How-Possibly Explanations in Quantum Computer Science

Michael E. Cuffaro

Ludwig-Maximilians-Universität München,

Munich Center for Mathematical Philosophy

Abstract

A primary goal of quantum computer science is to find an explanation for the fact that quantum computers are more powerful than classical computers. In this paper I argue that to answer this question is to compare algorithmic processes of various kinds, and in so doing to describe the possibility spaces associated with these processes. By doing this we explain how it is possible for one process to outperform its rival. Further, in this and similar examples little is gained in subsequently asking a how-actually question. Once one has explained how-possibly there is little left to do.

Word count: 4,927.

1 Introduction

There is a distinction that is sometimes made in the scholarship on scientific explanation between explaining *why* and explaining *how-possibly*. In the ontic context, where the explanations one gives aim at describing salient features of actual physical systems, the former is sometimes also called *how-actually* explanation.[1] That how-actually explanation actually explains is uncontroversial; however it is less clear just what if any explanatory merit there is in explaining how some event possibly came about. Partly for this reason, the literature on how-possibly explanation is comparatively sparse, and the few who have commented on the topic are of varying opinion with regard to its virtues. While some view how-possibly explanation as genuinely explanatory, others have argued that how-possibly 'explanation' is better thought of as a mere heuristic device and not as constituting genuine explanation at all. Still others have thought of how-possibly explanation as a kind of incomplete how-actually explanation—a stepping stone on the way to the how-actually explanation that one ultimately seeks.

Below I will consider a question which I will argue sheds light on this issue. It is drawn from the science of *quantum computation*. Quantum computation is a fruitful merger of the fields of physics and computer science, and one of the goals of this science is to determine the source of the power of quantum computers; i.e., to search for the explanation of the fact that quantum computers can in general (and sometimes dramatically) outperform classical computers. What I will argue is the following: to answer this question is to compare algorithmic processes of various kinds, and in so doing to describe the possibility spaces

---

[1]For more on the distinction between ontic, epistemic, and modal forms of explanation, see Salmon (1984).

associated with these processes. By doing this we explain how it is possible for one process to outperform its rival. Further, and importantly, in examples like these little if anything is gained in subsequently asking a how-actually question. Once one has answered the how-possibly question there is little left to do.

I will close by suggesting that the search for the explanation of the power of quantum computation is just one example of a species of how-possibly question that is likely to be found in many other sciences as well.

2 How-possibly Explanation

The first mention of how-possibly explanation is likely that of Dray (1957). Dray's primary goal in that book is to assess the adequacy of the 'covering law model' of explanation for characterising historical explanation. The model is so-called because it involves the subsumption of a particular set of initial conditions under a law or a set of laws (Hempel and Oppenheim, 1948). Dray's verdict is that the covering law model fails to capture many interesting senses of historical explanation. One of the ways in which it is inadequate, according to Dray, is that the covering law model insists that any explanation of a given fact must show why, necessarily, that fact had to occur, since the statement of the fact to be explained must be deductively entailed by the statements of the relevant laws and initial conditions.[2] Dray insists, however, that not all historical explanations are why-necessarily explanations.

An announcer broadcasting a baseball game from Victoria B.C., said: "It's a long

---

[2]This is the case for Hempel and Oppenheim's *Deductive-Nomological* (D-N) model of explanation. Statistical explanations require only inductive support (Hempel, 1965). For our purposes we need only detain ourselves with the former.

fly ball to centre field, and it's going to hit high up on the fence. The centre

fielder's back, he's under it, he's caught it, and the batter is out." Listeners who

knew the fence was twenty feet high couldn't figure out how the fielder caught the

ball. Spectators could have given the unlikely explanation. At the rear of centre

field was a high platform for the scorekeeper. The centre fielder ran up the ladder

and caught the ball twenty feet above the ground (from *Maclean's Magazine*, as

cited in Dray 1957, 158).

What is explained here, for Dray, is not exactly why the ball landed in the centre fielder's

glove. Rather, what is dispelled is the initial puzzlement on the part of the listener upon

hearing about the catch. This puzzlement is removed once she is told of the scorekeeper's

ladder, for the ladder explains how the catch was possible: it opens up a range of possibilities

that would not have been present without it. Of course one can still ask: "why, exactly, did the

ball land in his glove?" However to do so, for Dray, is to ask a logically *different* question.

The how-possibly question is answered once we have been told about the ladder.

Hempel and Oppenheim's covering law model, with its exclusive emphasis on

why-necessarily questions, was, for many years, the 'received view' on scientific explanation.

But although similar conceptions continue to be defended, there is no longer a near consensus,

and indeed many have taken a pluralistic attitude (or at any rate remained agnostic), on the

question of whether an all-encompassing model of scientific explanation exists.[3] Despite this,

how-possibly explanation has received comparatively little attention (as compared with, say,

causal explanation). But it has received some. There are those, for instance, who reject

outright the very idea of how-possibly explanation—Reiner (1993, 68) goes so far as to call

---

[3]See, for instance, Woodward (2003), who productively focuses his energies on

explicating one particular type of explanation.

its promotion and proliferation a "sociological risk"—however the majority of the debate surrounding how-possibly explanation centres around the sense and extent to which it (in Dray's or perhaps some other formulation[4]) is explanatory.

Thus even Hempel grants to Dray that there is some sense in which a how-possibly account explains. Nevertheless, he argues, upon hearing it the questioner will invariably desire to be told why the event necessarily occurred if he is to be fully satisfied (Hempel, 1965, 429). For Hempel, the role of how-possibly explanation is primarily pragmatic: it motivates the questioner to ask a further why-necessarily question. For Resnik (1991, 143), on the other hand, how-possibly explanation and how-actually explanation are of the same kind, and differ only in the degree to which they are empirically supported. That is, a how-possibly explanation is a how-actually explanation that enjoys no more than speculative supporting evidence, yet nevertheless displays other explanatory virtues like fruitfulness. Salmon's (1989, 137) conception is similar.

Forber (2010), in contrast, views how-possibly and how-actually explanation as different in kind. What Resnik refers to as how-possibly explanation is, for Forber, no more than an

---

[4]Dray's account is sometimes thought to rely overmuch on psychology: for Dray, recall, one explains how-possibly to dispel a questioner's puzzlement at having witnessed or been told of some event. One can do away with this psychological element, however, by explicating surprise in epistemic terms; i.e., as a prima facie tension between the fact to be explained and the questioner's body of knowledge absent some additional piece of information. This latter is what is provided by a how-possibly explanation. Other approaches to modifying Dray's view so that it is more conformant to ontic conceptions of explanation have also been considered (see Persson, 2012).

incomplete how-actually explanation.[5] Explaining how-possibly, for Forber, is not this but a kind of formal inquiry: given a set of relevant background assumptions, one deduces (e.g., via computer simulation) a particular set of outcomes reachable from them. For instance, let the assumptions consist of known biological laws relevant to a particular population, plus a specification of a set of variable parameters, and let the different outcomes represent various genotypes associated with that population. Then, when one runs such a simulation, the different paths by which it arrives at a particular outcome carve up the possibility space for that outcome—they represent a set of how-possibly explanations corresponding to it.[6] Explaining how-actually, in contrast, is a form of empirical inquiry: its aim is to determine which of these possible explanations is the actual one.

   A further mode of how-possibly explanation, described by Persson, "aims to establish the existence of a mechanism by which X could be, and was, generated without filling in all the details" (Persson, 2012, 275). Key to Persson's conception is the empirical determination of some actually existing mechanism responsible for X. It is thus distinct from Resnik's conception of how-possibly explanation as inadequately supported how-actually explanation. It is also distinct from Forber's conception of how-possibly explanation, which recall is not a form of empirical inquiry at all. Nevertheless it is how-possibly and not how-actually explanation because, although one describes an actual mechanism responsible for X,

---

[5]One could be forgiven for thinking this merely a dispute over labels. In Forber's defence, his view is more in the spirit of Dray's original account, who recall, viewed how-possibly and how-actually questions as logically different.

[6]Forber distinguishes between *global* and *local* how-possibly explanations. The former utilise highly idealised background assumptions. The latter are directed at real populations and utilise richer, empirically grounded, assumptions.

information is missing from our description of the mechanism which would allow us to determine the precise (typically causal) pathway by which X was brought about. Thus one has not given an account of how the event *actually* occurred.

In the modes of explaining how-possibly identified so far, the questioner is required, or at any rate it is perfectly sensible for her, to continue on to ask the how-actually question. Thus for Hempel she (at least in interesting cases; see Hempel 1965, 429) will not be fully satisfied until she answers the how-actually question. On Resnik's conception, how-possibly questions are just how-actually explanations that have not been adequately confirmed, and it goes without saying that we should try and confirm them. For Persson it is not confirming but "filling in" of the mechanism which remains to be done. On Forber's conception, explaining how-possibly is in no way inferior to explaining how-actually, yet both play an essential role in our inquiries into phenomena. For Dray, pace Hempel, sometimes one is thoroughly satisfied with a how-possibly explanation. Nevertheless it is perfectly sensible to go on and ask exactly how the centre fielder caught the ball once he was at the top of the ladder.

The kind of how-possibly explanation I describe in the next section bears certain resemblances to both Persson's and Forber's conceptions. As in Persson's conception it involves the description of some mechanism actually responsible for producing an outcome. Unlike in Persson's conception, in this case there are no relevant details of the mechanism left to fill in. On the other hand, the way the mechanism explains, as in Forber's conception, is that it carves out a particular possibility space for an outcome. But unlike all of the conceptions reviewed above, once one has answered the how-possibly question in this case, it is doubtful that a how-actually explanation can give us anything more of substance.[7]

_____

[7]Which of these conceptions is right? With Persson I would maintain these are all legitimate senses of explaining how-possibly. Which one is 'correct' will be relative to the

3 Explaining Quantum Speedup

A basic distinction, in Computational Complexity Theory, is between those computational problems amenable to an efficient solution in terms of time and/or space resources, and those that are not. Easy (or 'tractable', 'feasible', 'efficiently solvable', etc.) problems have solutions which involve resources bounded by a polynomial in the input size, n (n is typically the number of bits used to represent the input). Hard problems are those which are not easy; i.e., they require resources that are 'exponential' in n, i.e., that grow faster than any polynomial in n (Nielsen and Chuang, 2000, p. 139).[8] For example, a problem which requires $\approx n^c$ time steps to solve in the worst case (for some constant c) is polynomial in n and thus tractable according to this definition. A problem that requires $\approx c^n$ steps, on the other hand, is exponential in n therefore intractable according to this definition.

'Quantum speedup' refers to the fact that some computational problems can be solved exponentially faster with a quantum computer than with any known classical computer.[9] For example, the fastest known classical algorithm for factoring the product of two unknown primes is exponential in n. Shor's quantum algorithm, astoundingly, solves the problem in polynomial time (Nielsen and Chuang, 2000, 216). But while the fact of quantum speedup is almost beyond doubt,[10] its source is still a matter of debate within the scientific community.

specific question asked.

[8]The term 'exponential' is being used rather loosely here. Functions such as $n^{\log n}$ are called 'exponential' since they grow faster than any polynomial function, but they do not grow as fast as a true exponential such as 2^n.

[9]Research into quantum computing is still largely in the theoretical stage. However there is good reason to be optimistic that practical implementations will be realised eventually (see Aaronson, 2013, Ch. 15).

[10]There is currently no proof that the class of problems efficiently solvable by quantum

```
void SelectionSort(int intsToSort[], int lengthOfList) {
  // Declare list indices:
  int i, j, indexOfLowestNum;
  // For each position in the list,
  for (i = 0; i < lengthOfList − 1; i++) {
    // provisionally assert that it points to the lowest number,
    indexOfLowestNum = i;
    // and then for each of the other list positions,
    for (j = lengthOfList − 1; j > i; j−−) {
      // if the number pointed to by it is less than the number
      // pointed to by indexOfLowestNum,
      if (intsToSort[j] < intsToSort[indexOfLowestNum]) {
        // then make this the new provisional minimum index.
        indexOfLowestNum = j;
      }
    }
    // At the end of the ith iteration, put the number that is in the
    // indexOfLowestNum position into the ith position (and vice versa).
    Swap(&intsToSort[i], &intsToSort[indexOfLowestNum]);
  }
}
```

Figure 1: A set of instructions (in C) implementing the 'SelectionSort' solution to the problem of sorting a list of given integers.

According to Fortnow (2003), for instance, the explanation of quantum speedup lies in the ability of quantum systems to exhibit 'interference'. For Ekert and Jozsa (1998), on the other hand, it is their ability to exhibit 'entanglement'.

We will consider these explanations in a little more detail later. For now let us ask: what kind of question is one asking when one asks to have quantum speedup explained? It is clearly an ontic question, for it aims to identify particular characteristics of physical systems. It is also a how question, if anything is, for it asks, specifically, for the distinctive mechanism by which quantum computers operate. It remains to consider whether it is a how-possibly or a how-actually question.

Consider figure 1. Depicted there is an instance of the 'SelectionSort' algorithm for sorting

computer is larger than the class efficiently solvable by classical computer, however other results make the truth of this statement very likely (see Aaronson, 2013, Ch. 10).

a given list of integers. If given, say, the numbers (25, 12, 13, 19, 8), then after a certain time a computer running this code will produce: (8, 12, 13, 19, 25). Now if we examine the algorithm, we notice that in the worst case (indeed, in any case) there will be $n - 1$ comparisons in the first iteration, $n - 2$ comparisons in the second, $n - 3$ in the third, and so on (where n is the number of list items). This gives a total of $n(n - 1)/2$ comparisons; thus our total worst-case running time is proportional to n^2. SelectionSort is not the only algorithm for sorting integers. Both faster and slower algorithms exist. For instance, the 'MergeSort' algorithm has a worst-case running time $\propto n \log(n)$ (Mehlhorn and Sanders, 2008).

Suppose we are comparing the running times of various algorithms. I feed some random permutation of a list of n integers to my algorithm (which implements SelectionSort) and you feed one to yours (which implements MergeSort). After a time we both obtain the list in sorted order. Yours finishes after $k \leq n \log(n)$ steps. Mine finishes after $n \log(n) < l \leq n^2$ steps. What is the explanation for this difference in performance? Well, one way to explain it is just to point to the differences in the code for the two algorithms. But what does this code represent? Certainly it does not represent some one particular linear causal sequence of transitions, for the **if** as well as the **for** loops encode conditional statements. Rather, it represents a *space of possibilities*: a set of pathways by which the computer can arrive at a particular result. It turns out that the pathways available to a computer implementing MergeSort allow a solution to be reached in fewer time steps than the pathways available to one implementing SelectionSort.

Something similar can be said when comparing different classes of machine. Consider, for instance, figure 2. 'State transition diagrams' such as these are essentially just another way of representing algorithms, although the representation they afford is somewhat 'closer to the hardware', so to speak. The machine depicted is an example of a deterministic finite automaton (DFA). It is not the only kind of computing machine. There are also, for instance,
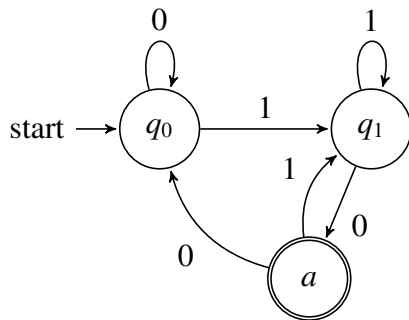
Figure 2: A state diagram representation of a deterministic finite automaton (DFA). Binary strings of variable length are input to the automaton. They are 'accepted' if the machine is found to be in the state a after the last character has been read. This particular machine will accept any string ending in '10'.

nondeterministic finite automata, deterministic and nondeterministic 'pushdown' automata, and deterministic and nondeterministic Turing machines, to name a few. To define these various classes of machine, we describe the possible states and state transitions which they are capable of. For example, DFAs are characterised by a finite set of states, deterministic transitions between states, and the lack of any form of external storage (see Martin, 1997).

Given our characterisations of different types of machine, we can inquire about the set of problems computable by the machines of a particular class. It turns out, for example, that DFAs are severely limited with respect to the class of problems they are capable of solving, while Turing machines, in contrast, are capable of solving any effectively calculable function. We can similarly ask about the resources required to solve particular classes of computational problems by machines of a particular sort. We can ask, for instance, about the class of problems solvable by a deterministic Turing machine in polynomial time, about those solvable by a nondeterministic Turing machine in exponential time, and so on. Answering these and other similar questions will involve appealing to the states and to the state

transitions which are possible for a particular class of machine. This state space, we will say, allows us to construct such a machine to realise an algorithm that will solve the problem in a given amount of time.

The question, "what is the source of quantum speedup?", is a question of just this sort. Quantum computers are just another type of computational machine, and just as for Turing machines and DFAs, quantum computers have associated with them a particular space of states and a particular way of transitioning between states. In order to answer the question "what is the source of quantum speedup?", therefore, we will appeal precisely to the quantum mechanical state space and to the allowable transitions within it, and we will consider these in comparison to the space of states and state transitions possible for a classical computer. And when we do so we will be explaining how-possibly.

We see this, in fact, when we examine the approaches that those in the scientific community have taken to answering this question. Consider Fortnow (2003), for example, who develops an abstract mathematical framework for representing both the computational complexity classes associated with classical and with quantum computing. In Fortnow's framework, both kinds of computation are represented by transition matrices which determine the allowable transitions between possible configurations of a particular kind of machine. To represent the quantum case, Fortnow allows matrix entries to be negative as well as positive, while for the classical case they may only be positive. As a result, in the quantum case matrix entries will sometimes cancel each other out when summed; not so in the classical case. Fortnow shows that this suffices to capture the computational complexity classes associated with classical and quantum computing. According to Fortnow, this means that the fundamental difference between quantum and classical computation is that in quantum computation there can sometimes be interference between computational paths: "The strength of quantum

computing lies in the ability to have bad computation paths eliminate each other thus causing some good paths to occur with larger probability" (Fortnow, 2003, p. 606).[11] Ekert and Jozsa (1998), on the other hand, argue that the fact that quantum systems can sometimes be in entangled states yields a state space for combined quantum systems that is exponentially larger than the state space associated with combined classical systems.[12] And while it is possible to, in a roundabout way, simulate this larger state space with a classical computer, the resource cost of doing so scales exponentially (ibid., 1771).[13]

---

[11]In the 'many worlds' interpretation (Hewitt-Horsman, 2009), of course, all of these paths would be actual and not merely possible. Perhaps. But I do not think it prudent to hinge one's views on scientific explanation on a particular interpretation of quantum mechanics (further, see Cuffaro 2012 for some strong reasons to be skeptical of the many worlds view in the context of quantum computing). This is moot in any case. When a quantum computer finds itself in a state like $|\psi\rangle = |000\rangle_i|f(000)\rangle_o + |001\rangle_i|f(001)\rangle_o + \cdots + |111\rangle_i|f(111)\rangle_o$ we do not, in order to make sense of Fortnow's analysis, need to take the terms in this superposition to represent either actual or possible computational paths. Rather, what is important is only that the possible states of a quantum computer, *unlike* those of a classical computer, include superpositions like $|\psi\rangle$ which have interfering terms.

[12]A system of two or more particles is said to be entangled when one cannot describe one of the particles in the system without referring to all of the others.

[13]There is disagreement here between Fortnow and Ekert and Jozsa. Does this undermine my view? I would say not. We have here two potential how-possibly explanations of quantum speedup. Further empirical research, presumably, will help to decide which of these how-possibly explanations is correct. One might investigate, for instance, whether cases exist in which a quantum algorithm is efficiently classically simulable despite the fact that it utilises entangled states.

But is this really explaining how-possibly? Isn't it the case, one might object, that an algorithm like SelectionSort just is a description of how a system actually goes about solving a problem, and likewise that a description of the state space and state transitions associated with a particular class of system just is a description of the actual resources used by those systems? These mechanisms are *actually* being employed, are they not? Of course this is true. In the same way, Dray's centre fielder *actually* used the ladder to make the catch that he did. But that is not a how-actually explanation, and neither are these. For Dray the role played by a description of the ladder in the explanation of the fielder's catch is to dispel the questioner's puzzlement regarding how the catch was possible, without explaining exactly how it happened (or why it was necessary). But why does the ladder dispel her puzzlement? It does so because pointing out that a ladder was present opens up a whole new range of possibilities for the questioner that simply weren't there before. Likewise in the cases we are considering: the algorithms, or the state spaces, as the case may be, explain by explicitly specifying the set of possibilities open to them.

But is the how-possibly explanation fully satisfactory? Shouldn't we feel the urge to continue our investigation until we have found the actual path taken by the computer through its state space? Let us consider what this would mean in the case of the performance comparison between SelectionSort and MergeSort. In this case we would presumably (assuming the precise input value was known) produce a how-actually explanation by giving a detailed description of the state of the computer after each time step, and in this way we would see exactly how it was that mine took l steps and yours took k. And yet, without referencing the possibility spaces carved out by each algorithm—the alternatives encoded in the conditional statements—it is hard to see how such an answer can be very informative with respect to the actual question that was asked. At best it is to answer a very different question

than the one originally asked. But in the context of a discussion of the performance characteristics of different types of computer it is not clear what an answer to such a question will add to our understanding of these processes. Such an answer, in that context, seems to do little more than restate the original question. The information about the performance characteristics of my and your computer is most crucially contained in the description of their possibility spaces. Such questions are therefore most appropriately answered by appealing to those possibility spaces; i.e., they are most appropriately answered with how-possibly explanations.

4 Conclusion

The kind of how-possibly explanation I have described in this paper bears certain resemblances to both Forber's and to Persson's conceptions of explaining how-possibly. As in Persson's conception, an explanation of the comparative performance characteristics of quantum and classical computers involves, I have argued, a description of the actual mechanisms associated with these machines. The description of a mechanism serves to carve out a particular possibility space for a machine, and as on Forber's view, this possibility space plays a crucial role in a how-possibly explanation of the computationally relevant characteristics of particular observed outcomes. I have further argued that, unlike other interesting examples previously given in the literature on this form of explanation, once one has answered this how-possibly question it is doubtful that continuing on to ask a how-actually question can yield anything more of substance.

The kind of how-possibly question I have described here is characteristic, not only of the inquiry into the source of quantum speedup, but of the science of computability and computational complexity generally—and not just this. Algorithmic processes abound in

nature: in biological systems, in cognitive systems, and also in physical systems, to name but a few. Questions regarding their comparative performance characteristics likewise abound. And I would argue, if I had the space, that all of these questions are most appropriately answered by explaining how-possibly.

References

Aaronson, Scott. *Quantum Computing Since Democritus*. New York: Cambridge University Press, 2013.

Cuffaro, Michael E. "Many Worlds, the Cluster-state Quantum Computer, and the Problem of the Preferred Basis." *Studies in History and Philosophy of Modern Physics* 43 (2012): 35–42.

Dray, William. *Laws and Explanation in History*. Oxford: Oxford University Press, 1957.

Ekert, Artur, and Richard Jozsa. "Quantum Algorithms: Entanglement-enhanced Information Processing." *Philosophical Transactions of the Royal Society A* 356 (1998): 1769–1782.

Forber, Patrick. "Confirmation and Explaining How Possible." *Studies in History and Philosophy of Biological and Biomedical Sciences* 41 (2010): 32–40.

Fortnow, Lance. "One Complexity Theorist's View of Quantum Computing." *Theoretical Computer Science* 292 (2003): 597–610.

Hempel, Carl G. *Aspects of Scientific Explanation And Other Essays in the Philosophy of Science*. New York: The Free Press, 1965.

Hempel, Carl G., and Paul Oppenheim. "Studies in the Logic of Explanation." *Philosophy of Science* 15 (1948): 135–175.

Hewitt-Horsman, Clare. "An Introduction to Many Worlds in Quantum Computation." *Foundations of Physics* 39 (2009): 869–902.

Martin, John C. *Introduction to Languages and the Theory of Computation*. New York: McGraw-Hill, 1997, second edition.

Mehlhorn, Kurt, and Peter Sanders. *Algorithms and Data Structures*. Berlin: Springer, 2008.

Nielsen, Michael A., and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2000.

Persson, Johannes. "Three Conceptions of Explaining How Possibly—and One Reductive Account." In *EPSA Philosophy of Science: Amsterdam 2009*, edited by Henk W. de Regt, Stephan Hartmann, and Samir Okasha. Dordrecht: Springer, 2012, 275–286.

Reiner, Richard. "Necessary Conditions and Explaining How-Possibly." *The Philosophical Quarterly* 43 (1993): 58–69.

Resnik, David B. "How-Possibly Explanations in Biology." *Acta Biotheoretica* 39 (1991): 141–149.

Salmon, Wesley C. "Scientific Explanation: Three Basic Conceptions." *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association* 1984 (1984): 293–305.

———. "Four Decades of Scientific Explanation." In *Scientific Explanation (Minnesota Studies in the Philosophy of Science, Volume XIII)*, edited by Philip Kitcher, and Wesley C. Salmon. Minneapolis: University of Minnesota Press, 1989, 3–219.

Woodward, James. *Making Things Happen*. Oxford: Oxford University Press, 2003.