Errata, second printing
Nancy Lynch

Note: Page and line numbers below are in the FIRST PRINTING.

p. xv:

Section title for 19.2.2 should have ChandyLamport in italics, and without a hyphen.

Section title for 20.1.5 should have LogicalTimeME in italics.

p. 35:

p. 35, line 8: "are left as an exercise (Exercise 3.6)".

p. 48:

Add a new exercise, after the current 3.5:

"Prove that the total time complexity of the HS algorithm is at most 3n if n is a power of 2, and 5n otherwise."

p. 49:

Exercise 3.19: Change the last line: "...messages required by a comparison-based algorithm in which all the processes compute n mod 2."

p. 61:

p. 61, line 13: "and the number of messages is $O(diam|E|)$, again because at most $|E|$ messages can be sent at each of the *diam* rounds. The number of bits is at most $O(n|E|b)$, where b is the maximum number of bits used to represent a single UID."

p. 61, line 19: "time is $O(diam)$, the number of messages is $O(diam|E|)$, and the number of bits is $O(n|E|b)$."

p. 91:

p. 91, line 8: Replace line of code with: "if, for some $j, k_j neq$ undefined then..."

p. 91, line 10: Replace line of code with: "if, for some $i', V_{i'}(j) neq$ undefined then ..."

p. 91, line 11: Replace line of code with: "if, for some $i', L_{i'}(j) > level(j)$ then ..." Also, put a subscript i' after the max in this line.

p. 120:

p. 120, line 16: Remove italics from "majority".

p. 147:

p. 147, line 13: "...input 1, ... input 0, and..." (The 1 and 0 should be interchanged.)

p. 154:

After Exercise 6.5, add another exercise: If the FloodSet algorithm is run for only f rounds instead of f+1, what is the largest number of different decisions that can be reached by nonfaulty processes?

p. 155:

After Exercise 6.13, add a new exercise: If the EIGStop algorithm is run for only f rounds instead of f+1, what is the largest number of different decisions that can be reached by nonfaulty processes?

p. 159:

Exercise 6.41 should have periods at the end of each of parts (a) and (b).

p. 179:

Lemma 7.16: "Suppose that ...ConvergeApproxAgreement, where i is a nonfaulty process. Then..."

p. 212:

line 6-7: Replace four occurrences of Σ with Φ.

p. 234:

Add a new exercise, after 8.16:

"Consider a channel D, which is similar to channel C on p. 204, except that it allows internal message duplication.

More specifically, in addition to the *send* and *receive* actions, D has two internal actions, *duplicate* and *discard*. When a $send(m)$ occurs, the message m is added to the end of the queue along with a Boolean tag. Tags for successive messages that are sent alternate, $1, 0, 1, 0, \ldots$. A *duplicate* causes an arbitrary message in the queue to be duplicated in place, along with its tag. The channel also keeps track of the tag of the last message delivered. A *receive* delivers the first message on the queue, as before, but only if the tag is unequal to that of the last message delivered. A *discard* discards the first message on the queue, provided the tag is the same as that of the last message delivered. (a) Give formal code for automaton D, in the same style as the other code in this chapter. (b) Prove carefully that D implements C, in the sense of inclusion of sets of traces. Use a simulation relation."

p. 296:

line -7: Replace "It does not exactly guarantee bounded bypoass, but rather guarantees a related condition—" with "It guarantees bounded bypass, and also a related condition—"

p. 330:

Add another exercise, after the current Exercise 10.23, saying:

"Prove that the Bakery algorithm guarantees bounded bypass."

p. 335

line -11: "of users that include at least two users with overlapping resource requirements."

p. 368

Exercise 11.2(a): Should refer to Exercise 10.13, not 10.30.

p. 425

line 6: "returned by the second OF THESE TWO SETS of reads (which..."

line 7: "by the first OF THE TWO SETS of reads"

p. 435: [[[A major simplification. We can omit condition 1 in the statement of Lemma 16. It is implied by condition 2, because beta is a sequence.

There are many changes in the book that result from this observation. They all need to be made to make this change consistent:

p. 434, line -1: Replace "four" by "three".

p. 435, line 1: Replace "four" by "three".

Statement of Lemma 13.16: Remove the first condition and renumber the others.

After the Lemma, remove the first sentence, explaining Condition 1, and renumber the conditions 2, 3, and 4 in the next few lines as 1, 2, and 3.

Modify the proof to begin with:

"We begin by claiming that for any operation pi in Pi, there are only finitely many operations phi such that phi ¡ pi. This is because, if there were infinitely many such operations phi, then by Condition 1, all of their invocation events would have to precede the response event of beta. (The fact that beta contains no incomplete operations implies that such a response event does in fact appear in beta.) But beta is a sequence, so no element can have infinitely many predecessors, a contradiction.

Now we describe how to insert a serializaation ...."

p. 435, line -6: Instead of "Condition 1 implies", say "The claim at the beginning of this proof implies".

p. 436, line 13: Condition 1

p. 436, line 18: Condition 2

p. 436, line 19: Condition 3

p. 439, line 14: Replace “four” with “three”.

p. 439: Remove the entire proof numbered 1, which spans from line 16 to line -7 on the page. Renumber the remaning three parts as 1, 2 and 3 instead of 2, 3 and 4.

p. 443, line -3: Replace “four” with “three”.

p. 443, line -2 and -1: “Conditions 2 and 3 are immediate, so all we must show is Condition 1. For this, the following...”

p. 444, line -15 and -14: Omit the tiny paragraph “Using Claims 13.24 and...an exercise”

p. 453: Omit Exercise 13.27

p. 444, line 13: Replace “For Condition 2,” with “To show Condition 1,”

p. 580, line 1: Replace “four” with “three”.]]]

p. 437:

line 3 below figure: For the “index” line, say “initially 1” instead of “initially arbitrary”.

p. 483:

line 23: “remain active at...” Remove “s”.

p. 496:

line 6: Rewrite point 1. as: “Asynchronous broadcast and convergecast, based on a searching algorithm (Section 15.3).”

p. 506

line 9: First “l” is in the wrong font.

p. 513:

In the “merge” definition, the last line should read: “The new component is assigned level = level(C) + 1.” The k should be replace by “level(C)”.

p. 536

line -9: “dependencies in the fair trace β to ensure...”

p. 540

p. 540, line -11: “occurs in α, and so, by the correspondence, in α', a contradiction.”

p. 553

line 2

"is much less than the height of ..." (Omit word "maximum".)

p. 599:

line 4: Change period after "N+" to a semicolon, and add "the ordering of triples is lexicographic."

line -18: "locally controlled, non-clock step," (Add "non-clock".)

p. 614:

Problem 18.15, line 5: "should should" should be "should"

p. 627:

Section title for 19.2.2 should have ChandyLamport in italics, and without a hyphen.

p. 647:

p. 647, line 3: "logical time, and moreover, that there never will be one.)" (Add the phrase "and moreover...one" inside the parens.)

p. 649:

Section title for section 20.1.5: LogicalTimeME should be in boldface italics.

p. 678:

In diagram, "broadcast channel" should be plural, "broadcast channels".

p. 679:

In transition definition for $bcast(w, I)_i$, add a third line to the precondition: "$val(i) neq null$"

p. 693:

p. 693, line -4:

"message (repeatedly) by sending its..." (Add word "repeatedly".)

p. 698, line 12: "message (repeatedly) by sending its..." (Add word "repeatedly".)

p. 710:

p. 710, line -12: Omit word "Then" at the beginning of the sentence. It should read simply: "The fact that P_2 keeps sending..."

p. 720:

p. 720, line 16: Remove subscript of 1 from $alpha'_1$. This should just say: "Also, $alpha'$ contains no unmatched CRASH events."

p. 720, line -14: Parenthetical comment beginning with "That is": Remove the parens, and restate this comment as: "More precisely, there are no following SEND, RECEIVE, CRASH or RECOVER events, and there are no unmatched CRASH events."

p. 731:

Add a new exercise, after the current Exercise 22.13:

"Prove that the Probe protocol is not message bounded. Do this by showing, for each k, how to produce a complete execution α that has no k-extensions."