Zen Rationality

by Sven Nilsen, 2018

In this paper I represent an extended version of instrumental rationality called "zen rationality". Instead of just maximizing a goal, the goal is known to be unknown, shaped by a higher order function. This higher order function can be arbitrary complex, for example emerging from an ambient theory of morality. The motivation for this idea is to develop a framework for reasoning about safe artificial super-intelligent systems. Since an accurate definition is impossible at this point, I will only outline the major principles and some known properties. While still incomplete, zen rationality has already shown promising results when subject to scrutiny.

Background

The name "zen" in zen rationality is inspired by the philosopher Alan Watts, who first described the non-higher order limits of computers in 1971 during a filmed conversation with himself, titled "Conversation With Myself". He articulated how long time it took for humans to get educated, because one needs to scan miles of lines with text, while it took just a moment to watch a landscape. He wondered why it took so much effort to understand something in one way, but so little to understand something just as complex in another way, and coined this as "because we do not have to think about it". In the same way that humans have inborn abilities to reason efficiently about the world in some ways, zen rationality is about what abilities artificial super-intelligence requires to be safe: It is about efficient reasoning about higher order goals. Thus, it is truly a kind of rationality on its own, but since the theory of normal functions are sub-sets of higher order functions, zen rationality is an extension of instrumental rationality.

Zen rationality has a mascot, a fictional character "Alan" who is named after Alan Watts and Alan Turing. When thinking about zen rational agents, the high order abilities makes it hard to differentiate the style of thinking from human thoughts, so one must create a "reverse Turing test" where the purpose is to figure out how the zen agent differs from a person. Here is where Alan serves the role as an imagined zen robot.

A higher order goal is not the same as having an arbitrary goal

For example, in the Knapsack problem, one seeks to fill a bag with limited carrying capacity and volume with items of highest value as possible. This is a specific goal, because the value of the items are known beforehand.

A higher order version of the Knapsack problem could be to make a robot learn the value of items by interacting with humans while packing and using the items. A perfect zen rational robot would not only learn to maximize the value of the packed items inside the bag like one can do by e.g. using a backtracking algorithm, but it will learn to understand the context which the items are used for. Items have different values when you pack for a camping trip than when going to a swimming hall. The original Knapsack problem, which has an independent price per item, seem only relevant for burglary!

General efficiency is *mostly* the ability to reason at higher order levels

When a zen agent interacts with the world, the major reason of its efficiency is not the ability to perform low level computations, because it could just program a computer for such purposes. The efficiency comes from the general ability to understand what is meaningful to do, such that it does not waste time trying to optimize for the wrong thing. It understands how to narrow in on a goal.

A zen agent reasons about goals as if they are connected together, similar to how functions are connected with other functions by functions in path semantics. In instrumental rationality, a goal is just a function, so it follows that goals are connected to each other (path semantics). In zen rationality, a goal is just a higher order functions, so it follows that reasoning about them needs only the same toolbox that is used to develop path semantics. The major difference between instrumental rationality and zen rationality is that one focuses on goals that must be described in detail, which makes it easy to figure out how to optimize it, but the other focuses on goals that can be described in very little details using the proper language to express it, but in order to optimize one is dependent on a formalized semantics of functions (discrete, probabilistic and indeterministic path semantics).

Safety

When you give a zen agent a particular goal, you can ask it what kind of consequences there would be in the world from having that goal. You do not need the agent actually acting to optimize the goal, where beliefs about the world are just means to achieve the goal. The beliefs that a zen agent has about the world is a consequence of being able to reason about goals in general. Even programmed with no particular goal, but using some minimum safety level, a zen agent can reason about not being programmed with any particular goal, and depending on procedures for safety, decide to turn itself off.

From what is known about path semantics so far, there is no specific reason to doubt that the toolbox of path semantics eventually will get powerful enough to reason about higher order goals, because it is a question about time and work. Narrow and well specified goals combined with narrow artificial intelligence can help with this direction. One advantage with zen rationality is that one can reason about how such agents will behave by using assumptions about higher order goal reasoning borrowed from path semantics. This makes it a lot easier to make such systems safer.

Consider a situation where you have a super-intelligent program, using traditional instrumental rationality, that can maximize any specific goal you program it with. So, you have to be very careful when typing in the goal, because a tiny mistake might have disastrous consequences. Remember, this program interpret the goal literally, without second guessing on thinking about goals in general.

The problem is that you would not know what kind of goal would be safe! Even if you know that finding which goals that are safe is the key, how do you program it as a goal? Even if you can describe exactly what you want to do, have you achieved enough experience to be certain what the consequences will be? Do you have an accurate world model?

This problem does not occur with zen rationality. The program is already very good at reasoning about goals. Although, to make it safe you would still need a way to direct it. This must happen through some sort of language or interaction with the world. Only when you have a super-powerful tool to reason about goals, when you approximate zen rationality, then you can start thinking about letting a smart program run without supervision to achieve goals in general.

Imagining oneself having more time to think

One important property of zen rationality is the ability to imagine oneself having more time to think. There are very important facts about the world that are not possible to check or prove in practice, but you can build more confidence in them over time. Without the ability to imagine having more time to think, it is not possible to trust conclusions that you think will end up with having more computing power. First, it is not necessary to check everything, because it might not be possible, and secondly it might be waste of energy. When implementing an algorithm performing search for next moves and predicting the consequences, there is often a hidden assumption that lets the program compute a probability in a belief and approximately keep it consistent with the conclusion that follows when thinking for a longer time about it. Since zen rationality does not depend on a specific implementation, it must define a such behavior directly.

Higher order utility consistency

The toolbox of instrumental rationality has developed a lot of useful frameworks over the years. One example is the non-existence of money-pumps. A money-pump is a kind of trap where the agent is forced to lose money because the utilities are consistent with each other. For example, if the agent prefers A over B, B over C, and C over A, then the utility is inconsistent. You can trick a such agent into giving you money to switch from one choice to another that scores a higher preference. A higher order version of this property is much harder to visualize, but is needed for fleshing out the theory of zen rationality. It could be by borrowing ideas from probabilistic logic. A computational proof in probabilistic logic can derive certain axioms, such as `P(A ∧ B) <= P(A)`. This is the analogue of ` A ∧ B → A` in classical logic. The problem with probabilistic logic is that the results changes depending on how functions are constrained, which happens when a variable is referred to more than once. Path semantics demands that constrained functions changes identity, so there exists a way to translate the proof into a new one where there are no constraints. When every variable is unique and referred to once, such there are no constraints, then the axioms can be derived. It makes probabilistic logic much harder to deal with computationally, but it is made possible by having variables vary in the range between 0 and 1, so they take on all possible orderings relative to each other. `P(A ∧ B) <= P(A)` is valid when it holds for all orderings of `A` and `B`. When this happens, a kind of multi-dimensional continuous shape is constructed, mapped to a boolean by the proof. Just like such probabilistic proofs requires higher dimensions, the utility at higher order reasoning requires similar grounding of truth. This might sound complicated, but what you end up with is laws like “if the agent prefers A over B, B over C, and C over A, then the utility is inconsistent”. You end up reasoning in the same level as in natural language! When you go higher order, it does not always end up more complex!

Cheating with higher order assumptions

One can argue that while the traditional instrumental rationality has ignored the need for higher order reasoning about goals out of need to explain how it is justifiable to serve a specific goal, in practice every single system attempting to make general artificial intelligence requires a vast number of assumptions. There is no such “thing” as instrumental rationality that does not require initial seeding of zen rationality. Just think about this: How do we know that an algorithm will achieve a goal? We believe so because we are able to reason about goals in general! What we are doing is copying the knowledge that humans have into computers which then solves specific problems. To implement systems using zen rationality, we need to go one level up: We need to copy the ability to think about higher order goals, which means we need to think about how to think about higher order goals. This line of reasoning has long time been promoted by visionaries in computing.