Feedback to HW9

Aurora

December 7, 2016

In this solution, denote `Int1` with i_1 and `Int2` with i_2.

1 Semaphore

1.1 How many threads will exit from `wait(c)`

$i_1 \mod 4 + 4$

1.2 Most blocks accessed

$i_1 \mod 4 + 3$

1.3 Filling the blanks in 2nd version

In `init`: $c.h = \text{Semaphore}(0)$
In `wait`: $\text{V}(c.h)$
In `signal`: $\text{P}(c.h)$
In `broadcast`: $\text{P}(c.h)$

2 Stable Network

2.1 Stable solution?

Yes, A: ABE, B: BE, C: CDE, D: DE

2.2 Stable solution 2?

No. Consider the following case:

A: AE

B: BAE (AE is found to be valid)
D: DE
C: CDE
A: ADE (DE is found to be valid and ADE is preferred to AE)
B: BE (AE is found to be invalid)
C: CBE (BE is found to be valid and CBE is preferred to CDE)
D: DCBE (CBE is found to be valid and DCBE is preferred to DE)
A: AE (DE is found to be invalid)
... (infinitely loop)

Any initialization is acceptable. All will fall into this loop.

2.3 Temporary forwarding loop

$i_1 = 0$: A and D will see temporary forwarding loop.
$i_1 = 1$: A and D will see temporary forwarding loop. or A, B, D
$i_1 \geq 2$: A, B, D

3 New Product

Students are supposed to answer this question with one of following assumptions:

- **Each block has only one bit** i.e. all bits in a block flip at the same time

  In this case each block only has two states: `correct` or `failed`.

  The answer for this case is: Q3.1: 1, Q3.2: 1.

  Analysis graph are shown in **appendix**.

- **Each block has a lot of bits, and each of them can be flipped independently**
  - 1 block fails: can be detected and (located + recovered);
  - 2 blocks fail:
    - in same line or same row: can be detected and (located + recovered);
    - in different lines and different rows: can be detected, perhaps can be located and recovered by trail and error;
  - 3 blocks fail: (If students are not considering case as detectable, i.e. giving answer 2, I think is also reasonable.)
    - all in same line or same row: can be detected and located;
    - a and b in same line, b and c in the same row: can be detected, perhaps can be located and recovered by trail and error;
    - a and b in the same line/row, c in another line/row and in the third row/line: can be detected (there is error), the number cannot be determined(there may be 3 to 6 failure), cannot be located, cannot be recovered;
    - all three are in diff lines and rows: can be detected (there is error), the number cannot be determined(there may be 3 to 9 failure), cannot be located, cannot be recovered;
  - more than 4 blocks fail: can be detected (the exact number may not be detected), cannot be recovered.

4 Appendix

Case analysis: 1 bit per block

WLOG, assume the original state to be like this:

0	0	0	√
0	0	0	√
0	0	0	√
√	√	√	

- **One block failure**

1	0	0	×
0	0	0	√
0	0	0	√
×	√	√	

Table 1: 1 failure - detectable and recoverable

- **Two block failure**

1	**1**	0	√
0	0	0	√
0	0	0	√
×	×	√	

0	0	0	√
1	**1**	0	√
0	0	0	√
×	×	√	

0	0	0	√
0	0	0	√
1	**1**	0	√
×	×	√	

Table 2: 2 failures - indistinguishable

- **Three block failures**
- **Four failures**

You can do this for 5 or more than 5 failures.

1	0	0	×
0	**1**	0	×
0	0	0	√
×	×	√	

0	**1**	0	×
1	0	0	×
0	0	0	√
×	×	√	

Table 3: 2 failures - indistinguishable

1	**1**	**1**	×
0	0	0	√
0	0	0	√
×	×	×	

0	0	**1**	×
1	**1**	0	√
0	0	0	√
×	×	×	

Table 4: 3 failures - indistinguishable

1	0	0	×
1	**1**	0	√
0	0	0	√
×	√	√	

1	0	0	×
0	0	0	√
0	0	0	√
×	√	√	

Table 5: 3 failures - indistinguishable from 1 failure case

1	**1**	0	√
1	**1**	0	√
0	0	0	√
√	√	√	

Table 6: 4 failures - indistinguishable from original case

1	**1**	**1**	×
1	0	0	×
0	0	0	√
√	×	×	

0	0	**1**	×
1	0	0	×
1	**1**	0	√
√	×	×	

Table 7: 4 failures - indistinguishable

1	**1**	0	√
1	0	**1**	√
0	0	0	√
√	×	×	

0	**1**	**1**	√
0	0	0	√
0	0	0	√
√	×	×	

Table 8: 4 failures - indistinguishable from 2-failure case