QEST2021 - Supplemental Material

Riley Roberts[1] [0000−0002−8676−3767], Thakur Neupane[2] [0000−0002−1870−4079], Lukas Buecherl[3] [0000−0002−4844−6605], Chris J. Myers[3] [0000−0002−8762−8444], and Zhen Zhang[1] [0000−0002−8269−9489]

[1] Utah State University, Logan, UT, USA
riley.roberts@aggiemail.usu.edu, zhen.zhang@usu.edu
[2] The MathWorks, Inc., Natick, MA, USA, tneupane@mathworks.com
[3] University of Colorado Boulder, lukas.buecherl@colorado.edu, chris.myers@colorado.edu

1 Algorithm

This presents a detailed algorithm for STAMINA 2.0. Refer to the QEST 2021 submission for information on the symbols used. $explored$ is a set used to indicate the states that have been explored using one particular value for κ.

Algorithm 1: Improved state re-exploration aglorithm in STAMINA 2.0.

Input : A PRISM CTMC model file, a CSL property, and w.
Output: $\mathsf{P}_{\mathsf{min}}$ and $\mathsf{P}_{\mathsf{max}}$.
$\mathsf{P}_{\mathsf{min}} := 0.0$; $\mathsf{P}_{\mathsf{max}} := 1.0$; $\pi(s_0) := 1.0$; $\mathbf{S} := \{s\}$; $\mathbf{T} := \{s\}$;
while $\mathsf{P}_{\mathsf{max}} - \mathsf{P}_{\mathsf{min}} > w$ **do**
  $\Pi := 1.0$;
  while $\Pi > \frac{w}{m}$ **do**
    $enqueue(queue, s_0)$; $explored := \emptyset$;
    while $queue \neq \emptyset$ **do**
      $s := dequeue(queue)$;
      if $s \notin \mathbf{T} \ \vee \ \pi(s) \geq \kappa$ **then**
        if $\pi(s) == 0$ **then**
          ▷ $post(s)$ is computed from the PRISM file
          forall $s' \in post(s)$ **do**
            $enqueue(queue, s')$;
        else
          if $s \in \mathbf{T}$ **then**
            $\mathbf{T}.remove(s)$;
          forall $s' \in post(s)$ **do**
            $\pi(s') := \pi(s') + \pi(s) \cdot p(s, s')$;
            if $(s' \in \mathbf{S} \wedge s' \notin explored) \vee (s' \notin \mathbf{S})$ **then**
              $explored := explored \cup \{s'\}$;
              $enequeue(queue, s')$;
              if $s' \notin \mathbf{S}$ **then**
                $\mathbf{T} := \mathbf{T} \cup \{s'\}$; $\mathbf{S} := \mathbf{S} \cup \{s'\}$;
          $\pi(s) := 0$;
    $\Pi := \sum_{s_i \in \mathbf{T}} \pi(s_i)$;
    $\kappa := \frac{\kappa}{r_\kappa}$;
  Instruct PRISM to build the proper statespace based on the states in $\mathbf{S}$ and $\mathbf{T}$, and the original inputted PRISM model;
  Compute $\mathsf{P}_{\mathsf{min}}$ and $\mathsf{P}_{\mathsf{max}}$ of the inputted CSL propery, using PRISM;
  if $\mathsf{P}_{\mathsf{max}} - \mathsf{P}_{\mathsf{min}} > w$ **then**
    Increase m proportionally to the difference between $\mathsf{P}_{\mathsf{max}} - \mathsf{P}_{\mathsf{min}}$ and w;

2 Tables

This section presents tables with the full, detailed results of the comparison between STAMINA 1.0 and STAMINA 2.0

Table 1: Performance of STAMINA 2.0 relative to STAMINA 1.0 on the hazard circuit. A $\star$ by the improvement indicates STAMINA 1.0 could not achieve the desired w within 10 iterations; † indicates STAMINA 1.0 did not complete due to memory constraints.

Glitch	Transition	STAMINA 2.0				STAMINA 1.0				Improvement	
		$\lvert\mathcal{G}\rvert$ (K)	T (C/A)	P_{min}	P_{max}	$\lvert\mathcal{G}\rvert$ (K)	T (C/A)	P_{min}	P_{max}	$\lvert\mathcal{G}\rvert$ $(\%)$	T $(\%)$
Glitch Zero	$(0,1,0) \rightarrow (1,1,1)$	$3,527$	$1006/2872$	0.0166	0.0168	$8,382$	$5708/41,592$	0.0060	0.9218	$\star$**57.9**	$\star$**91.8**
	$(0,1,0) \rightarrow (1,0,0)$	85	$13/35$	0.3950	0.3951	933	$224/706$	0.3951	0.3960	**90.9**	**94.8**
	$(1,1,1) \rightarrow (1,0,0)$	406	$86/184$	0.7357	0.7358	$3,464$	$1,936/4,595$	0.7351	0.7361	**88.3**	**95.9**
	$(1,1,1) \rightarrow (0,1,0)$	468	$97/219$	0.6947	0.6948	$6,929$	$829/1,464$	0.6947	0.6947	**93.2**	**86.2**
	$(1,0,0) \rightarrow (0,1,0)$	165	$30/76$	0.4550	0.4551	$3,408$	$2,133/9,892$	0.4550	0.4555	**95.2**	**99.1**
	$(1,0,0) \rightarrow (1,1,1)$	$3,569$	$1021/2850$	0.0166	0.0168	$9,280$	$6,029/42,336$	0.0125	0.5405	$\star$**61.5**	$\star$**92.0**
Glitch One	$(0,1,1) \rightarrow (1,0,1)$	$2,813$	$818/1988$	0.9895	0.9897	$11,462$	$8,102/65,546$	0.8608	0.9990	$\star$**75.5**	$\star$**96.2**
	$(0,0,0) \rightarrow (0,1,1)$	$2,544$	$590/2034$	0.8260	0.8262	$14,408$	$7,611/54,156$	0.6661	0.9669	†**82.3**	†**95.8**
	$(0,0,0) \rightarrow (1,0,1)$	$2,830$	$810/2,165$	0.9902	0.9905	$9,406$	$4,976/46,219$	0.9477	0.9998	$\star$**69.9**	$\star$**94.2**
	$(1,0,1) \rightarrow (0,1,1)$	$3,006$	$821/2,170$	0.9895	0.9898	$17,994$	$6,124/41,362$	0.8498	0.9981	†**83.3**	†**93.7**
	$(0,1,1) \rightarrow (0,0,0)$	381	$70/174$	0.8574	0.8575	$7,077$	$3,868/23,541$	0.8574	0.8580	**94.6**	**99.1**
	$(1,0,1) \rightarrow (0,0,0)$	328	$59/151$	0.8644	0.8645	$8,165$	$3,611/24,023$	0.8642	0.8652	**96.0**	**99.2**

Table 2: Performance STAMINA 2.0 relative to STAMINA 1.0 on the benchmarks.

Model	Params	STAMINA 2.0				STAMINA 1.0				Improvement	
		$\lvert\mathcal{G}\rvert$ (K)	T (C/A)	P_{min}	P_{max}	$\lvert\mathcal{G}\rvert$ (K)	T (C/A)	P_{min}	P_{max}	$\lvert\mathcal{G}\rvert$ (%)	T (%)
Robot (n/K)	32/ 64	474	51/ 165	0.9755	0.9756	696	38/ 321	0.9756	0.9756	**31.9**	**39.8**
	32/ 1024	474	51/ 167	0.9755	0.9756	696	37/ 329	0.9756	0.9756	**31.9**	**40.4**
	64/ 64	$1,562$	139/ 354	$2.94e$-5	$1.78e$-4	$2,273$	123/ 870	$1.46e$-4	$1.68e$-4	**31.3**	**50.4**
	64/ 1024	$1,562$	138/ 375	$2.94e$-5	$1.78e$-4	$2,273$	121/ 829	$1.46e$-4	$1.68e$-4	**31.3**	**46.0**
Jackson (N/λ)	4/ 5	187	15/ 19	0.8654	0.8655	167	18/ 41	0.8653	0.8657	**-10.7**	**42.4**
	5/ 5	$1,480$	176/ 273	0.8194	0.8202	$6,141$	$1,852$/ $2,606$	0.8197	0.8197	**75.9**	**89.9**
Polling (N)	12	0.001	0.016/ 0.018	1.0	1.0	19	3/ 24	1.0	1.0	**99.9**	**99.9**
	16	0.001	0.017/ 0.012	1.0	1.0	57	17/ 79	1.0	1.0	**99.9**	**99.9**
	20	0.001	0.018/ 0.019	1.0	1.0	113	25/ 149	1.0	1.0	**99.9**	**99.9**
Tandem (c)	2047	21	1/ 13	0.4990	0.4990	25	0.3/ 26	0.4990	0.4990	**16**	**46.8**
	4095	42	2/ 51	0.4993	0.4993	50	1/ 117	0.499	0.499	**16**	**55.1**
Toggle (c)	0	4.2	0.3/ 0.9	0.0131	0.0131	4.2	0.3/ 3.2	0.0131	0.0136	**0**	**65.7**
	100	6.7	0.5/ 1.9	0.9918	0.9918	7.6	0.4/ 5.4	0.9917	0.9918	**11.8**	**58.6**