Best of LessWrong: April 2014

Best of LessWrong: April 2014

1. [Botworld: a cellular automaton for studying self-modifying agents embedded in their environment](#)
2. [A brief summary of effective study methods](#)
3. [Be comfortable with hypocrisy](#)
4. [How long will Alcor be around?](#)
5. [Business Networking through LessWrong](#)

Botworld: a cellular automaton for studying self-modifying agents embedded in their environment

On April 1, I started working full-time for MIRI. In the weeks prior, while I was winding down my job and packing up my things, Benja and I built *Botworld*, a cellular automaton that we've been using to help us study self-modifying agents. Today, we're publicly releasing Botworld on the new MIRI github page. To give you a feel for Botworld, I've reproduced the beginning of the technical report below.

This report introduces *Botworld*, a cellular automaton that provides a toy environment for studying self-modifying agents.

The traditional agent framework, used for example in Markov Decision Processes and in Marcus Hutter's universal agent AIXI, splits the universe into an agent and an environment, which interact only via discrete input and output channels.

Such formalisms are perhaps ill-suited for real self-modifying agents, which are embedded within their environments. Indeed, the agent/environment separation is somewhat reminiscent of Cartesian dualism: any agent using this framework to reason about the world does not model itself as part of its environment. For example, such an agent would be unable to understand the concept of the environment interfering with its internal computations, e.g. by inducing errors in the agent's RAM through heat.

Intuitively, this separation does not seem to be a fatal flaw, but merely a tool for simplifying the discussion. We should be able to remove this "Cartesian" assumption from formal models of intelligence. However, the concrete non-Cartesian models that have been proposed (such as Orseau and Ring's formalism for space-time embedded intelligence, Vladimir Slepnev's models of updateless decision theory, and Yudkowsky and Herreshoff's tiling agents) depart significantly from their Cartesian counterparts.

Botworld is a toy example of the type of universe that these formalisms are designed to reason about: it provides a concrete world containing agents ("robots") whose internal computations are a part of the environment, and allows us to study what happens when the Cartesian barrier between an agent and its environment breaks down. Botworld allows us to write decision problems where the Cartesian barrier is relevant, program actual agents, and run the system.

As it turns out, many interesting problems arise when agents are embedded in their environment. For example, agents whose source code is readable may be subjected to Newcomb-like problems by entities that simulate the agent and choose their actions accordingly.

Furthermore, certain obstacles to self-reference arise when non-Cartesian agents attempt to achieve confidence in their future actions. Some of these issues are raised by Yudkowsky and Herreshoff; Botworld gives us a concrete environment in which we can examine them.

One of the primary benefits of Botworld is *concreteness*: when working with abstract problems of self-reference, it is often very useful to see a concrete decision problem ("game") in a fully specified world that directly exhibits the obstacle under consideration. Botworld makes it easier to visualize these obstacles.

Conversely, Botworld also makes it easier to visualize suggested agent architectures, which in turn makes it easier to visualize potential problems and probe the architecture for edge cases.

Finally, Botworld is a tool for communicating. It is our hope that Botworld will help others understand the varying formalisms for self-modifying agents by giving them a concrete way to visualize such architectures being implemented. Furthermore, Botworld gives us a concrete way to illustrate various obstacles, by implementing Botworld games in which the obstacles arise.

Botworld has helped us gain a deeper understanding of varying formalisms for self-modifying agents and the obstacles they face. It is our hope that Botworld will help others more concretely understand these issues as well.

Overview

Botworld is a high level cellular automaton: the contents of each cell can be quite complex. Indeed, cells may house robots with register machines, which are run for a fixed amount of time in each cellular automaton step. A brief overview of the cellular automaton follows. Afterwards, we will present the details along with a full implementation in Haskell.

Botworld consists of a grid of cells, each of which is either a *square* or an impassable *wall*. Each square may contain an arbitrary number of *robots* and *items*. Robots can navigate the grid and possess tools for manipulating items. Some items are quite useful: for example, *shields* can protect robots from attacks by other robots. Other items are intrinsically valuable, though the values of various items depends upon the game being played.

Among the items are *robot parts*, which the robots can use to construct other robots. Robots may also be broken down into their component parts (hence the necessity for shields). Thus, robots in Botworld are quite versatile: a well-programmed robot can reassemble its enemies into allies or construct a robot horde.

Because robots are transient objects, it is important to note that players are not robots. Many games begin by allowing each player to specify the initial state of a single robot, but clever players will write programs that soon distribute themselves across many robots or construct fleets of allied robots. Thus, Botworld games are not scored depending upon the actions of the robot. Instead, each player is assigned a home square (or squares), and Botworld games are scored according to the items carried by all robots that are in the player's home square at the end of the game. (We may imagine these robots being airlifted and the items in their possession being given to the player.)

Robots cannot see the contents of robot register machines by default, though robots *can* execute an inspection to see the precise state of another robot's register machine. This is one way in which the Cartesian boundary can break down: It may not

be enough to choose an optimal *action*, if the way in which this action is *computed* can matter.

For example, imagine a robot which tries to execute an action that it can prove will achieve a certain minimum expected utility u_min. In the traditional agent framework, this can imply an optimality property: if there is *any* program p our robot could have run such that our robot can prove that p would have received expected utility ≥ u_min, then our robot will receive expected utility ≥ u_min (because it can always do what that other program would have done). But suppose that this robot is placed into an environment where another robot reads the contents of the first robot's register machine, and gives the first robot a reward *if and only if the first robot runs the program "do nothing ever"*. Then, since this is not the program our robot runs, it will not receive the reward.

It is important to note that there are two different notions of time in Botworld. The cellular automaton evolution proceeds in discrete steps according to the rules described below. During each cellular automaton step, the machines inside the robots are run for some finite number of ticks.

Like any cellular automaton, Botworld updates in discrete *steps* which apply to every cell. Each cell is updated using only information from the cell and its immediate neighbors. Roughly speaking, the step function proceeds in the following manner for each individual square:

The output register of the register machine of each robot in the square is read to determine the robot's *command*. Note that robots are expected to be initialized with their first command in the output register.

The commands are used in aggregate to determine the robot *actions*. This involves checking for conflicts and invalid commands.

The list of items lying around in the square is updated according to the robot actions. Items that have been lifted or used to create robots are removed, items that have been dropped are added.

Robots incoming from neighboring squares are added to the robot list.

Newly created robots are added to the robot list.

The input registers are set on all robots. Robot input includes a list of all robots in the square (including exiting, entering, destroyed, and created robots), the actions that each robot took, and the updated item list.

Robots that have exited the square or that have been destroyed are removed from the robot list.

All remaining robots have their register machines executed (and are expected to leave a command in the output register.)

These rules allow for a wide variety of games, from NP-hard knapsack packing games to difficult Newcomb-like games such as a variant of the Parfit's hitchhiker problem (wherein a robot will drop a valuable item only if it, after simulating your robot, concludes that your robot will give it a less valuable item).

Cartesianism in Botworld

Though we have stated that we mean to study non-Cartesian formalizations of intelligence, Botworld does in fact have a "Cartesian" boundary. The robot parts are fundamental objects, the machine registers are non-reducible. The important property of Botworld is not that it lacks a Cartesian boundary, but that the boundary is *breakable*.

In the real world the execution of a computer program is unaffected by the environment *most* of the time (except via the normal input channels). While the contents of a computer's RAM *can* be changed by heating it up [with a desk lamp](#), they are usually not. An Artificial General Intelligence (AGI) would presumably make use of this fact. Thus, an AGI may commonly wish to ensure that its Cartesian boundary is not violated in this way over some time period (during which it can make use of the nice properties of Cartesian frameworks). Botworld attempts to model this in a simple way by requiring agents to contend with the possibility that they may be destroyed by other robots.

More problematically, in the real world, the internals of a computer program will always affect the environment—for example, through waste heat emitted by the computer—but it seems likely that these effects are usually unpredictable enough that an AGI will not be able to improve its performance by carefully choosing e.g. the pattern of waste heat it emits. However, an AGI will need to ensure that these unavoidable violations of its Cartesian boundary will *in fact* not make an expected difference to its goals. Botworld sidesteps this issue and only requires robots to deal with a more tractable issue: Contending with the possibility that their source code might be read by another agent.

Our model is not realistic, but it is simple to reason about. For all that the robot machines are not reducible, the robots are still embedded in their environment, and they can still be read or destroyed by other agents. We hope that this captures some of the complexity of naturalistic agents, and that it will serve as a useful test bed for formalisms designed to deal with this complexity. Although being able to deal with the challenges of Botworld is presumably not a good indicator that a formalism will be able to deal with *all* of the challenges of naturalistic agents, it allows us to see in concrete terms how it deals with some of them.

In creating Botworld we tried to build something implementable by a lower-level system, such as Conway's [Game of Life](#). It is useful to imagine such an implementation when considering Botworld games.

Future versions of Botworld may treat the robot bodies as less fundamental objects. In the meantime, we hope that it is possible to picture an implementation where the Cartesian boundary is much less fundamental, and to use Botworld to gain useful insights about agents embedded within their environment. Our intent is that when we apply a formalism for naturalistic agents to the current implementation of Botworld, then there will be a straightforward translation to an application of the same formalism to an implementation of Botworld in (say) the Game of Life.

The [full technical report](#) goes on to provide an implementation of Botworld in Haskell. You can find the source code on the [MIRI Botworld repository](#). Sample games are forthcoming.

Benja and I will be writing up some of the results we've achieved. In the meantime, you're encouraged to play around with it and build something cool.

A brief summary of effective study methods

EDIT: Reworked and moved to Main following [Gunnar_Zarncke's advice](#).

Related to: [Book Review: How Learning Works](#), [Build Small Skills in the Right Order](#), [What are useful skills to learn at university?](#)

This article is organized into three sections focusing on attention, processing and recall respectively. The advice in each section is roughly organised in order of usefulness, although your mileage may vary. It's best to view this as a menu of study techniques rather than an in depth guide.

Just follow at the links provided if you wish to learn more about any point. Links with a lettered superscript[a] generally link to a part of a YouTube video while those with a numbered superscript[1] link to an article. Links without any superscript generally link to another LessWrong page.

Paying Attention

Attention is *very* important for learning. Where you spend it [directly determines](#) which areas of your brain you'll develop while studying and learning new skills.

1. [Split your study up into 25 minute chunks, separated by five minute breaks](#)[a]
   Also known as the [Pomodoro Technique](#)[b]. This one is simple to implement but tremendously effective. It will protect you from attention burnout, increase your useful study-time, and help prevent distractions from becoming procrastination by setting up a [Schelling fence](#) around your breaks.
2. [Focus on one task at a time](#)[1]
   Multitasking is one of the worst things you can do while studying, it can reduce your productivity by up to 40% and divides your attention up unnecessarily which will impair your ability absorb new information. If social media and the internet is of a particular distraction to you, tools such as [Stay Focused](#) can help you stay on track.
3. [Set up your study environment](#)[a]
   Exploit [situational psychology](#) by making your environment more conducive to study; identify cues that cause you to procrastinate, remove them if possible, and set up cues for studying. Mentioned in the video, a 'study lamp' can make an effective cue providing it is only ever used for studying. Additionally joining a study group can be an effective way to do this (a good example being the [LessWrong Study Hall](#)).
4. [Choose the right music](#)[2]
   There are a few rules of thumb to follow here. Avoid listening to music while trying to absorb new information, though if your aural environment is particularly distracting then music without lyrics or white noise can be useful. Don't use unfamiliar music or music with lyrics in as this will unnecessarily tax your ability to focus. Music can increase productivity for mundane or well-practiced tasks involving low mental effort.

Learning Material

Before going any further I'd advise you to watch [this video](c)[c]. It's an excellent explanation of why just going over material isn't enough to actually learn it and additionally dispels a few myths about the important factors in learning.

1. [Understand the principles behind 'deep processing'](c)[c]
   The key thing to understand here is that the more you relate a new concept to ones previously learned, the more likely you are to remember it. This is *far* more effective than learning by rote, not only does it improve recall but it also improves your ability to apply the material. A study strategy that forces you to process things deeply is called to as an [orienting task](c)[c].
2. [Develop your metacognition](c)[c]
   Metacognition refers to your beliefs about how well you know the material you're studying. Overconfidence here is negatively correlated with academic success (see the video) and [can prevent you from updating on new material](d)[d]. One of the reasons for this negative correlation is that overconfident learners spend less time on material than they should. Being sure to test yourself on your knowledge regularly can go a long way to combating this.
3. [Understand the difference between recognition and recollection](a)[a]
   Related to the previous point, a sense of recognition is one of the biggest causes of overconfidence when reviewing material. A good solution is to test yourself on your ability to recall material before you review it. Not only will doing so help you avoid mistaking recognition for recollection, but knowing what you don't know will help target your revision.
4. [Troubleshoot your understanding](e)[e]
   In most subjects, concepts have a chain of dependencies with advanced concepts depending on the more fundamental ones (in mathematics this chain is particularly long). If you're having trouble understanding a new concept it is very unlikely that you're inherently bad at understanding that concept, rather there's a flaw in your understanding of the more fundamental concepts that lead up to it. Target your understanding of those and understanding the concept in question will become much easier.

Holding onto Information

Once you've processed the material effectively you need to be able to recall it efficiently. While deep processing helps you get information into long term memory, getting it to stay there is a different matter entirely. Memory follows what's known as the [forgetting curve](3)[3]. Forgetting has not so much to do with losing the information, but rather having trouble retrieving it – and as far as learning goes you haven't really learned something until you can effectively retrieve the information.

1. [Test yourself on material](4)[4]
   Practicing retrieval has a dramatic effect on your ability to recall information. Key to this method is ensuring your cues are appropriate to the way you're going to be test, so past paper questions tend to be best. When using flashcards it is

important to make sure that the cues require you to not only recall the information, but process it on a deep level too.
  2. [Make use of spaced repetition](#)[4]
Spaced repetition is testing yourself on material over incrementally larger periods of time (an hour, a day, a week, a month and so on). The idea is to test yourself on information just as you're about to forget it and as it turns out, it is far more efficient than just blindly testing yourself on material over and over. Keeping track of when to review information can be a pain, fortunately there's [plenty of spaced repetition](#) software out there to do that for you (I personally find [Mnemosyne](#) is simple to implement and use).
  3. [Get some sleep](#)[a]
Sleep is absolutely crucial for retention. If you must cram, make sure you do it the night before the exam, if you do things the other way round your memory will be considerably worse off for it. In general make sure you get a good nights sleep every day that you've studied. If you're having trouble sleeping due to spending a lot of time at a computer [f.lux](#) might be helpful to you.

Video Sources:
[a] *Study Less, Study Smart* | M. Lobdell - Pierce College
[b] *What is the Pomodoro Technique?* | F. Crillo
[c] *How to Get the Most Out of Studying* | S. Chew - Samford University
[d] *Khan Academy and the effectiveness of science videos* | D. Muller - Veritasium
[e] *Let's use video to reinvent education* | S. Khan - Khan Academy

Article Sources:
[1] *Multitasking: The Cognitive Costs of Multitasking* | K. Cherry - About.com
[2] *Does Listening to Music While Working Make You Less Productive?* | A. M. Paul - Time.com
[3] *The Forgetting Curve* | Wikipedia
[4] *Spaced Repetition* | gwern

Be comfortable with hypocrisy

Neal Stephenson's *The Diamond Age* takes place several decades in the future and this conversation is looking back on the present day:

> "You know, when I was a young man, hypocrisy was deemed the worst of vices," Finkle-McGraw said. "It was all because of moral relativism. You see, in that sort of a climate, you are not allowed to criticise others-after all, if there is no absolute right and wrong, then what grounds is there for criticism?" [...]
>
> "Now, this led to a good deal of general frustration, for people are naturally censorious and love nothing better than to criticise others' shortcomings. And so it was that they seized on hypocrisy and elevated it from a ubiquitous peccadillo into the monarch of all vices. For, you see, even if there is no right and wrong, you can find grounds to criticise another person by contrasting what he has espoused with what he has actually done. In this case, you are not making any judgment whatsoever as to the correctness of his views or the morality of his behaviour-you are merely pointing out that he has said one thing and done another. Virtually all political discourse in the days of my youth was devoted to the ferreting out of hypocrisy." [...]
>
> "We take a somewhat different view of hypocrisy," Finkle-McGraw continued. "In the late-twentieth-century Weltanschauung, a hypocrite was someone who espoused high moral views as part of a planned campaign of deception-he never held these beliefs sincerely and routinely violated them in privacy. Of course, most hypocrites are not like that. Most of the time it's a spirit-is-willing, flesh-is-weak sort of thing."
>
> "That we occasionally violate our own stated moral code," Major Napier said, working it through, "does not imply that we are insincere in espousing that code."

I'm not sure if I agree with this characterization of the current political climate; in any case, that's not the point I'm interested in. I'm also not interested in moral relativism.

But the passage does point out a flaw which I recognize in myself: a preference for consistency over actually doing the right thing. I place a lot of stock--as I think many here do--on self-consistency. After all, clearly any moral code which is inconsistent is wrong. But dismissing a moral code for inconsistency or a person for hypocrisy is lazy. Morality is hard. It's easy to get a warm glow from the nice self-consistency of your own principles and mistake this for actually being right.

Placing too much emphasis on consistency led me to at least one embarrassing failure. I decided that no one who ate meat could be taken seriously when discussing animal rights: killing animals because they taste good seems completely inconsistent with placing any value on their lives. Furthermore, I myself ignored the whole concept of animal rights because I eat meat, so that it would be inconsistent for me to assign animals any rights. Consistency between my moral principles and my actions--not being a hypocrite--was more important to me than actually figuring out what the correct moral principles were.

To generalize: holding high moral ideals is going to produce cognitive dissonance when you are not able to live up to those ideals. It is always tempting--for me at least-

-to resolve this dissonance by backing down from those high ideals. An alternative we might try is to be more comfortable with hypocrisy.

Related: [Self-deception: Hypocrisy or Akrasia?](#)

How long will Alcor be around?

The Drake equation for cryonics is pretty simple: work out all the things that need to happen for cryonics to succeed one day, estimate the probability of each thing occurring independently, then multiply all those numbers together. Here's one example of the breakdown from Robin Hanson. According to the 2013 LW survey, LW believes the average probability that cryonics will be successful for someone frozen today is 22.8% assuming no major global catastrophe. That seems startlingly high to me – I put the probability at at least two orders of magnitude lower. I decided to unpick some of the assumptions behind that estimate, particularly focussing on assumptions which I could model.

EDIT: This needs a health warning; here be overconfidence dragons. There are psychological biases that can lead you to estimating these numbers badly based on the number of terms you're asked to evaluate, statistical biases that lead to correlated events being evaluated independently by these kind of models and overall this can lead to suicidal overconfidence if you take the nice neat number these equations spit out as gospel.

Every breakdown includes a component for 'the probability that the company you freeze with goes bankrupt' for obvious reasons. In fact, the probability of bankruptcy (and global catastrophe) are particularly interesting terms because they are the only terms which are 'time dependant' in the usual Drake equation. What I mean by this is that if you know your body will be frozen intact forever, then it doesn't matter to you *when* effective unfreezing technology is developed (except to the extent you might have a preference to live in a particular time period). By contrast, if you know safe unfreezing techniques will definitely be developed one day it matters very much to you that it occurs *sooner* rather than later because if you unfreeze before the development of these techniques then they are totally wasted on you.

The probability of bankruptcy is also very interesting because – I naively assumed last week – we must have excellent historical data on the probability of bankruptcy given the size, age and market penetration of a given company. From this – I foolishly reasoned – we must be able to calculate the *actual* probability of the 'bankruptcy' component in the Cryo-Drake equation and slightly update our beliefs.

I began by searching for the expected lifespan of an average company and got two estimates which I thought would be a useful upper- and lower-bound. Startup companies have an average lifespan of four years. S&P 500 companies have an average lifespan of fifteen years. My logic here was that startups must be the most volatile kind of company, S&P 500 must be the least volatile and cryonics firms must be somewhere in the middle. Since the two sources only report the average lifespan, I modelled the average as a half-life. The results really surprised me; take a look at the following graph:

Probability that company you sign with will survive x years
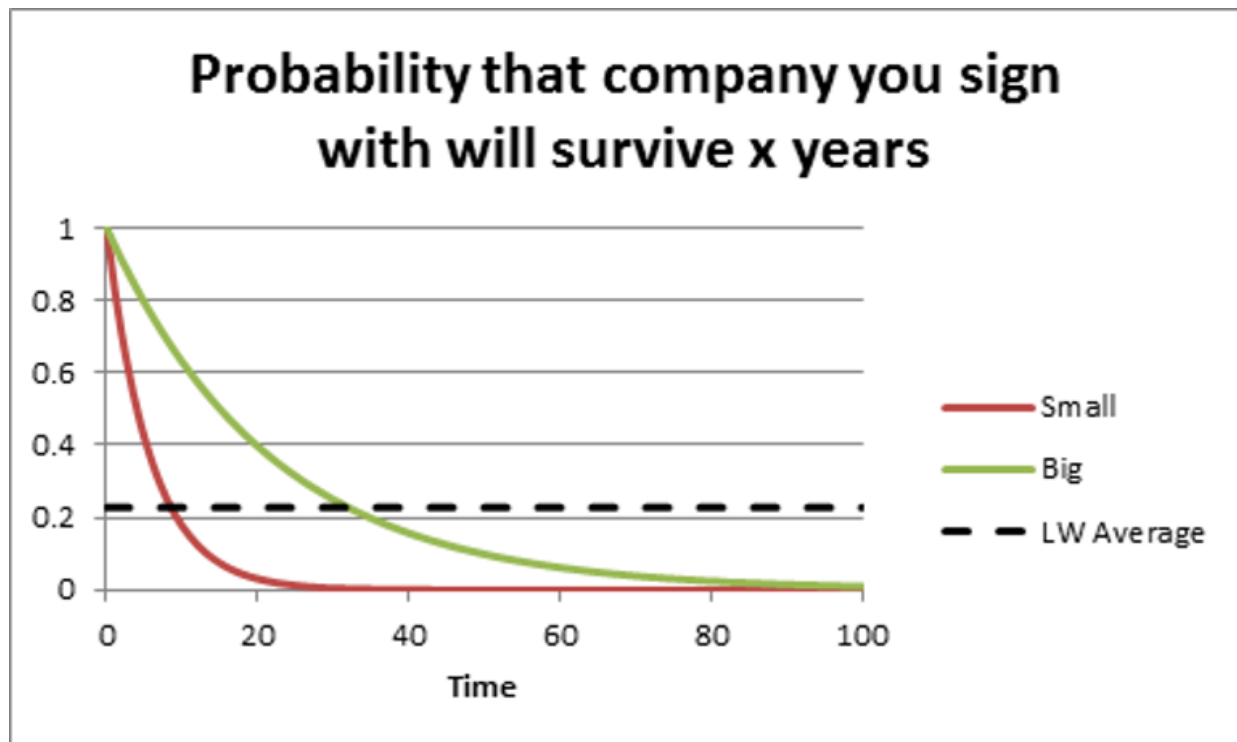
Even assuming cryonics firms are as well managed as S&P 500 companies, a 22.8% chance of success depends on every single other factor in the Drake equation being absolutely certain AND unfreezing technology being developed in 37 years.

But I noticed I was confused; Alcor has been around forty-ish years. Assuming it started life as a small company, the chance of that happening was one in ten thousand. That both Alcor AND The Cryonics Institute have been successfully freezing people for forty years seems literally beyond belief. I formed some possible hypotheses to explain this:

1. Many cryo firms have been set up, and I only know about the successes (a kind of anthropic argument)
2. Cryonics firms are unusually well-managed
3. The data from one or both of my sources was wrong
4. Modelling an average life expectancy as a half-life was wrong
5. Some extremely unlikely event that is still more likely than the one in billion chance my model predicts – for example the BBC article is an April Fool's joke that I don't understand.

I'm pretty sure I can rule out 1; if many cryo firms were set up I'd expect to see four lasting twenty years and eight lasting ten years, but in fact we see one lasting about five years and two lasting indefinitely. We can also probably rule out 2; if cryo firms were demonstrably better managed than S&P 500 companies, the CEO of Alcor could go and run Microsoft and use the pay differential to support cryo research (if he was feeling altruistic). Since I can't do anything about 5, I decided to focus my analysis on 3 and 4. In fact, I think 3 and 4 are both correct explanations; my source for the S&P 500 companies counted dropping out of the S&P 500 as a company 'death', when in fact you might drop out because you got taken over, because your industry became less important (but kept existing) or because other companies overtook you – your

company can't do anything about Facebook or Apple displacing them from the S&P 500, but Facebook and Apple don't make you any more likely to fail. Additionally, modelling as a half-life must have been flawed; a company that has survived one hundred years and a company that has survived one year are not equally likely to collapse!

Consequently I searched Google Scholar for a proper academic source. I found one, but I should introduce the following caveats:

1. It is UK data, so may not be comparable to the US (my understanding is that the US is a lot more forgiving of a business going bankrupt, so the UK businesses may liquidate slightly less frequently).
2. It uses data from 1980. As well as being old data, there are specific reasons to believe that this time period overestimates the true survival of companies. For example, the mid-1980's was an economic boom in the UK and 1980-1985 misses both major UK financial crashes of modern times (Black Wednesday and the Sub-Prime Crash). If the BBC is to be believed, the trend has been for companies to go bankrupt more and more frequently since the 1920's.

I found it really shocking that this question was not better studied. Anyway, the key table that informed my model was this one, which unfortunately seems to break the website when I try to embed it. The source is Dunne, Paul, and Alan Hughes. "Age, size, growth and survival: UK companies in the 1980s." *The Journal of Industrial Economics* (1994): 115-140.

You see on the left the size of the company in 1980 (£1 in 1980 is worth about £2.5 now). On the top is the size of the company in 1985, with additional columns for 'taken over', 'bankrupt' or 'other'. Even though a takeover might signal the end of a particular product line within a company, I have only counted bankruptcies as representing a threat to a frozen body; it is unlikely Alcor will be bought out by anyone unless they have an interest in cryonics.

The model is a Discrete Time Markov Chain analysis in five-year increments. What this means is that I start my hypothetical cryonics company at <£1m and then allow it to either grow or go bankrupt at the rate indicated in the article. After the first period I look at the new size of the company and allow it to grow, shrink or go bankrupt in accordance with the *new* probabilities. The only slightly confusing decision was what to do with takeovers. In the end I decided to ignore takeovers completely, and redistribute the probability mass they represented to all other survival scenarios.

The results are astonishingly different:

Markov-Estimated Survival Probability

(http://imgur.com/CkQirYD.jpg)
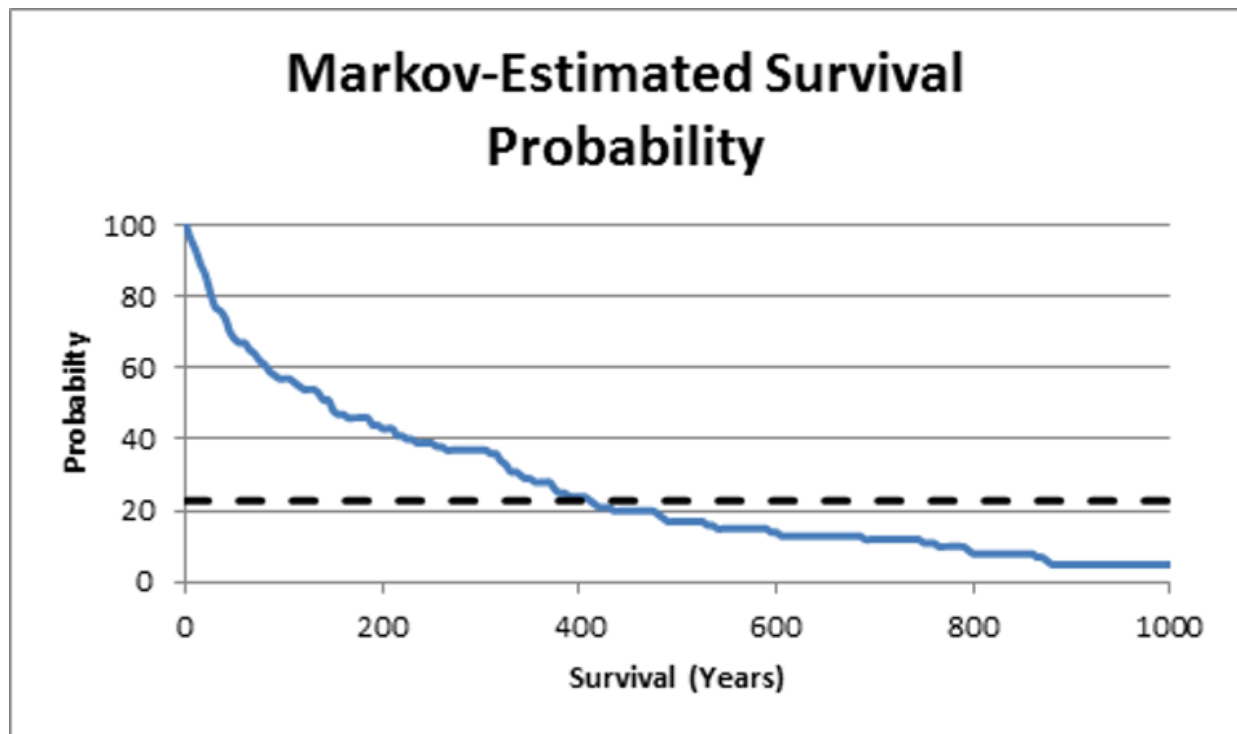
Now your body can remain alive 415 years for a 22.8% chance of revival (assuming all other probabilities are certain). Perhaps more usefully, if you estimate the year you expect revival to occur you can read across the x axis to find the probability that your cryo company will still exist by then. For example in the OvercomingBias link above, Hanson estimates that this will occur in 2090, meaning he should probably assign something like a 0.65 chance to the probability his cryo company is still around.

Remember you don't actually need to estimate the actual year YOUR revival will occur, but only the first year the first successful revival proves that cryogenically frozen bodies are 'alive' in a meaningful sense and therefore recieve protection under the law in case your company goes bankrupt. In fact, you could instead estimate the year Congress passes a 'right to not-death' law which would protect your body in the event of a bankruptcy even before routine unfreezing, or the year when brain-state scanning becomes advanced enough that it doesn't matter what happens to your meatspace body because a copy of your brain exists on the internet.

My conclusion is that the survival of your cryonics firm is a lot more likely that the average person in the street thinks, but probably a lot less likely that *you* think if you are strongly into cryonics. This is probably not news to you; most of you will be aware of over-optimism bias, and have tried to correct for it. Hopefully these concrete numbers will be useful next time you consider the Cryo-Drake equation and the net present value of investing in cryonics.

Business Networking through LessWrong

Is anyone interested in contacting other people in the LessWrong community to find a job, employee, business partner, co-founder, adviser, or investor?

Connections like this develop inside ethnic and religious groups, as well as among university alums or members of a fraternity. I think that LessWrong can provide the same value.

For example, LessWrong must have plenty of skilled software developers in dull jobs, who would love to work with smart, agenty rationalists. Likewise, there must be some company founders or managers who are having a very hard time finding good software developers.

A shared commitment to instrumental and epistemic rationality should be a good starting point, not to mentioned a shared memeplex to help break the ice. (Paperclips! MoR!)

Besides being fun, working together with other rationalists could be a good business move.

As a side-benefit, it also has good potential to raise the sanity waterline and help further develop new rationality skills, both personally and as a community.

Naturally, such a connection is not guaranteed to produce results. But it's hard to find the right people to work with: Why not try this route? And although you can cold-contact someone you've seen online, you don't know who's interested in what you have to offer, so I think more effort is needed to bootstrap such networking.

I'd like to gauge interest. (Alexandros has volunteered to help.) If you might be interested in this sort of networking, please fill out this short Google Form [Edit: Survey closed as of April 15]. I'll post an update about what sort of response we get.

Privacy: Although the main purpose of this form is to gauge interest, and other details may be needed to form good connections, the info might be enough to get some contacts going. So, we might use this information to personally connect people. We won't share the info or build any online group with it. If we get a lot of interest we may later create some sort of online mechanism, but we'll be sure to get your permission before adding you.

Edit April 6: We're still seeing that people are filling out the form, so we'll wait a week or two, and report on results.

Edit April 15: See a summary of the results below.