LFSR Reseeding for Stochastic Circuit Repairing and Minimization

Chen Wang and Weikang Qian

University of Michigan-SJTU Joint Institute, Shanghai Jiao Tong University

Outline

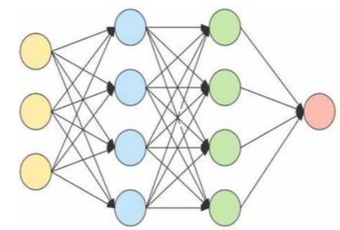
- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

Outline

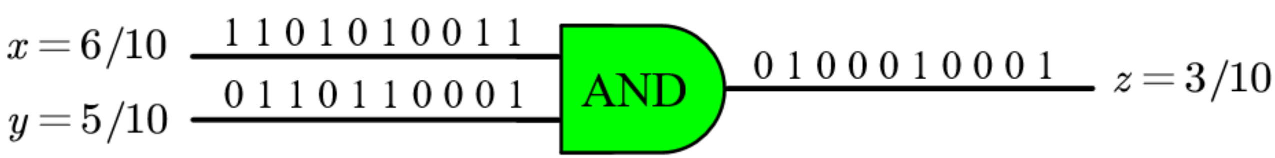
- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

Introduction to Stochastic Computing (SC)

- Stochastic computing (SC) is a re-emerging computing paradigm.
  - Early proposed in 1960s.
  - Recently applied in image processing and neural networks.

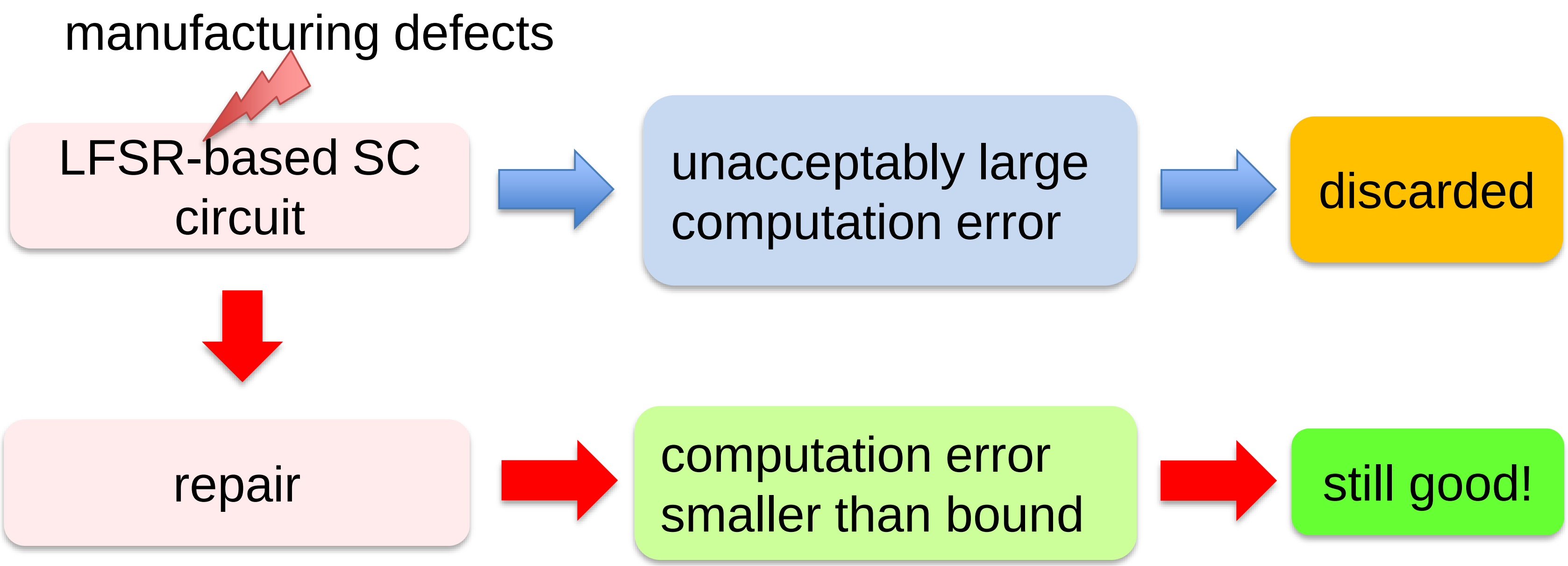


- SC computes on bit streams rather than radix-2 numbers.
- Value of a bit stream: probability of 1 in it.
- SC realizes complicated computation with simple circuit.



Motivations of This Study

- SC circuit is currently implemented by CMOS technology, and it can still be faulty.



- Question: can we repair the faulty SC circuit?

Motivations of This Study

SC benefits

Simple circuit → Small
- critical-path delay
- circuit area
- power consumption

Fault tolerance → Fault-tolerant applications

SC drawbacks

Long bit streams to ensure accuracy i.e., 512 or 1024 → Large
- computation latency
- energy consumption

- Question: can we further minimize SC circuit during design time?

Observation and Contributions

- We make an important observation

> The computation error of the faulty LFSR-based SC circuit can be reduced by LFSR reseeding without any hardware modification.

- Our contributions

➢ Repair the faulty SC circuit by LFSR reseeding.

➢ Minimize the SC circuit by constant replacement with accuracy recovery through LFSR reseeding.

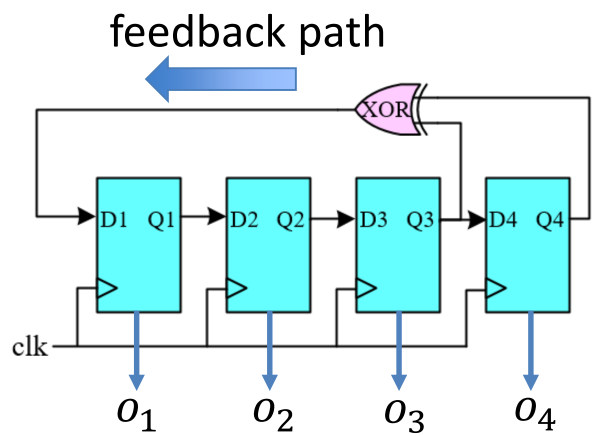
Outline

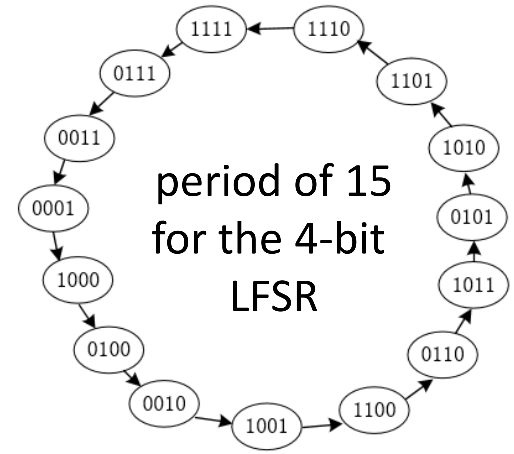
- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

Preliminaries: LFSR Reseeding

- An LFSR is a basic primitive for both digital circuit design and test.

Possible LFSR seeds
(0,1,0,1), (1,1,1,0),...





- LFSR seed: initial states of the registers

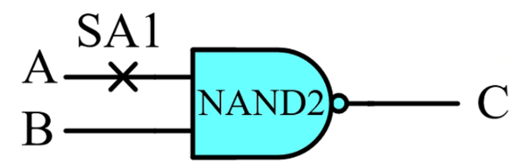
- An *n*-bit maximum-length LFSR provides an output sequence with a period of $2^n - 1$.

- LFSR reseeding: change the LFSR seed → change LFSR output sequence

- Each LFSR has extra ports to load a given seed.

Preliminaries: Stuck-At Fault and Constant Replacement

- Stuck-at fault
  - ➢ Caused by manufacturing defects
  - ➢ A signal fixed to logic 0 or 1
  - ➢ Usually causes output error of the circuit

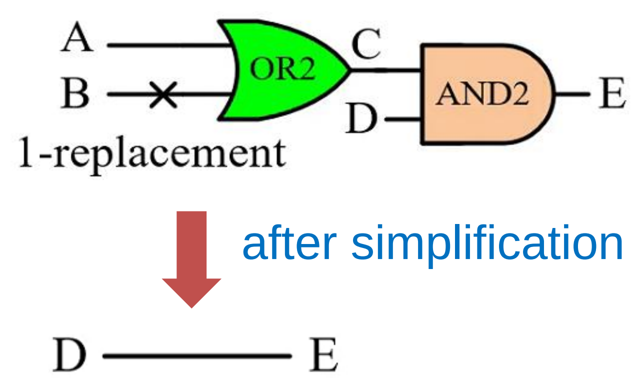


A\B	0	1
0	1	1
1	1	0

output error!

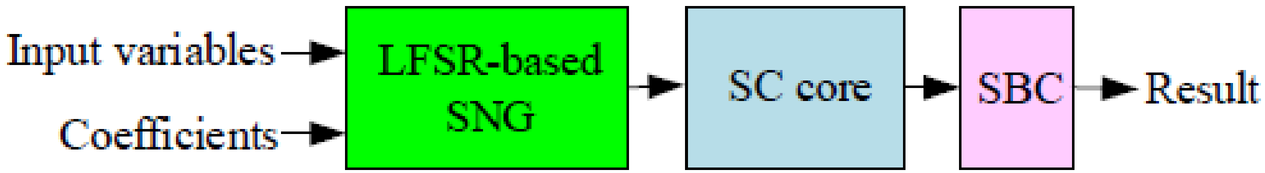
A\B	0	1
0	1	0
1	1	0

- Constant replacement
  - ➢ Replace an internal signal by a constant logic 0 or 1.
  - ➢ Effect is similar to a stuck-at fault, but deliberately applied at design time.
  - ➢ An approach used in approximate computing to simplify circuit.
  - ➢ Also leads to output error.

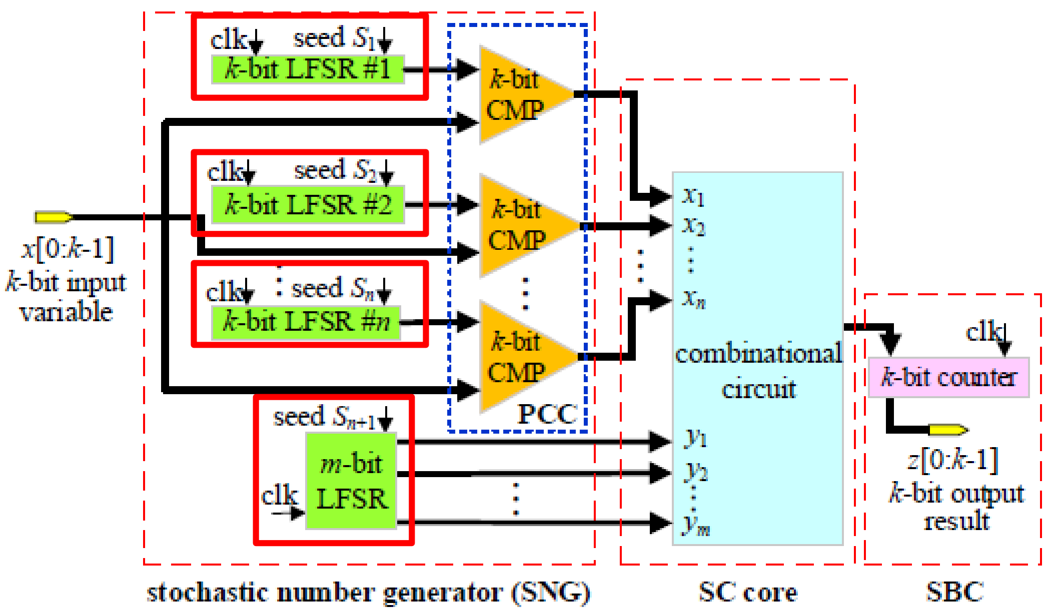


Preliminaries: Architecture of the SC Circuit Applicable for this Study

- A general architecture of the SC circuit considered in this study



- SC circuit studied here (proposed by Zhao et. al., 2015)



LFSR seed vector:

$$SV = (s_1, s_2, \cdots, s_{n+1})$$

Outline

- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

Outline

- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

SC Circuit Repair by LFSR Reseeding
Basic Idea

- Find faults may need repair
- Collect related info.

Stage 1

offline characterization

- Refined seed vector set for repair
- Try to repair each faulty SC circuit

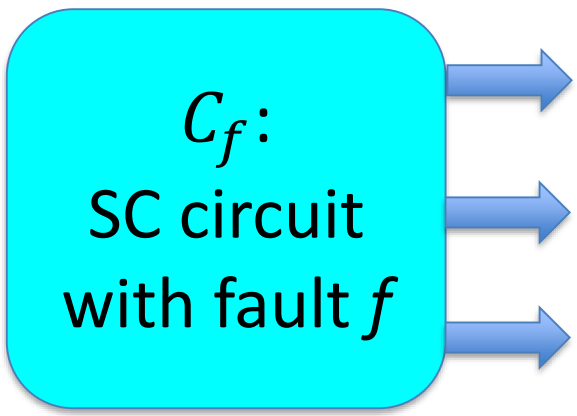
Stage 2

online test and pair

- e^*: reference output error (ROE), minimum output error of fault-free circuit
- v^*: reference seed vector (RSV)
- Basically, obtained by exhaustive search over all seed vectors for the fault-free SC circuit.

Stage 1: Offline Characterization

- Error bound is not given. Enumerate all possible faults.



e_{v^*} output error of C_f under RSV v^*

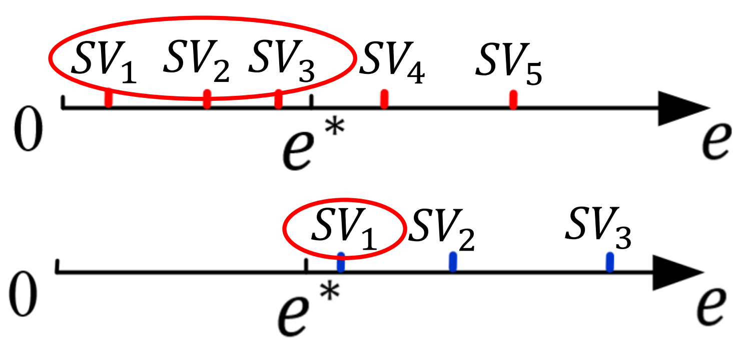
v_r decisive seed vector (DSV)

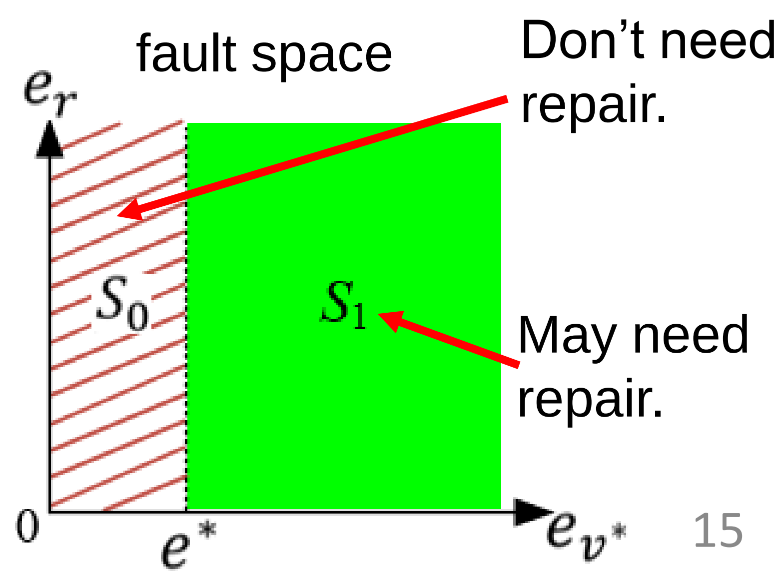
e_r output error of C_f under DSV v_r

- Decisive seed vector: in set V

  (1) leads to $e \leq e^*$

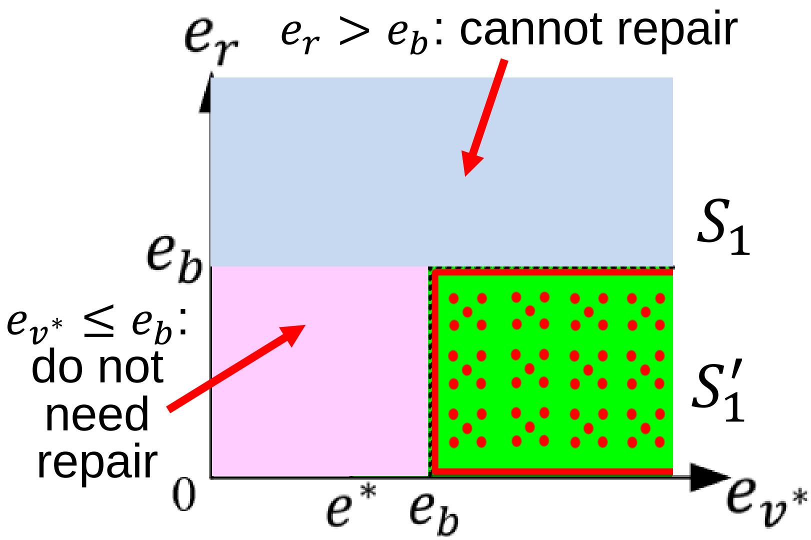
  (2) gives minimum error if all $e > e^*$

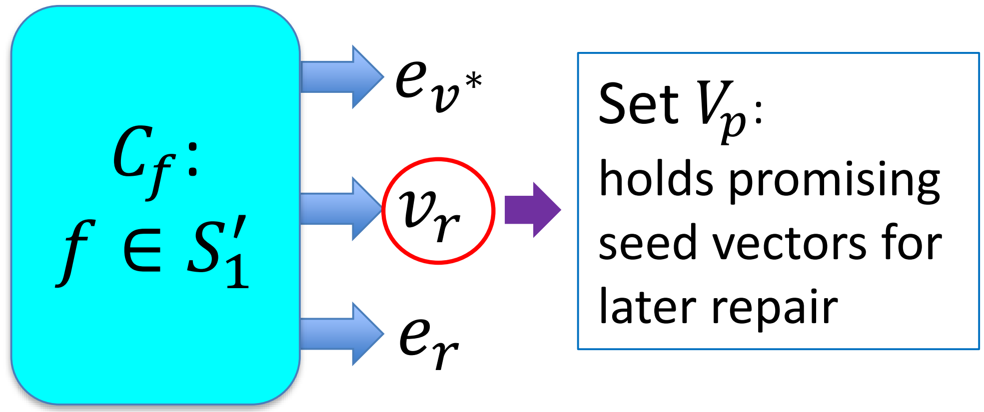


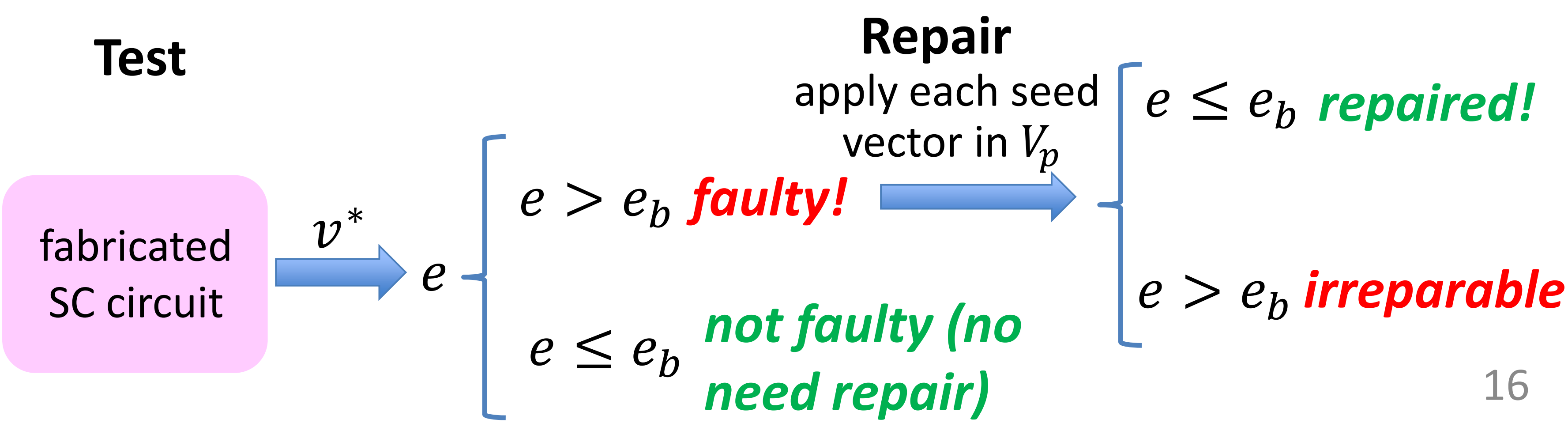


Stage 2: Online Test and Repair

- The user-defined error bound e_b is now given.







Outline

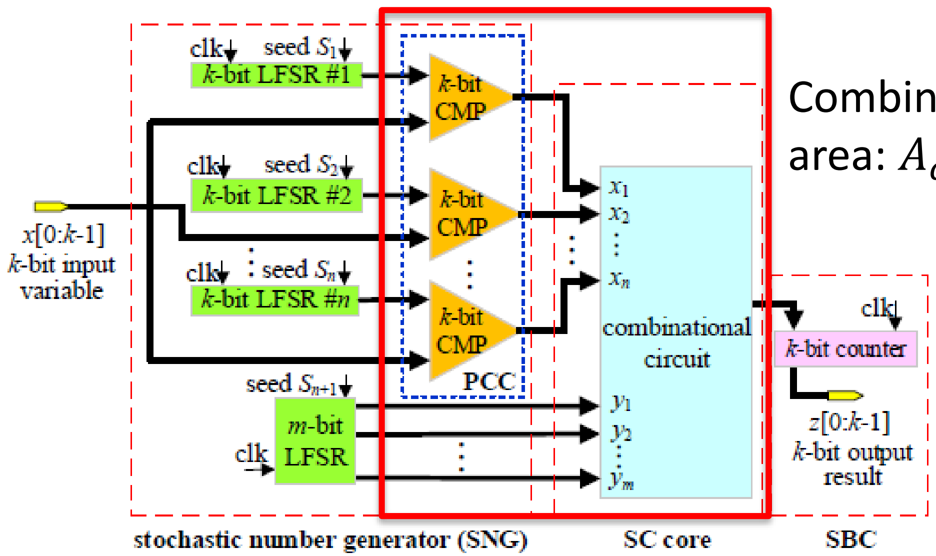
- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

Minimize SC Circuit by Constant Replacement with LFSR Reseeding

- For LFSR-based SC circuit, probability conversion circuit (PCC) consumes a large portion of hardware cost.

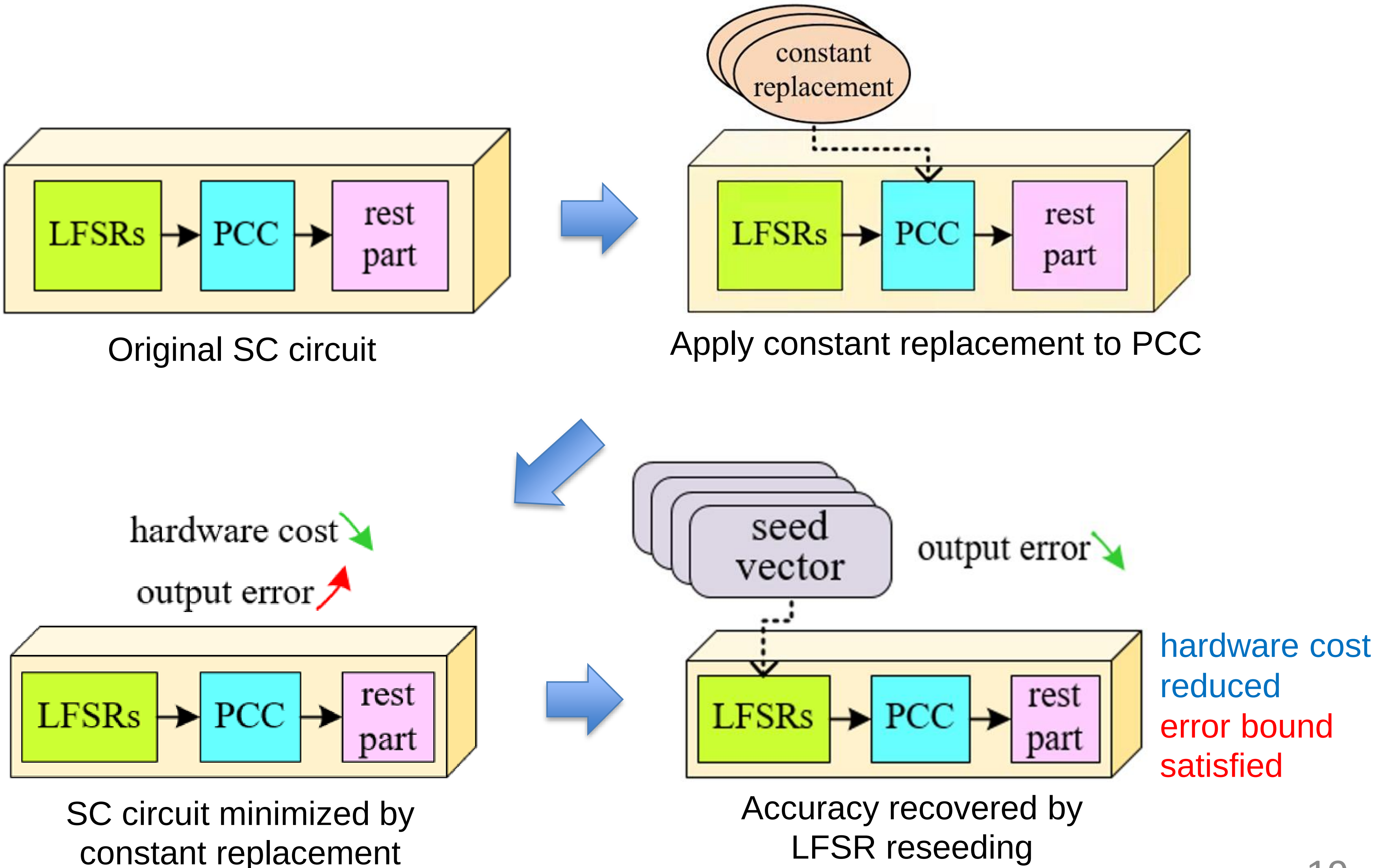
- Area-delay product (ADP) is used to measure the hardware cost.

$$\left(\sum_{i=1}^{n+1} A_{LFSR,i} + A_{comb} + A_{CNT}\right) \cdot (D_{comb} + D_{CNT})$$



Combination of PCC and SC core area: A_{comb}, delay: D_{comb}

Basic Idea


constant replacement
LFSRs
PCC
rest part
Original SC circuit
Apply constant replacement to PCC
hardware cost
output error
SC circuit minimized by constant replacement
seed vector
output error
Accuracy recovered by LFSR reseeding
hardware cost reduced
error bound satisfied

Outline

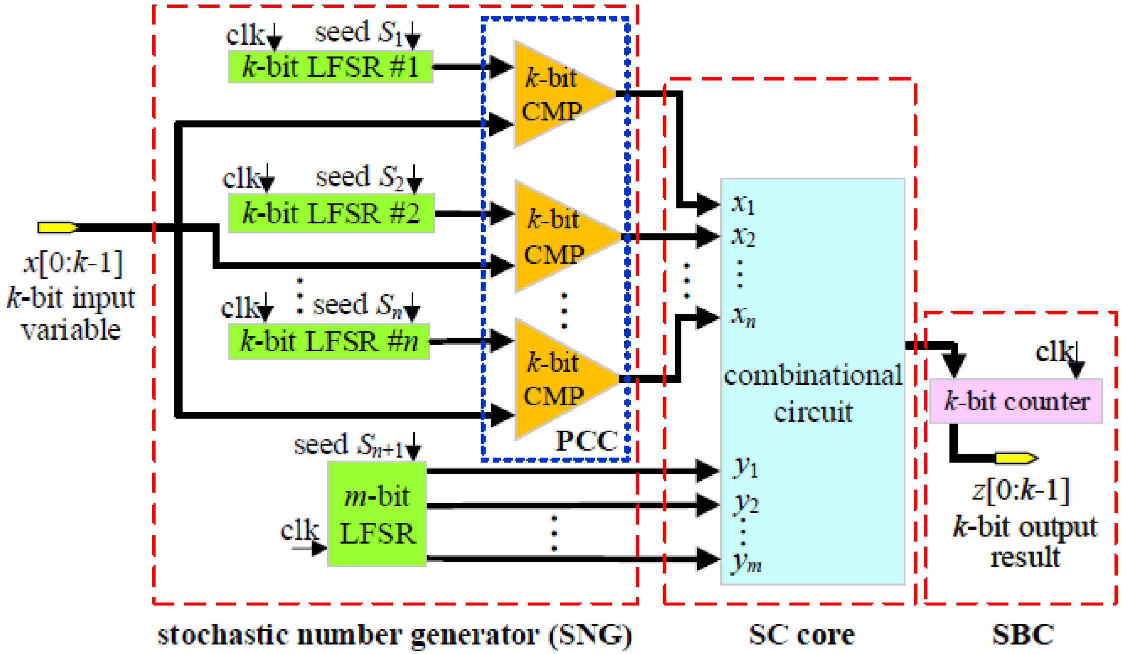
- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

Experiment 1: Unshared-LFSR Design

- Benchmarks used in this experiment.

ID	(n,m)	function	ID	(n,m)	function	ID	(n,m)	function
1	(4,4)	$\cos(x)$	5	(6,6)	$\cos(x)$	9	(6,6)	$\tanh(x)$
2	(4,4)	$\exp(-x)$	6	(6,6)	$\exp(-x)$	10	(6,6)	$\exp(-2x)$
3	(4,4)	$\tanh(4x)$	7	(6,6)	$\tanh(4x)$	11	(6,6)	$1/(1+\exp(-x))$
4	(4,4)	$x^{2.2}$	8	(6,6)	$x^{2.2}$	12	(6,6)	$0.5\cos(\pi x)+0.5$

- SC circuit architecture. Independent LFSRs are used.



Experiment 1: Unshared-LFSR Design
(1) Faulty SC Circuit Repair

- Repairing rates over different error bounds.

$$repairing\ rate\ (\%) = \frac{\#repaired\ faults}{\#faults\ violating\ e_b\ before\ reseeding} \times 100\%$$

The larger, the better!

BM ID	WCAE bound 0.02	0.04	0.06	0.08	0.1	MAE bound 0.02	0.04	0.06	0.08	0.1
1	15.5	15.3	12.7	5.8	9	18.8	13.6	8.4	8.3	13.6
2	–	15	18.4	15.8	7.2	–	13.6	22.3	19.2	20.7
3	0	3.5	8.2	4.8	2.7	0	3.8	8.9	5.3	3
4	32	20.1	5.3	8.6	9.3	35.6	27.1	9.7	9.4	8.8
5	44.2	28.8	8.3	29.9	39.3	11.2	20.8	37.6	32.4	14.8
6	19.4	10.6	24.3	18.3	17.3	14.6	21.5	13.5	4.4	8.1
7	28.7	32.4	51.1	40.9	28.1	40.2	20.2	7.1	12	11.5
8	37.9	40.6	39.8	30.4	42.3	33.5	28.8	33.3	20	35.1
9	31.1	30	34	31.8	26.7	25.4	32.4	25.9	26.6	42.3
10	26.7	11.5	13.2	17.8	25.3	26.5	27.4	20.2	10.6	2.3
11	24.8	25	8.4	6.4	10.5	21.6	23.9	33.7	27.3	32.1
12	38.2	23.7	27.9	25.7	9.1	25.4	24.5	9.1	46.1	37.7

WCAE: worst-case absolute error

MAE: mean absolute error

Repairing rates are relatively large! *Repairing rates > 30%*

- For WCAE, 585× average cost reduction for online repair over a straightforward method (reseeding over 10k random seed vectors).

Experiment 1: Unshared-LFSR Design
(2) SC Circuit Minimization

- Minimize SC circuit by constant replacement with LFSR reseeding (1k random seeds). $e_b = \alpha \cdot e_{ref}$ relative error bound $\alpha = 1.5$

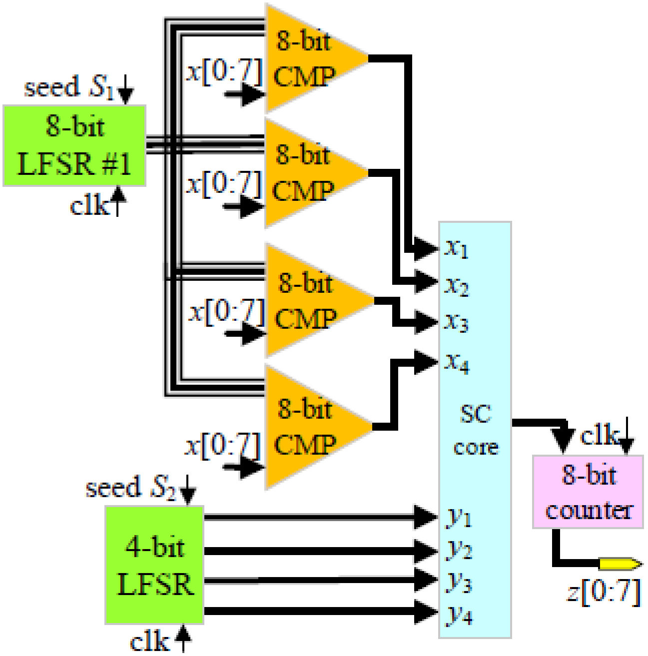
BM ID	SC circuit under test	WCAE bound	minimum WCAE	#8-bit LFSR	area (μm^2)	delay (ps)	total ADP (improvement)	
1	fault-free	-	8.84E-3	4	337.3	408.8	137892	
	$\alpha = 1.1$	9.73E-3	8.84E-3	4	337.3	408.8	137892 (0%)	
	$\alpha = 1.5$	1.33E-2	1.26E-2	2	195	335.3	65367 (52.6%)	
2	fault-free	-	2.95E-2	4	331.4	415.2	137597	
	$\alpha = 1.1$	3.24E-2	2.92E-2	2	192.3	331.3	63721 (53.7%)	
	$\alpha = 1.5$	4.42E-2	4.16E-2	2	186.7	316.4	59086 (57.1%)	
3	fault-free	-	1.84E-2	4	326.4	399.6	130409	
	$\alpha = 1.1$	2.02E-2	1.84E-2	4	326.4	399.6	130409 (0%)	
	$\alpha = 1.5$	2.75E-2	1.84E-2	4	326.4	399.6	130409 (0%)	
4	fault-free	-	7.99E-3	4	337	394.7	133033	
	$\alpha = 1.1$	8.78E-3	7.99E-3	4	337	394.7	133033 (0%)	
	$\alpha = 1.5$	1.20E-2	1.15E-2	3	267	374.6	100046 (24.8%)	
5	fault-free	-	7.89E-3	6	492.6	458.1	225666	
	$\alpha = 1.1$	8.68E-3	7.89E-3	6	492.6	458.1	225666 (0%)	
	$\alpha = 1.5$	1.18E-2	1.10E-2	2	213.3	353.1	75319.7 (66.6%)	> 66%
6	fault-free	-	7.54E-3	6	492.9	468.2	230778	
	$\alpha = 1.1$	8.29E-3	7.54E-3	6	492.9	468.2	230778 (0%)	
	$\alpha = 1.5$	1.13E-2	1.06E-2	2	216.3	359.1	77658 (66.3%)	> 66%
7	fault-free	-	1.56E-2	6	508.3	443.5	225437	
	$\alpha = 1.1$	1.72E-2	1.72E-2	5	383.8	381.5	146439 (35%)	
	$\alpha = 1.5$	2.34E-2	2.15E-2	5	365.2	363.9	132907 (41%)	
8	fault-free	-	9.60E-3	6	482.3	535.8	258368	
	$\alpha = 1.1$	1.06E-2	9.60E-3	6	482.3	535.8	258368 (0%)	
	$\alpha = 1.5$	1.44E-2	1.42E-2	6	439.7	436.6	191979 (25.7%)	

Experiment 2: Shared-LFSR Design

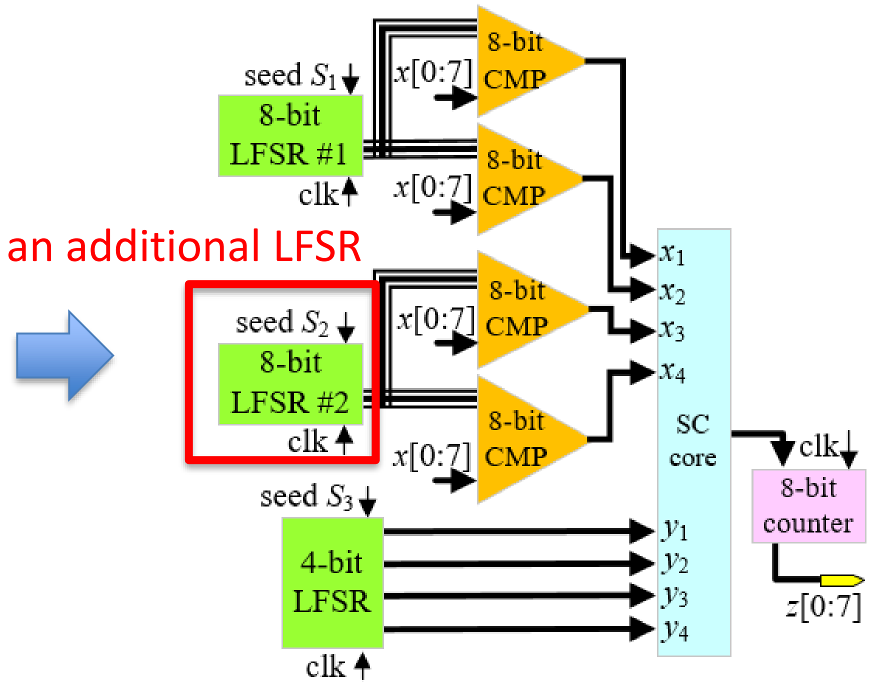
- Partially-shared design vs. fully-shared design
  - 1 more LFSR is added
  - Increased area is further reduced by PCC minimization.
  - LFSR reseeding space is enlarged, leads to lower output error

    # LFSR seed vector choice:

    - fully-shared: **15**
    - partially-shared: **3825**



fully-shared LFSR design



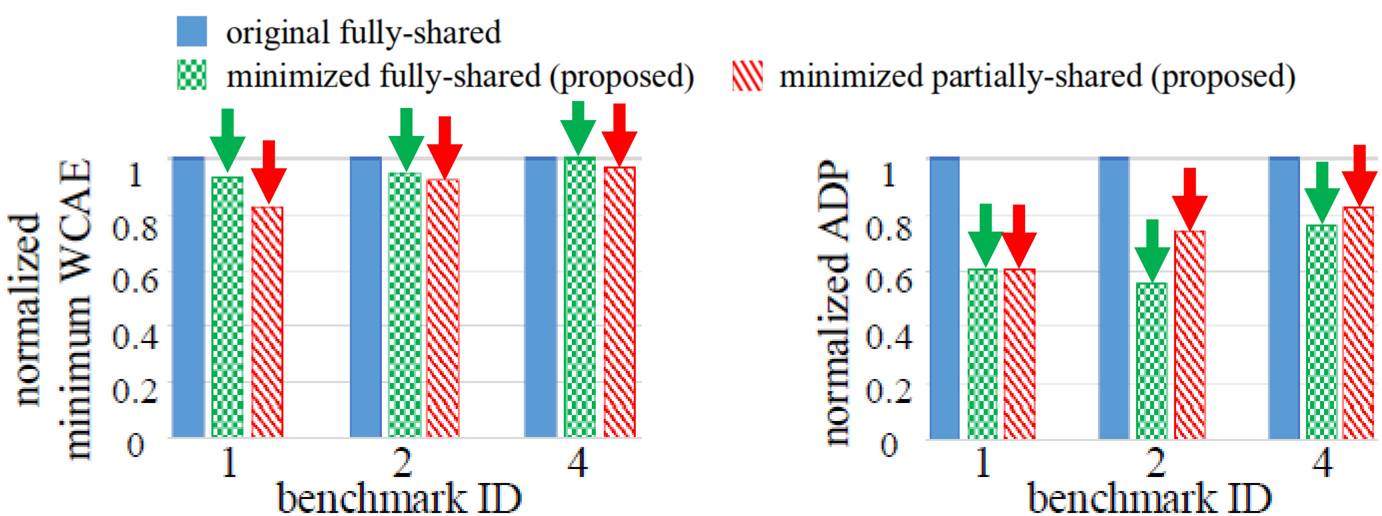
partially-shared LFSR design

Experiment 2: Shared-LFSR Design

- For partially-shared design, better repairing rates over different WCAE bounds.

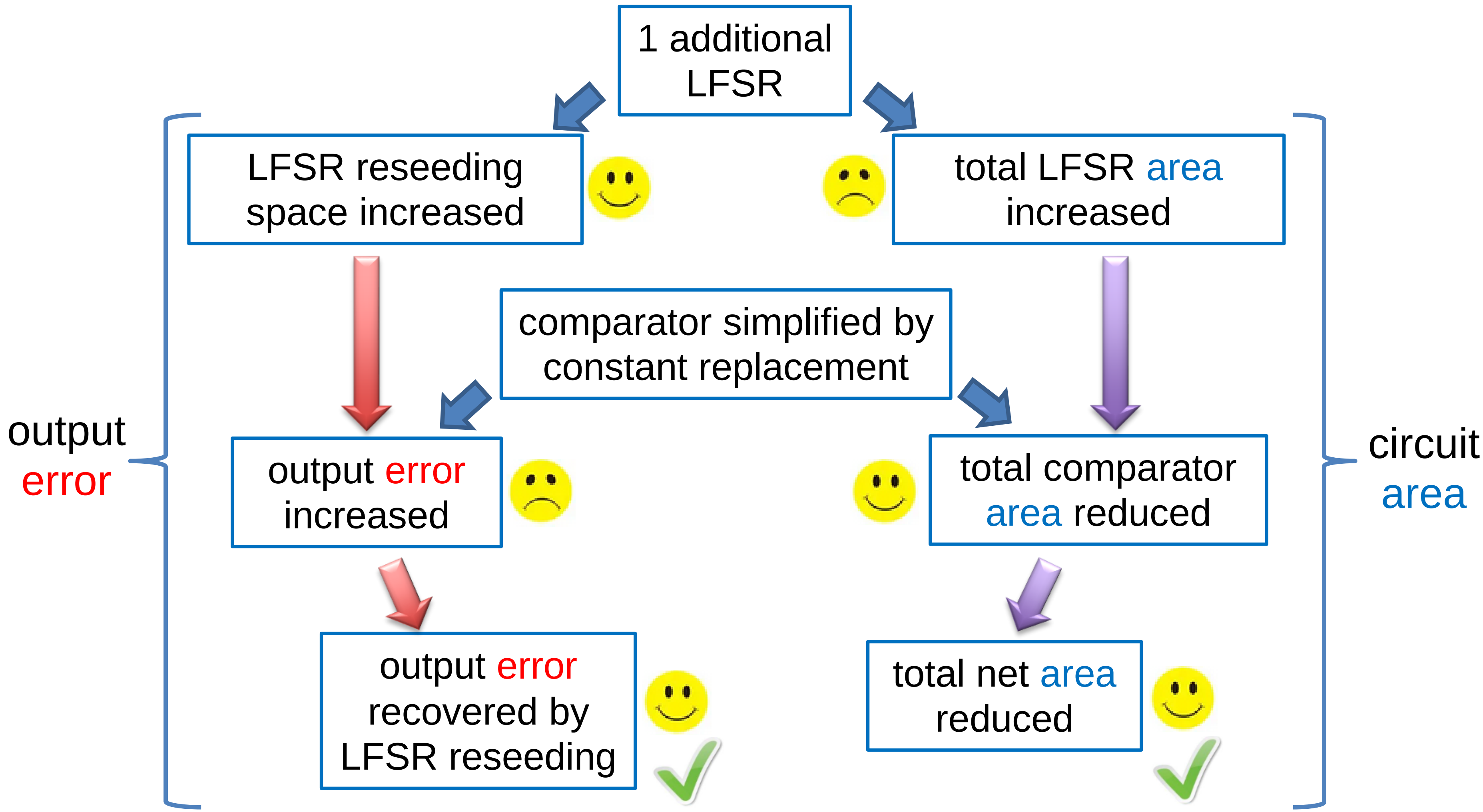
	BM	WCAE bound									
	ID	0.02	0.04	0.06	0.08	0.10	0.12	0.14	0.16	0.18	0.20
fully shared	1	–	7.32	3.31	3.42	4.17	4.35	0	0	0	1.69
	2	–	7.69	4.17	0.69	2.11	2.22	0	0	0.76	0
	4	–	4.52	2.25	0	4.88	0	0.67	0.67	2.03	10.64
partially shared	1	25.55	10.19	10.96	7.44	5.08	20.35	6.32	0	0	2.27
	2	–	13.5	6.63	13.64	15.65	13.53	5.04	4.27	4.35	5.26
	4	14.35	8.99	8.44	5.71	6.57	1.54	2.5	5	17.24	18.92

- Compared to fully-shared, partially-shared has smaller error, larger ADP, but both less than the original case.



Experiment 2: Shared-LFSR Design

- Partially-shared design: a new tradeoff between area and error.



Outline

- Background and Motivations
- Preliminaries
- Proposed Methods
  - Repair Faulty SC Circuits by LFSR Reseeding
  - Minimize SC Circuit by Constant Replacement with LFSR Reseeding
- Experimental Results
- Conclusions

Conclusions

- We observe the output error of the faulty LFSR-based SC circuit can be reduced by LFSR reseeding.
- An efficient LFSR reseeding method is proposed to repair the faulty SC circuit.
- At design time, we minimize SC circuit by deliberate constant replacement, followed by accuracy recovery through LFSR reseeding.
- Experiment indicates a new way to design the low-cost high-accuracy partially-shared LFSR-based SC circuit.

THANK YOU！

Q and A