Kripke's Infinity Argument

Abstract: The paper explores whether Kripke's infinity argument defies a dispositional approach to meaning. The conclusion is that it does not. The infinity argument fails to challenge a dispositional analysis of perceptual terms. It is also far from obvious that the argument challenges an analysis of arithmetical terms, such as '+'. The significance of the argument should be seen, rather, in the context of the claim that we follow a rule much like a symbol machine executes programs. Kripke forcefully demonstrates that this rule-following idea conflates abstract programs and concrete physical objects. Unlike abstract programs, concrete physical symbol systems obey empirical principles, and, as such, are not disposed to return the value of $m+n$ for arbitrary m and n.

In *Wittgenstein on Rules and Private Language*, Saul Kripke (1982) advances two arguments against dispositional accounts of meaning. One of them – the normativity argument – has been widely discussed, and ever since has set much of the agenda in philosophy of language and philosophy of mind. The other one – the infinity argument – has been less influential. My aim here is to investigate the infinity argument, and to take a first step in clarifying its scope. In part 1, I highlight an important difference between infinity and normativity. In part 2, I argue that the infinity argument fails to challenge even simple dispositional accounts of meaning of perceptual terms, such as 'horse'. In part 3, I contend that the argument also does not challenge dispositional accounts of arithmetical terms, such as '+'. In section 4, I suggest that the argument should be located in the context of symbol machines, and is effective against the idea that these machines are disposed to compute infinite arithmetical functions, such as *addition*. The moral is that the dispositionalist cannot fully rely on symbol machines to account for the meaning of arithmetical terms.

In what follows, I assume the standard reading of Kripke's arguments, which has been advanced, for example, by Crispin Wright (1984, 1989), Warren Goldfarb (1985), Simon

Blackburn (1985), and Paul Boghossian (1989). On this reading, the target of Kripke's

arguments is *reductive* theories of meaning: dispositional theories that attempt to account for

meaning in non-semantic and non-intentional terms. Among them are versions of

behaviorism, functionalism, and teleological and causal-informational theories. The crux of

Kripke's arguments is ontological, not epistemological. The claim in these arguments is that

there are no facts, to which the reductive theories can appeal, that constitute the correct

ascription of meaning to people. Lastly, while Kripke discusses the arithmetical term '+', his

arguments apply to all terms and expressions, including perceptual terms. Whether this

reading is faithful to Kripke, and to what extent Kripke's arguments reflect Wittgenstein's, is

something beyond the scope of this paper.

1. Infinity and normativity

Kripke's skeptic wonders what fact could constitute my meaning *addition* by '+' and not

quaddition (quaddition is the two-place function whose values agree with addition when both

arguments are smaller than 57, and 5 otherwise (Kripke, p. 9)). One possible answer is that

meaning is dispositional:

> To mean addition by '+' is to be disposed, when asked for any sum 'x+y', to give the sum of
> x and y as the answer (in particular, to say '125' when queried about '68+57'); to mean quus
> is to be disposed when queried about any arguments, to respond with their quum (in particular
> to answer '5' when queried about '68+57'). True, my actual thoughts and responses in the
> past do not differentiate between the plus and the quus hypotheses; but, even in the past, there
> were dispositional facts about me that did make such a differentiation. To say that in fact I
> meant plus in the past is to say – as surely was the case! – that had I been queried about
> '68+57', I *would* have answered '125'. By hypothesis I was not in fact asked, but the
> disposition was present none the less (Kripke, pp. 22-23).

Kripke, however, resists to this account. He says (p. 24) that the relation between my

meaning *addition* and my use of '+' is a normative relation; that is, if I mean *addition* by '+',

then there is a unique thing that I *ought* to do when I use '+'. The dispositionalist, however, provides only a descriptive account of this relation; she does not point to how I should use '+', but to how I in fact do and will use it. A dispositional analysis would work, therefore, only if there is no gap between my dispositions to perform and the correct applications of predicates. Kripke, however, provides two arguments to the effect that there is such a gap. His first argument – known as the infinity argument – contrasts my finite nature with the requirement for infinite application of predicates. On the one hand, as a finite being, existing for a finite time "not only my actual performance, but also the totality of my dispositions, is finite" (p. 26). My dispositions, that is, "extend to only finitely many cases" (p. 28). In particular:

> It is not true, for example, that if queried about the sum of any two numbers, no matter how large, I will reply with their actual sum, for some pairs of numbers are simply too large for my mind – or my brain – to grasp. When given such sums, I may shrug my shoulders for lack of comprehension; I may even, if the numbers involved are large enough, die of old age before the questioner completes his question. (pp. 26-27).

On the other hand, if I mean *addition* by '+', I should be disposed to apply the term to any *m* and *n*, even if they are very large. It thus follows that there is a gap between the way I ought to use '+', which extends to infinitely many numbers, and the way I do and will use '+', which extends to finitely many numbers. We can summarize the argument, in a more general form, as follows:

1. If my meaning *P* by a predicate S is dispositional, then I should be disposed to apply S in infinitely many cases.
2. "The totality of my dispositions is finite". My dispositions "extend to only finitely many cases."

Conclusion (by Modus Tollens): My meaning *P* by S is not dispositional.

The normativity argument is quite different. It points out the gap between my meaning *addition* by '+' and the misapplications of '+'. There is no dispute that I sometimes make mistakes in the sense that my actual use of '+' differs from the usual addition table: I

sometimes forget to carry, lose attention, and so forth. But if meaning is equated with actual use, the dispositionalist cannot say that I mean *addition* by '+'. Since I am also disposed to make mistakes, the meaning of '+' could be also equated with the erroneous use. Thus the dispositionalist has to say why the meaning is equated with one disposition and not another, something he cannot do. "According to him, the function someone means is to be *read off* from his dispositions; it cannot be presupposed in advance which function is meant" (pp. 29-30). Yet, there is no fact, other than my intentions, that favors one disposition over another.

An important difference between normativity and infinity is this: The normativity argument does not challenge my disposition to provide the correct answer for all *types* of input pairs (m,n). The argument is that since I am *also* disposed to make mistakes, there is nothing to appeal to, other than my intentions, that can tell that '+' means *addition* and not *quaddition*. Kripke does not deny, for example, that I will usually reply '125' when queried about '67+58'. The argument is that given that I will sometimes reply '5', when I'm very tired, distracted, and so forth, there is nothing the dispositionalist can appeal to, other than my intentions, that could explain my meaning *addition* by '+' and not *quaddition*. In contrast, the infinity argument does challenge my disposition to provide the right answer to all types of pairs (m,n). It challenges the very possibility of a disposition that applies to infinitely many numbers. Of course, Kripke grants that we are disposed to reply '125' to '67+58'. But he denies that we are disposed to reply the value of $m+n$ if m and n are large enough.

To summarize the difference, the normativity argument questions how it is possible to distinguish, in dispositional terms, between correct and incorrect usage. How can the meaning of '+' be constituted in one disposition, given that I also have the other disposition. The infinity argument questions how it is possible to apply '+' to infinitely many numbers.

How can the meaning of '+' be constituted in a disposition, given that my disposition to reply the value of *m+n* extends to finitely many numbers?

2. The 'horse' case

On the standard reading, Kripke's arguments are meant to defeat dispositional analyses of all terms, even the analyses of what is known as perceptual terms, such as 'horse' or 'green'. In particular, the infinity claim is taken to be that there is an unbridgeable gap between my finite dispositions and the required, infinite, application of 'horse'. Since the totality of my dispositions "extend to only finitely many cases" (p. 28), and assuming that there are infinitely many actual and possible horses, including horses in the far past and on Alpha Centauri, I cannot be disposed to apply 'horse' to *all* horses. Thus my meaning *horse* by 'horse' cannot be dispositional.

Many think that the argument does not convince (see, e.g., Blackburn pp. 289-290, Boghossian, pp. 528-530). This is not merely because the number of relevant horses might be finite, as one may argue (see e.g., Goldfarb, pp. 478-479). Even if the number is finite, it is very large, and perhaps unbounded, so the argument could be still made. The contention rather is that the infinity argument fails even if there were infinitely many horses. The true claim that I can apply 'horse' only finitely many times is consistent with being disposed to apply 'horse' to any horse, no matter how many horses there are.

Indeed, the infinity argument fails to defy even the simplest dispositional analyses of 'horse'. Take a simple causal-informational dispositional analysis. According to this account, a speaker means by the term 'horse' a property *horse* just in case this speaker is disposed to

apply the term 'horse' to objects with that property, namely to horses. The disposition of the speaker to apply 'horse' to horses is grounded, in turn, in a causal nomological relation between the term 'horse' and the property *horse*. In particular, it is a counterfactual supportive truth that:

> (1) If I were to encounter (get into a causal contact with) *any* particular
> horse, and asked what animal it is, I would call it 'horse'.

Or in a more abbreviated form:

> (1') $\forall x((Hx \ \& \ Ex) > Cx)$, where Hx = x is a horse, Ex = I encounter x, and
> asked what animal x is, Cx = I call x 'horse'.

On influential versions of this account, advanced, for example, by Dretske (1981, 1988) and Fodor (1990), my meaning *horse* by 'horse' is associated with the content of a mental symbol **horse**. Whereas, the content of the mental symbol **horse** expresses the property *horse* by virtue of causal-nomological relations between **horse** and horses (objects that exemplify the property *horse*).

We should note that (1) does not require that I *actually* apply 'horse' to *all* horses. The requirement is just that I would apply 'horse' to *any* horse that I encounter in my lifetime. It is therefore consistent with Kripke's contention that I can apply 'horse' in finitely many cases. In fact, (1) is consistent with the stronger claim that I can *possibly* apply 'horse' in finitely many cases. To see why, assume that there are infinitely many horses, and that it is not possible for me to apply 'horse' more than ten times in my lifetime. This assumption is consistent with (1) because it is open for *any* horse to be in the class of ten applications of 'horse'.

It is now clear how the dispositionalist could reply to the infinity argument, at least with respect to perceptual terms. If the second premise – that my dispositions extends to

finitely many cases – means that I can apply 'horse' in finitely many cases in my lifetime, then this premise is true. But it is also consistent with being disposed to apply 'horse' to all horses. Even if there are infinitely many horses, I can be disposed to apply 'horse' to any one of them. In that case, the first premise will be false. The meaning of 'horse' can be dispositional, without me being disposed to apply 'horse' infinitely many times. And if the second premise means that I am disposed to apply 'horse' to finitely many horses, then this premise is false. I am disposed to apply 'horse' to *any* horse, even if there are infinitely many horses, and even though I can apply 'horse' finitely many times.

Let us consider possible objections to this response. One may object that (1) does not fully capture the notion of disposition. The truth of (1) does not ensure that I am disposed to apply 'horse' to all horses. To be disposed to apply 'horse' to all horses is to actually apply 'horse' to any horse at some point in my lifetime. My disposition to apply 'horse' to all horses requires, in other words, the truth of the following counterfactual:

(2) $\forall x(Hx > (Ex \ \& \ Cx))$

Unlike (1), (2) does require that I call any of the infinitely many actual and possible horses by the word 'horse'. But since I cannot possibly accomplish this task, (2) is clearly false. Thus, my meaning *horse* by 'horse' cannot be dispositional.

My reply is that a dispositional analysis need not satisfy a strong condition of the magnitude of (2). Consider the disposition of a sample of salt to dissolve in water. We associate water-solubility with the claim "If this sample of salt were put in water, then it would dissolve". But surely we understand this counterfactual in the weaker sense, analogous to (1), and not in the stronger sense analogous to (2). Surely we do not expect a particular sample of salt to satisfy the counterfactual "If this sample of salt were put in *all* samples of

water, it would dissolve in each of them". In most cases, there will be nothing to put in water after our sample of salt has dissolved. Similarly, we associate the fragility of a particular piece of glass with the counterfactual "If (once!) dropped, the glass would break". I thus see no reason why my disposition to apply 'horse' to horses is different from water-solubility and fragility, and requires the further constraint that I *actually* have to apply 'horse' to all horses. I may be disposed to apply 'horse' to horses without actually applying 'horse' to horses even once.[1]

There is yet another worry about (1) not capturing the notion of disposition. One may say that even if I am disposed to apply 'horse' to infinitely many horses, as (1) permits, I am still not disposed to apply 'horse' to *all* horses. I am not disposed to apply 'horse', for example, to the horses living far away on Alpha Centauri, since I cannot even survive the trip to Alpha Centauri to get into causal contact with the horses there. The objection, in other words, is that the notion of my horsy disposition requires the truth of the following counterfactual:

(3) $\forall x(Hx > \Diamond(Ex \ \& \ Cx))$,

This counterfactual is, however, false.

My reply is similar to the one before. A dispositional analysis need not satisfy a condition like (3). I am disposed to apply 'horse' to the horses on Alpha Centauri, even though I cannot arrive there. This sample of salt is water-soluble, even if it is not possible for it to survive the trip to Alpha Centauri, to dissolve in water there. We just require the truth of the counterfactual "If this sample of salt were put in the water on Alpha Centauri it would dissolve". Likewise, we need not require that I survive the trip to Alpha Centauri. We just

---

[1] See also Blackburn, pp. 289-290.

need to require the truth of the counterfactual "If I were to arrive to Alpha Centauri, I would call any horse I encounter there 'horse'".[2]

One may point out, correctly, that (1) is true only if read under a *ceteris paribus* clause. The truth of (1), that is, depends on certain ideal background conditions: that I am alive and awake, that the lighting is good, etc. If it is too dark outside, I may not call the horse in front of me 'horse', but 'cow'. On a first glance, there is nothing wrong about a *ceteris paribus* reading. Even Kripke agrees that "*ceteris paribus* notions of dispositions, not crude and literal notions, are the ones standardly used in philosophy and in science" (p. 27). Kripke, however, also thinks that the appeal to *ceteris paribus* clauses is illegitimate in the analysis of meaning. Whether I would have called the horses on Alpha Centauri 'horses' is something I don't know: "How in the world can I tell what would happen if my brain were stuffed with extra brain matter, or if my life were prolonged by some magic elixir? Surely such speculation should be left to science fiction writers or futurologists" (p. 27). Since we do not know how I am to act on Alpha Centauri, we specify ideal conditions that are in accord with what is meant. Under these conditions, (1) becomes true, but the specification is obviously circular. The meaning of 'horse' resides in a *ceteris paribus* clause that gives rise to ideal conditions whose specifications rest on the prior assumption that 'horse' means *horse* (p. 28).

I think that Kripke's worries about the *ceteris paribus* notions of dispositions are correct, and that they challenge a possible reply to the normativity argument. But they do not really challenge my reply to the infinity argument. Let me explain. The aim of the normativity argument is not to deny that I am disposed to apply 'horse' to horses. It is agreed

---

[2] See also Boghossian, p. 529.

that there are conditions under which I will call a horse I never encountered before, and perhaps never will, 'horse'. The concern is that since I am also disposed to apply 'horse' to cows – when the lighting conditions are bad and so forth – we cannot favor, without circularity, the *ceteris paribus* reading that picks out the application of 'horse' to horses over the application of 'horse' to horses-or-cows. Since I am *also* disposed to call cows 'horse', the *ceteris paribus* clause cannot tell which disposition is in accordance with the meaning of 'horse'. We simply favor the conditions that are in accordance with what we take to be the meaning of 'horse'.

The claim in the infinity argument is quite different: Granted that I have finite dispositions, and assuming that there are infinitely many horses, there must be horses to which I am not disposed to apply 'horse'. This would mean that there is a horse, perhaps one that lives on Alpha Centauri, to which, *under no conditions*, I will call 'horse'. If this were the case, then Kripke's complaint would have been just: we are taking the conditional "If I were to arrive to Alpha Centauri, I would call any horse I encounter there 'horse'" to be true, under *ceteris paribus* reading, even though the ideal conditions implicit in the *ceteris paribus* clause do not and could not exist. Kripke, however, is in no position to claim that under all conditions the conditional is false. If the antecedent of the conditional is always false – there are no conditions under which I can encounter the horse on Alpha Centauri – then the conditional itself is always true. And if the antecedent is sometimes true – there are conditions under which I encounter the pertinent horse – it is certainly conceivable that under some of these conditions I will call this horse 'horse'. If Kripke thinks it is inconceivable, it is he who has to prove why, under all circumstances, I cannot call a horse I encounter on Alpha Centauri 'horse'. Short of an argument showing why such a case is inconceivable,

which Kripke does not provide, there is no reason to suspect that the *ceteris paribus* reading of (1) is somehow illegitimate.

Lastly, one may object that (1) cannot be true because the infinitely many actual and potential horses differ from each other in many, perhaps infinitely many, respects. The worry is that a finite creature like me cannot have the appropriate powers to apply 'horse' to any of these very different horses. But, actually, I could have these powers. In fact, there are already many simple information-processing systems with such powers. Consider a flip-detector that flips '+' whenever the input voltage is higher than 2.5 volts, and '-' otherwise. There are infinitely many types of currents that can be detected. Hence, the '-' state of the detector has the powers to detect different voltages on the scale of [0,2.5], though there are infinitely many potential-inputs with different currents. Similarly, there is no special reason why my mental symbol **horse** cannot detect any of the actual and potential horses, even if all these horses differ from each other.

3. The '+' case

We saw that the infinity argument is ineffective when applied to simple perceptual terms. But does the argument work against a dispositional analysis of arithmetical terms, such as '+'? One might think that it does; that the analysis of '+' might require a far more powerful disposition than the analysis of 'horse'. Thus one can argue that my criticism with respect to the 'horse' case is not immediately applicable to the '+' case, which is the case considered by Kripke. If the infinity argument works with respect to '+', it would still show that the meaning of *some* terms cannot be dispositional. It would still show that we cannot provide an all-

inclusive account of meaning, solely in dispositional terms.

Here is why one might think that the analysis of '+' calls for a more complex disposition. In the 'horse' case, I should be able to discriminate between horses and non-horses. I should apply 'horse' to any horse I encounter, and just to horses. But I need not discriminate between types of horses. Nor is there a need to distinguish between previous horses I encountered and the one I encounter now. I apply 'horse' to only one type of objects, namely to horses. In the '+' case, besides the ability to discriminate between numbers and non-numbers, I should also be able to discriminate between different types of numbers (or, *representations of numb*ers, for example, decimals). I should be able to answer '7' when asked for the value of '3+4', but '17' when asked for the value of '11+6'. The problem, however, is that there are infinitely many *types* of numbers. So I must be able to discriminate between infinitely many types (m,n), and to produce for them infinitely many types $m+n$.

Put differently, the infinity problem in the 'horse' case is about tokens. The problem is that there are infinitely many tokens, namely horses, to which I have to be able to apply 'horse'. The infinity problem in the '+' case is *also* about types. In order to compute addition, it is not enough to tell that tokens of '7' from other tokens. In addition, I should be able to discriminate between the other types of numbers. I should be able to tell tokens of '3' from tokens of '4'. But since there are infinitely many numbers, I must have the powers to discriminate between and to produce tokens of infinitely many types. But how can a finite object have this ability? How can a finite object have the required disposition given that, on the one hand, its dispositions extend to finitely many numbers, and on the other, that it should give as a reply the sum of m and $n,$ when asked for the value of $m+n$, for any m and n?[3]

---

[3] The ability to entertain or produce an unbounded number of types of expressions with finite means is known

But the infinity argument is not very effective, even with respect to '+'. Even if the '+'

case might require a more complex disposition, there are finite-size objects with the required

dispositional powers. To take just one example, consider the following neural network. The

net consists of two input units, i_1 and i_2, one output unit, o, and perhaps a number of hidden

units. Assuming that the activation values of the input units range from 0 to 1, we will

uniquely encode the numbers by the rational-values Q_i in [0,1], for example, $Q_i = 1/i$ (where i

is a natural number different from 0; if i=0 then $Q_0 = 0$). The net can be seen as performing

addition if and only if it computes the function f('x','y') = ('x'×'y')/('x'+'y') with the

interpretation I('x') = 1/x. There is no need to expand the net for large *m* and *n*. The relevant

memory is stored in the rational coefficients. Thus the very same net can produce *m+n* for

any inputs m and n, and so is disposed to produce the value of *m+n* for arbitrary *m* and *n*.[4]

That the abstract network computes addition does not undermine the infinity

argument. Kripke would not deny that, in the abstract space of possibilities, the network is

---

as productive or generative capacity. The ability to compute arithmetical functions is one example of
productivity. Another celebrated example is the ability to recognize and generate an unbounded number of
grammatical sentences (e.g., Chomsky 1965). Kripke's argument, then, seems to have a wider scope,
challenging the idea that the mind/brain, as a finite object, has productive capacities, arguing that the
dispositions of an object with finite means "extend to finitely many cases". But this challenge should be
qualified. Productivity is sometimes viewed not just in terms of dispositions, but also as a *theoretical* maxim.
For example, Fodor and Pylyshyn (1988) suggest that there is no need to assume that we possess infinite
generative capacities. "Infinite generative capacity can be viewed, instead, as a consequence or a corollary of
theories formulated so as to capture the greatest number of generalizations with the fewest independent
principles" (1988, p. 33, note 22; see also Chomsky 1986). Theories of cognition generalize over individuals
whose actual performance to produce *m+n* varies quite dramatically. In addition, it is known that individuals
can improve their performance by "relaxing time constraints, increasing motivation, or supplying pencil and
paper" (p. 34). Thus healthy theoretical principles such as generality, simplicity, and economy motivate us to
attribute to individuals the capacity to produce *m+n* for any *m* and *n* (see also Fodor (1990: 94-95)). Kripke,
indeed, emphasizes that his argument is not meant to target this theoretical understanding of productivity (see
pp. 38-39, and note 25 on pages 39-40, and also Wright (1989)). His argument, rather, is that this theoretical
claim is not grounded in *disposition*. If our alleged infinite powers are grounded in principles such as generality,
simplicity, and economy, then, dispositionally speaking, we are not really adders but quadders.

[4] It is not necessary to specify here the inner structure of the net. A result by Siegelmann & Sontag (1995)
indicates that for every function that is Turing-machine computable, there is such a corresponding network with
rational coefficients that computes the values of the function. There is even a finite network, with less than 900
neurons, which is as powerful as a universal Turing machine.

possible. His point, rather, is that this "abstract mathematical object, gets us no further" (p. 33). To answer the skeptic, we would need to show that the network is *also* in the *physical* space of possibilities (p. 34). Is there a physically possible network that is disposed to produce the value of *m+n* for arbitrary *m* and *n*? The difficulty in implementing the abstract net is that it assumes unbounded discriminative powers. The net is finite in the sense that it consists of finitely many neurons and connections whose implementation occupies a finite and bounded physical space. But the price is paid elsewhere. The net is not finite in the sense that its logical definition requires an unbounded number of *types* of atomic components. In particular, it requires the ability to support and discriminate between an unbounded number of types of rational values. If, for example, we encode each natural number *n* by an activation value $1/n$, the net must be able to produce precise activation values for any n, as well as to distinguish between $1/n$ and $1/(n+1)$. Thus if the numbers encoded are very large, the distance between the rational activation values that represent these numbers is very small. In that case, the physical components must support and discriminate between any two rational activation values.

Is it possible to have such a physical net with the proper discriminative powers? It might be. Nothing in current physics excludes the possibility of matter that is infinitely divisible, or objects with infinitely many states. Physical systems, for example with chaotic dynamics, may get into infinitely different conditions, and produce different behaviors from each condition. It is true that we, the observers, cannot observe the difference between the initial values. It is also true that we cannot engineer a reliable enough machine with these abilities. The point, however, is that if the machine can have infinitely different conditions, one of its dispositions may be the production of *m+n* for arbitrary *m* and *n*. Again, the

machine may also have the disposition to produce other values. It may often make mistakes. There is also the normativity challenge of identifying correctness with the disposition to compute addition and not quaddition. But all this is compatible with the machine being able to produce the right sums.

Of course, I do not think that my brain implements this network, nor that my meaning *addition* by '+' is rooted in its dispositional powers. Rather my point is this: since this simplistic network meets the infinity challenge, it is reasonable to think that there might be another, and perhaps more sophisticated, dispositional explanation to my meaning *addition* by '+'. Again, perhaps there is no such explanation. Perhaps my mind/brain cannot be disposed to discriminate between infinitely many types of numbers. But it is surely up to Kripke to show why it cannot be. And the infinity argument falls short of showing that.

4. Symbol machines

I have argued that the infinity argument does not fulfill its task. It does not endanger the dispositional approach to meaning. But does the argument have any value at all? I think it does. My suggestion is to locate the argument in the context of symbol machines. More specifically, I shall argue that the infinity argument challenges the idea that symbol machines are *disposed* to compute functions that are defined over infinitely many types, for example, to produce as outputs $m+n$ for any m and n.

The case of symbol machines is of interest for various reasons. First, Kripke dedicates a lengthy discussion to machines as a variant of the dispositional account (pp. 32-37), apparently as a response to Dummett. Kripke's obvious concern is that his objector will

wonder why I cannot be disposed to compute addition, given that familiar finite machines, which embody programs for addition, have this ability:

> We can interpret the objector as arguing that the rule can be *embodied* in a machine that computes the relevant function. If I build such a machine, it will simply grind out the right answer, in any particular case, to any particular addition problem. The answer that the machine gives is, then, the answer that I intended (p. 33).

In these passages, Kripke advances two criticisms of the use of machines as a way out of skepticism – its finitude and the possibility of malfunction – that are versions of the infinity and normativity arguments. There is also an interesting exegetical point here. Wittgenstein, who apparently had the same concerns, also discusses machines explicitly in the *Investigations* (193-195) and in the *Remarks on the Foundations of Mathematics* (I, 118-130, II (III), 87, and III [IV], 48-49). Kripke not only refers to these paragraphs in Wittgenstein, but declares that, in fact, his "criticisms of dispositional analysis and of use of machines to solve the problems are inspired by these sections" (p. 35, note 24). It would be therefore even more interesting to review the arguments about machines that Kripke finds in Wittgenstein.

The case of symbol machines is of interest for another reason. The view that some symbol machines are disposed to compute addition is widespread, and explicitly advocated, for example, by Fodor and Pylyshyn (1988), Schwarz (1992) and Van der Velde (1996). The view is also used by Chomsky (1986) and Fodor (1990) to reply to Kripke's argument. Moreover, the view is often associated with another thesis, nowadays paradigmatic in cognitive science, that our mind/brain is a sort of such a symbol machine, and that this fact can account for mental capacities that are involved with entertaining and producing infinitely many types. If my mind/brain is a sort of a physical symbol system, then it can account for my alleged ability to reply the value of $m+n$, for any m and n, and perhaps even for my

meaning *addition* by '+'. It would also be interesting therefore to see to whether Kripke's argument contests this picture.[5]

What is the (symbol) machine that Wittgenstein and Kripke discuss in their arguments? Here is a short summary of its main architectural properties:

(a) The operations of the machine are defined over a symbol system, namely a system of representations whose rules of syntax and semantics are recursive. A paradigm example of a symbol system is the decimal system whose members (decimals) represent numbers. The decimal system includes infinitely many types of symbols, each of which is a finite string of primitive or atomic symbols (digits). There are, overall, a finite and bounded number of types of atomic symbols, i.e., ten, and all the other types of decimals are strings of digits, composed according to a finite number of construction and interpretation rules.[6]

(b) There is a functional distinction between the encoded program, which is fixed and finite, and memory, where other symbols are stored, and that can be extended, at least in principle, indefinitely. A paradigm example of a program/memory system is a

---

[5] The hypothesis that our minds/brains are "physical symbol systems" appears in its most explicit form in Newell and Simon (1976). Fodor and Pylyshyn (1988) further argue that symbol machines are the *only* ones – as far as we know – that have the ability to entertain and produce an unbounded number of types of numbers. The argument of Fodor and Pylyshyn to which I refer here is about the productivity of thought (1988: 33-36). Assuming that the human mind/brain has productive or generative capacities, Fodor and Pylyshyn thus infer that the human mind/brain must be a symbol machine. Fodor (1990: 94-95) also invokes the idea of symbol machines to demonstrate that my meaning of '+' can be dispositional. Here my claim is that Kripke's argument threatens the premise that our symbol systems have the required productive powers (unless "productivity" is understood as a theoretical maxim, see notes 3 and 7). Elsewhere, I also argue that Fodor and Pylyshyn's other premise – that symbol machines are the only ones that can account for productivity – is false even when productivity is understood as a theoretical maxim.

[6] The rules of syntax and semantics may be defined as following:

Rules of syntax:
1) '0', '1', ...'9' are primitive decimals (digits).
2) If T is a string, and x is a digit, then Tx is a string.

Rules of semantics:
1) I('0') = 0; I('1') = 1; ...I('9') = 9
2) I(Tx) = 10×T + I(x).

Turing machine. In a Turing machine, the program consists of a finite and fixed number of operations, and the memory tape, which is divided into squares, can be extended indefinitely from both ends.

(c)   The computation process consists of a series of single steps, whereas each step is involved with a bounded change in the configuration of the symbols on the tape and the state of the program. In each computation step of a Turing machine, a symbol is read from the tape (through a read/write finite mechanism), and then the machine operates according to this symbol and the current state in the program. In a Turing machine the possible types of operations are: to replace the symbol by another, to move one square to the left or to the right, or to change its current state.

(d)   The program is sensitive to the syntactic structure of the symbols. This means, for example, that the operations of a program for addition are not defined over whole strings, but over the single digits from which the strings are composed.

These features enable a symbol machine to compute the value of $m+n$ for arbitrary m and n, using only finite means. Given arbitrary m and n, the machine goes through a stepwise process that produces at the end the value of $m+n$. This process is finite in three respects. First, the process consists of finitely many steps. Since the decimals m and n are finite strings of tokens of digits, and given that the value of the computed function is defined, the process will terminate at some point, namely, within a finite number of steps. Second, there are only finitely many *types* of single steps in all the computations the machine can perform (as the program consists of a finite number of types of operations). Nevertheless, the program can be effective with respect to an infinite number of decimals. In particular, a program for addition, whose primitive operations are defined over the digits, can be applied to any pair of numerals

whatsoever. And, third, each type of single step is a change in a finite and bounded number of resources. In a Turing machine, as we saw, each single step is mostly involved with a change in one digit on the tape, in one state of the program, and in the location of the read/write mechanism. In sum, then, producing the value of $m+n$ for any inputs m and n is finite in the sense that (a) the process consists of finitely many steps, (b) there are finitely many types of steps, and (c) each type of step requires finite resources.

All this, however, does not yet show that the machine is disposed to compute addition. Though producing the value of $m+n$ for arbitrary m and n is a finite process, it is not bounded. If the strings are long enough, the machine must be able to use more memory and goes through more computation steps. Since there are infinitely many decimals – some of which are very long strings – no bound can be put on these resources. There will always be sums $m+n$ whose production requires more of these resources. This may not raise a difficulty for the *abstract* Turing machine whose resources can be extended indefinitely. But it does raise a difficulty if the machines we deal with are *physical* objects, whose resources are arguably bounded. If the input strings are long enough, then the *physical* symbol machine will run out of memory space or break down before it produces the sum of these inputs. The concrete physical symbol machine, one may argue, is not disposed to produce the value of $m+n$ for arbitrary m and n.

There is a standard strategy to take care of this difficulty. Since a symbol system maintains a functional distinction between program and memory, we can add more memory to the machine without altering its computational structure, that is, without altering the program executed by the machine: "In a system such as a Turing machine, where the length of the tape is not fixed in advance, changes in the amount of available memory can be

affected without changing the computational structure of the machine; viz. by making more tape available" (Fodor and Pylyshyn 1988: 34-35); "Memory is external to the program, so that the memory capacity can be increased without changing the program, that is without changing the computed function itself" (Van der Velde 1995: 250). Thus if the physical machine is identified with the program it executes, it is disposed to entertain any input pair (m,n) and produce the value of $m+n$: under ideal conditions, when the working memory is unbounded, the machine will produce $m+n$ for any m and n. Consequently, if *we* (or our brains) are physical symbol machines, we are also disposed to entertain any input pair (m,n) and produce for it the sum $m+n$: "if we did have unbounded memory, then, ceteris paribus, we would be able to compute the value of $m+n$ for arbitrary m and n" (Fodor 1990: 95).

This strategy is often rephrased in terms of a distinction between competence and performance. The distinction has been forcefully employed by Chomsky (1965: 3-5) to emphasize that hearers/speakers have the *competence* to hear/understand arbitrarily long expressions even if they do not actually *perform* well when the expressions are too long. In his discussion of Kripke's arguments, Chomsky (1986) puts the distinction as follows:

> But the account of "competence"… deals with the configuration and structure of the mind/brain and takes one element of it, the component of L, to be an instantiation of a certain general system that is one part of the human biological endowment. We could regard this instantiation as a particular program (machine), although guarding against the implications that it determines behavior (p. 238).

The performance of the physical symbol machine is identified with its actual and potential behavior that depends, among other things, on the size of a memory, which is bounded. As a result, its performance is "correct" with respect to a bounded number of input and output types. The competence of the machine, however, is associated with its program for *addition*. As such, the competence of the machines is revealed only when the program operates under ideal conditions, i.e., with unbounded memory: "Competence simply is

performance under ideal processing conditions; i.e. without memory constraints. Competence

will differ from performance when the system does not realize the required amount of

memory" (Schwarz, p. 215). And, under these ideal conditions, the machine entertains any

type of input pair (*m,n*) and produces for it the value of *m+n*.

Kripke, however, criticizes this view. He does not deny the theoretical and practical

virtues of the distinctions between program and memory and competence and performance,

but he denies that the distinctions can help the dispositionalist.[7] Why? Simply because the

disposition to entertain any pair (*m,n*) and produce for it the sum *m+n*, for arbitrary *m* and *n,*

requires not only unbounded memory, but also unbounded iterative powers. To see this,

assume that the state q_2 in my program is tokened whenever '1' occurs in the string. This

requires the tokening of q_2 not only in the cases where the machine encounters the single

string '1', but also in the cases where the machine encounters any string that consists of '1'

(e.g., '101'). But this requires the tokening of q_2 over and over again in the cases where the

machine encounters strings that consist of many '1's. It requires the tokening of q_2 an

unbounded number of times. It requires, in other words, a *perpetuum mobile* working

program.

This is often ignored since we think about the program as an *abstract* entity. And,

indeed, there is no dispute that an abstract Turing machine can entertain arbitrarily long

strings of decimals. But the situation is quite different with "the actual physical machine,

which is subject to breakdown" (Kripke, n. 24, p. 35). A concrete physical machine made of

metal and gears, transistors and wires or neural tissues, "is a finite object, accepting only

---

[7] In fact, Kripke explicitly emphasizes that he does not reject Chomsky's competence-performance distinction. "On the contrary, I personally find that the familiar arguments for the distinction…have great persuasive force" (note 22, p. 30). Kripke also does not think that "Wittgenstein himself would reject the distinction" (note 22, p. 31), provided that we take, as we should do, the notion of competence as *normative*, not descriptive.

finitely many numbers as inputs and yielding only finitely many as output – others are simply too big" (p. 34). A physical machine, unlike an abstract machine, is constrained by physical laws, and, according to these laws, the machine is bound to break down, before it completes the addition of very long strings. A physical symbol machine cannot produce the sum $m+n$ for arbitrary m and n, even if its memory were unbounded. If the two input-strings are too long, the working program will disintegrate before completing the mission. The physical symbol machine is, therefore, not disposed to produce $m+n$ for any m and n, even under ideal memory conditions.[8]

I think that Kripke's comments on symbol machines are right on the mark. The appeal to ideal memory conditions is both unnecessary and unhelpful in solving the infinity problem. It is unnecessary for two reasons. First, computing addition does not require an internal memory tape at all. As Kripke notes, *addition* can be computed by a finite state automaton (Kripke note 24, pp. 35-37). Unbounded memory is required for other arithmetical functions such as multiplication, but not for addition. It is thus surprising that the critics think that the

---

[8] Compare Kripke's argument to the pertinent passages in Wittgenstein:

"But might it not be said that the *rules* lead this way, even if no-one went it? For that is what one would like to say - and here we see the mathematical machine, which, driven by the rules themselves, obeys only mathematical laws and not physical ones." (*Remarks on the Foundations of Mathematics*, IV, 48).

"The machine as symbolizing its actions: the action of a machine – I say at first – seems to be there in it from the start. What does it mean? If I know the machine, everything else, that is its movements, seems to be already completely determined.

We talk as if these parts could only move in this way, as if they could not do anything else. How is this – do we forget the possibility of their bending, breaking off, melting, and so on?

…But when we reflect that the machine could also have moved differently it may look as if the way it moves must be contained in the machine-as-symbol far more determinately than in the actual machine... And it is quite true: the movement of the machine-as-symbol is predetermined in a different sense from that in which the movement of any given actual machine is predetermined." (*Philosophical Investigations*, 193)

"We say, for example, that a machine *has* (possesses) such-and-such possibilities of movement; we speak of the ideally rigid machine which *can* only move in such-and-such a way... We say: "It's not moving yet, but it already has the possibility of moving" – 'so possibility is something very near reality'. Though we may doubt whether such-and-such physical conditions make *this* movement possible, we never discuss whether this is the possibility of this or of that movement... 'so it is not an empirical fact that this possibility is the possibility of precisely this movement'" (*Philosophical Investigations*, 194).

appeal to unbounded memory can meet Kripke's argument, though the argument is not about memory constraints. Second, the assumption that there are ideal memory conditions is not only dubious, but also not necessary even for computing multiplication. It is dubious because the number of particles in each possible physical world may be bounded. And it is unnecessary because we can appeal, instead, to the weaker and more plausible assumption that for every input pair (m,n), there is a possible physical world with enough, but finite, memory, such that this machine (program) can produce the value of $m \times n$.

The appeal to ideal memory conditions is unhelpful because the main problems are with the program, not memory. One problem has to do with normativity. Granted that "actual machines can *malfunction*: through melting wires or slipping gears they may give the wrong answer" (p. 34), the dispositionalist should tell why the competence of the machine is computing *addition* and not *quaddition*. Associating the competence with a program, even if it were justified, is of no help. If 'program' refers to the abstract rule we take the machine to implement, and not to the physical part that implements this rule, then "the physical object is superfluous for the purpose of determining what function is meant" (p. 35). The competence of the machine is really determined by the intentions of the designer. And if the 'program' refers to the implementing physical object, then we are back in square one, as the physical part can malfunction too.

The other problem has to do with infinity. Even if we had the means to sort out proper functioning from malfunctioning, the dispositionalist cannot identify proper functioning (competence) with a function that the system is not disposed to compute. And since the working program cannot produce $m+n$ for any m and n – even if memory is unbounded – the competence of the machine is not computing addition. "Usually this is ignored because the

designer of the machine intended it to fulfill one program". But, as Kripke emphasizes, "the appeal to the designer's program makes the physical machine superfluous; only the program is really relevant. The machine as a physical object is of value only if the intended function can somehow be read off from the physical object alone" (p. 34). Yet the problem is that we cannot read off the addition function from the physical object alone, since there are no conditions under which the physical object can compute the value of $m+n$ for arbitrary m and n.

Kripke's argument against symbol machines is powerful, even if not decisive. It is not decisive because we cannot rule out completely the physical possibility of a symbol machine that is disposed to produce the right sums.[9] But it is powerful because the argument demonstrates that it is highly unlikely that such a machine can be found in our physical environment, if it can exist at all. Even if my brain is a physical symbol system, this would not show that it has the right disposition. If the input strings are long enough, my brain – as the biological organ it is – will cease to exist before it produced the value of $m+n$, even if the memory is unbounded. And even if my meaning *addition* by '+' is dispositional, it is very unlikely that this disposition is rooted, solely, in a symbol system that is disposed to produce the value of $m+n$ for arbitrary m and n.

5. Conclusion

The infinity argument clearly fails to challenge a dispositional analysis of perceptual terms

---

[9] Physical laws can be compatible with the existence of very powerful machines. We recently showed that General Relativity is consistent with the existence of a machine that completes infinitely many steps in a bounded span of time.

(section 2). It is also far from obvious that the argument challenges all analyses of

arithmetical terms (section 3). Still, it is important to note that Kripke himself discusses a

specific family of analyses of the meaning of '+'. The analyses that Kripke considers are

rooted in the idea that we follow a rule much like a symbol machine executes programs.

Following Wittgenstein, Kripke forcefully demonstrates that this rule-following idea

conflates between abstract programs and concrete physical objects. Unlike abstract programs,

concrete physical symbol systems obey empirical principles, and, as such, are not disposed to

return the value of $m+n$ for any m and n (section 4).

References

Blackburn, S. 1984: 'The Individual Strikes Back', *Synthese*, 58, 281-301.

Boghossian, P.A. 1989: 'The Rule-Following Considerations'. *Mind*, 98, 508-549.

Chomsky, N. 1965: *Aspects of the Theory of Syntax*. Cambridge, Mass.: MIT.
    – 1986: *Knowledge of Language*. New York: Praeger.

Dretske, F.I. 1981: *Knowledge and the Flow of Information*. Cambridge, Mass.: MIT.
    – 1988: *Explaining Behavior*, Cambridge, Mass.: MIT.

Fodor, J.A. 1990: 'A Theory of Content II: The Theory'. In his: *A Theory of Content and Other Essays*. Cambridge: MIT: 89-136.

Fodor, J.A. and Pylyshyn Z.W. 1988: 'Connectionism and Cognitive Architecture: A Critical Analysis'. *Cognition*, 28, 3-71.

Goldfarb W. 1985: 'Kripke on Wittgenstein on Rules'. *Journal of Philosophy*, 82, 471-488.

Kripke, S.A. 1982: *Wittgenstein on Rules and Private Language*. Oxford: Blackwell.

Newell, A. and Simon, H.A. 1976: 'Computer Science as Empirical Inquiry: Symbols and Search'. *Communications of the Association for Computing Machinery*, 19 (3), 113-126.

Pitowsky, I. 1990: 'The Physical Church Thesis and Physical Computational Complexity'. *Iyuun*, 39, 81-99.

Schwarz, G. 1992: 'Connectionism, Processing, Memory'. *Connection Science*, 4, 207-226.

Siegelmann, H.T and Sontag E.D. 1995: 'On the Computational Powers of Neural Nets'. *Journal of Computer System Sciences*, 50, 132-150.

Van der Velde, F. 1995: 'Symbol Manipulation with Neural Networks: Production of a Context-Free Language Using a Modifiable Working Memory'. *Connection Science*, 7, 247-280.

Wittgenstein, L. 1953: *Philosophical Investigations*. Oxford: Blackwell.
    - 1956: *Remarks on the Foundations of Mathematics*. Oxford: Blackwell.

Wright, C. 1984: 'Kripke's Account of the Argument Against Private Language'. *Journal of Philosophy*, 81, 759-778.
    – 1989: 'Wittgenstein's Rule-Following Considerations and the Central Project of Theoretical Linguistics'. In A. George (ed.), *Reflection on Chomsky*, Oxford: Blackwell.