REPORT ON: "SIMD-oriented Fast Mersenne Twister: a 128-bit Pseudorandom Number Generator", by Saito and Matsumoto, Received March 2007.

The paper proposes new fast RNGs designed specifically for recent SIMD computers. These RNGs generate their numbers by batches, filling an entire array at a time. They are very fast, much faster than the standard (sequential) implementation of the Mersenne twister (MT), and also faster than a version of MT that generates one array at a time. One drawback is that the new generators are not fully portable; however, there are C implementations that behave in the same way on the most popular high-end processors. This paper would fit well the MCQMC'06 Proceedings after a revision that addresses the issues listed below.

Detailed Comments:

1. In the abstract, replace "at one time" by "in one call".
2. Page 2, line 8: I suggest to remove "and consequently it is not so important to improve the speed..." I did not find any statement like this in [11].
3. Page 3, lines 1–2: spell out MSB and LSB (these abbreviations are undefined).
4. Line 2: remove the "and".
5. Page 5, line 2: "we searched"
6. Replace "LFSR by" by "An LFSR that obeys".
7. Mid-page: replace "discard some fixed r-bits" by "discard r bits" or "discard r specific bits".
8. Mid-page: Using a reducible characteristic polynomial seems to be equivalent to using a combined (or compound) LFSR generator as in references [101, 102, 105, 106] given below. See also [103] for a discussion of this equivalence. This should be clarified in the paper.
9. Page 5, last two lines: Should say explicitly that "Ker" means the kernel, and also use parentheses: "$\mathrm{Ker}(\phi_p(f))$".
10. Page 6, lines 1–2: This decomposition should be better explained. You may want to make the correspondence with the case where we combine two generators, say with periods $2^p - 1$ and $2^r - 1$. Then the kernel of ϕ_p can be viewed essentially the set of states where the r bits of the second component are at zero (or are just neglected).

Thus, V_p has dimension p and corresponds to the state space of the first component, and the decomposition $s = s_p + s_r$ matches with $S = V_p \oplus V_r$. Also: replace the comma by a period after V_r.

11. Pages 6, 7, and bottom of page 8: similar notions of equidistribution have been defined and used earlier; please provide references. For example, you should compare your definitions with those in [101].

12. Page 7: where are the expressions "little-endian" and "big-endian" coming from? What do they mean exactly? Reference?

13. Page 8, Eq. (3): The notation used there is confusing.

14. Line 2: "Take an initial state..."

15. Page 11, Section 6, line 1: read "For an LFSR with ..... g, we observe the following...". A phenomenon is something that we observe, it does not "belong" to a function.

16. Line 3: "for considerable generations" should be rewritten; perhaps "for many steps". The next two lines should be readjusted accordingly.

17. Page 12, line 1 after Figure 2: "among the compared generators."

18. Next line: replace "previously-computed" by "two most recently computed". You can also remove the "i.e." (twice).

19. Page 12, lines 3–7 from below: I am not completely convinced about this. There are other potential problems; for instance if two initializations are very similar for some reason, then the states may stay similar for a long time. Most users do not know (and do not care) about the initialization problems; slow recovery could be dangerous for this reason.

20. Section 7 and the second paragraph of Section 8.2 are on implementation details that should appear earlier. The "block generation" method is compared in Tables 1 and 2, so it should be precisely defined before. All these implementations details should probably appear at the end of section 2, in a special subsection or section.

21. Section 8, line 1: "the SFMT".

22. Section 8.1, line 4: "the PRNG".

23. Section 8.1, line 7: "WELL is much slower" is not true for all the computers in your Tables 1 and 2. (Note that one must look at the two tables to make comparisons.)

24. Section 8.1, line 6 from below: I suppose that this is for one particular statistical test. There are other statistical tests that would require a smaller sample size to detect the linearity. See [104].

25. Section 8.2, line 1: "We prepared"; line 4: "require the icl...".

26. Line 7: "There is a problem of the endian..." should be better formulated. Perhaps "Due to the fact that computers use different endian systems to store 128-bit integers, the 128-bit integers must be converted to 32-bit integers in a different way on these different computers in order to maintain portability. ..."

27. References 1 and 3: read "Mersenne" and "Tausworthe".

28. I have marked several additional minor English corrections on a copy of the manuscript and gave it directly to the authors.

References

[101] P. L'Ecuyer. Maximally equidistributed combined Tausworthe generators. *Mathematics of Computation*, 65(213):203–213, 1996.

[102] P. L'Ecuyer. Tables of maximally equidistributed combined LFSR generators. *Mathematics of Computation*, 68(225):261–269, 1999.

[103] P. L'Ecuyer and F. Panneton. $\mathbf{F}_2$-linear random number generators. Submitted for publication, 2007.

[104] P. L'Ecuyer and R. Simard. TestU01: A C library for empirical testing of random number generators. *ACM Transactions on Mathematical Software*, 2006. to appear.

[105] S. Tezuka and P. L'Ecuyer. Efficient and portable combined Tausworthe random number generators. *ACM Transactions on Modeling and Computer Simulation*, 1(2):99–112, 1991.

[106] D. Wang and A. Compagner. On the use of reducible polynomials as random number generators. *Mathematics of Computation*, 60:363–374, 1993.