Hardware ODE Solvers Using Stochastic Circuits

Siting Liu and Jie Han, Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada

Abstract

❑ Hardware ordinary differential equation (ODE) solvers are designed by using stochastic circuits.
❑ The stochastic ODE integrators serve as unbiased Euler solution estimators.
❑ Several strategies are proposed to reduce the error of stochastic ODE solvers; the designs are verified.
❑ The stochastic ODE solvers have lower energy cost, higher throughput and shorter minimum computation time than their binary counterparts.

Introduction

Stochastic Computing (SC)

In SC, information is carried by a stochastic bit stream. For example: {0101100}, coding 3/7.

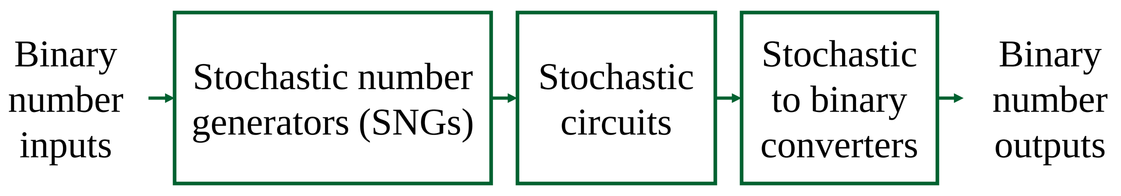


Fig. 1. A stochastic computing system.

Stochastic number generators (SNGs)

The component that converts a real number to a stochastic bit stream is usually referred to as an SNG.

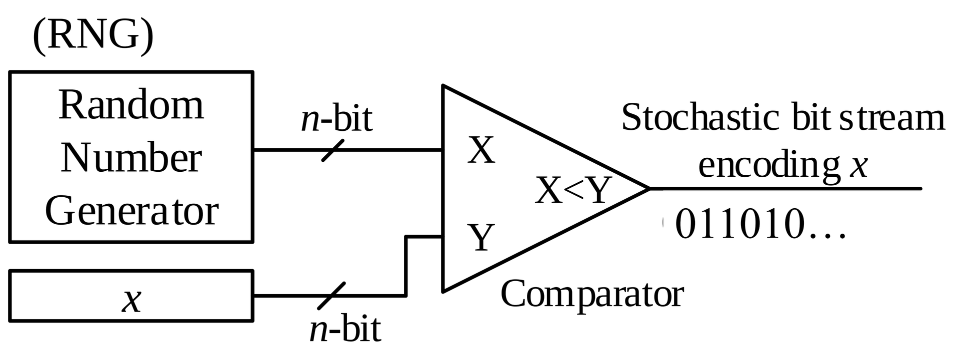


Fig. 2. A stochastic number generator (SNG).

Stochastic Integrator and its Formulation

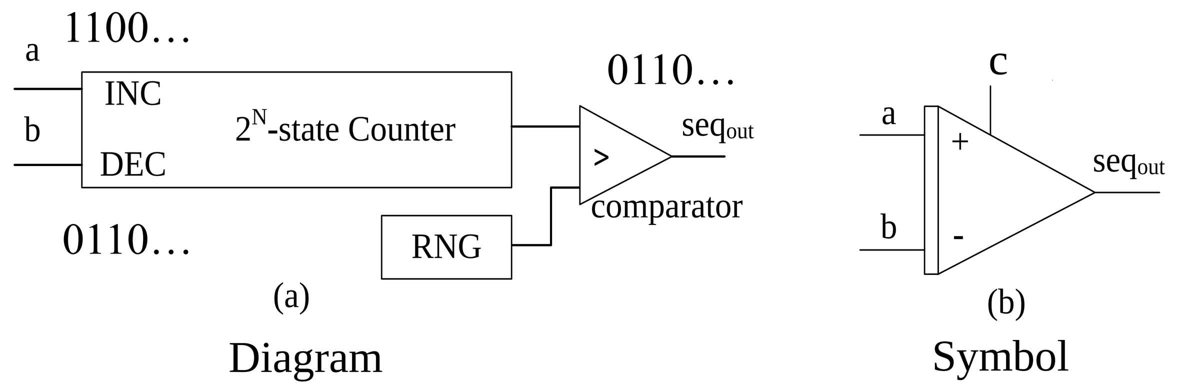


Fig. 3. A stochastic integrator.

❑ A stochastic integrator approximates the integration of the difference of two input stochastic sequences: $\int (a-b)dt$.

Table 1. An example of stochastic integrator with bit width 8 (N=8)

i	a_i	b_i	c_i	RN$_i$	seq_{out}
0	1	0	$(0.10000000)_2$	0.75	0
1	1	1	$(0.10000001)_2$	0.20	1
2	0	1	$(0.10000001)_2$	0.19	1
3	0	0	$(0.10000000)_2$	0.62	0

$$c_{i+1} = c_i + \frac{1}{2^N}(a_i - b_i). \quad (1)$$

Accumulating (1) for $i = 0, 1, 2, \dots, k-1$

$$c_k = c_0 + \frac{1}{2^N}\sum_{i=0}^{k-1}(a_i - b_i). \quad (2)$$

Taking expectation of (2)

$$\mathbb{E}[c_k] = c_0 + \frac{1}{2^N}\sum_{i=0}^{k-1}(\mathbb{E}[a_i] - \mathbb{E}[b_i]). \quad (3)$$

❑ Recall Euler method for solving $\frac{dy(t)}{dt} = f(t)$,

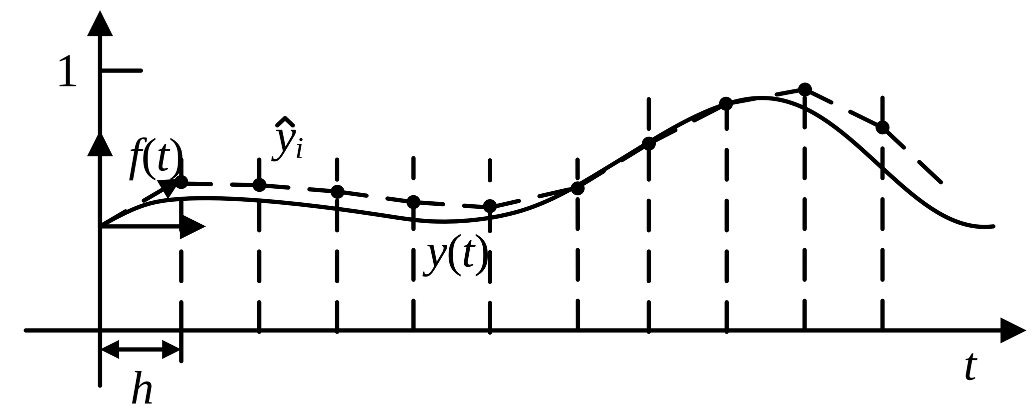


Fig. 4. The Euler method.

$$\hat{y}_{i+1} = y_i + hf(t_i), \quad (4)$$

where h is the step size and $t_i = h \cdot i$.

(4) can be converted to

$$\hat{y}_{i+1} = y_i + hf(hi) \approx y(h(i+1)). \quad (5)$$

By accumulating (2) for $i = 0, 1, 2, \dots, k$

$$\hat{y}_k = y_0 + h\sum_{i=0}^{k-1} f(h \cdot i) \approx y(h \cdot k). \quad (6)$$

For (3), let $c_0 = y_0$, $\mathbb{E}[a_i] - \mathbb{E}[b_i] = f(h \cdot i)$, then

$\mathbb{E}[c_k] = \hat{y}_k \approx y(\frac{k}{2^N})$, with $h = \frac{1}{2^N}$.

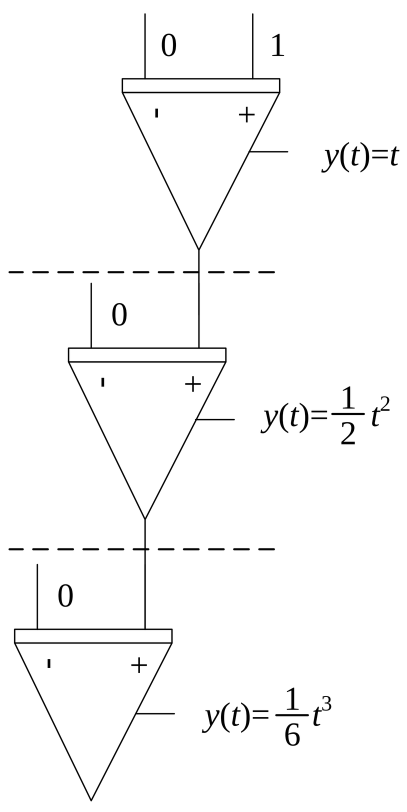
The stochastic integrator provides an unbiased estimate of the Euler solution with step size $h = \frac{1}{2^N}$.

Reference:

[1] J. C. Butcher. 1987. *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods*. Wiley-Interscience.
[2] B. R. Gaines. 1969. *Stochastic Computing Systems*. Springer US, 37–172.
[3] N. Saraf, K. Bazargan, D. J. Lilja, and M. D. Riedel. 2014. IIR Filters using Stochastic Arithmetic. In *DATE*.
[4] J. P. Hayes. 2015. Introduction to Stochastic Computing and Its Challenges. In *DAC*.

Stochastic ODE Solvers

Nonhomogeneous ODE solvers

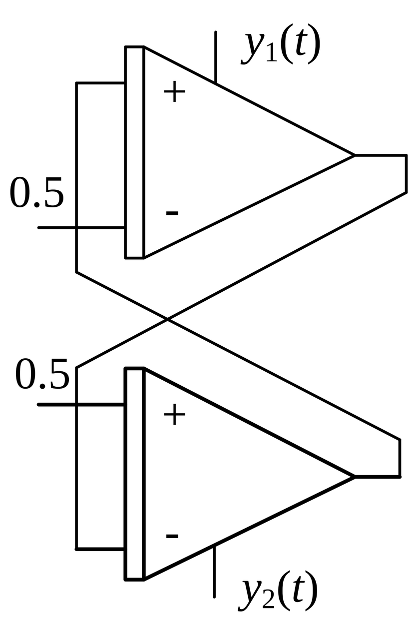


$$\frac{dy(t)}{dt} = 1 - 0 \quad (7)$$

$$\frac{dy(t)}{dt} = t \quad (8)$$

$$\frac{dy(t)}{dt} = \frac{1}{2}t^2 \quad (9)$$

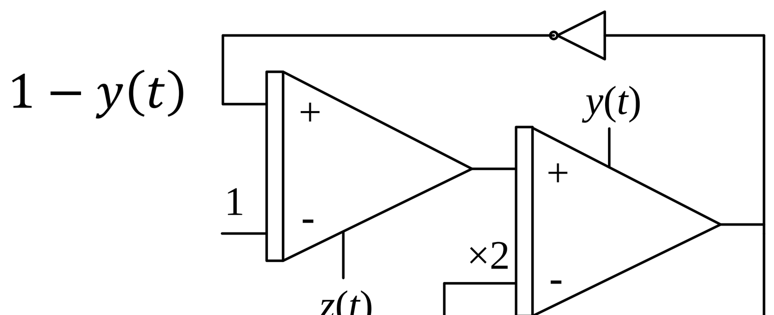
Systems of ODEs



$$\begin{cases} \frac{dy_1(t)}{dt} = y_2(t) - 0.5 \\ \frac{dy_2(t)}{dt} = 0.5 - y_1(t) \end{cases} \quad (10)$$

Higher-order ODEs

$$\frac{d^2y(t)}{dt} + \frac{2dy(t)}{dt} + y(t) = 0 \quad (11)$$



$$\frac{dz(t)}{dt} = -y(t) \qquad \frac{dy(t)}{dt} = z(t) - 2y(t)$$

An auxiliary function z(t) is introduced to satisfy $\frac{dz(t)}{dt} = \frac{d^2y(t)}{dt} + \frac{2dy(t)}{dt}$ to reduce order.

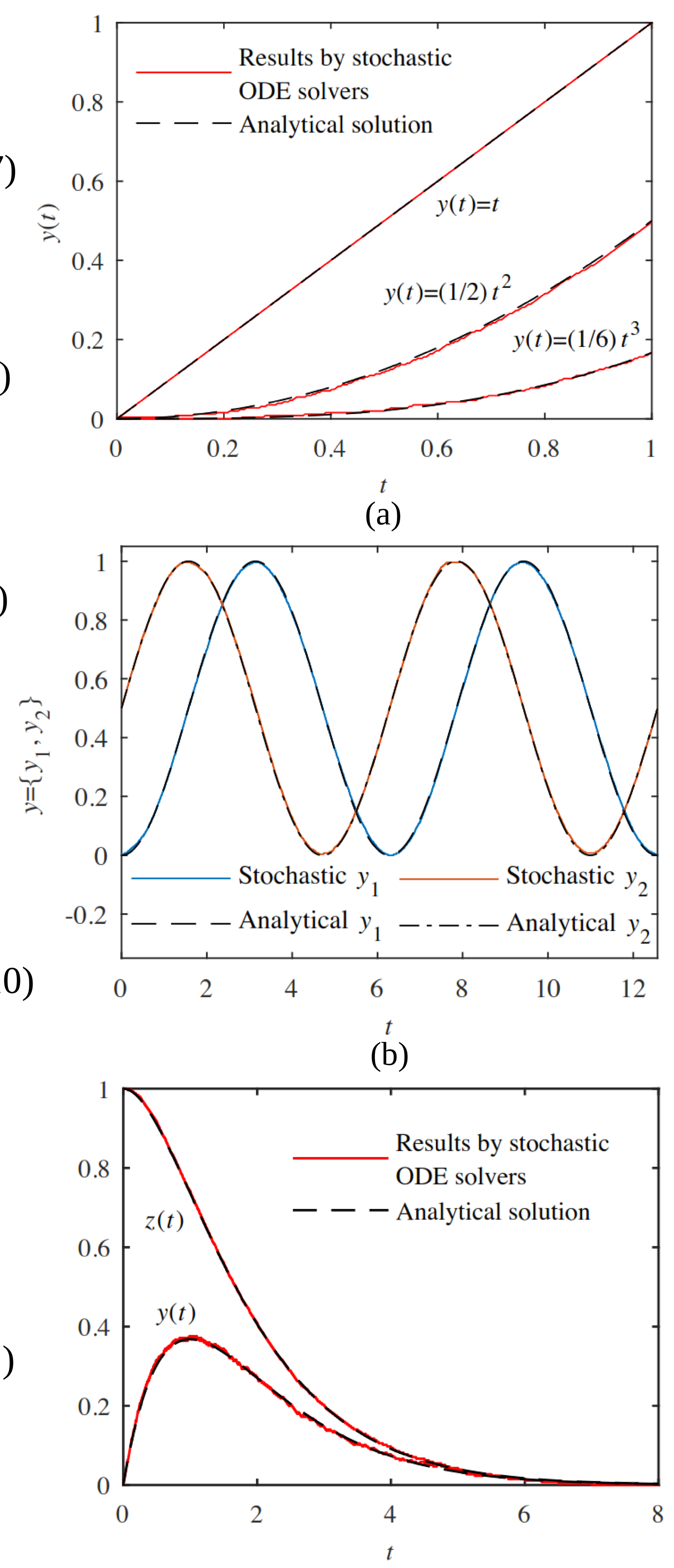
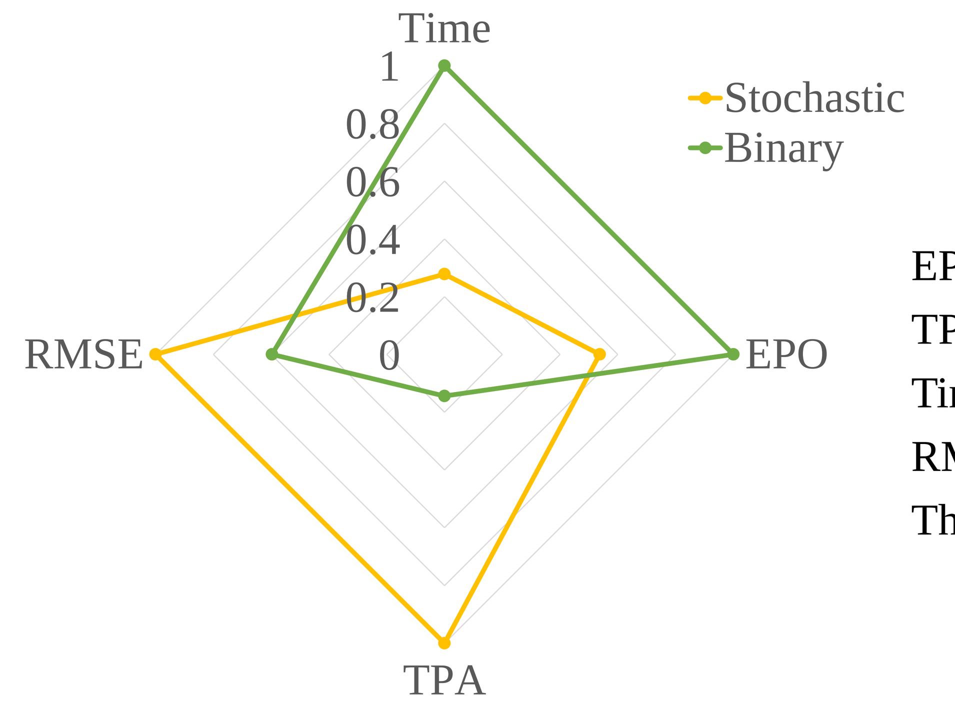


Fig. 5. Results produced by stochastic ODE solvers vs. analytical solution for: (a) (7), (8) and (9); (b) (10); (c) (11).

Hardware Performance



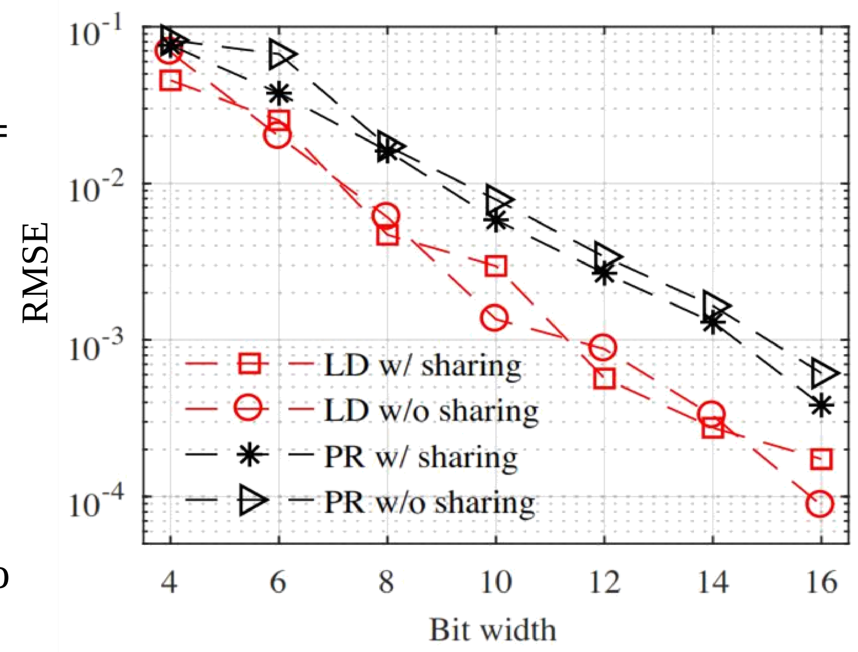
EPO: Energy per operation
TPA: Throughput per area
Time: Total minimum computation time
RMSE: Root-mean-square error
The measurements are all normalized.

Fig. 6. Normalized average performance for stochastic and binary ODE solvers.

Error Assessment

As per $\mathbb{E}[c_k] = c_0 + \frac{1}{2^N}\sum_{i=0}^{k-1}\mathbb{E}[(\boldsymbol{a_i} - \boldsymbol{b_i})] = \hat{y}_k \approx y(\frac{k}{2^N})$, one can

❑ Increase N to decrease the step size $1/2^N$;
❑ Share the RNGs to generate a and b;
❑ Use low-discrepancy (LD) sequences to generate a and b.



*PR: pseudorandom sequences generated by the linear feedback shift registers (LFSRs).

Conclusion

❑ **The stochastic integrator provides unbiased estimation for Euler solution.**
❑ **The stochastic ODE solvers have lower energy consumption, higher throughput and shorter minimum computation time than their binary counterparts with high calculation accuracy.**
❑ **Sharing the RNGs for generating the input stochastic sequences of the stochastic integrator can reduce the random variation.**