Errata regarding

On the Time-Complexity of Broadcast in Radio Networks: An Exponential Gap Between Determinism and Randomization

Reuven Bar-Yehuda* Oded Goldreich† Alon Itai*

December 3, 2002

Summary: As pointed out by Kowalski and Pelc (*FOCS*, 2002), there is an error in our paper (which appeared in the *JCSS*, 1992). The error is due to a *gap between two reasonable models of radio communication* without collision-detection mechanisms. Specifically, this effects the linear-time lower-bound claimed in our paper, which does hold for one model but not for the other related model (which unfortunately is the model stated in the paper).

The difference between the two models is in the treatment of the case in which several neighbors of a potential receiver transmit at the same time. In one model (formulated below), the result may be arbitrary (i.e., either one transmission is received or nothing is received (like in case of no transmission)). In the second model (formulated in the original paper), the result is that nothing is received.

1 High level description and discussion

The errata refers to our paper *On the Time-Complexity of Broadcast in Radio Networks: An Exponential Gap Between Determinism and Randomization*, which has appeared in the Journal of Computer and system Sciences, Vol. 45, (1992), pages 104–126. Specifically, we refer to the linear lower-bound on the *deterministic* time-complexity of broadcast (in radio networks), which is claimed in that paper.

1.1 Two models

The said lower-bound is valid in a reasonable model of radio communication, but (as shown by Kowalski and Pelc, *FOCS'02*) not in the model stated in the original work. The different between these two models refers to what is postulated to happen in case several neighbors of a potential receiver choose to transmit in the same time (or round). Recall that there are three possible cases (w.r.t the number of transmitting neighbors):

*Department of Computer Science, Technion – Israel Institute of Technology, Haifa, ISRAEL.
†Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, ISRAEL.

1. None of the neighbors transmits.

   Clearly, in this case, the receiver obtains no message.

2. Exactly one of the neighbors transmits.

   In this case, the receiver obtains the message.

3. Several (i.e., at least two) of the neighbors transmits.

   The issue is what happens in this case.

If *conflict detection* mechanisms are available in the network (and are used by the receiver), then the receiver can distinguish the above three cases (and in particular may distinguish the third case from the first two cases that are definitely distinguishable). Our work was aimed at modeling networks in which conflict detection mechanisms are not available. Intuitively, in such networks, the third case *may be* indistinguishable from the first case. But the question we raise here is whether this means that the third case is *always* indistinguishable from the first case. Two possible models emerge.

Model A: as in the original work. In this model it is postulated that the case in which two or more neighbors transmit is always indistinguishable from the case in which no neighbor transmits. The rational is that in both cases, the potential receiver hears noise, which is always present also in case nobody transmits.

Model B: different than in the original work. In this model it is postulated that in case two or more neighbors transmit the result may be either that one of these transmissions is received (like in a case of a transmission by a single neighbor) or that nothing is received (as in case that no neighbor transmits). The rational is that the case of multiple transmission is a bad event and it is unpredictable what its outcome may be. Furthermore, postulating the the outcome is *always indistinguishable* from the case in which no neighbor transmits means that one can distinguish the case of single transmission from the case of multiple transmission. This seems to be a strong assumption, which may not hold in some communication networks.

Clearly, an execution under Model A is also a valid execution under Model B, but the converse does not hold. Note that in Model A the message delivery events are fully determined by the number of neighbors that transmit, whereas in Model B in some cases (i.e., multiple transmitters) delivery is decided non-deterministically (i.e., by an adversary).

We note that our original intuitions about radio networks were more along the lines of Model B. However, since Model A is simpler to formulate and negative results regarding Model A certainly hold for Model B, we preferred at the time to state our negative results with respect to Model A. Such a choice would have been justified if the negative results were to hold also for Model A.

1.2 The main facts

The main facts are as follows:

1. The lower-bound claimed in the original work *does not hold for Model A*.

   (For details see the work of Kowalski and Pelc, *FOCS'02*.)

2. In contrast, the lower-bound claimed in the original work *does hold for Model B*.

The flaw in the original lower-bound proof is due to a single point; specifically, to the proof of Lemma 7. The rather laconic proof of Lemma 7 actually refers to executions that are consistent with Model B (but not with Model A). Consequently, although Lemma 7 is wrong (as stated w.r.t Model A) it is valid w.r.t Model B (see details below). The rest of the proof of the lower-bound remains unchanged and correct.

1.3 An after-thought

In retrospect, we prefer Model B over Model A. In a sense, Model B postulates than when a bad event (i.e., multiple transmission) occurs the result may be arbitrary. In contrast, Model A postulates than when a bad event (i.e., multiple transmission) occurs the result is always as in case of a different bad event (i.e., no transmission). Although none of these models seems totally realistic, Model B feels more adequate because it assumes less about reality. In general, abstract models may carry reality to an unreasonable extreme, but it seems better to be overly pessimistic than overly optimistic. Consequently, it is better to have unrealistic events justify negative results than have them justify positive ones (e.g., see the use of ECHO in the work of Kowalski and Pelc).

Still, for sure, the flaw pointed by Kowalski and Pelc is a significant contribution to the clarification of the issues involved. Furthermore, the discovery of the distinction between the two models is very interesting and may lead to further improvements in our understanding of these issues.

2 Technical details

The linear lower-bound on the time-complexity of broadcast is proven by considering broadcast on a very simple class of networks and reducing the problem of broadcast on these networks to a combinatorial game. Specifically, for any integer n, we prove a $n/8$ lower-bound for a class of $(n+2)$-vertex networks of radius 2. Each network in the class is identified by a non-empty subset $S \subseteq \{1, ..., n\}$ and consists of the vertex set $\{0, 1, ..., n, n+1\}$ and the edge set

$$\{(0, i) : i = 1, ..., n\} \cup \{(i, n+1) : i \in S\}$$

where vertex 0 (resp., $n+1$) is called the source (resp., sink), and we consider broadcast initiated by the source and ending when the sink receives the message.

The reduction proceeds via a sequence of simplification steps (i.e., considering simplified communication models) and culminates in the reduction of broadcast via abstract protocols (as in Def. 4) to the hitting game (of Def. 5). The reduction is given in Lemma 7, which (as explained above) is flawed as stated. The analysis of the hitting game (provided in Sec. 3.3) is correct as stated.

We thus focus on obtaining a valid version of Lemma 7. All that is needed is to modify the definition of a broadcast protocol and its simplifications such that they all refer to Model B rather than to Model A. That is, in Item 3 of Definition 1, we should postulate that a processor, acting as receiver in a certain time-slot, is guaranteed to receive a message in this time-slot if exactly one of its neighbors transmits, but may receive a message (of one of its neighbors) also if more than one of its neighbors transmits. (Indeed, message delivery is not guaranteed in the latter case, but it may occur nevertheless.) A similar modification applies implicitly to Definition 2 and should be applied explicitly to Definition 4. It is easy to see that the simple reductions among these protocol models remain valid. All that remains is to show that the modified Lemma 7 is valid, where this modified lemma reduces broadcast as formulated in the modified Def. 4 to the game (as stated in Def. 5, with no change here).

2.1 Proof of the modified Lemma 7

We assume that the reader is familiar with the definition of the abstract broadcast model (as stated in Def. 4 and modified above) and with the hitting game (as in Def. 5). The rest of the text refers to these two definitions.

The proof of Lemma 7 describes how to use a t-round broadcast strategy π in order to derive a $2t$-move strategy for the game. Recall that π determines for each processor, S-indicator bit and execution prefix, whether the processor is to transmit in the current round; that is, $\pi(p, \sigma, H)$ determines whether processor p transmits, on execution prefix H and when $\chi_S(i) = \sigma$, where $\chi_S(p)$ is 1 iff $p \in S$. Each round in the protocol (executed in a network identified with the set S) is used to determine two moves in the game, and the referees answers (w.r.t this set S) are used to determine the outcome of this communication round.

In each round, we use the sets T^1 and T^0 (defined by π), as the two next moves in the game, where $T^\sigma = \{p : \pi(p, \sigma, H) = 1\}$ and H is the corresponding execution prefix. Recall that a move M wins in the game if its intersection with S is a singleton (i.e., $|M \cap S| = 1$), and otherwise is answered with $ref_{\overline{S}}(M)$, which is p if $\{p\} = \overline{S} \cap M$ and $\perp$ otherwise.[1] If either one of the two moves wins then we halt and declare the broadcast as completed.[2] Otherwise, we need to determine the answer to be given to the protocol π (i.e., to specify whether and what massage is delivered). It suffices to deliver the message $(0, p)$ iff $ref_{\overline{S}}(T^0) = \{p\}$ (which happens iff $\overline{S} \cap T^0 = \{p\}$).[3] Otherwise, no message is delivered at this round. Note that we only use the second referee answer (i.e., $ref_{\overline{S}}(T^0)$) to determine the delivery event in the protocol.

Observe that in case we have delivered the message $(0, p)$, it holds that the set of transmitters (i.e., $T \stackrel{\text{def}}{=} (T^1 \cap S) \cup (T^0 \cap \overline{S})$) contains p. We stress that this delivery is consistent with Model B, but not necessarily with Model A (because the set T may contain additional transmitters on top of p, which must certainly be in $T \supseteq T^0 \cap \overline{S} = \{p\}$).[4] On the other hand, if no message is delivered at this round and the broadcast is not completed, then it must be that both $T^1 \cap S$ and $T^0 \cap \overline{S}$ are not singletons (because the move T^1 would have won the game in the first case and a message would have been delivered in the second case), and so the decision not to deliver a message is justified.[5] It follows that the way we deliver messages (and determine the protocol's completion) is consistent with Model B.

Note that in case the abstract protocol is proclaimed completed in the ith round, it must be that one of the two corresponding (i.e., $2i$th or $2i-1$st) moves wins in the game (i.e., either $T_i^1 \cap S$ or $T_i^0 \cap S$ is a singleton). Thus, if the protocol always complete broadcast in t rounds (on any network from the class C_n) then there exists a $2t$-move winning strategy for the nth hitting game. The lemma follows. ■

---

[1] We replace the notation $\emptyset$ (used in the paper) by $\perp$.

[2] Actually, only a win by the move T^1 implies a successful broadcast, but it does not hurt to define the protocol successful also in case it is not necessarily so. Alternatively, one may halt and declare the broadcast successful iff $|T^1 \cap S| = 1$.

[3] This description is identical to the more cumbersome form of the paper, which actually has a typo. The original text should have been "let $S_i \leftarrow g(ref_S(T_i^1), ref_{\overline{S}}(T_i^0))$, where $g(A, B) = \{p\}$ iff $A \cup B = \{p\}$", which in turn (since $ref_S(T_i^1)$ cannot be a singleton) implies that $|ref_{\overline{S}}(T_i^0)| = 1$. Thus, in fact, $S_i \leftarrow \{p\}$ iff $ref_{\overline{S}}(T_i^0) = \{p\}$ and $S_i \leftarrow \perp$ otherwise, where S_i is the delivery event.

[4] Indeed, our emulation of Model B does not allow to distinguish the case that $T = T^0 \cap \overline{S} = \{p\}$ (i.e., $T^1 \cap S = \emptyset$) from the case that $T \setminus \{p\} = T^1 \cap S \neq \emptyset$.

[5] If two sets (i.e., $T^1 \cap S$ and $T^0 \cap \overline{S}$) are not singletons then neither is their union (i.e., T).

Relation to the original text: The above description is similar to the original text, except that it was not noticed there that the delivery rule is not consistent with Model A (but is rather consistent with Model B).

A digest: The proof of Lemma 7 effectively decouples transmissions by parties in S from transmission by parties in $\overline{S} \stackrel{\rm def}{=} [n] \setminus S$. Viewed in a different way, each round in the original (abstract) protocol is split into two consecutive rounds such that only processors in $T^1 \cap S$ (resp., $T^0 \cap \overline{S}$) transmit in the first (resp., second) new round. (Recall that the set of transmitters in the original round is $T = (T^1 \cap S) \cup (T^0 \cap \overline{S})$.) We highlight two key points regarding this transformation.

1. One key observation is that in Model B (but not in Model A), we lose nothing (other than a factor of 2 in the round complexity) by employing the above transformation. On one hand, if $|T^1 \cap S| = 1$ then the original protocol Π completes broadcast but so does also the resulting protocol Π'. Otherwise (i.e., $|T^1 \cap S| \neq 1$), determining the delivery events in Π' according to Model A (and in particular consistently with Model B), we can perfectly emulate the delivery events in Π in a way that is consistent with Model B. Specifically, we let the original protocol Π deliver a message iff it was delivered in the resulting protocol Π'. This means that we deliver a message in Π iff $|T^0 \cap \overline{S}| = 1$, regardless of whether $|T^1 \cap S| = 0$ or $|T^1 \cap S| > 1$. (Indeed, this is consistent with Model B but not with Model A.)
2. Referring to the resulting protocol Π', we may just analyze it under Model A (which is consistent with Model B). The key observation is that communication rounds are split to rounds in which only parties is S may transmit and rounds in which only parties in $\overline{S}$ may transmit. This means that Π' may essentially test whether sets determined by it have a single element in either S or $\overline{S}$ (i.e., by instructing a corresponding transmission in a corresponding round). But (unlike Π) protocol Π' cannot test whether sets determined by it have a single element in sets that contain elements from both S and $\overline{S}$ (e.g., the set $S \cup \{p\}$, where p is known to be in $\overline{S}$).[6] Thus, the analysis of Π' reduces easily to the analysis of the hitting game (which allows only queries regarding (singleton-intersection with) either S or $\overline{S}$, but not queries regarding a mix of elements from S and $\overline{S}$).

2.2 A comment regarding the analysis in Section 3.3

The analysis of the hitting game effectively reduces general strategies to oblivious ones (i.e., to strategies in which the sequence of moves is fixed before the actual execution starts). This is done by choosing the adversary set in a way that allows to determine all referee answers from the moves themselves. That is, given an arbitrary game strategy, we consider the moves it takes when all non-singleton moves are answered $\perp$ (and all singletons are answered with the corresponding element, which is declared not to be in S). Thus, the sequence of moves is fixed (i.e., is independent of S, which is rather defined to fit this sequence).

[6]In contrast, *in Model A*, Π may test whether $R \cap (S \cup \{p\})$ is a singleton, by setting $T^0 = \{p\}$ and $T^1 = R' \stackrel{\rm def}{=} R \setminus \{p\}$, where p is known to be in $\overline{S}$, because $R \cap (S \cup \{p\}) = (T^1 \cap S) \cup (T^0 \cap \overline{S})$. This yield ability to test whether $R' \cap S = \emptyset$, which in turn allows to implement a binary search for an element in S. We stress that this is possible for the original protocol Π operating in Model A, but not when operating in Model B. Furthermore, the resulting protocol Π' cannot conduct such queries (and such a search) even in Model A.