Ideal Negative Conceivability and the Halting Problem

Manolo Martínez

Abstract

Our limited a priori-reasoning skills open a gap between our finding a proposition conceivable and its metaphysical possibility. A prominent strategy for closing this gap is the postulation of *ideal conceivers*, who suffer from no such limitations. In this paper I argue that, under many, maybe all, plausible unpackings of the notion of ideal conceiver, it is false that ideal negative conceivability entails possiblity.

1 Introduction

The foundational problem of modal epistemology is that of explaining, or explaining away, our apparent epistemic access to modal facts – *i.e.*, facts about what is necessary, or possible. A very popular proposal has it that such access is mediated by the faculty of *conceiving*. This faculty can be understood as being a special kind of imagining, such that:

> p is conceivable for me if I can imagine a world that I take to verify p. (Yablo 1993, p. 29)

That is, if I can imagine a situation of which I believe that p with truth (Yablo 1993, p. 26).

Our finding some propositions conceivable, the proposal goes, is in *some* way linked to their being metaphysically possible. Just in which way, though, is a matter of dispute, and we have known at least since Kripke that the relation between the conceivable and the possible cannot be one of entailment. One reason for this[1] is that we are not cognitively ideal creatures: our inferential capabilities, our attention and memory have a limited span. Exercising these cognitive faculties when conceiving, thus, may sometimes mislead us into judging that something is possible when it is not or vice versa. As-yet-unproven mathematical or logical statements, such as

> $p\left(2^{18}\right) + 1$ *may be expressed as the sum of two primes.*

---

[1] The only reason I will be concerned with in this paper, but there are other, probably more important ones – *cf.* Chalmers (2002).

– where $p\left(2^{18}\right)$ is the 2^{18}th prime – are a case in point: someone imagining a world that she takes (justifiedly) to verify such a statement is compatible with the statement being false. But, according to the most common opinion, if it is false it is necessarily false.

One prominent neo-rationalist strategy to tackle this problem (*cf.*, *inter alia*, Chalmers (2002) and Menzies (1998)) involves introducing an ancillary notion of *ideal conceivability*, conceivability not subject to our contingent cognitive shortcomings, as a means to bridge, even if partly, the gap between the conceivable and the possible. I will call defenders of this theoretical move *ideal conceivabilists*. So, *e.g.*, Menzies:

> Under what circumstances do our corrective practices discount acts of conceiving as not being veridical indicators of possibility? The answer is simple: when they suffer from one kind of cognitive limitation or other. ...
>
> Let us call a subject who does not suffer any of the limitations recognised in our practice as discounting acts of conceiving an *ideal conceiver*. (Menzies 1998, p. 268f)

Such an ideal conceiver, it is hoped, will close the gap between conceivability and possibility opened by our imperfect a priori-reasoning skills. More concretely, a popular thesis regarding the connection between ideal conceivability and possibility is the following:

CHALMERS: Ideal primary negative conceivability entails primary possibility. (Chalmers 2002, p. 172)

In what follows I will leave the "primary" restriction aside: it is designed to deal with a posteriori necessities, and this type of truths will play no role in my discussion[2]. The idea, then, is that negative ideal conceivability entails (metaphysical) possibility, where negative conceivability is such that

> [A statement] S is negatively conceivable when S is not ruled out a priori, or when there is no (apparent) contradiction in S. (Chalmers 2002, p. 149)

Negative conceivability is to be contrasted with *positive* conceivability: for the latter, a positive act of "modally imagining" (Chalmers 2002, p. 151) the proposition expressed by the target statement is required by the conceiver. For the former, it is enough that the conceiver finds no contradiction (hence the *negative* tag) in the proposition in question.

---

[2] This should not be taken to mean that the *primary/secondary* distinction (closely related to the distinction between epistemic and metaphysical possibility, although I will not be discussing it any further) is unimportant or innocuous. On the contrary, it does crucial work in Chalmers's philosophy of mind, where it is used, for example, to derive anti-materialist conclusions with respect to the mind-body problem – cf. Chalmers (2009).

There are probably many things to say about the cogency of this distinction, when it is applied to the philosophically interesting cases, but I will steer clear of this important complication by restricting the discussion to logico-mathematical statements, whose primary and secondary modal profiles are supposed to match.

As regards the "ideal" modifier, Chalmers relates it to the idea of undefeatability by better reasoning:

> S is ideally conceivable when there is a possible subject for whom S is prima facie conceivable, with justification that is undefeatable by better reasoning. (Chalmers 2002, p. 148)

Finally, in ascertaining what counts as better reasoning in a particular context, Chalmers suggests that we should respect what I will call the *independence requirement*:

> [i]t is important that 'better reasoning' about conceivability not be defined, even in part, as reasoning that better tracks possibility. Such a criterion would trivalize the link between ideal conceivability and possibility. (Chalmers 2002, p. 149)

In the case of logico-mathematical statements, which, I am assuming, are necessary (and possible) iff true, and impossible iff contradictory iff false, the independence requirement asks that we do not tie better reasoning to truth-tracking.

Thus, ideal conceivers are to be thought of as follows:

IDEAL CONCEIVER: A subject is an ideal conceiver of a statement S iff she finds S conceivable with justification that is undefeatable by better reasoning.

We should bear in mind, though, that in this sense being an ideal conceiver does not entail being a perfect conceiver or reasoner; it only implies that, as far as the target proposition is concerned, any better conceiver/reasoner will give the same conceivability verdict as the ideal conceiver[3].

Putting all of this together we can render CHALMERS as the more perspicuous IDEAL:

IDEAL: If an ideal conceiver finds no a priori contradiction in p, then p is metaphysically possible.

Chalmers asserts that such a thesis is "very likely true". The main contention of this paper is that, under a number of plausible and attractive ways of spelling out the notion of undefeatability by better reasoning, IDEAL is false[4].

Before that, though, I will quickly review another, inconclusive worry about the ideal-conceivability move which will be useful as an introduction to my own, hopefully more conclusive, objection.

2   An Inconclusive Objection

One may worry that we do not have as clear a grasp of the idea of idealising away from limitations in memory, attention, time and the like as we would like to think. Thus, Kripke:

---

[3] And, again, the notion of better reasoning at play should not depend in any way on truth-tracking reliability. As we go along, I will signal the places at which this *caveat* is relevant.

[4] I should also say that, although there may be other reasonably clear ways to spell out this notion, I don't know of any equally attractive ones.

> If my brain had been stuffed with sufficient extra matter to grasp
> large enough numbers ... and if my life (in a healthy state) were
> prolonged enough, then given an addition problem involving two
> large numbers, m and n, I would respond with their sum. But ...
> [h]ow 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? (Kripke 1982, p. 27)

Leaving aside worries about how to fix which rule is the addition rule – which
were, of course, the worries that Kripke was raising in the book I have just
cited, but which need not detain us now – there is a problem here for the ideal
conceivabilist. Suppose that a putatively ideal conceiver is invited to give a
verdict of conceivability about instances of the following statement schema:

Addition(m,n,o): The sum of m and n is o.

If IDEAL is true, our ideal conceiver[5] has to deem negatively conceivable – i.e.,
has to fail to find a priori contradiction in – only such statements that are
possible (because true). In order to calculate this, if m, n and o are sufficiently
big, she will need lots of extra brain matter, if only to make room for a sufficient
amount of working memory. But now, Kripke's worry kicks in: "how in the world
can we tell what would happen if her brain...?"

I believe that the ideal conceivabilist has a compelling response to the worry:
we do not need to deal with brains and their matter because, as far as the con-
ceivability of instances of *Addition(m,n,o)* is concerned, an ideal conceiver may
be thought of as an ideal adding device, and we have a perfectly general, per-
fectly formal characterisation of ideal adding devices: *universal Turing machines*
running an addition program.

Turing machines are notional machines (i.e., independent of any implemen-
tation details), which can be fully characterised in a way that leaves no room
for the kind of worries that Kripke raised regarding brain matter. They are to
be conceived of as having a tape with adjacent cells and a head (that can move,
one cell at a time, along the tape) able to read the symbol written at one of
these cells and, if so instructed, erase it and write a new one. A program for the
machine is a set of quadruples: in each quadruple we find, first, a specification
of the state the machine is in; second, the symbol the head is reading; third, the
action that the head has to take – erase and write one of a number of symbols
in its current position, or do nothing; then stay put or move left/right –; finally,
the new state the machine is going to move to. A computation starts with the
machine in the *Initial* state, and some information – its input – written on its
tape, and progresses until the machine reaches the *Final* state (if it ever does),
at which point the machine halts and the calculation is over – the information

---

[5] In fact, for all IDEAL claims, different ideal conceivers might be necessary in order to
conceive in the undefeatable-reasoning way different substitutions of a statement schema. In
the *Addition(m,n,o)* case, though, a single ideal conceiver can handle them all, as we are
about to see.

written at a certain designated area of the tape is the calculation's output[6].

In the case at hand, we may instruct the ideal conceiver (a Turing machine) to do as follows: take the three input quantities, m, n and o, as written on the tape, calculate the sum of the first two and then compare the result to the third. If they are not the same quantity, it has to erase all the tape and write a 1 at the left of the head. 1 should be taken to mean 'I find an a priori contradiction in this statement'. It can be proven that the ideal conceiver will only write 1 when o is not the sum of m plus n.

Although, in general, it might be difficult to provide a satisfactory characterisation of what counts as failing to find a priori contradiction in a statement, the logico-mathematical case is easy in that such a statement is true iff necessary iff possible and false iff contradictory iff impossible; so that checking for truth and falsity is enough to check for contradictoriness[7]. In this context, therefore, failing to find a substitution of *Addition(m,n,o)* contradictory amounts to *not* writing a 1 at the left of the head – maybe because a 0 is written instead by the time we reach the *Final* state, maybe because nothing at all is written[8]. All in all, the ideal conceiver provably behaves as IDEAL claims it should.

In describing the workings of such an ideal conceiver I haven't had to mention brains at all. But if someone insists on having an answer in terms of brain matter, the following can be said: not just any stuffing with extra brain matter will do; only such stuffing as implements the computational equivalent of a Turing machine that performs addition with as long a tape as is needed to deal with the m, n and o in question. And not just any life-extending magic elixir will do; only such elixir that leaves the Turing machine untouched and provides enough time to complete the program.

One compelling answer to this Kripkean worry about ideal conceivers, then, is that recursion theory (the theory, among other things, of Turing machines) provides an informative way in which such an ideal conceiver may be thought of. This answer, though, points to another, more serious problem.

3   Ideal Prediction of Haltings

Consider now the following statement schema:

Halt(P,i): A Turing machine running program P, given input i, eventually halts.

As far as the conceivability of instances of *Halt(P,i)* is concerned, an ideal conceiver is an ideal predictor of haltings. The problem now is that there is

---

[6] This very brief summary can be supplemented by any of a number of textbooks, such as Davis et al. (1994, chapter 6).

[7] It might be that pressing for a specification of the peculiar propositional attitude of *finding a priori contradiction in* p is already a way to put the ideal conceivabilist into trouble. Concentrating in an area of discourse in which checking for truth and falsity is enough, therefore, has the bonus feature of making things as easy as possible for her.

[8] Nothing is written, perhaps, because the *Final* state is never reached. This will not happen in the simple case of *Addition(m,n,o)*, but will happen for semi-decidable problems such as the ones that figure prominently in the sequel.

no Turing machine that, given any program P and input i, resolves whether that program working on that input will halt or not – *cf.* Turing (1936). This result may be extended to any computational equivalent of Turing machines. This means that no Turing machine, running any program, will be able to find an a priori contradiction in many substitutions of the *Halt(P,i)* schema which, in fact, are about programs P that do *not* halt given input i[9]. That is, any possible Turing machine will find ideally negatively conceivable – i.e., will fail to write a 1 at the designated square of the tape, maybe because it never reaches its *Final* state – many propositions which are impossible (because false). IDEAL, under the construal of ideal conceivers as Turing machines, is false.

In summary: ideal conceivers should fail to find a priori contradiction only in those logico-mathematical propositions that are true. This is what is needed for IDEAL to hold; but a (group of) ideal conceiver(s) that is able to deal with arbitrary instances of *Halt(P,i)* – and each of them is, determinately, either true or false – cannot do so by computing. Recursion theory, I suggested above, provides a way in which we may think of ideal conceivers, but this way breaks down when it gets to not decidable (semi-decidable, in this case) problems[10].

On the other hand, the theory of Turing machines goes beyond standard Turing machines such as the ones I have been discussing: for example, oracle

---

[9] *Many* and not *all* substitutions, because Turing machines are able to calculate the halting problem for a restricted class of programs and inputs. For example, an adding device, given any two numbers, will calculate their sum and halt, always. So, when ADD is the addition program it implements, the restricted halting problem is trivial:

Halt(ADD,i): A Turing machine running program ADD, given input i, eventually halts.

The answer is 'yes' for every pair i of numbers.

[10] Again here, IDEAL is compatible with more than one ideal conceiver (Turing machine) being needed in order to deal with different substitutions of the *Halt(P,i)* schema. An ideal conceivabilist may try to exploit this fact by arguing as follows:

> All I have claimed is that, for any statement S, there is a conceiver who has achieved her conclusions by way of undefeatable reasoning, and who finds contradiction in S only if S is false (and hence impossible). Now, such a pairing of statement and conceiver can be trivially made to work for substitutions of the *Halt(P,i)* schema, in the following way:
>
> 1. Consider two Turing machines: the YES machine is such that, for any pair (P,i) provided as input, it immediately prints a 1 at the left of its head and halts. The NO machine, instead, immediately prints a 0 and halts.
>
> 2. Feed all pairs (P,i) that correspond to a program P that halts when supplied with input i to the NO machine, and all other pairs to the YES machine.

It is, indeed, easy to see that this pair of machines can deal with the halting problem faultlessly. It is equally clear that, if this way of allocating conceivers to statements were intended by ideal conceivabilists as one which vindicates IDEAL, we should not really care much about this thesis, which would have turned out to be entirely vacuous.

I should quickly point out that real-life ideal conceivabilists are under no illusion about this; and, in particular, Chalmers advocates for the much more substantial independence requirement of undefeatable reasoning, which in this case rules out the allocation of conceivers to statements based on (antecedent) information about the truth or falsity of the statement in question.

The fact remains that any allocation of conceivers (Turing machines) to statements, such that it respects the independence requirement, is unable to solve the halting problem.

Turing machines and hypercomputers are reasonably well understood theoretical constructs that vastly outperform (standard) Turing machines. Couldn't ideal conceivers be fruitfully thought of as *those*? In fact, as I will now argue, IDEAL is false for these more exotic machines, too.

4   Beyond Turing Machines

4.1   Oracles

Recursion theory has seen it useful to introduce the notion of an *oracle*: a black box that can provide answers to non Turing-computable problems, such as whether any given pair of Turing machine and input will eventually halt. An *oracle Turing machine* is a standard Turing machine hooked up to a halting oracle, such that it can use these oracular capabilities in its own calculations. Couldn't we now suggest that ideal conceivers are to be modelled as oracle Turing machines?

Such machines would, indeed, be able to provide the correct conceivability verdict about any substitution of the *Halt(P,i)* schema above; on the other hand, they are totally unsuited for the role of ideal conceiver. First, they are inherently mysterious: the oracle is a *black* box, and there is no inkling of what may be happening inside that gives it its oracular powers. This is very clear in Turing's original formulation:

> Let us suppose that we are supplied with some unspecified means of solving number-theoretic problems; a kind of oracle as it were. We shall not go any further into the nature of this oracle apart from saying that it cannot be a machine. (Turing 2001, p. 172f)

Relatedly, and crucially, they do not satisfy Chalmers's independence requirement of undefeatable reasoning: they have been introduced precisely as *whatever things that provide the right verdict when confronted to certain number-theoretic problems*. This is just about the least subtle way in which the independence requirement can be flouted, and it would make a modal epistemology based on oracle-reasoning almost perfectly vacuous. None of this means that oracles are useless theoretical constructs, but it does mean that they are useless in alleviating our worries about the nature and eventual coherence of ideal conceivers.

The bottomline is that oracle Turing machines, not meeting Chalmers's independent requirement, don't count as displaying *better reasoning* than Turing machines. For all IDEAL is concerned, they bring nothing new to the table. I turn now to consider other notional machines that do not have this problem: several kinds of so-called *hypercomputers* can be described in a way that exquisitely respects the independence requirement; and each of them is more powerful than a standard Turing machine. Unfortunately, as we are about to see, this is not powerful enough.

4.2   Hypercomputation

Forget black boxes. Today we have a flourishing industry of modifications to Turing machines that make them quite capable of performing supertasks[11]. Let me go quickly through some among these proposals:

Infinite time Turing machines   (ITTM – cf. Hamkins and Lewis (2000), Hamkins (2002)) are allowed to keep going, once they've computed ω steps, on to step $\omega + 1$, $\omega + 2$, ..., $\omega + \omega$, etc.[12] Whenever they reach a limit ordinal stage (ω, $\omega + \omega$, etc.) the value in each cell is substituted by a 'limit' value (the current value of the cell, if it has stabilised, and 1 if the cell has alternated from 0 to 1 unboundedly often (Hamkins 2002, p. 527)), and computation is resumed. ITTMs can decide the halting problem in ω many steps (Hamkins 2002, p. 529).

ITTMs are – despite of their name – not Turing machines: the state of a Turing machine at any stage is a function solely of its state at the previous stage, and the state of ITTMs at a limit stage is, instead, a function of all of the previous ω stages. Indeed, the same kind of classical diagonalisation arguments that prove that the Turing-machine halting problem is not decidable can be straightforwardly extended to prove that, although ITTMs can solve the Turing-machine halting problem, they cannot solve the *supertask halting problem*[13].

The upshot of the supertask halting problem for our actual concerns is that no ITTM ideal-conceivers[14] are able to give the right verdict of negative conceivability about arbitrary substitutions of a corresponding *Supertask-Halt(P,i)* schema:

Supertask-Halt(P,i): An ITTM machine running program P, given input i, eventually halts.

This supertask halting problem is equally non decidable by other notional machines described in the literature on hypercomputation:

Accelerating Turing machines   (ATM – cf. Copeland (2002), Copeland and Shagrir (forthcoming)) are Turing machines that compute each step in half the time it takes to compute the previous stage. This means that a whole computation, possibly with infinite steps, takes place in less than double the time it takes to compute its first step. ATMs have a designated square in which a 0 is written at the beginning of the computation, and which is overwritten with a 1 if the computation the ATM is doing halts. In the way of undertanding ATMs

---

[11] Tasks with an infinite number of steps, cf. Benacerraf (1962).

[12] I will concede that computing during transfinitely many steps makes sense.

[13] Halting problem*s*, really – see (Hamkins 2002, p. 535), (Hamkins 2004, p. 153) for details.

[14] That is, if they have not been cherry-picked in a way that violates the independence requirement in a manner analogous to the one described in footnote 10. Incidentally, we may now note that an effective cherry-picking must be done by something that is more powerful computationally than an oracle Turing machine.

that makes them able to compute the halting problem for Turing machines, they are not Turing machines, and cannot compute their own (supertask) halting problem[15].

Infinitely parallel machines   (IPM – cf. Ziegler (2005) for details) are arrays of infinitely many Turing machines that compute independently and whose outputs are, subsequently, integrated. Chalmers seems to have IPMs in mind when he writes that

> Our inability to know a given Gödel sentence plausibly results from a contingent cognitive limitation ...   our contingent inability to evaluate a predicate of all integers simultaneously (Russell's 'mere medical impossibility') (Chalmers 2002, p. 180)

IPMs are, indeed, able to solve the Turing machine halting problem, but they cannot solve the IPM-halting problem.

In conclusion, all of these hypercomputers fall prey to their own halting problem. To solve this problem, we would need a supertask-halting oracle. This oracle can be introduced[16]; but, crucially for our current interests, only as *whatever things that solve the supertask halting problem*. Not, therefore, in a way that respects the independence requirement. The upshot is that hypercomputers, even if they provide a remedy to Russell's 'mere medical impossibility', still fall short of IDEAL.

4.3   Meta-Oracles

In conclusion, the entailment from negative ideal conceivability to possibility – under the different understandings of ideality I have been reviewing – is falsified by the different halting problems each (hyper-)machine is unable to compute.

Before wrapping up, I want to look again into the idea of interpreting the notion of better reasoning in terms of ever more powerful oracle Turing machines[17]. Let me quickly reiterate why this is not the intended sense of better reasoning, and cannot be. Suppose we propose that

CONCEIVER HIERARCHY: A statement S is ideally negatively conceivable if it is not ruled out a priori by *any* oracle Turing machine.

The problem is that each oracle in the hierarchy is introduced as whatever it is that can give a solution to its intended class of undecidable statements; so that CONCEIVER HIERARCHY is simply a convoluted way of ruling that a statement

---

[15] There is another way of understanding ATMs which makes them Turing machines – but unable to solve the Turing machine halting problem. For discussion, see Copeland and Shagrir (forthcoming).

[16] See (Hamkins 2004, p. 153f) for a discussion of oracles in the hypercomputer context.

[17] The least powerful oracle Turing machine is the one that can solve the halting problem for standard Turing machines; the following in the hierarchy is the one that can solve the halting problem for the least powerful oracle Turing machine, etc.

is ideally negatively conceivable only if it is possible – a very unsubtle way of flouting the independence requirement.

Apart from this unacceptable feature of the proposal, Conceiver Hierarchy is unsatisfactory in another (lesser) respect. Recall Menzies's original sense of ideal conceivability as conceivability not affected by any cognitive limitation. We have seen that a conceiver that is not cognitively limited in any obvious respect can still fall short of the goal of linking negative conceivability and possibility. The idea, which helps making ideal conceivers attractive, that they are *just like us but flawless* is not really tenable.

Now, one could try to rebuild the link between oracle-conceiving and (a flawless version of) us in the following way: introduce a "meta-oracle", an aggregator that checks each oracle machine in the hierarchy to see whether it has been able to rule out the target statement. Such an aggregator would channel the information in the hierarchy for us to use, say, in our own modal appraisals. But no such channeling would help us meet IDEAL, for already familiar reasons: the meta-oracle has its own halting problem.

5 Concluding Remarks

As Hamkins puts it,

> Any notion of computation naturally provides a corresponding halting problem. (Hamkins 2004, p. 153)

There simply is no way around the halting problem.

Chalmers (2002, p. 180) considers briefly the case of mathematical truths as a potential counterexample to IDEAL. His suggestion is that it is plausible that every mathematical statement is either ideally knowable or untrue. As evidence for this dichotomy he cites, on the one hand, all truths of arithmetic (which are true, but also ideally knowable by an ITTM, for example); and, on the other hand unprovable statements of set theory (maybe unknowable, even ideally, but also dubiously true).

The (supertask) halting problem falls under neither side of the dichotomy: it provides a set of statements, all of them determinately true or false, and such that infinitely many of them are unknowable by a number of radically idealised cognisers – Turing machines, and the notional hypercomputers of section 4.2.

A line of response to this family of counterexamples to IDEAL by the ideal conceivabilist could be to work out an intelligible way to further idealise these machines that still respects the independence requirement. For example (the only relevant example I can think of, although of course there might be others I have overlooked), the ideal conceivabilist might resort to defending that the reasoning of ideal conceivers is better than that of Turing machines in that the former have ideal *creativity* (creativity, that is, that cannot be bettered) in coming up with proofs of the halting statement for particular instances of P and i. Ideal creativity, moreover, that cannot be reduced to or otherwise explained in algorithmic terms.

It is, I think, very reasonable to doubt that we have a firm grip on the notion of ideal, non-algorithmic creativity: note that this is a faculty which should allow a (team of) ideal conceiver(s) to ascertain the falsity of false statements in the (not decidable) halting problem so that they can refrain from issuing a verdict of inconceivability regarding them. Moreover, it has to be specifiable in terms which are independent from the goal of proving the truth or falsity of logico-mathematical statements. But it is plausible that whatever notion we have of ideal creativity depends precisely on its theorem-proving and -disproving goals: something along the lines of 'ideal creativity is an ability to come up with (possibly unexpected and elegant) proofs of, or counterexamples to, its target theses that cannot be bettered'. Relying in ideal creativity so described would trivialise the claim that ideally creative conceivers can bridge the gap between the conceivable and the possible.

In this paper I have argued for the following conditional claim: if we idealise by identifying ideal conceivers with Turing (hyper-)machines, then IDEAL is false. I also think that – as is sometimes the case with this kind of conditional results, when the antecedent of the conditional is sufficiently plausible and attractive – the failure of Turing (hyper-)machines in securing IDEAL goes some way towards moving the burden of proof to the ideal conceivabilist's shoulders. It is now more urgent than before that she provides a clear characterisation of the further respects in which idealisation is needed, and of what counts as idealised[18].

Until then, that is, until the ideal conceivabilist can come up with a characterisation of the extra ingredient in ideal conceivability, there are only two options for her: accepting that IDEAL is false or embracing quietism about the nature of ideal conceivers. The former is bad news at least for a very influential line of work in the neo-rationalist project; the latter is unlikely to convince those sceptic about ideal conceivers[19].

References

Benacerraf, P.: 1962, Tasks, super-tasks, and the modern eleatics, *Journal of Philosophy* **59**(24), 765–784.

Chalmers, D.: 2002, Does Conceivability Entail Possibility?, *in* T. Gendler and J. Hawthorne (eds), *Conceivability and Possibility*, Oxford University Press, pp. 145–200.

---

[18] There are very many aspects of rationality that cannot, or not clearly, be modelled by Turing machines: say, a certain kind of subtlety in weighing pros and cons in everyday life. This is, I believe, the kind of abilities that make Chalmers talk of rationality as open ended and as resisting complete characterisation. On the other hand, these abilities are irrelevant to the issue at hand: the (ideal negative) conceivability of logico-mathematical statements.

Chalmers, D.: 2009, The two-dimensional argument against materialism, *in* B. McLaughlin and A. Beckermann (eds), *The Oxford Handbook of Philosophy of Mind*, Oxford University Press.

Copeland, B. J.: 2002, Accelerating turing machines, *Minds and Machines* **12**(2), 281–300.

Copeland, B. J. and Shagrir, O.: forthcoming, Do accelerating turing machines compute the uncomputable?, *Minds and Machines* .

Davis, M., Sigal, R. and Weyuker, E.: 1994, *Computability, Complexity and Languages*, Academic Press.

Hamkins, J. D.: 2002, Infinite time turing machines, *Minds and Machines* **12**(4), 567–604.

Hamkins, J. D.: 2004, Supertask computation, *in* B. Löwe, B. Piwinger and T. Räsch (eds), *Classical and New Paradigms of Computation and their Complexity Hierarchies*, Kluwer Academic Publishers, pp. 141–58.

Hamkins, J. D. and Lewis, A.: 2000, Infinite time turing machines, *Journal of Symbolic Logic* **65**(2), 567–604.

Kripke, S.: 1982, *Wittgenstein on Rules and Private Language*, Harvard University Press.

Menzies, P.: 1998, Possibility and Conceivability: A Response-Dependent Account of Their Connections, *in* R. Casati (ed.), *European Review of Philosophy, Volume 3: Response-Dependence*, Stanford, pp. 255–277.

Turing, A.: 1936, On computable numbers, with an application to the entscheidungsproblem, *Proceedings of the London Mathematical Society* pp. 230–265.

Turing, A.: 2001, Systems of logic based on ordinals, *in* R. Gandy and C. Yates (eds), *Collected Works*, Vol. 4 (Mathematical Logic), North Holland.

Yablo, S.: 1993, Is Conceivability a Guide to Possibility?, *Philosophy and Phenomenological Research* **53**(1), 1–42.

Ziegler, M.: 2005, Computational power of infinite quantum parallelism, *International Journal of Theoretical Physics* **44**, 2059–2071.