Letters to the Editor

From Eric Ristad:

The purpose of this letter is to advise the journal's readership of some misstatements in Alexis Manaster Ramer's review of my book *The Language Complexity Game* (*LCG*) [The MIT Press, 1993] in *Computational Linguistics* 21(1), March 1995, 124–131 ("the review," henceforth).

LCG presents a theory, at the highest level of computational abstraction, of the internal computations performed by language users. The inputs to the *LCG* computations are mental representations, as are the outputs. Thus, the review is mistaken to claim that "the problem studied [in *LCG*] is ... that of determining what meanings are possible for a given set of sentences" (p. 124). No computational problem studied in *LCG* has sentences as its inputs or as its outputs. The input to the Anaphoric Agreement problem considered in Chapter 3 consists of sets of anaphoric elements and available antecedents. The output is a valid relation of immediate antecedence (pp. 33–34). No sentences are involved. The input to the Referential Dependencies problem considered in Chapter 4 is a linguistic representation lacking only relations of obviation and antecedence. The output is the missing relations of obviation and antecedence (pp. 49–50). Again, no sentences are involved. The same holds for the Anaphoric Copying and Anaphoric Sharing problems considered in Chapter 5: no sentences. This may help explain why the reviewer is unable to find even a "single complete example" of an input sentence for the *LCG* complexity analysis of the Referential Dependencies problem (p. 127) or why "the translation from Boolean formulas to English sentences is not fully specified" (p. 128).

The review's presentation of the *LCG* complexity analysis is not wholly accurate either. For example, on page 125, the review states that examples such as (1) are "completely irrelevant to the NP-hardness argument" and that "to get NP-hardness one would need to consider examples in which pronouns with different features are available." Moreover, the review claims that "an infinite set of inflectional features" are required "to get NP-hardness."

(1) Before Bill, Tom, and Jack were friends, he wanted him to introduce him to him.

As explained in section 4.1.1 of *LCG*, example (1) exemplifies a local c-command configuration. Configurations of local c-command are essential in the reduction from 3SAT to the Referential Dependencies problem (lemma 3, p. 64). They are used in both the variable gadget (Figure 4.3) and the clause gadget (Figure 4.4). And contrary to the review's second claim, absolutely no inflectional features are used in this reduction. In the foreword to *LCG*, Robert Berwick correctly notes that examples such as (1) play an important role in establishing the NP-hardness of the Anaphora Problem (p. xiii). So the review's claim that "Berwick misinterprets what the result is all about" (p. 125) is also unfounded.

Nor does the review accurately convey the statement of the relevant lemma. The review claims "that [this] proof does not state what the reduction from 3-SAT is to, which is a serious matter for a reduction proof" (p. 127). Turning to the middle of page 64 in *LCG*, we see "LEMMA 3 3SAT $\leq_{\mathcal{P}}$ Referential Dependencies." As is standard in computer science—and explained on page 135 in the appendix on mathematical background—the symbol $\leq_{\mathcal{P}}$ denotes the relation of polynomial time reducibility between computational problems. So Lemma 3 states that the 3SAT problem reduces to the Referential Dependencies problem in polynomial time. A quick check of the index reveals that 3SAT is defined on page 137 and Referential Dependencies is defined on page 49.

The technical content of *LCG* is to define language computations involving anaphora, in the form of problem statements, and to analyze their computational complexity. The review misstates the computational problems involved and does not accurately report the lemmas or their proofs. So I am concerned that the review may not have accurately represented *LCG* to the journal's readership.

After raising these (and other) technical issues, the review questions the distinction between a computational theory and a generative theory. In particular, the review states that "it is by no means clear ... that this [computational approach] is ... any different from, for example, Chomsky's [generative] approach" (p. 130). As summarized on page 113 and pages 115–6 of *LCG*, a generative theory of language is an explicit procedure for enumerating the set of linguistic representations. The computations specified by a generative theory enumerate linguistic representations; these computations are called derivations. Derivations are not directly attributed to language users, contra the Derivational Theory of Complexity. In contrast, a computational theory of language is a theory of the mental computations performed in the comprehension, production, and acquisition of human languages. Thus, the computations specified by a computational theory are indeed attributed to language users. For this reason, a computational theory such as that of *LCG* offers a more internalized view of language than is offered by a generative theory.

The review goes on to claim that *LCG* "does not attempt ... to show that there are factual reasons for preferring this [computational] view of language" to the generative approach (p. 130). In Section 6.3, *LCG* argues that, unlike the generative approach, the computational approach more naturally results in language models that are accurate in computationally significant respects. Thus, the complexity analysis of anaphoric agreement in Chapter 3 leads to an empirical critique of the standard "nondistinctness" theory of agreement as well as a simpler theory of agreement. The complexity analysis of referential dependencies in Chapter 4 leads to a better understanding of obviation in elliptical contexts. The complexity analysis of ellipsis in Chapter 5 raises some empirical problems for the popular "copy" theory of ellipsis, and proposes a simpler alternative based on the sharing of thematic functions. And so the review's claim that *LCG* "gives no basis for preferring this [computational] view of language to what [*LCG*] takes to be Chomsky's version of the I-language perspective" (p. 130) is misleading.

In addition to these technical and conceptual issues, Manaster Ramer's review also raises some questions of style and proportion.

The review opens with the statement that "Ristad consistently and mistakenly uses the words *prove* and *proof* when speaking not only about the results concerning well-defined mathematical objects ... but also about the claims that (some) NLs are properly modeled by these mathematical objects ... " (p. 124). By my count, the words *prove* and *proof* appear a total of 66 times in *LCG*. One instance is nontechnical: "prove useful" (p. 11). The remaining 65 instances refer to mathematical proofs, typically to proofs of the computational complexity of precisely stated computational problems based on particular formal models of language. I am unable to find a single case in *LCG* where either *proof* or *prove* is used in an empirical argument about human languages. So the review is half right here: my usage is consistent, but not in the manner that the review claims.

Shortly thereafter, the review states that "the whole argument about the NP-hardness of the anaphora problem in agreement situations ... was the centerpiece of [Ristad's] dissertation" (p. 125). My doctoral dissertation includes complexity analyses of segmental phonology, autosegmental phonology, morpho-syntax, ellipsis, and anaphoric relations. The entire discussion of anaphoric agreement occupies only one page out of more than 140 pages in the dissertation, less than one section out of the more than 40 sections. So it does not seem entirely accurate to call the discussion of anaphoric agreement the "centerpiece" of the dissertation.

Eric Ristad
Department of Computer Science
Princeton University
Princeton, NJ 08544
ristad@cs.princeton.edu

From Alexis Manaster Ramer:

Dr. Ristad claims that my review misrepresented both him and Robert C. Berwick, the author of the effusive introduction to his book (Ristad 1993; henceforth *LCG*). In my review, I praised *LCG*'s insights into certain linguistic problems with agreement, anaphora, and ellipsis, but I criticized some of its claims about English, its treatment of mentalism, and its argumentation about the complexity-theoretic properties of natural languages, specifically, the claims that English or even NLs in general are NP-complete (that is, NP-hard and included in the class $\mathcal{NP}$). Dr. Ristad does not take issue with the most important of my critical comments, dealing with the fundamentally empirical nature of his claims and the extreme fragility of his claims about English. A laborious examination of the points which he does appeal will show that I was right on those points too.

1. The first point of contention concerns the term 'sentence'. Dr. Ristad's complaint is that I should never have mentioned "sentences," since, for example, "[t]he input to the Referential Dependencies problem considered in Chapter 4 is a linguistic representation lacking only relations of obviation and antecedence. The output is the missing relations of obviation and antecedence ... no sentences are involved," and likewise for the other problems he studied. However, the examples in *LCG* of English structures illustrating these problems are usually sentences, such as (1), and Ristad (1993) himself refers to them as "sentence"s (e.g., p. 54, lines 13, 15, 16, 18). The "proof" of the NP-hardness of Referential Dependencies (p. 64) involves a reduction that translates a Boolean formula into a "linguistic representation ... shown schematically in figure 4.2"; the figure shows us a *highly* schematic tree structure of an English sentence, a *part* of which would be (1).

(1) True met him$_2$, who$_A$ he$_1$ expected him$_3$ to want him$_4$ framed, who$_B$ he$_5$ believed he$_6$ did [e] with t$_B$, after exposing himself$_5$ to [e] for t$_A$, before telling himself$_1$ to [e].

The only genuine issue I can see here, therefore, is whether the term 'sentence' may be used just for a bare string or whether it can also apply to a string provided with some kind of structure. As it is, some of the example sentences in *LCG* are bare strings (e.g., (5), p. 31), some contain bits of structure (such as the *[e]*'s in (1) above), and some have (unlabeled) brackets marking some constituent boundaries (e.g., (22), p. 54). I see no reason to call these three different kinds of objects 'sentences' and yet to refuse this term to a more complete (labeled) bracketing, which is equivalent to a tree structure, like those in Figures 4.2 and 4.4 in *LCG*. To me the term 'sentence' is systematically ambiguous on this point.

The real issue, of course, is not whether Dr. Ristad's arguments operate in terms of bare strings, partial structures like (1), or complete ones, but rather that he did not specify the set of linguistic objects (whatever we call them) that is involved in his argument. Of course, if he had, I would not have complained about his failure to define the set of "sentences" involved. For, given a set of trees, one can deduce the relevant set of "sentences" (if, that is, we are going to insist that only a structure like (1) or one with even less structure is to be called a "sentence"), by simply pruning as much of the structural information as is desired.

In any event, the polemic about the term 'sentence' (which *LCG* also uses, as noted) and the rhetoric about "internal computations performed by language users, at the highest level of computational abstraction" on "mental representations" (and not on "sentences," it is claimed) does not in the least "help explain why the reviewer is unable to find even a 'single complete example' of an input sentence for the *LCG* complexity analysis of the Referential Dependencies problem (p. 127) or why 'the translation from Boolean formulas to English sentences is not fully specified' (p. 128)." The reason I could not find such an example or specification is that *LCG* never gave a complete example of a relevant "linguistic representation" (since (1), even with its complete tree structure, is just a part of what is required), not to mention a definition of the relevant set of such "representations," and as a result did not, and could not, fully specify the reduction.

2. Dr. Ristad's second complaint has to do with my discussion of Berwick's use of (2) as an example of the complexity that leads to the *LCG* NP-hardness result:

(2) Before Bill, Tom, and Jack were friends, he wanted him to introduce him to him.

As noted in the review, (2) is not an example from *LCG* but from Ristad's dissertation (1990, p. 69). What appears in *LCG* is a very similar but differently worded example (p. 54, example (22)), and the reasons for these examples being irrelevant are given by Ristad in both places (1990, p. 69; *LCG*, p. 55). I even quoted the statement from Ristad (1990: 69), to wit, "The configurations used to construct the sentence [(2)] can only give rise to very simple obviation graphs on their own, and therefore the proof of [NP-hardness] must build obviation graphs using the agreement condition." As Dr. Ristad notes, I then went on to observe that the "agreement condition" in question—involving (as do all agreement phenomena) a variety of contrasting inflectional features—does not figure in this example, where all the pronouns are masculine singular. This, and what I said about the number of features involved in the argument, was correct. Dr. Ristad's equally correct observation that (2) is an example of local c-command, which together with other phenomena is supposed to lead to NP-hardness, doesn't change anything. The point was, and remains, that Berwick would have us believe that Ristad's NP-hardness result explains why "the links between the different *hims*" in (2) are difficult to "figure out," and that this is untrue.

As noted in *LCG* (p. 55), "[t]he local c-command configurations used in (22) only give rise to simple obviation graphs that are easily colored." Since example (22) in *LCG* is isomorphic to (2), and since "easily colored" means "in $\mathcal{P}$," Berwick's confidence is misplaced. If such examples are hard for people to interpret, this cannot have any connection with Ristad's result. Not even close. The sentences that illustrate the NP-hardness are vastly more complicated than this—indeed so complicated that Ristad (1993), as noted, never constructed a complete example. Sentence (1), which is already so complicated as to be quite unintelligible, is just a part of a considerably longer sentence that would have to be constructed as the *shortest* example of the kind of complexity we are talking about.

The broader context here is that *LCG* (p. 31–48) refutes Ristad's 1990 attempt to show the NP-hardness of English on the basis of agreement phenomena (which is now presented as a straw man) and chooses to rest it on the interaction of c-command, control, strong crossover, and invisible obviation. Berwick's claim cannot be salvaged by changing the reference from the 1990 argument (where these examples are irrelevant because the pronouns are all the same) to the 1993 one (where they are again irrelevant because only local c-command is involved). Moreover, the confusion suggested by Berwick's use of an example from the 1990 work is very real, and Berwick muddies the waters still further by talking about "agreement and syntactic category ambiguity." This refers to yet another putative NP-hardness argument, entirely different from either that in Ristad (1990) or that in *LCG*, and instead to be found in Chapter 3 ("Agreement and Ambiguity") of Barton, Berwick, and Ristad (1987). A potential reader (or purchaser) of *LCG* has the right to be warned that there are three different NP-hardness arguments floating around, and that Berwick's high-power endorsement of *LCG*'s argument seems to confuse it with two earlier ones, the first of which (1987) was never asserted to hold for any actual NL (Barton, Berwick, and Ristad 1987, p. 96) and the second of which (1990) has now been given up.

3. Next, Dr. Ristad returns to my criticism of his incompletely specified reduction arguments:

> Turning to the middle of page 64 in *LCG*, we see "LEMMA 3 3SAT $\leq_{\mathcal{P}}$ Referential Dependencies." As is standard in computer science—and explained on page 135 in the appendix on mathematical background—the symbol $\leq_{\mathcal{P}}$ denotes the relation of polynomial time reducibility between computational problems. So Lemma 3 states that the 3SAT problem reduces to the Referential Dependencies problem in polynomial time. A quick check of the index reveals that 3SAT is defined on page 137 and Referential Dependencies is defined on page 49.

Much of this is correct, but there *is* no precise definition of Referential Dependencies on page 49 or anywhere else in *LCG*. "Referential Dependencies" are certain dependencies that exist (or are claimed to exist) in English, but *LCG* never specifies fully just what the claims about English are. The "definition" on pages 49–50 reads as follows:

> The input consists of a linguistic representation R lacking relations of referential dependency but containing the 3-tuple $\langle A^A, A^B, A^C \rangle$ of disjoint sets of arguments, where A^A consists of reflexives and reciprocals, A^B consists of pronouns, and A^C consists of referring-expressions....

> The output is a correct referential dependency graph $G = \langle A, O, L \rangle$ that is compatible with R.

We know from earlier on page 49 that O is the **obviate** relation and L is the **link** relation, but we lack a precise definition of the conditions under which these relations are supposed to hold in a well-defined set of sentences (or tree structures of sentences) in some variety of English. It is only by extrapolating from examples and by reading the relevant (informal) linguistic literature that one can gauge roughly what the extensions of these relations are, and hence which referential dependency graphs are claimed to be "correct" in English and which not. Perhaps all that *LCG* needs here is an empathetic exegete, but it certainly needs something.

Dr. Ristad concludes his discussion of this point by saying:

> The technical content of *LCG* is to define language computations involving anaphora, in the form of problem statements, and to analyze their computational complexity. The review misstates the computational problems involved and does not accurately report the lemmas or their proofs. So I am concerned that the review may not have accurately represented *LCG* to the journal's readership.

Since he gives no examples of such supposed inaccuracies, I do not know how to respond to this beyond a flat, curt denial.

4. Next, Dr. Ristad raises the issue of how I handled his mentalist view of language. His first complaint is:

> ... the review questions the distinction between a computational theory and a generative theory. In particular, the review states that "it is by no means clear ... that this [computational approach] is ... any different from, for example, Chomsky's [generative] approach" (p. 130).

This, however, is not what what I wrote, which was:

> Now, it is by no means clear to me or to Chomsky (personal communication) that this is original, that is, any different from, for example, Chomsky's approach.

The bracketed words "computational approach" and "generative," which Dr. Ristad sneaks into my quotation, misrepresent utterly what I said. There are three, and not two, approaches at issue. What Dr. Ristad calls "generative" is what Chomsky calls the "E-language" approach (found in some of Chomsky's writings between roughly 1956 and 1965). What he calls "computational" is basically the same as Chomsky's (e.g., 1986) own I-language approach. What I questioned was not the difference between the E-language ("generative") and the I-language ("computational") approaches, but rather between Ristad's formulation of the I-language approach and Chomsky's, as I made clear by saying, among other things:

> Moreover, Ristad gives no basis for preferring this view of language to what he takes to be Chomsky's version of the I-language perspective.

Thus, Dr. Ristad's complaint is groundless and misleading: Chomsky has not followed the "generative approach" for decades, and I, having worked extensively on characterizing Chomsky's notion of I-language computationally and mathematically (Manaster Ramer 1993, to appear), would not very likely have said that he has.

Dr. Ristad's second complaint is that he did advance the kinds of reasons I claimed he did not. Yet, even if there are differences between Chomsky's and Ristad's versions of the I-language perspective, it is clear from Dr. Ristad's letter that his arguments, rehashed here, refer not to these differences but to those between I-language and E-language. Moreover, if Ristad's view of language is roughly the same as Chomsky's, then there cannot logically be very many arguments for one over the other.

The only difference I actually found is that Chomsky continues to uphold the competence/performance distinction, whereas Ristad claims not to, although he gives no argument at all on this score, preferring to merely anathemize this distinction. Moreover, the *LCG* results actually presuppose the competence/performance distinction, since, if there were no such distinction, then sentences like (1) could not be both grammatical and unacceptable. Their status in the "language itself, exactly as it is and how it operates" (*LCG*'s description of what there is in place of competence and performance, p. 117) would then have to be either unequivocally good (which surely no speaker would assent to) or unequivocally bad (in which case they are not English, and the NP-hardness argument collapses). Finally, *LCG*, (p. 13, 117–120) appears to concede that its claims carry over to the E-language approach (as we would expect, since complexity work is usually done in those terms anyway).

5. Next, Dr. Ristad objects to my having said that "as in far too many other books and articles in our field ... Ristad consistently and mistakenly uses the words *prove* and *proof* when speaking not only about the results concerning well-defined mathematical objects ... but also about the claims that (some) NLs are properly modeled by these mathematical objects and hence share their properties." Dr. Ristad does not appear to question this distinction, which I consider so important because the status of a theorem about a formal language has an entirely different status from an argument about an NL. Results about formal languages, if done correctly, are presumably irrefutable. Claims about NLs can, and often are, refuted when new discoveries occur (e.g., when we find out that some sentence we thought was ungrammatical is fine, or that it can have a meaning we thought impossible). To be sure, Dr. Ristad says that he never used the words *proof* and *prove* in the way that I am objecting to. However, in *LCG prove* is used this way (e.g., on page 48, line 4, and on page 71, line 1, and *proof* on page 66, line 12), referring to arguments (as I would call them) that the "Anaphora Problem" and/or the "Anaphoric Preference Problem" are NP-hard (or NP-complete). Now, these two problems are the problems of finding, given a sentence with a number of anaphors in it, those interpretations (specifically, relations of referential dependency) that it has (or preferentially has), as a matter of linguistic fact, in English. The NP-hardness or otherwise of these problems thus depends crucially on empirical claims about which NPs can and which cannot antecede (or preferentially antecede) which anaphors in certain English sentences (like (1) but more complex), although, as noted above, Ristad never presents all this with sufficient rigor. But, whatever the claims are, they are up to a linguist or an informant, not a mathematician, to evaluate. Which is precisely what makes the use of the terms *proof* and *prove* inappropriate.

Of course, there must be a mathematical component to the arguments, namely, a proof that the formal model (had this been fully specified) of the *LCG* theory of one or the other of these problems is NP-hard (or NP-complete). But this does not alter the fact that we may not say that anyone has *proved,* or given a *proof,* that any NL, or a linguistically significant fragment or problem, is NP-hard. Such a result can never be proved because it is always subject to empirical challenge. The complex story of the attempts to show the NP-hardness of NLs shows how true and relevant this is. As noted above, the 1987 argument appears to have been still-born, and the 1990 argument was refuted in *LCG,* which now offers us a third argument. How can we accept the latest result as a "proof," knowing that the earlier ones failed, not because of any flaw in the mathematics, but on empirical grounds? What happened each time is that it turned out that some empirical claim or assumption about the relevant linguistic facts did not hold. This can happen again, and indeed in my review I provided a detailed discussion of the reasons why *LCG*'s claims about English are to be rejected, vitiating the latest argument as well.

I might add that it is particularly objectionable when we are told of a "proof" of NP-completeness, as opposed to just NP-hardness (since the existence of a nondeterministic polynomial time algorithm solving one problem in English can in no imaginable way tell us that all linguistically significant problems in English have such algorithms) or of a "proof" that NL *tout court,* rather than some specific NL like English, is NP-hard or NP-complete (since, even if English were, it would not follow that other languages are). There are, one finds, more pernicious and less pernicious misuses of the words *proof* and *prove,* although they are all pernicious.

6. Finally, Dr. Ristad contends that I misrepresented his 1990 dissertation by describing the (now-refuted) anaphora/agreement argument as the "centerpiece" of that work. He points out that he also dealt with various other problems in his dissertation, namely, "complexity analyses of segmental phonology, autosegmental phonology, morpho-syntax, [and] ellipsis," in addition to anaphora and agreement, and that "[t]he entire discussion of anaphoric agreement occupies only one page out of more than 140 pages in my dissertation."

While much of this is true, some of the other complexity analyses appear in an appendix, and the section of the dissertation dealing with the argument at issue (pp. 68–70) is a bit longer than a page. More importantly, dissertations usually save the best for last, and the problem of anaphoric agreement is indeed the last chapter before the references and the appendices start (Chapter 4, entitled "Complexity of Pronoun Antecedence"). Even more importantly, the very first sentence of the dissertation reads: "The central thesis of this dissertation is that human language is NP-complete" (Ristad 1990, p. *i*), which can only refer to this chapter. There is a subtle but legitimate issue Dr. Ristad could have raised here, but does not. He referred to the whole NP-completeness result as the "central thesis," whereas I described just the lower-bound (NP-hardness) part of it as the "centerpiece." This is not the issue he raises, but it may be worth noting that the discussion of the upper-bound part of the argument still only occupies a small fraction of the total page count (pp. 70–87). Moreover, given the fact, alluded to above, that an upper-bound result of this nature is, by the nature of things, much more fragile than a lower-bound one, I think I was right to focus on the latter result. Had it stood the test of time, instead of getting refuted in *LCG,* this would have been a revolutionary achievement—and one as robust as is possible given the empirical nature of the underlying issues.

7. My evaluation of *LCG* stands. Berwick compares its claims to Chomsky's (1956) classic center-embedding argument against finite-state models for NLs. What I said in my review, and what I still believe, can perhaps best be summarized by saying that Ristad's work is like what Chomsky's would have been (a) had the crucial grammaticality judgments Chomsky invoked been incorrect, (b) had Chomsky failed to make it clear just what fragment of English he was talking about, (c) had Chomsky denied the performance/competence distinction, and thus been unable to treat examples of repeated center-embedding as sentences of English, and worst of all, (d) had Chomsky then claimed to have "proved" that NLs are context-free on the grounds that he could construct a context-free grammar handling those center-embeddings (a hypothetical upper-bound result quite analogous to Ristad's).

Of course, it is entirely possible that English really is NP-hard and perhaps NP-complete as well, but it is important to realize that these are not the kinds of things we can ever know with mathematical certitude. And, that as of now, we still lack any solid grounds for accepting either claim even as a working hypothesis or an empirical theory.

Alexis Manaster Ramer
Department of Computer Science
Wayne State University
Detroit, MI 48202
amr@cs.wayne.edu

References

Barton, G. Edward, Robert C. Berwick, and Eric Sven Ristad (1987). *Computational Complexity and Natural Language.* Cambridge, MA: The MIT Press.

Chomsky, Noam (1956). "Three models for the description of language." *IRE Transactions on Information Theory* IT-2, 113–124. Reprinted (1965) in: R. D. Luce, R. R. Bush, and E. Galanter (editors), *Readings in Mathematical Psychology*, v. 2, 105–124, New York: John Wiley.

Chomsky, Noam (1986). *Knowledge of Language: Its Nature, Origins, and Use.* New York: Praeger.

Manaster Ramer, Alexis (1993). "Towards transductive linguistics." In: Karen Jensen, George E. Heidorn, and Stephen D. Richardson (editors), *Natural Language Processing: The PLNLP Approach*, 13–27. Boston: Kluwer Academic Publishers.

Manaster Ramer, Alexis (To appear). *The Uses and Abuses of Mathematics in Linguistics.* Tarragona: Universitat Rovira i Virgili.

Ristad, Eric Sven (1990). *Computational Structure of Human Language.* Ph.D. dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.

Ristad, Eric Sven (1993). *The Language Complexity Game.* Cambridge, Massachusetts, and London, England: The MIT Press.