And there is one bigger item that needs to be corrected. The technical discussion in Section 5.4.1 on page 71 is flawed. The updated discussion is the following:

To avoid the sources of timing effects in Theorem 5.6, we need to ensure that there are no stalls from instruction \( I_1 \), and that \( I_1 \) is not parallel to any of the successor instructions. This is ensured by working with “rigid” instruction models, so that no stalls can occur in the concatenation of instructions, and by making each instruction (or node) use all pipeline stages. The following pictures shows an example of such a conservative model, first the model without extra pipeline stages, and then with:

![Conservative model](image1)

The result is an overapproximation: node A would not be allowed to completely overlap node B (as shown for the sequence AB), which makes the timing effect between the nodes minus two, instead of the minus three if we did not use conservative assumptions. Also, B and C do not overlap as
much as in the precise model. Thus, the potential positive timing effect of the interference between A and C is taken early, on the edge between A and B, which is safe but pessimistic. In a precise model, the execution time for ABC would be 11 cycles, but here we get 12.

If a processor has instructions that can execute for quite a long time in parallel to other instructions, such a pairwise model is likely to give very high overestimations. It should also be noted that the precise placement of the extra pipeline stage uses for each instruction can have a big impact on the actual extent of the overapproximation.