PNF7 and Living Motion

'Engaging the world with transforms'

A universal tool

If we want to solve a problem, we should make a map of the processes in the outer world around that particular problem. To achieve this we can make use of an AI tool that can map/describe/engage the real world.

A problem is first defined in the causal diagram, but the connection with the real world/outer world happens in the 'scanner' tool as described in this document.
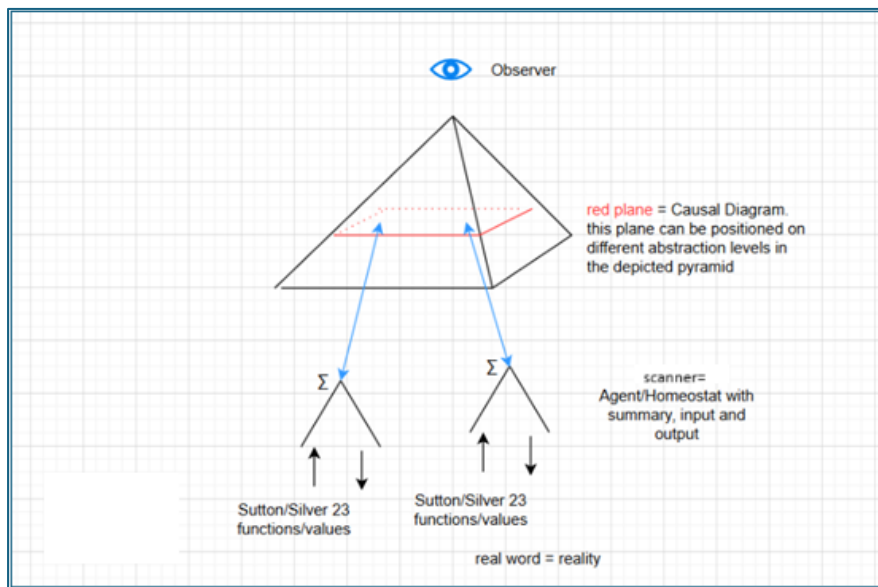

Figure 1. The relation between the causal diagram and the scanner

How does It work

The scanner scans the outer world (the real world) and it maps the local processes.

For this it uses transforms, because transforms are better suited to 'engage' with the processes that occur in the real world.

An example of a single transform is:

Effort -> price

For instance, a negotiation process transforms effort into a price variation.

The AI, for instance an LLM, can find the transforms that have the best fit with the local processes.

Transforms are constructed using a limited set of 'quantities' to define them. The set of quantities is small because we need to avoid the so-called combinatorial explosion. First the scanner has a belief about the outer world and it tests that by measuring the signals that return from the outside world. If the measurement differs from the belief, the tool has to adjust the belief.

The summary at the top of the scanner is connected to a factor in the causal diagram.
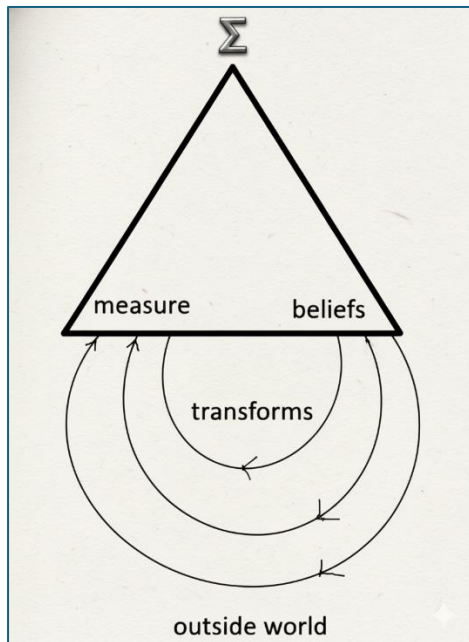

Figure 2. Scanner

The active transforms help to 'minimize surprise'. They probe the world.

Examples of different configurations

Situation/Problem in causal diagram	Number of factors in the causal diagram	Number of abstraction levels	Number of transforms per scanner/factor (each factor has a connected scanner)
Robotic arm	3	1	5
Emphatic Health Assistant	5	2	5
Grandmother-granddaughter	7	2	5

Story

The first step is to make a story about the model we want to describe. The nice thing about this approach is that it is universal and works in any language. After entering the story the tool immediately distills the top-k signal words. We can click on two of those top-k words to fill the goal/topic1 and the subject/object/topic2. The tool will use these fields to make a prompt.

Transforms

we can see the world as a set of transforms. And we want to check (measure) our belief of those transforms. An example of a transform is:

Effort -> price

For instance, a negotiation process transforms effort into a price variation.

Making software programs

Using the tool we have collected the goal, topic, subject/object and the transforms. The tool now has enough information to create the prompt for making software programs.

➔ Click on 'Make Prompt'

Dividing up the world in sectors

Because the world is so large, it helps to select a sector where the problem is located. This takes care that the created prompt is better defined.

We can select a sector for example with a rotary selector as shown in this picture:

Figure 3. Different sectors that divide up the world

The relationship between situations and problems

When is a situation a problem? This depends on the observer.

What is a problem for one person, does not have to be a problem for another person.

Abstraction levels

You can 'enter' any problem at different abstraction levels. An abstraction level can be selected in the tool before making the prompt.

Causality

We imagine that there is a causality diagram positioned above the scanner. (ptd: elaborate further)

Trying different paths; exploring the world

A living being continuously tries multiple paths. An AI will do the same. A node in such a path can be a thing or a process that is or will be tried out.

The AI will in this way explore the world, see the picture below:
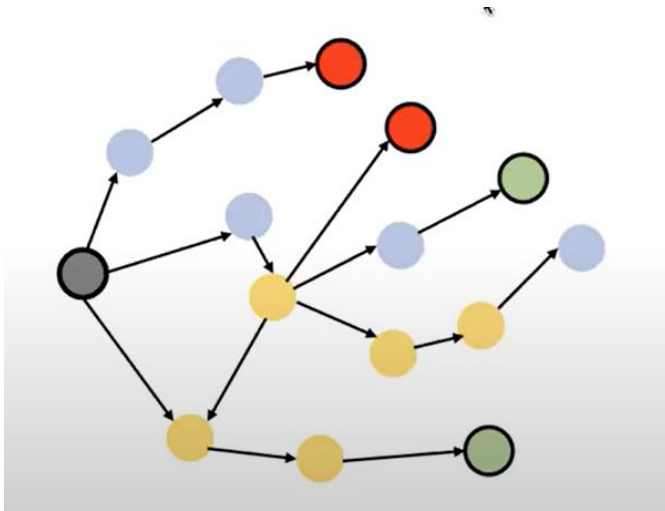
Figure 4. A graph that depicts possibilities for an AI or program

A complex program also has this structure: the different tries or possibilities or abstractions.

Constraints

A constraint ensures that when -for instance- a robot is modeled, the robot does not fall over because of instability. So 'being stabile' is the constraint. Ptd: this is a common-sense rule: objects must be stable. Common-sense rules can also be stored in a scanner. The common sense rules/transforms can be tested in the outer world, this is the same transform-testing mechanism.

Convergence

Convergence is applicable to any situation: robot gripper, robot arm, health bot.

A homeostat-like device has convergence.

'Minimize surprise' is the slogan for homeostat-like devices.

More examples of transforms

Repair, redo etc (from paper1) are also popular transforms! Redo = Effort -> I, repair = effort-> I

Redo/Repair is time travel-like?

AI will become stronger in the near future

And thus it will be possible that the AI writes the python code that the prompts specify.

Reinforcement learning (RL)

Continuously the AI tries to check its beliefs by measuring in the real world. If the belief is correct, it will reinforce it. This is called reinforcement learning (RL).

Uncertainty

We can encode uncertainty in the belief (see figure1). If the belief corresponds with the measurement, the uncertainty is decreased.

World model

We are building a world model of the real world using the transforms. In other words: because we engage with the world using the transforms, we build the world model.

Examples

The robot hand

A robot hand must converge (='minimize surprise'), have RL, causality, constraints, a goal and a subject/object.

Transforms 'engage' with the processes that take place in the outer world for the robot hand.

A robot hand can transform:

effort-> location.

effort-> orderedness (entropy)

A robot hand can repair an object: effort-> X,I,P

The empathic health assistant

The empathic health assistant must converge (='minimize surprise'), have RL, causality, constraints, a goal and a subject/object. Transforms 'engage' with the processes that take place in the outer world for the emphatic health assistant.

A health assistant can transform:

Transform: effort -> replace route (to 'zorgwinkel')

Transform: effort -> replace route (to 'apotheek')

Transform: effort -> orderedness

Idea: the commonality between all these examples is that transforms 'engage' with processes in the real world. Constraints are also transforms that 'engage' with processes in the real world!

Enforce or measure transforms?

This is an interesting nuance: when the AI wants to change the world , it will try to influence it using transforms.

If the AI only wants to get information (be informed), it will 'measure' or 'sense'. It is the distinction between influencing and measuring.