The Degree of Masking Fault Tolerance vs. Temporal Redundancy - Erratum -

Nils Müllner and Oliver Theel
Department of Computer Science
Carl von Ossietzky Universität Oldenburg
26111 Oldenburg, Germany
Email: {nils.muellner|oliver.theel}@informatik.uni-oldenburg.de
Phone: +49-441-798-2364
Fax: +49-441-798-2756

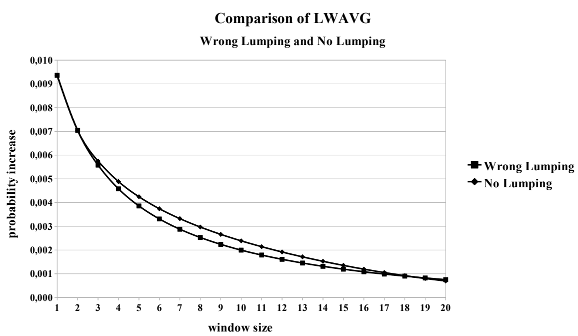


Fig. 1. Wrong Lumping vs. No Lumping

I. Erratum

The contributions of the paper "'The Degree of Masking Fault Tolerance vs. Temporal Redundancy"' [1, ch.4] are 1) limiting window availability (LWA), 2) a method to compute limiting availability, and 3) a lumping method for use with the computation method to reduce the complexity. For the lumping of states we assumed that those states that have the same amount of corrupted processes can be lumped, and that the resulting Markov chain still computes the LWA. In this erratum we show how this assumption is wrong in and present an example, for which we discuss correct lumping.

A. Error in the Lumping Assumption

a) The Proposed Assumption is Wrong: The proposed lumping method [1] results in a reduced Markov chain that does not compute the LWA of the system.

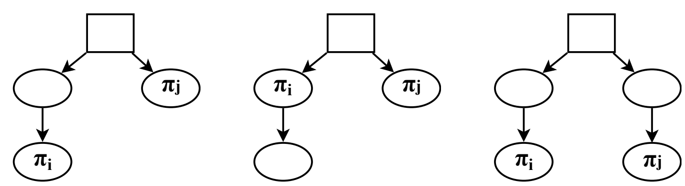
b) Example: We compute the first twenty instances of the LWA, once without lumping and once with the lumping proposed, for the example given in the paper. We derive the LWA vector and its gradient for both instances. The limiting window availability vector gradients the shown in Figure 1 (for demonstration we present the graph and not the numbers). The graphs are different, and hence the lumping proposed is wrong.

B. A Correct Lumping Assumption

Having the same amount of corrupt states is not a sufficient condition for states to be lumped. We refrain to provide the definition of the whole legally lumpability criterion. The definition thereof is too complex for a short erratum paper. We show that correct lumping (in the sense that lumped Markov chains have an equal potential to compute the correct LWA) can be possible.

C. A Suitable Example

For correct lumping we seek processes that mimic each other's behavior. By mimicking we refer to the property of two processes (or connected sets of processes) that are affected by the system in the same fashion, and affect the system in the same fashion.



(a) Different Input (b) Different Output (c) Both

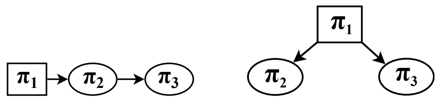
Fig. 2. Equivalence Class and Parent Treatment

Figure 4 shows three topologies. We cannot lump processes π_i and π_j in Figure 2(a), or in Figure 2(b) respectively. The former case is not lumpable as both processes are treated by the system in different ways. The latter case is not lumpable as the processes have different effects on the system. Figure 2(c) shows a topology in which processes π_i and π_j are lumpable (as are their respective predecessors), given each non-root process executes the same algorithm and has the same fault and execution model.

Generally, we seek system symmetries in parallelisms. Serial topology (cf. Figure 3(a)) in general cannot be lumped (while maintaining the property to compute the LWA). Parallel systems (cf. Figure 3(b)) on the other hand, in which processes mimic each others behavior, are the point of attack. We use the exact same algorithm and fault model presented in [1]. Instead of using a serial topology (as presented in paper [1]) we employ the parallel topology shown in Figure 3(b), in which processes π_2 and π_3 mimic each other's behavior.

For the computation of the LWA, it is irrelevant whether π_2 or π_3 is corrupt. Hence, we lump

- states $\langle 0,0,2\rangle$ and $\langle 0,2,0\rangle$ into $\langle 0,\mathbb{1}\rangle$, and afterwards



(a) Serial Topology (b) Parallel Topology

Fig. 3. Lumping Restrictions: Symmetry

↓from/to→	$\langle 0,0,0 \rangle$	$\langle 0,0,2 \rangle$	$\langle 0,2,0 \rangle$	$\langle 2,0,0 \rangle$
$\langle 0,0,0 \rangle$	0.99	$0.00\overline{3}$	$0.00\overline{3}$	$0.00\overline{3}$
$\langle 0,0,2 \rangle$	0.33	$0.66\overline{3}$		
$\langle 0,2,0 \rangle$	0.33		$0.66\overline{3}$	
$\langle 2,0,0 \rangle$	0.33			$0.00\overline{3}$
$\langle 0,2,2 \rangle$		0.33	0.33	
$\langle 2,0,2 \rangle$		0.33		
$\langle 2,2,0 \rangle$			0.33	
↓from/to→	$\langle 0,2,2 \rangle$	$\langle 2,0,2 \rangle$	$\langle 2,2,0 \rangle$	$\langle 2,2,2 \rangle$
$\langle 0,0,2 \rangle$	$0.00\overline{3}$	$0.00\overline{3}$		
$\langle 0,2,0 \rangle$	$0.00\overline{3}$		$0.00\overline{3}$	
$\langle 2,0,0 \rangle$		$0.\overline{3}$	$0.\overline{3}$	
$\langle 0,2,2 \rangle$	$0.33\overline{6}$			$0.00\overline{3}$
$\langle 2,0,2 \rangle$		$0.33\overline{6}$		$0.\overline{3}$
$\langle 2,2,0 \rangle$			$0.33\overline{6}$	$0.\overline{3}$
$\langle 2,2,2 \rangle$	0.33			0.67

TABLE I
MARKOV CHAIN OF THREE PARALLEL PROCESS SYSTEM

- states $\langle 2,0,2 \rangle$ and $\langle 2,2,0 \rangle$ into $\langle 2,\mathbb{1} \rangle$.

In the further proceedings we use numerical values $p = 0.99$ and $e_i = \frac{1}{n}, 1 \leq i \leq n$.

A strong indicator for processes to be mimicking each others behavior is an exactly equal steady state probability distribution. Having an equal distribution is a mandatory precondition. Yet, solely it is not sufficient.

The lumped Markov chain is shown in Table III.

State	Steady State Probability
$\langle 0,0,0 \rangle$	0.96222934665
$\langle 0,0,2 \rangle$	0.01297015335
$\langle 0,2,0 \rangle$	0.01297015335
$\langle 2,0,0 \rangle$	0.00321815835
$\langle 0,2,2 \rangle$	0.00183034665
$\langle 2,0,2 \rangle$	0.00168234165
$\langle 2,2,0 \rangle$	0.00168234165
$\langle 2,2,2 \rangle$	0.00341715835

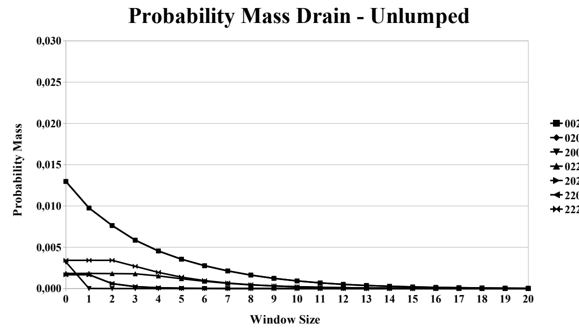
TABLE II
STEADY STATE PROBABILITY OF THREE PARALLEL PROCESS SYSTEM

↓from/to→	$\langle 0,0,0 \rangle$	$\langle 0,\mathbb{1} \rangle$	$\langle 2,0,0 \rangle$	$\langle 0,2,2 \rangle$	$\langle 2,\mathbb{1} \rangle$	$\langle 2,2,2 \rangle$
$\langle 0,0,0 \rangle$	0.99	$0.00\overline{6}$	$0.00\overline{3}$			
$\langle 0,\mathbb{1} \rangle$	0.33	$0.66\overline{3}$		$0.00\overline{3}$	$0.00\overline{3}$	
$\langle 2,0,0 \rangle$	0.33		$0.00\overline{3}$		$0.\overline{6}$	
$\langle 0,2,2 \rangle$		0.66		$0.33\overline{6}$		$0.00\overline{3}$
$\langle 2,\mathbb{1} \rangle$		0.33			$0.33\overline{6}$	$0.\overline{3}$
$\langle 2,2,2 \rangle$				0.33		0.67

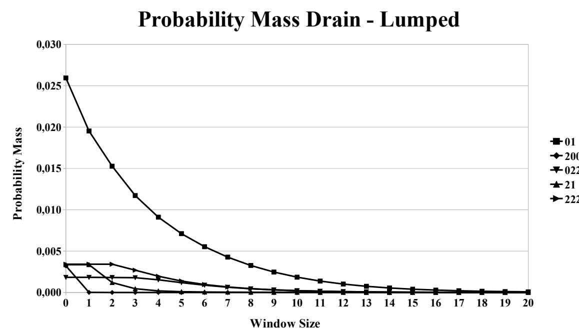
TABLE III
LUMPED MARKOV CHAIN OF THREE PARALLEL PROCESS SYSTEM

We now compute the LWA for both with window size 20 and compare the results.

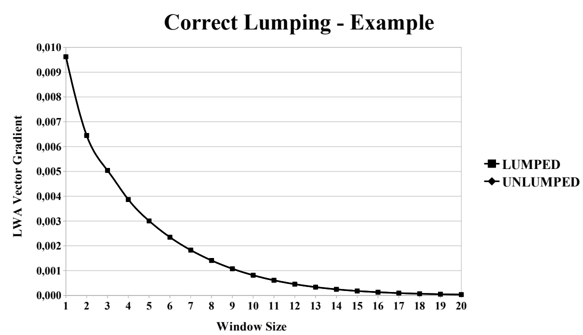
Figure 4(a) shows the probability mass drain for each state (except $\langle 0,0,0 \rangle$) for the first 20 time steps for the original chain (Figure 4(b) respectively for the lumped chain). In Figure 4(a) we observe, that states $\langle 0,0,2 \rangle$ and $\langle 0,2,0 \rangle$ (states $\langle 2,0,2 \rangle$ and $\langle 2,2,0 \rangle$ respectively) show the same trajectory. In Figure 4(b) their respective trajectories add up as expected.



(a) Probability Mass Drain - Original Chain



(b) Probability Mass Drain - Lumped Chain



(c) Limiting Window Availability Vector Gradient Comparison

Fig. 4. Results

Finally, we compare the LWA vector gradients and observe, that the lumped Markov chain exhibits the exact same behavior as the original chain. We have presented an example for which lumping is a legal method to compute the LWA.

REFERENCES

[1] N. Müllner and O. Theel. The Degree of Masking Fault Tolerance vs. Temporal Redundancy. In *Proceedings of the 2011 IEEE 25th International Conference on Advanced Information Networking and Applications Workshops*, WAINA '11, Washington, DC, USA, 2011. IEEE Computer Society.