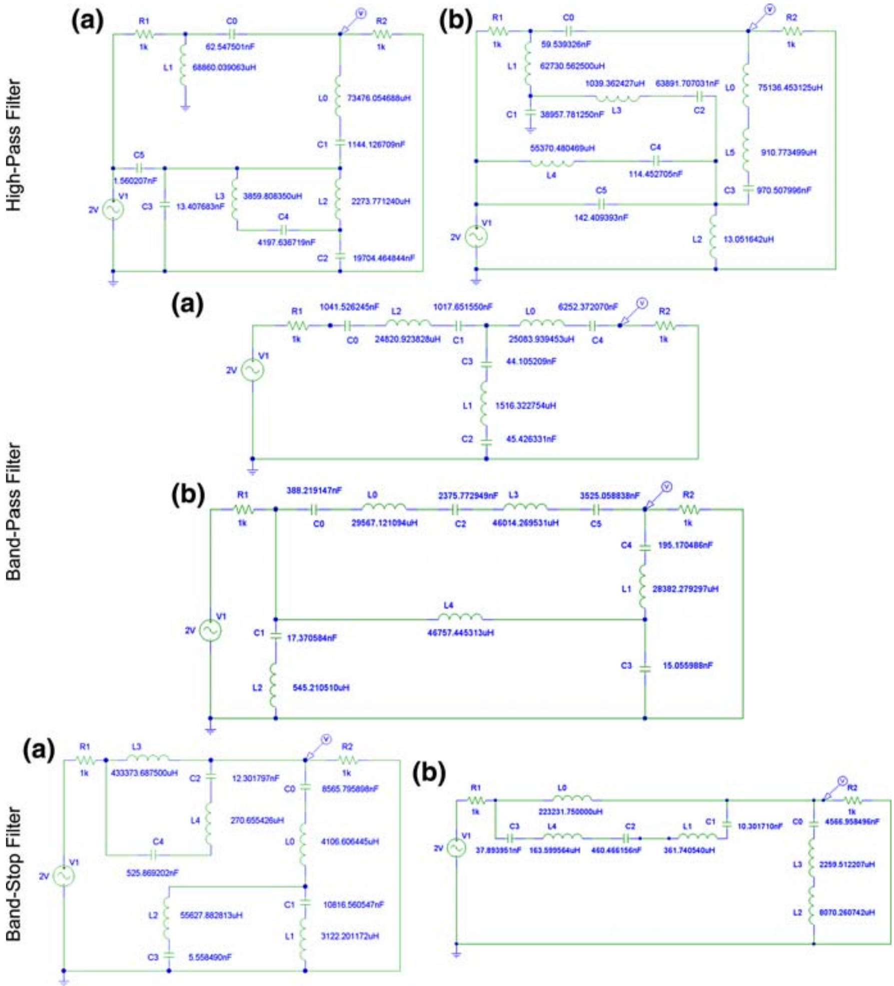


Fig. 11 Robust circuits for other tasks. In the high-pass Þlter, output is 1 V after 2 kHz and 0 V before 1 kHz. In band-pass Þlter, the output is 1 V between 100 Hz and 10 kHz. In band-stop Þlter, the output is opposite to the band-pass Þlter. *a* Partial short damage, *b* partial disconnection damage

commercially available. In previous work addressing physical implementation [10], the values of component were restricted to the commercially available E-12 series values represented as {10,12,15,18,22,27,33, 39,47,56,68,82} $\times 10^{A}$. Alternatively, it is possible to approximate the real values using serial and parallel combinations of standard components. In this paper, however, we replaced the real values in the evolved circuits with the closest values in the E-12 series without signiÞcant loss of performance. Figures 12 and 13 shows a comparison of performance between evolved and approximated circuits. In real circuit implementation, instead of 180 and 220 mH inductors, the combinations of multiple inductors (100, 15, and 10 mH) were used.

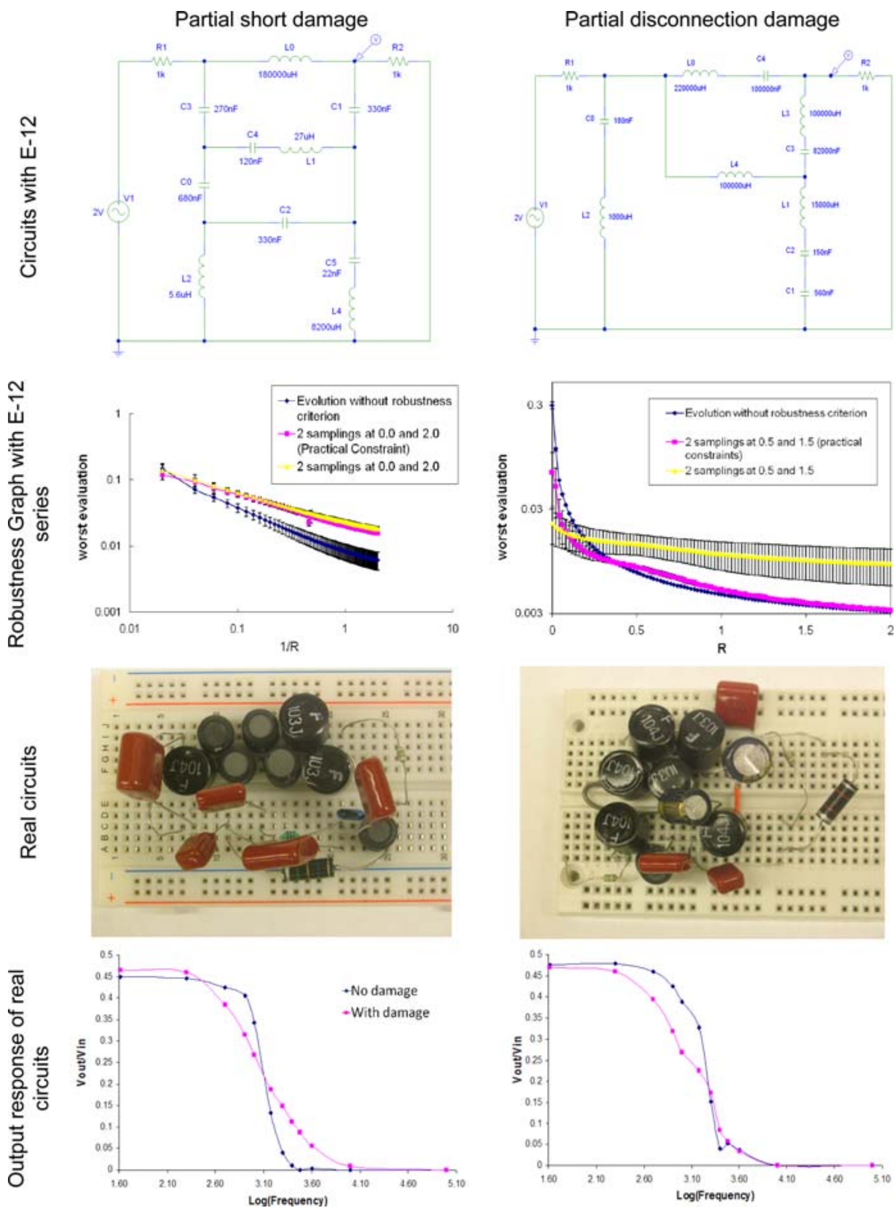


Fig. 12 Physical implementation of robust circuits. Desired V_{out}/V_{in} is 0.5 in low frequency area and 0.0 in high frequency one

To test circuit performance, we used a 2 V sinusoidal signal generator as a source and an oscilloscope to measure amplitude attenuation. The output responses were recorded at 12 different frequencies ranging from 41 Hz to 100 kHz. The Y axis is V_{out}/V_{in}. For the robust circuits, the results were similar to the one of simulations with SPICE.

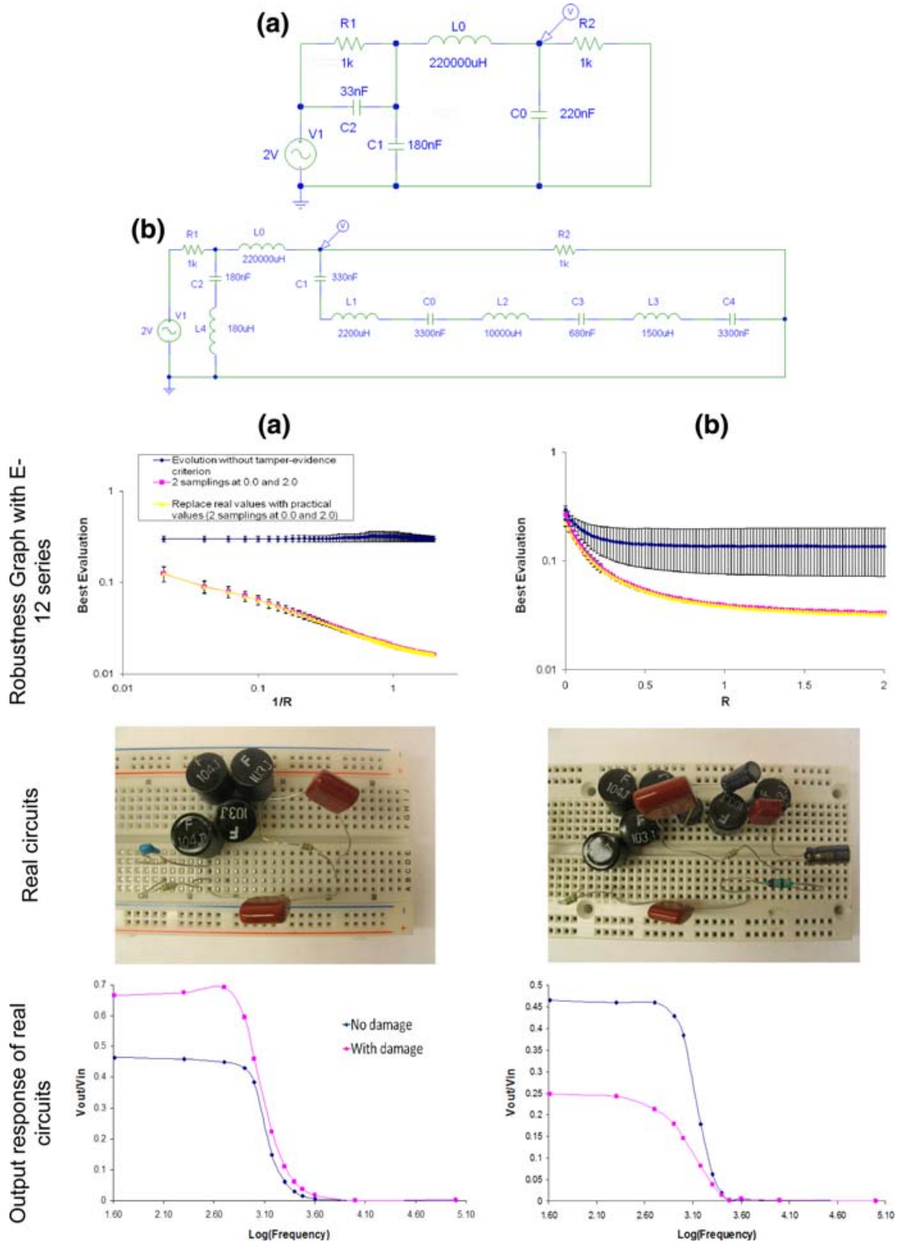


Fig. 13 Physical implementation of tamper-evident circuits. Desired V_{out}/V_{in} is 0.5 in low frequency area and 0.0 in high frequency one. a Partial short damage, b partial disconnection damage

A similar approach was difficult to use for the tamper-evident circuits (Fig. 13), as they are evolved to be sensitive to variations. Once evolved components were replaced with standard ratings, the performance changed. That performance, however, was highly sensitive to any additional changes: for example, in the tamper-evident circuit for partial short damage, $L0$ was the least sensitive component when $R = 0.5$-K. However, once the circuit values were standardized, the least sensitive component became $C2$, for the real physical circuit. In case of partial disconnection damage, least-sensitive component $L1$ became $L0$. Regardless of the change, however, the tamper-evident circuits showed sensitivity to the damage. For the all four circuits, the output response in low frequency area was smaller (910Ð960 mV) than expected in simulation (1 V).

5 Conclusions and future work

In this paper, we proposed a method for evolving a robust analog circuit against partial short and disconnection damages. The computational cost was minimized by using a compact evolutionary strategy and pruning method. The evolutionary process required two or three times more computational effort than the evolution of standard circuits, but it produced highly robust circuits compared to standard evolution. Using the inverse deÞnition of robustness, tamper-evident circuits were evolved and showed successful sensitivity to modiÞcation or reverse engineering. Finally, we tested the evolved circuit using a real physical implementation.

While the damage representation used in this paper can cover a wide variety of faults, we realize that it is not universal, and more elaborate, component-speciÞc failure modes should be considered in the future. To offset the extensive computational cost associated with more extensive damage models, more elaborate sampling methods should be considered as well.

The key result of this study is the production of resilient circuits that have no single point of failure. Surprisingly, this robustness did not come at a signiÞcant increase in circuit complexity, suggesting that design of passively robust circuits may be practical for more complex tasks.

Future work is needed to address tolerance considerations for manufacturability and yield. Similar to several attempts in digital circuit evolution [11, 12], it would also be interesting to use data mining techniques to extract speciÞc robust design rules and motifs from the plethora of circuits generated by this automated system.

Acknowledgments This work was supported in part by US National Science Foundation (NSF) CAREER grant number DMI 0547376. Co-author K.-J.K. was supported by the Korea Research Foundation Grant (KRF-2007-357-D00220) funded by the Korean Government (MOEHRD) and Korea Health 21 R&D Project, Ministry for Health, Welfare and Family Affairs (A040163).

References

1. G.A. Hollinger, D.A. Gawaltney, Evolutionary design of fault-tolerant analog control for a piezo-electric pipe-crawling robot. In: *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation* (2006), pp. 761Ð768
2. J. Hu, X. Zhong, E.D. Goodman, Open-ended robust design of analog Þlters using genetic programming. In: *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation* (2005), pp. 1619Ð1626

3. R.S. Zebulum, A. Stoica, D. Keymeulen, L. Sekanina, R. Ramesham, X. Guo, Evolvable hardware system at extreme low temperature. Lect. Notes Comput. Sci. 3637, 37Ð45 (2005)
4. J. Torresen, A scalable approach to evolvable hardware. Genet. Program Evolvable Mach. 3(3), 259Ð282 (2002)
5. J.R. Koza, M.A. Keane, J. Yu, F.H. Bennett, W. Mydlowec, Automatic creation of human-competitive programs and controllers by means of genetic programming. Genet. Program Evolvable Mach. 1(1Ð2), 121Ð164 (2000)
6. A. Ciccazzo, P. Conca, G. Nicosia, G. Stracquadanio, An advanced clonal selection algorithm with Ad-Hoc network-based hypermutation operators for synthesis of topology and sizing of analog electrical circuits. Lect. Notes Comput. Sci. 5132, 60Ð70 (2008)
7. R.S. Zebulum, M.A. Pacheco, M. Vellasco, H.T. Sinohara, Evolvable hardware: on the automatic synthesis of analog control systems. Proc. IEEE. Aerosp. Conference 5, 451Ð463 (2000)
8. WINSPICE, http://www.winspice.com/
9. J.R. Koza, F.H. Bennett III, D. Andre, M.A. Keane, F. Dunlap, Automated synthesis of analog electrical circuits by means of genetic programming. IEEE. Trans. Evol. Comput. 1(2), 109Ð128 (1997)
10. C. Goh, Y. Li, GA automated design and synthesis of analog circuits with practical constraints. In: Proceedings of the 2001 congress on evolutionary computation 1, 170Ð177 (2001)
11. J.F. Miller, D. Job, V.K. Vassilev, Principles in the evolutionary design of digital circuits-Part I. Genet. Program Evolvable Mach. 1(1Ð2), 7Ð35 (2000)
12. S. Zhao, L. Jiao, Multi-objective evolutionary design and knowledge discovery of logic circuits based on an adaptive genetic algorithm. Genet. Program Evolvable Mach. 7(3), 195Ð210 (2006)
13. J.R. Koza, M.A. Keane, M.J. Streeter, Routine automated synthesis of Þve patented analog circuits using genetic programming. Soft. Comput. 8, 318Ð324 (2004)
14. F. Wang, Y. Li, L. Li, K. Li, Automated analog circuit design using two-layer genetic programming. Appl. Math. Comput. 185, 1087Ð1097 (2007)
15. T. Sripramong, C. Toumazou, The invention of CMOS ampliÞers using genetic programming and current-ßow analysis. IEEE. Trans. Comput. Aided Des. Integr. Circuits Syst. 21(11), 1237Ð1252 (2002)
16. D. Keymeulen, R.S. Zebulum, Y. Jin, A. Stoica, Fault-tolerant evolvable hardware using Þeld-programmable transistor arrays. IIEEE. Trans. Reliab. 49(3), 305Ð316 (2000)
17. J.D. Lohn, S.P. Colombano, Automated analog circuit synthesis using a linear representation. In: Proceedings of the 2nd international conference on evolvable systems, pp. 125Ð133 (1998)
18. P. Layzell, A. Thompson, Understanding inherent qualities of evolved circuits: evolutionary history as a predictor of fault tolerance. Lect. Notes Comput. Sci. 1801, 133Ð144 (2000)
19. M. Natsui, N. Homma, T. Aoki, T. Higuchi, Topology-oriented design of analog circuits based on evolutionary graph generation. Lect. Notes Comput. Sci. 3242, 342Ð351 (2004)
20. T.R. Dastidar, P.P. Chakrabarti, P. Ray, A synthesis system for analog circuits based on evolutionary search and topological reuse. IEEE. Trans. Evol. Comput. 9(2), 211Ð224 (2005)
21. S. Ando, H. Iba, Analog circuit design with a variable length chromosome. In: Proceedings of the 2000 congress on evolutionary computation, vol. 2, pp. 994Ð1001 (2000)
22. C. Mattiussi, D. Floreano, Analog genetic encoding for the evolution of circuits and networks. IEEE. Trans. Evol. Comput. 11(5), 596Ð607 (2007)
23. X. Xia, Y. Li, W. Ying, L. Chen, Automated design approach for analog circuit using genetic algorithm. Lect. Notes Comput. Sci. 4490, 1124Ð1130 (2007)
24. J.B. Grimbleby, Hybrid genetic algorithms for analogue network synthesis. In: Proceedings of the 1999 congress on evolutionary computation, vol. 3, pp. 1781Ð1787 (1999)
25. D. Berenson, N. Estevez, H. Lipson, Hardware evolution of analog circuits for in-situ robotic fault-recovery. In: Proceedings of NASA/DOD conference on evolvable hardware, pp. 12Ð19 (2005)
26. Y. Sapargaliyev, T. Kalganova, Constrained and unconstrained evolution of ÔÔLCRÕÕ low-pass Þlters with oscillating length representation. In: Proceedings of IEEE congress on evolutionary computation, pp. 1529Ð1536 (2006)
27. T. Biondi, A. Ciccazzo, V. Cutello, S. DÕAntona, G. Nicosia, S. Spinella, Multi-objective evolutionary algorithms and pattern search methods for circuit design problems. J. Univ. Comput. Sci. 12(4), 432Ð449 (2006)
28. G. Nicosia, S. Rinaudo, E. Sciacca, An evolutionary algorithm-based approach to robust analog circuit design using constrained multi-objective optimization. Knowl.-Based Syst. 21(3), 175Ð183 (2008)

29. L. Zinchenko, H. Muhlenbein, V. Kureichik, T. Mahnig, A comparison of different circuit representations for evolutionary analog circuit design. Lect. Notes Comput. Sci. 2606, 13Ð23 (2003)
30. M.D. Schmidt, H. Lipson, Coevolution of Þtness predictors, IEEE Trans. Evol. Comput. 12(6), 736Ð749 (2008)
31. J.B. Hagen, *Radio-Frequency Electronics-Circuits and Applications* (Cambridge University Press, Cambridge, 1996)