KOMEGA REQUIREMENTS: Start with a Political Program Version V3

*

Gerd Doeben-Henisch (gerd@doeben-henisch)
triggered by a case-discussion with
Philipp Westermeier
and
Rudolf Bauer

November 23-25, 2020

Abstract

Applying the original P-V-Pref Document structure to real cases it became clear that the everyday logic behind the classification of facts into *problems [P]* or *visions [V]* follows a kind of logic hidden in the semantic space of the used expressions. This text explains this hidden logic and what this means for our application.

1 Actors, Expression, and the Real World

To be able to talk about *problems* P and *visions* V one has to clarify the context of talking.

Actors, Real World, Expressions: In this text it is assumed that there are some *actors* A living in some part RS of the *real world* RW, $RS \subseteq RW$ and these actors are using an *everyday language* L realized by *expressions* E related to this language L.

Meaning: These expressions E are with the aid of some *internal meaning function* μ related to some internal states called *concepts* C, written as $\mu : E \longleftrightarrow C$. Cognitive concepts C are representing *cognitive facts* F^C.

Truth: Some of these internal cognitive facts F^C are related to some assumed *real facts* F being part of the real world $F \subseteq RW$, some of the internal cognitive facts F^C are not related to real facts F. To be able to decide whether cognitive facts F^C are actually belonging to real facts or not a *cognitive truth function* τ^C is here assumed which does this work: $\tau^C : F^C \times \kappa \longmapsto F^{C,r} \times F^{C,p}$. The factor κ denotes some internal cognitive criterion which enables such a decision, and the expressions $F^{C,r}, F^{C,p}$ denote those cognitive facts which are assumed to be associated with real facts F or not. The *possible cognitive facts* $F^{C,p}$ can internally be classified to become real cognitive facts $F^{C,r}$ in some *future* associated with some *probability* π that this could happen.

Classified Meaning: While the normal (internal) meaning function μ can map expressions E of an everyday language L into cognitive concepts C which are related to cognitive facts F^C, can the internal cognitive truth function τ decide which of these cognitive facts F^C are actually related to real facts F and which are not. In this text a meaning which is decided as being associated with real facts or not is called a *classified meaning*: $\tau(\mu(e)) \in \{True, False\}$ with $e \in E$ tells us that an expression e receives a meaning by the meaning function mu identifying some concepts C with their cognitive facts F^C and these are classified by the truth function τ whether these cognitive facts are actually associated with real facts F or not.

Preferences: In our application scenario we can observe so-called *preferences* manifested in the behavior of actors. It is assumed that every single actor $\alpha \in A$ holds some *Preferences* $Pref$, which are understood as *pairs of expressions* (v, r) where the expression v is representing *cognitive facts* which are not actually associated with real facts F – written as: $\tau(\mu(v) \subseteq F^{C,p}) = False$ – and expressions r representing *cognitive facts* which are actually associated with real facts – written as: $\tau(\mu(r) \subseteq F^{C,r}) = True$ – . This can be written as follows:

$$Pref_{\alpha \in A} = \{(v, r) | \tau(\mu(v) \subseteq F^{C,p}) = False \,\&\, \tau(\mu(r) \subseteq F^{C,r}) = True \,\&\, v >_\alpha r\}$$

Preferences *located* in some *real actor* $\alpha \in A$ as a pair (v, r) enable in the actor a kind of *ranking* stating that the expressions v should be higher rated $>_\alpha$ than the expressions r. And because these expressions are further related to cognitive concepts and cognitive facts this enables an individual actor α to show an observable habit *prefering* the *substitution* of the real cognitive facts $F^{C,r}$ by the *possible, but not yet real* cognitive facts $F^{C,p}$.

Example 1: Assume there is an actor α being a citizen of the city Frankfurt in Hessen, Germany. If he is looking to the traffic in Frankfurt he can easily observe that the expression e1 = *The traffic in Frankfurt is dense* is true. And there is some chance that this citizen can imagine a possible state expressed by e2 = *The traffic in Frankfurt is reduced by 50%*. And it is conceivable that this citizen generates a preference like this v=$\{$e2$\}$ $>_\alpha$ r=$\{$e1$\}$. Having done this the fact represented by the expression e1 can be declared a *problem* P based on his *vision* V represented by the expression e2.

2 Political Citizens

Equipped with the before introduced concepts one can outline a model of *political communication* between citizens which eventually can enhance the *rationality* of acting inspired by thinking.

A Given Situation: At every point of time real citizens – here considered as real actors – are always embedded in some real part RS of the city RW. Usually they are able to describe properties of their real environment with some expressions E of the everyday language L they are using. Let us call such a collection of finitely many expressions a *state description* or simply a *state* S.

Generate a Preference: It is further conceivable that each of these citizens is able to imagine some ideas in his head which can be communicated with expressions E^V_α telling that these expressions describe for him/ her/ x some ideas which are *not yet real*, which are not yet part of the state S – which one can call a *vision* V –, and saying this he will usually point to those expressions r which are part of the actual state S, which he/ she/ x thinks have to be *replaced* by this vision. In that moment it is clear that this citizen α has established a *preference* (V,r) which eventually will guide his decisions and thereby his behavior. The important point is that an expression as such is not a *problem* P, but such an expression can *become a problem* in the presence of other expressions, which are not yet real, but which possible could become real if an actor would establish an appropriate preference in his head which can *guide* the actor to try to replace the old expressions of the state description S with the new ones.

A Political Program: If citizens will start to write down sets of preferences $VP = \{(v,r)_1, (v,r)_2, ..., (v,r)_k\}$ we can understand this as a first rough outline of a *political program*. A program can shed some light what should be changed and into which direction it should be change. But a program usually does not yet tell you *how* this change can happen in detail.

Planning Change: Seriously minded citizens will not be satisfied to have only a first political program; they want that this program *will become real*. To enable this one has to develop a *plan* how one possibly can proceed from some *given real situation*, the *initial state* S_{start}, to some *envisioned state in the future*, the goal state S_{goal}. Basically this means that one constructs a finite *chain of states* or *sequence of states* $\langle S_{start}, S_{start+1}, ..., S_{goal} \rangle$, where the *transition* from one state S_i to the follow-up state S_{i+1} is described by a set of *change rules* X.

3 The komega-SW

To do real planning based on a political program is'nt really easy. But with the aid of an appropriate *software [SW]* this can be achieved much easier. For this together with a really inspiring team I am developing not only a theory (this actual text is part of this theory) but also a software with the working label *komega-SW*. What can this komega-SW do for you as a political minded citizen?

Install Your Political Program: (See for this figure 1) As you can imagine from the preceding paragraphs you must start with some friends to define a political program consisting of a set of preferences. A single preference is a pair (v,r) where v and r each represents a set of expressions. Let us call the r-expressions the *real* part of a preference and the v-expressions the *vision* part. If you have more than one preference like $VP = \{(v,r)_1, (v,r)_2, ..., (v,r)_k\}$ then all the real parts together $S_{kernel} = \{r | r \in (v,r)\}$ built up a *kernel situation* S_{kernel} which should be part of the *start situation* of your change process. Usually the start situation S_{start} is much larger than the kernel situation – $S_{kernel} \subseteq S_{start}$ – because you will usually not change *everything* but only some *identified problems*.

If you proceed in this way then, at the end of some change process, you will reach a state where you can analyze, whether this actual final state is a *goal state* G or not. Thus You will need some kind of an *evaluation* $EVAL$. A very basic version of an evaluation goes like this (cf. figure 1): You count the number of expressions which have been classified in the presence of a vision as *problem statement* $\sum p_i$ and which are still there, and you count all expressions which have been classified as *vision statements* $\sum v_i$. If the difference $\sum v_i - \sum p_i > \theta$ is bigger than some threshold θ than the actual end state can be declared a goal state; otherwise you can continue the change process, if there is still some hope to reach the goal in the future.

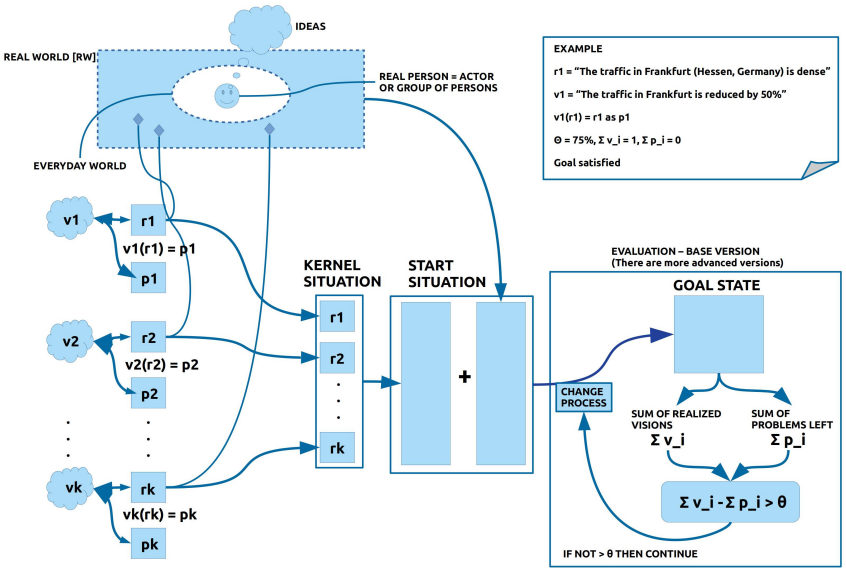

Figure 1: Generate a political program and how to evaluate this at the end

Generate a Change Plan: Thus having written down a political program in the format of a vision-problem set VP you have to extend your kernel state S_{kernel} to a *start state* S_{start}. If your political program is an ambitious program then it can make sense that you generate in the beginning several kernel states $\{S_{kernel,1}, S_{kernel,2}, ..., S_{kernel,k}\}$ and extend each kernel state to a separate start state getting a set of different start states $\{S_{start,1}, S_{start,2}, ..., S_{start,m}\}$. The komega-SW allows the citizens to develop as many as necessary start states in parallel. On demand it is possible to unify these different start states again. The next step is to generate for each start state a set of change rules $\{X_1, X_2, ..., X_m\}$. As in the case of the start states it is possible to unify sets of change rules on demand. Thus if you want to investigate how a unified start state $S_{start,i} \cup S_{start,s}$ will develop with the unified set of change rules $X_i \cup X_s$ then you can do this.

Continuation: This text describes only some part of the komega-SW. If You want to know more have a look to the komega-SW homepage at uffmm.org.