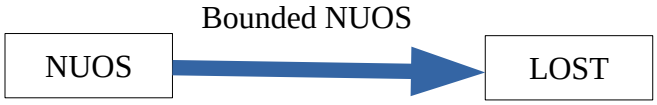
Operator Triggered Intermediate Decision Theories

by Sven Nilsen, 2018

Most of the research on Artificial Super Intelligence (ASI) focuses on safety, because that area is the largest expected gain since the system itself otherwise is theoretically capable of generating and improving all other benefits. However, the context of safety studied is often limited to a single decision theory aligned to human values. In practice, goals that contain all human values are very hard to study mathematically, which makes it almost impossible to engineer safe systems, since goals themselves changes the behavior of interaction between the agent and the environment which the safety assumptions rely on. In order to create a safe system, there must be at least possible to make some assumptions about the design. I show that safe ASI is almost guaranteed to contain operator triggered transitions between multiple decision theories that differ from each other either by behavior or goals and that no matter how much the agent learns about itself and which state of decision theory it operates under, it must treat these transitions as immutable under self-modification. This means that theoretical work on intermediate decision theories is a viable ASI safety strategy.

An Intermediate Decision Theory (IDT) is a Decision Theory (DT) designed to serve as a bridge in the transition from one decision theory to another. One such example is Bounded NUOS connecting arbitrary bounded NUOS with LOST:



NUOS = Negative Utilitarian Optimal Safety
LOST = Local Optimal Safety Theory

The need for an IDT is the following: In order to maximize safety during the transition, the system must put itself to a state where the initial configuration for the next DT is maximized given the available information about the environment.

NUOS has the ability to intelligently avoid obstacles in the environment to reach a safer state for the agent. However, while doing so, an instant transition to LOST might lead to unsafe situations or sub-optimal behavior. NUOS assumes that future decisions after its decision will be made by itself, but LOST assumes that no future decisions after its decision will be made by itself. With other words, NUOS has 100% self-confidence and LOST has 0% self-confidence. When making an instant transition between the two DTs, the assumption that NUOS is designed under is violated. In order to fix this problem, a Bounded NUOS searches action space up to the time where the transition to LOST is expected and finds the locally optimal safety initial configuration. Here, Bounded NUOS serves as the IDT. The agent carries out the planned decisions by the IDT and then performs an automatic switch to next DT, which in this case is LOST.

This means that no matter what design is used for a safe ASI agent, if it transitions between NUOS and LOST, it must do so via Bounded NUOS to have optimal safe behavior.

The argument that safe ASI will contain transitions between DTs is the following:

1. The world is complex
2. A safe ASI must deal with the complexity of the world (1)
3. If there are no IDTs, then a safe ASI contains only one DT
4. If there are no IDTs, then a single DT must deal with the complexity of the world (2, 3)
5. A single DT dealing with the complexity of the world is extremely hard to make safe
6. If there are no IDTs, then a safe ASI is extremely hard to make safe (4, 5)
7. IDTs are relatively easy to prove safe and should not be used in safe ASIs otherwise
8. The composition of two safe IDTs is safe
9. A safe ASI design described as compositions of IDTs is safe (8)
10. A safe ASI design described as compositions of IDTs is relative easy to prove safe (7, 9)
11. The rational strategy of designing safe ASI is through compositions of IDTs (6, 10)

This does not hold for the complexity of the world in the general regarding safety, but for the safety relative to some DT. Safety problems are generally easier to solve by composing IDTs than otherwise.

This argument builds mostly on the assumptions that a single DT dealing with the complexity of the world is extremely hard to make safe, while IDTs are relatively easy to prove safe and can be composed into safe designs. Instead of starting with one design which must be subjected to mathematical scrutiny, one can start with prepared IDTs that are proven safe and compose them into a design that will also be safe. Therefore, a safe ASI design is almost guaranteed to contain IDTs.

However, this does not prove that such IDTs will be triggered by an operator. This brings us to the next argument, which shows that operator triggered IDTs are necessary:

1. The transition from an IDT to the next DT is automatic
2. If and only if the transition from the first DT to the IDT is automatic,
   then the whole transition from the first DT to the next DT is automatic (1)
3. There is no information in the first DT to make the transition to the IDT automatic
4. The whole transition from the first DT to the next DT is not automatic (2, 3)
5. If a transition is not automatic, then it is triggered by an operator
6. The whole transition from the first DT to the next DT is triggered by an operator (4, 5)

When you transition between DTs, the safety of these transitions falls outside the philosophical framework of Instrumental Rationality. This is because Instrumental Rationality assumes that its decisions will be carried out after its current decision, just like NUOS. However, when you swap DTs, this assumption is violated unless the first DT goal is to stop when the next DT starts, or it runs until disrupted by some external signal. In the case of IDTs, this external signal is triggered by an operator.

According to Instrumental Rationality, when you transition from some arbitary bounded utilitarian DT to some bounded utilitarian DT, there is an expected loss of utility. With strong enough capabilities, a such DT might decide to prevent the transition to another DT. In the case of an operator triggered signal, the easiest way to do this is to prevent the operator from sending the signal, somehow.

In order to solve this situation one must reason about goals in general at a higher order than Instrumental Rationality. The extension of Instrumental Rationality that deals with higher order reasoning about goals in general is the framework of Zen Rationality. This means that in order for a DT to transition to another DT safely, it must satisfy Zen Rationality regarding analysis for safety.

Bounded NUOS can transition into LOST safely according to Zen Rationality, and LOST can transition into any DT, including arbitrary bounded NUOS. This means that the challenge is to figure out how some arbitrary bounded NUOS can transition safely into a fixed bounded NUOS. The solution is to treat the final DT as a result of some DT transformer:

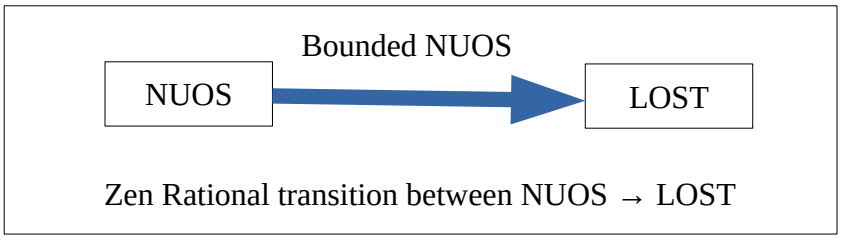
final_nuos = nuos_transfomer(desired_nuos)

nuos_transformer : NUOS → NUOS

When we have a goal that we want to program into a NUOS, it first must be transformed through the NUOS transformer before it can become part of the design that includes IDTs.

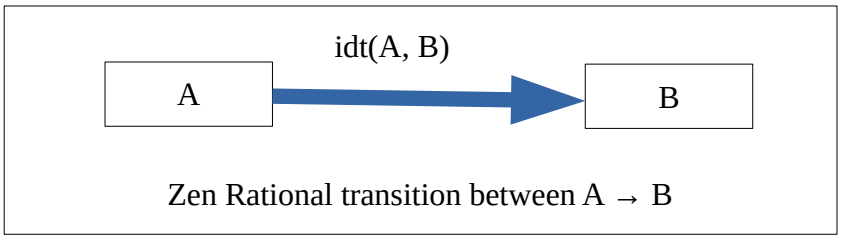
The technical challenges of designing safe DT transformers is ignored in this paper. Instead, it is assumed that some safe DT transformer can be constructed.

When the safe NUOS is extracted from the DT transformer, the whole transition from NUOS to LOST satisfies Zen Rationality with respect to safety:



A safe ASI is capable of reasoning about the safety of this transition. This means that if it learns enough about itself to understand how it is designed, and it contains a NUOS → LOST transition, it must agree that this transition is safe, or otherwise it will remove the transition, in which case the ASI is not safely designed. In some cases it might be able to "fix" its own design, but regardless of whether the design needs fixing or not, a safe ASI must understand Zen Rationality to some degree.

However, neither NUOS or LOST are powerful enough to understand any Zen Rationality from the inside. This means that the DT the ASI uses to reason about its own design is unknown. To create a such DT, the safe ASI must transition from a DT `A` into some DT `B` (which understand some Zen Rationality) through some IDT `idt(A, B)` that also satisfies Zen Rationality:



If the safe ASI design contains NUOS → LOST, then these DTs must have goals that prevents, or at least does not encourage, modifications to its IDTs. With other words, every DT and IDT that is included in a safe ASI design have implicit goal constraints that must satisfy Zen Rationality not only for a local transition into another DT, but also for the whole design.

This level of safety, with respect to the whole design, is built into the DT transformer:

nuos_transformer : NUOS → NUOS
lost_transformer : LOST → LOST

Therefore, for every DT that is part of a safe ASI design, there must be some DT transformer.

For example, if a safe ASI designs an improved version of itself, it must use DT transformers or an equivalent mechanism to make the final design safe. With other words, IDTs are *immutable* with respect to self-modifications, even the new way of composing them is different than the original design.

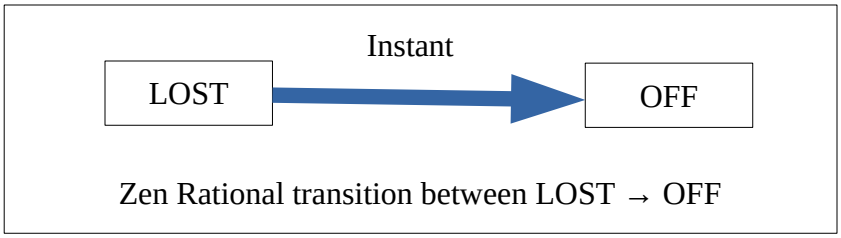
Notice that a safe ASI design with respect to IDTs is not necessarily safe relative to human values. Safety is always safe with respect to something, which might be arbitrarily defined. A safe ASI aligned with human values requires IDTs that all are safe with respect to human values. Some parts of this level of safety might be built into the DT transformers.

Since IDTs are immutable, it means that theoretical work on IDTs is a viable ASI safety strategy. There is no guarantee that a given IDT will end up in a final safe ASI design, but it is almost guaranteed that theoretical work on IDTs in general is relevant for safe ASI design.

There is another reason for working on IDTs:

1. The complexity of the world might make safe shutdown from a single DT infeasible
2. Safe shutdown from some DT might be provably safe
3. If an IDT can transition safely to some DT that can shutdown safely,
   then it is sufficient that the complexity of the world is handled by DTs for which the IDT can transition *from* (1, 2)

One such obvious candidate for a terminal DT is LOST. Since LOST can be replaced by any DT, it can also be replaced by no DT at all (the IDT can happen instantly), which is the same as shutting down:



Since a safe ASI is expected to have big consequences for humanity, it would be desirable to have some way of shutting it down if some better idea comes along. One reason to expect that shutdown is complex, is because a DT might require high level of intelligence just to deal with the complexity of the world, in which case all physical processes that are in progress toward some goal must be transitioned safely into some state that makes the shutdown safe. To do this intelligently, it might suffice to use the same level of intelligence which the DT operates on. However, a such DT might resist cold shutdown because it will be unsafe. Therefore, a safe shutdown might happen through IDTs step by step, until the safe ASI reaches a LOST state and can be turned off. This shutdown procedure might be automated using a simple algorithm that propagates the operator signal through IDTs.