

Analysis of multi-actor policy contexts using perception graphs

Pieter W.G. Bots

Cemagref, Groupement Montpellier / Delft University of Technology

pieter.bots@cemagref.fr

Abstract

Policy making is a multi-actor process: it involves a variety of actors, each trying to further their own interests. How these actors decide and act largely depends on the way they perceive the policy problem. This paper describes Dynamic Actor Network Analysis (DANA), a graph-based method/tool to analyze a policy context by modeling how actors view a policy issue. Each actor view is modeled as a perception graph, a type of causal map that represents the (probabilistic) relations between goals, policy actions and external influences. Cross-comparison of these perception graphs reveals properties of the multi-actor policy network, such as factor relevance, resource dependency, conflict, and possible tradeoffs. Although DANA models technically have the potential for simulating policy scenarios, some interesting methodological problems remain.

1. Introduction

Assuming that policy decisions ‘happen’ in a network of actors [7, 8, 17], the key to a full understanding of a policy problem lies in finding out which actors are involved, what interests they have, and what actions they might take to achieve their objectives, to maximize their influence on the policy process, and/or to avoid becoming overly dependent from other actors in the policy arena. The purpose of the method proposed in this paper – Dynamic Actor Network Analysis (DANA) – is to help a policy analyst gain such understanding, allowing him/her to identify conflicts, potential coalitions, and opportunities for successful coordination when actors have conflicting preferences but have similar cognitive representations of the decision context [22].

The actor network view assumes that the situations by which actors are influenced and to which they adapt themselves do not stem from a single ‘objective’ world, but from their own subjectively perceived world. A DANA model therefore comprises a set of perception graphs: representations of the problem perceptions of different actors in the form of causal maps from which the actors’ preferred strategies can be

inferred. Cross-comparison of perception graphs can reveal similarities in perceptions, while confrontation of strategies can reveal conflicts as well as opportunities for their resolution.

This paper focuses on the technical aspects of DANA. In the next section, the choice of type of causal map is motivated. Section 3 specifies the formal representation of these maps, Section 4 the causal inference mechanism, Section 5 strategy evaluation and selection, and Section 6 some of the more interesting analyses that are presently supported by DANA.

2. A choice of causal maps

Policy analysis [9, 21] typically involves some form of causal analysis. When asked to advise a client on a particular issue, a policy analyst will identify the client’s objectives as factors the client would like to see changed (e.g., more employment, better public health, lower emissions) as well as those that should not change (e.g., expenditure). The analyst then looks for factors that by some causal chain influence these ‘outcomes of interest’. For each factor, the analyst looks for actors (the client or others) who take an interest in this factor, or have the capability of changing it through some action. The result of this search is a system diagram as depicted in Figure 1. The dotted line symbolizes the system’s boundary, defined by the outcomes of interest, possible ‘tactics’ (single actions aimed at changing one outcome of interest, a ‘strategy’ being a combination of tactics [25]), and ‘external influences’ (factors that may change autonomously).

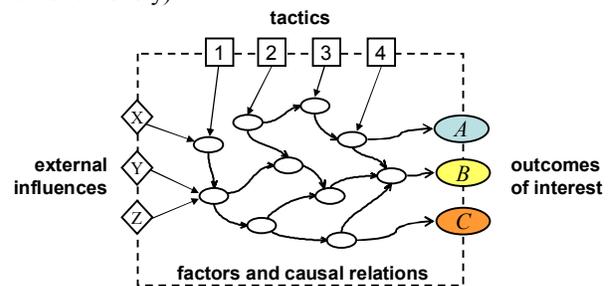


Figure 1. System diagram used by policy analysts

Depending on the style of policy analysis [18], the analyst may develop a single, integrated system diagram to be used as the basis for a quantitative modeling, or develop a range of different system diagrams to represent the variety of actor perspectives on the policy issue under investigation.

A system diagram is a type of causal map [2, 13]. Previous work on causal maps (see [1, 5, 11, 19] for excellent surveys) has produced a variety of formalisms for representing and making inference on causal maps. As DANA aims to support policy analysts in modeling and comparing different actor perceptions, its causal maps should be represented in such a way that:

1. the analyst can differentiate between stronger and weaker causal influence;
2. the analyst can represent an actor's uncertainty regarding external influences and causal relations;
3. the impact of tactics and external influences on outcomes of interest can be inferred from a map, and compared across maps;
4. the analyst can differentiate between more and less important outcomes of interest;
5. an actor's preferred strategy can be inferred from a map (to detect conflicting views on what is good policy).

The first two requirements pose an interesting problem: the formalisms found in the literature either combine a deterministic causal algebra with a discrete scale of strength levels [5, p. 194], or represent a causal relation between two factors as a fuzzy number [6, 15, 27] or using Bayesian probabilities [20]. There appears to be no prior work on causal maps in which the strength and certainty of a causal link are two separate dimensions, whereas decision makers do distinguish between impacts that are weak but certain, and impacts that may be very strong but have a low probability. This was the main reason for developing a new formalism for DANA.

The third requirement emphasizes that the causal maps are used primarily for comparing actor perceptions. The analyst wants to know whether actors expect different outcomes from the same action or external influence. Such comparison requires that an influence matrix can be inferred. How this is done efficiently depends on the representation. A relational algebra permits elegant matrix operations to calculate for all factor pairs (f_i, f_j) the total effect of f_i on f_j as the sum of the indirect effects of all paths from f_i to f_j [5, p. 187], whereas heuristic algorithms are used to compute the transitive closure for the NPN networks proposed in [27].

The association of utility with changes in certain factors implied by the fourth requirement is usually addressed obliquely [26, p. 145], if at all. Although

causal maps can be used in different ways to determine preferred strategies [19], the fifth requirement implies that the representation of the utility (as perceived by different actors) of changes in factors, and how these utilities are used in search for preferred strategies, taking into account uncertainty, needs to be elaborated in a form that is both intuitive and logically sound.

The 'perception graph' representation proposed in the following section meets all five requirements.

3. Representation of a perception graph

3.1. Change, and causal links as multipliers

A causal model should represent to what extent a change in one factor causes changes in other factors in the system. In DANA, changes are defined on a 7-point scale: a change in a factor may range from a strong decrease via 'no change' to a strong increase, graphically depicted as minuses and pluses of different sizes. Throughout this paper, the letter c is used to denote a change, with $c \in \{\ominus, \ominus, \ominus, \circ, \oplus, \oplus, \oplus\}$.

If a change in factor A is believed to cause a change in factor B , the change in B is a function of the change in A and the specific causal relation between A and B . The relation $A \rightarrow B$ may affect the direction of a change (an increase in factor A may cause a decrease in factor B) as well as amplify or reduce changes in chain of cause and effect. Thus, the causal relation ' \rightarrow ' functions as a change multiplier. Change multipliers are defined on a 7-point scale that ranges (in both directions) from 'no effect', via 'reduced effect' and 'same effect', to 'amplified effect'. Throughout this paper, the letter m is used to denote a change multiplier, with $m \in \{\ominus, \ominus, \ominus, \circ, \oplus, \oplus, \oplus\}$.

Because causal relations function as change multipliers, the change in B as the result of a change in A is defined as $c(B) = m \otimes c(A)$, where \otimes is a special multiplication operator. Causal influences are assumed to accumulate: when two relations $A \rightarrow C$ and $B \rightarrow C$ are assumed to exist, the change in C is defined as $c(C) = (m_1 \otimes c(A)) \oplus (m_2 \otimes c(B))$, where m_1 and m_2 are the multipliers of, respectively, $A \rightarrow C$ and $B \rightarrow C$, and \oplus is a special addition operator.

The 7-point scales raise questions when it comes to defining the result of adding and multiplying changes: if one intuitively expects $\oplus \otimes \oplus$ to produce \oplus , $\oplus \otimes \ominus$ to produce \ominus , and $\oplus \otimes \oplus$ to produce \oplus , what to expect of $\oplus \oplus \oplus$ or of $\ominus \otimes \ominus \otimes \oplus$? Unless some rules for truncation are defined, a change may 'run off the scale'. DANA originally (see [4]) defined algebraic laws for \oplus and \otimes , using tables similar to those in [5]. This led to the undesirable property that the sequence

$A \ominus \otimes B \oplus \otimes C$ produced different results from $A \oplus \otimes B \ominus \otimes C$, and that $(A \ominus \otimes B) \oplus \otimes C$ could differ from $A \ominus \otimes (B \oplus \otimes C)$. The present implementation of DANA therefore converts the ordinal scale values into ratio scale values before a sequence of operations, then performs standard multiplications (\cdot) and additions ($+$), and finally converts the result back to ordinal scale values. Tables 1 and 2 define the conversion functions for changes and change multipliers that are used in DANA, with the aim to preserve as much as possible the intuitive interpretation of \otimes and \oplus , as well as their associative and commutative properties, and the property that \otimes distributes over \oplus .

Table 1. Conversion table for changes

$\{r: c\text{-ord}(r) = c\}$	c	$c\text{-rat}(c)$
$\langle -\infty, -\gamma_3 \rangle$	■	$-\gamma_3$
$\langle -\gamma_3, -\gamma_2 \rangle$	▬	$-\gamma_2$
$\langle -\gamma_2, -\gamma_1 \rangle$	-	$-\gamma_1$
$\langle -\gamma_1, \gamma_1 \rangle$	○	0
$[\gamma_1, \gamma_2 \rangle$	+	γ_1
$[\gamma_2, \gamma_3 \rangle$	+	γ_2
$[\gamma_3, \infty \rangle$	+	γ_3

The function $c\text{-rat}$ returns for a change c the corresponding real number. The default values in DANA are $\gamma_1 = 1$, $\gamma_2 = 2$, and $\gamma_3 = 4$, which makes that $+\oplus+ = +$, and that $+\oplus+ = +$. The analyst can change the γ -parameters, as long as $0 < \gamma_1 < \gamma_2 < \gamma_3$. The function $c\text{-ord}$ acts like the inverse of $c\text{-rat}$, returning for any real number the corresponding change on the 7-point scale. The intervals in Table 1 show how small numbers are truncated to ○, and large numbers to + or ▬.

Table 2. Conversion table for change multipliers

$\{r: m\text{-ord}(r) = m\}$	m	$m\text{-rat}(m)$
$\langle -\infty, -\mu \rangle$	⊖	$-\mu$
$\langle -\mu, -1 \rangle$	⊖	-1
$\langle -1, -1/\mu \rangle$	⊖	$-1/\mu$
$\langle -1/\mu, 1/\mu \rangle$	○	0
$[1/\mu, 1 \rangle$	⊕	$1/\mu$
$[1, \mu \rangle$	⊕	1
$[\mu, \infty \rangle$	⊕	μ

The function $m\text{-rat}$ depends on a single parameter μ that by default in DANA has the value 2, so $+\otimes+ = +$, and $+\otimes- = -$. The function $m\text{-ord}$ follows the same logic as $c\text{-ord}$.

3.2. Representing uncertainty

Two sources of uncertainty can make that a change in a factor should be modeled as a stochastic variable: There may be uncontrollable forces outside the system

boundaries that cause probabilistic changes, and causal relations may be so uncertain that the multiplier acts as a probabilistic function. This uncertainty is modeled by representing a change as a *change vector*

$$\bar{c} = (p[\blacksquare], p[\blacksquare], p[\blacksquare], p[\circ], p[\oplus], p[\oplus], p[\oplus])$$

with the sum of the probabilities $p[c]$ equal to 1, and likewise a change multiplier as a *multiplier vector*

$$\bar{m} = (p[\ominus], p[\ominus], p[\ominus], p[\circ], p[\oplus], p[\oplus], p[\oplus])$$

with $\sum p[m] = 1$. The definition of a causal relation or *link* then follows:

Definition 1: A *link* is a 3-tuple $l = (f_{\text{cause}}, f_{\text{effect}}, \bar{m})$ where f_{cause} and f_{effect} are factors and \bar{m} is a multiplier vector.

To preserve the benefits of the ‘convert–compute–convert’ approach, the change operators \otimes and \oplus for change vectors must be defined in such manner that intermediary result values that do not figure on the 7-point scale are left intact until all computation has been done. For this purpose, the *probability–value set*, or *proval* for short, is introduced.

Definition 2: A *proval* is a set of ordered pairs (p_i, v_i) where probability p_i and value v_i both are real numbers, and $0 < p_i \leq 1$.

The function Φ_m that converts *multiplier vectors* into *provals*, the function Φ_c that converts *change vectors* into *provals*, and their inverse functions can be defined as follows.

Definition 3a: Let $m\text{-rat}$ and $c\text{-rat}$ be the functions that map ordinal scale values onto ratio scale values according to Tables 1 and 2, let \bar{m} be a multiplier vector, and let \bar{c} be a change vector, then

$$\Phi_m(\bar{m}) = \cup \{(p[m], m\text{-rat}(m))\} \quad (1)$$

for $m = \ominus, \dots, \oplus$ with $p[m] > 0$, and

$$\Phi_c(\bar{c}) = \cup \{(p[c], c\text{-rat}(c))\} \quad (2)$$

for $c = \blacksquare, \dots, \oplus$ with $p[c] > 0$.

Definition 3b: Let $m\text{-ord}$ and $c\text{-ord}$ be the functions that map ratio scale values onto ordinal scale values according to Tables 1 and 2, and let x be a *proval*, then

$$\Phi_m^{-1}(x) = (p[\ominus], p[\ominus], p[\ominus], p[\circ], p[\oplus], p[\oplus], p[\oplus])$$

with

$$p[m] = \sum p \text{ for } (p, v) \in x \text{ with } m\text{-ord}(v) = m \quad (3)$$

and

$$\Phi_c^{-1}(x) = (p[\blacksquare], p[\blacksquare], p[\blacksquare], p[\circ], p[\oplus], p[\oplus], p[\oplus])$$

with

$$p[c] = \sum p \text{ for } (p, v) \in x \text{ with } c\text{-ord}(v) = c \quad (4)$$

Assuming that the probabilities of the *provals* in a given expression are independent, the operators \otimes and \oplus can now be defined for *provals* as follows:

Definition 4: Let x and y be two *provals* $x = \{(px_1, vx_1), \dots, (px_m, vx_m)\}$ and $y = \{(py_1, vy_1), \dots, (py_n, vy_n)\}$ with integers $m \geq 0$ and $n \geq 0$, then

$$x \otimes y = \bigcup_{i=1}^m \bigcup_{j=1}^n \{(px_i \cdot py_j, vx_i \cdot vy_j)\} \quad (5)$$

and

$$x \oplus y = \bigcup_{i=1}^m \bigcup_{j=1}^n \{(px_i \cdot py_j, vx_i + vy_j)\} \quad (6)$$

As an illustration (again using the parameter values $\mu = 2$, $\gamma_1 = 1$, $\gamma_2 = 2$, and $\gamma_3 = 4$), let the change vector $\bar{c}(A) = (0, 0, 0, 0, 0.1, 0.2, 0.7)$ represent the notion of ‘an increase in A , most likely a strong one’, then conversion to a *proval* gives $x = \Phi_c(\bar{c}) = \{(0.1, 1), (0.2, 2), (0.7, 4)\}$. Furthermore, let the multiplier vector $\bar{m} = (0.9, 0.1, 0, 0, 0, 0, 0)$ in $A \xrightarrow{\bar{m}} B$ represent the notion ‘an increase in A will almost certainly cause a strong decrease in B ’, then conversion to a *proval* gives $y = \Phi_m(\bar{m}) = \{(0.9, -2), (0.1, -1)\}$, multiplication of the two *provals* gives $x \otimes y = \{(0.09, -2), (0.01, -1), (0.18, -4), (0.02, -2), (0.63, -8), (0.07, -4)\}$, and finally $c(B) = \Phi_c^{-1}(x \otimes y) = (0.88, 0.11, 0.01, 0, 0, 0, 0)$.

It is worth noting that although Definition 2 allows a *proval* to be an empty set and does not require that $\sum p_i = 1$, the \otimes and \oplus operators do conserve the condition that the sum of the probabilities in a *proval* be 1 if it holds for both operands x and y . It can easily be checked that the \otimes and \oplus operators for *provals* still have the commutative and associative properties.

3.3. Representing external influences, tactics, and outcomes of interest

In addition to factors and causal links, DANA perception graphs also represent the external influences, tactics, and outcomes of interests depicted in Figure 1.

External influences are represented as *prospects*. A prospect associates a change vector with a factor to express how the actor thinks this factor may change in the near future:

Definition 4: A *prospect* is an ordered pair $p = (f, \bar{c})$ where f is a factor and \bar{c} a change vector.

Tactics are represented as changes brought about by some actor’s *actions*. An action is represented as a factor associated with some actor a , where a may be a different actor than the one who’s perception is represented by the graph. The ‘feasible action range’ (by default the full set $\{\ominus, \dots, \oplus\}$) may be reduced to a

subset to represent that the actor has limited power or freedom to act.

Definition 5: A *tactic* is an ordered pair (f, c) where f is a factor that represents an action that can be taken by some actor, and c is a specific change value that falls within the feasible action range for f and represents the extent of the action, i.e., \ominus indicates no action, and $\omin�$ and \oplus indicate a change to the most influential actor’s maximum ability.

Outcomes of interests are represented as *goals*. A goal associates a utility value with each of the seven possible changes in a factor.

Definition 6: A *goal* is an ordered pair $g = (f, \bar{u})$ where f is a factor and \bar{u} a utility vector $(u[\omin�], u[\omin�], u[\omin�])$ with $u[c] \in \{\text{☹, 😞, 😟, 😐, 😊, 😄, 😁}\}$.

The utility values are defined on a 7-point scale that ranges from a strong disapproval via ‘neutral’ to a strong appreciation. The face icons have been chosen to reflect that the utility scale models the feeling experienced by the actor when a change occurs. A 😊 represents the highest positive emotion an actor can experience because of a change in a single factor. If an actor has n goals, then he may experience an emotion as strong as n times 😊.

The ordinal scale u ’s are converted to ratio scale values when computation of (cumulative) expected utilities is required. For this purpose, the function $u\text{-rat}$ maps the seven u ’s onto $\{-v_3, -v_2, -v_1, 0, v_1, v_2, v_3\}$, where $0 < v_1 < v_2 < v_3$. When utility values are output to the user, they are divided by v_3 , so the result is expressed in the unit 😊 on a ratio scale. The default parameter values in DANA are $v_1 = 1$, $v_2 = 2$, and $v_3 = 4$, but these can be changed by the analyst.

4. Causal inference

Using the concepts defined in the previous section, a perception graph can now be defined as follows:

Definition 7: A *perception graph* is a 6-tuple $\mathcal{P} = (A, a, \mathbf{F}, \mathbf{P}, \mathbf{L}, \mathbf{G})$ in which A is a policy arena, a is an actor with an interest in A , and \mathbf{F} is the set of factors, \mathbf{P} the set of prospects, \mathbf{L} the set of links, and \mathbf{G} the set of goals that together represent how actor a views the policy issue addressed in arena A .

Factors and links together constitute a directed graph. The idea is that the casual influence of any factor on any other factor in the system can be consistently inferred from this graph. When devising algorithms that perform this inference, two problems emerge: preserving probabilistic soundness, and dealing with cyclic graphs.

A first problem with perception graphs is that with changes and change multipliers being probability vectors, for some graphs the correct algebraic expression (using \otimes and \oplus) will not produce the correct result, as the following example demonstrates.

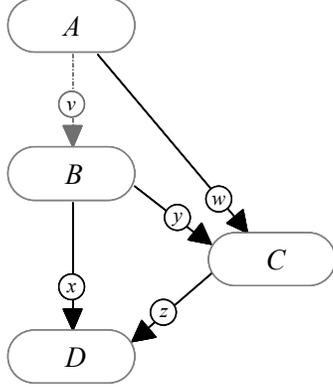


Figure 2. Complex causal influence of A on D

Consider the perception graph in Figure 2. A, B, C and D are factors, and v , w , x , y and z are *provals*. First, assume that the direct link from A to C (with *proval* w) does not exist. The total influence of A on D then is equal to the total influence of B on D, multiplied by v . This can be computed by the following expression, in which the parentheses force the proper order:

$$c(D) = v \otimes (x \oplus (v \otimes z)) \otimes c(A) \quad (7)$$

When the edge $A \rightarrow C$ is added to the graph, the total influence of A on D would be x ‘times’ the total influence of A on B ‘plus’ z ‘times’ the total influence of A on C:

$$c(D) = ((v \otimes x) \oplus (w \oplus (v \otimes y)) \otimes z) \otimes c(A) \quad (8)$$

Although expression (8) properly represents this, it makes that the uncertainty in the causal relation $A \rightarrow B$ is propagated twice. This means that, for example, for $v = \{(0.5, 0.5), (0.5, 1)\}$ and $w = x = y = z = \{(1, 1)\}$, the total influence of A on D is computed to be $\{(0.25, 2), (0.25, 2.5), (0.25, 2.5), (0.25, 3)\}$, while the proper answer should be $\{(0.5, 2), (0.5, 3)\}$. Although in this example the function Φ_m^{-1} will convert both results to the same multiplier vector $(0, 0, 0, 0, 0, 1)$, there will be differences when more causal relations are assumed to be uncertain.

The practical significance (in terms of validity of the insights obtained by the analyst) of errors in the uncertainty of causal influence is as of yet unclear. Presently, DANA reduces the error as much as possible by computing the total influence of factor f_i on factor f_j using the following algorithmic function Π :

Algorithm 1. For a given path set \mathbf{P} containing chains of causal relations $f_0 \xrightarrow{\bar{m}_1} f_1 \xrightarrow{\bar{m}_2} \dots \xrightarrow{\bar{m}_h} f_n$, the function $\Pi(\mathbf{P})$ returns a *proval* that represents the total influence of f_0 on f_n via these paths, computed as follows:

```

if  $\mathbf{P} = \emptyset$  then
  return  $\{(1, 0)\}$ 
else if  $\#\mathbf{P} = 1$  then
  return  $\Phi_m(\bar{m}_h) \otimes \dots \otimes \Phi_m(\bar{m}_1)$ 
else
  select the edge  $e = f_k \xrightarrow{\bar{m}} f_l$  that occurs in most
  paths in  $\mathbf{P}$ ;
  let  $x = \Phi_m(\bar{m})$ ;
  let  $\mathbf{P}_0 \subset \mathbf{P}$  contain all paths in  $\mathbf{P}$  that do not contain  $e$ ;
  let  $\mathbf{P}_1 = \mathbf{P} - \mathbf{P}_0$ ;
  remove  $e$  from all paths in  $\mathbf{P}_1$ ;
  return  $(x \otimes \Pi(\mathbf{P}_1)) \oplus \Pi(\mathbf{P}_0)$ 
fi

```

It can easily be checked that when applied to the graph in Figure 2 (without and with link $A \rightarrow C$), Algorithm 1 effectively calculates the corresponding expressions for $c(D)$.

Algorithm 1 implies that the set of paths between two factors must be finite. The path matrix is constructed using an algorithm similar to those discussed in [27]. To deal with cyclic dependencies when inferring the causal influence between any pair of factors, DANA therefore takes into account only ‘straight paths’, that is, sequences of causal relations $f_0 \xrightarrow{\bar{m}_1} f_1 \xrightarrow{\bar{m}_2} \dots \xrightarrow{\bar{m}_n} f_n$ in which all factors are different except that f_0 may be equal to f_n . Thus, the (indirect) influence that factors may have on themselves is not ignored, but path sets with an infinite number of elements are avoided. The consequences of cycles in terms of dynamic system behavior are something to be assessed by the analyst. Note that if the analyst believes that an actor is aware of the consequences of amplifying or damping feedback effects (taking into account the actor’s time horizon!), such strong effects can also be represented by replacing a feedback loop by a link with a strong/weak multiplier in case of positive/negative feedback, or by removing the link altogether.

Definition 8: Let $\mathbf{F} = \{f_1, \dots, f_n\}$ be a set of factors, then the *path matrix* $\text{PM}(\mathbf{F})$ is an $n \times n$ matrix of path sets with pm_{ij} the set of all straight paths from f_i to f_j , and the *total influence matrix* $\text{TM}(\mathbf{F})$ is an $n \times n$ matrix of *provals* with $tm_{ij} = \Pi(pm_{ij})$.

The two matrices permit automated comparison of the causal assumptions made by different actors that help the analyst find conflicting views. They also constitute the basis for strategy evaluation, as will be discussed in the next section.

5. Strategy evaluation and selection

In DANA, three measures can be used to evaluate changes: utility, satisfaction, and frustration. Each measure is a function of a goal and a change.

Definition 9: Let $g = (f, \bar{u})$ be a goal with u_{\max} the highest positive utility value and u_{\min} the lowest negative utility value, and let \bar{c} be a change vector, then

$$utility(g, \bar{c}) = \sum p[c] \cdot u[c] / v_3 \quad \text{for } c = \blacksquare, \dots, \blacksquare \quad (9)$$

$$satisfaction(g, \bar{c}) = 1 - \sum p[c] \cdot (u_{\max} - u[c]) / u_{\max} \quad (10)$$

for $c = \blacksquare, \dots, \blacksquare$ with $u[c] > 0$

$$frustration(g, \bar{c}) = 1 - \sum p[c] \cdot (u_{\min} - u[c]) / u_{\min} \quad (11)$$

for $c = \blacksquare, \dots, \blacksquare$ with $u[c] < 0$

The *utility* function returns a value on the ratio scale interval $[-1, 1]$ that corresponds to the ordinal scale interval $[\text{☹}, \text{😊}]$. Utility values can be added, so for multiple goals negative utility is balanced against positive utility. The *satisfaction* and *frustration* functions return a value on the interval $[0, 1]$ that should be read as a percentage. 100% satisfaction means that the change vector \bar{c} could not have been better, while 0% satisfaction means that there is 0% chance of a positive utility. Likewise, 100% frustration means that the change vector \bar{c} could not have been worse, while 0% frustration means that there is 0% chance of a negative utility. If an actor has multiple goals, satisfaction values and frustration values should be averaged, rather than summed. There is no point in adding satisfaction to frustration values; total expected utility is most appropriate as a balance indicator.

Using the total influence matrix of Definition 8 and the evaluation functions of Definition 9, DANA can generate evaluation tables that show the total utility, satisfaction and frustration for each tactic, and provide details on its impact on separate goals. The standard deviation for utility and the averages for satisfaction and frustration will be sensitive to the imperfect propagation of uncertainty discussed in Section 4.

By definition, a strategy is a combination of tactics. The example in Figure 3 demonstrates that the combination of the most promising tactics does not necessarily produce the best possible strategy. Since action A and action B both have a slight influence on factor C , for each action separate the preferred tactic would be to take strong action (☒) to achieve the preferred change in D (in the example in Figure 3 ☒, as the medium-sized upward triangle ▲ denotes a utility vector (☹, ☹, ☹, ☹, ☹, ☹, ☹, ☹, ☹, ☹)), but the combined strategy would obviously cause D to overshoot its target. In general, when two or more tactics affect the same goal, their combined impact is likely to be sub-optimal. To find the preferred strategy,

the total influence matrix can be used to formulate the optimization problem as the equation $(c(A) \otimes (\oplus \otimes \oplus)) \oplus (c(B) \otimes (\oplus \otimes \oplus)) = \oplus$, for which the solution space is constrained by the feasible action ranges of the actions (the white-on-gray changes ☐ indicate which actions are deemed unfeasible).

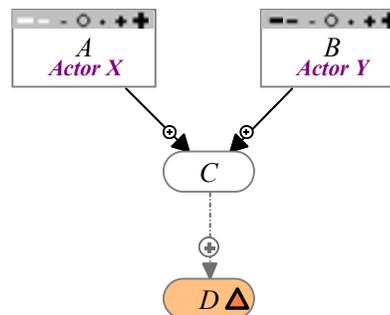


Figure 3. Two tactics affecting the same goal

If multiple goals are specified, the optimization problem will involve a set of equations. As the variables are *provals*, an aggregation function such as expected utility must be selected to find the optimum.

Presently, DANA finds solutions with ‘brute force’: it enumerates all combinations of feasible tactics and evaluates these by cumulating the impact of separate tactics on goals. Note how in the given examples this leads to the problem of ‘doubling’ the uncertainty in the link $C \rightarrow D$. The path from $A \rightarrow D$ and the path from $B \rightarrow D$ both contain an uncertain link, so the influences $A \rightarrow D$ and $B \rightarrow D$ in the impact matrix are both uncertain. When the uncertain impacts are added, the \oplus operator treats their uncertainties as independent, whereas they are not.

Another problem is that the ‘brute force’ approach is inadequate when the number of feasible tactics is large. For want of smarter algorithms, DANA samples the solution space when it is larger than a specified number. The analyst can also guide the evaluation process by reducing the feasible action range for actions.

The analyst can select optimization criteria to model different actor rationalities if such is deemed appropriate. By default, DANA generates ideal strategies by maximizing expected utility, but most satisfaction and least frustration may also be aimed for.

6. Some analyses supported by DANA

The previous section has shown that analysis of a single perception graph can reveal the promising tactics and preferred strategies from a single actor perspective. Comparison of perception graphs that model how to different actors view the same policy issue can yield insights that may be even more valuable to a policy analyst. In particular, it may help the

analyst to distinguish between different types of conflict:

- *goal conflict*: Actors may fundamentally disagree about what changes in a factor are desirable. For example, actor a may be satisfied when costs do not increase (☉), while actor b prefers a substantial decrease in cost (⊕).
- *action conflict*: Actors may disagree about what action should be taken. This may be because they have different goals, but also because they make different assumptions about the system’s behavior.

These two types of disagreement are labeled as ‘conflict’ because they involve differences in values. Other types of disagreement are:

- *disagreement on prospects*: Actors may make different assumptions about what changes will occur due to external influences.
- *disagreement on influence*: Actor who see the same causal link $X \rightarrow Y$ may disagree about its multiplier to the extent that actor a expects an increase in factor X to cause a strong increase in Y , whereas actor b expects it to cause a strong decrease in Y .
- *disagreement on causality*: Actors may disagree about cause and effect. For example, actor a may argue that obesity causes people to exercise less, which may lower their physical condition, while actor b argues that when physical condition is reduced (e.g., due to illness), people will exercise less and may become obese.

All types of disagreement can be precisely defined as functions. As an example, the following definition specifies how the function *g-conflict* computes goal conflict as the average (over all possible changes in the goal factor) of the absolute difference between the utilities these changes generate for each actor. Thus, *g-conflict* returns a numerical value on the utility scale, i.e., $1 = \text{☺}$.

Definition 10: Let $\mathcal{P}_1 = (A, a_1, \mathbf{F}_1, \mathbf{P}_1, \mathbf{L}_1, \mathbf{G}_1)$ and $\mathcal{P}_2 = (A, a_2, \mathbf{F}_2, \mathbf{P}_2, \mathbf{L}_2, \mathbf{G}_2)$ be the perception graphs that model how actors a_1 and a_2 view the issue addressed in policy arena A , let $f \in \mathbf{F}_1 \cap \mathbf{F}_2$ be a factor representing an action, and let $g_1 = (f_1, \bar{u}_1) \in \mathbf{G}_1, g_2 = (f_2, \bar{u}_2) \in \mathbf{G}_2$, then

$$g\text{-conflict}(g_1, g_2) = \sum_{c \in \mathbb{M}} |\bar{u}_1[c] - \bar{u}_2[c]| / 7 \quad (12)$$

The definitions of the other disagreement functions can be found on the DANA web site [28]. The disagreement functions can help the analyst to identify and understand the nature of conflicts in a policy arena. To assess action conflict at the level of strategies, rather than tactics, additional functions are needed. Using its capability to infer from a single perception graph the best possible strategies, DANA can also infer how actors would value each others preferred strategies. To this purpose, DANA computes the

utility, satisfaction and/or frustration of actor a_1 for the strategy that contains the tactics preferred by actor a_2 insofar as the actions occur in both perceptions. If actor a_1 perceives actions that are not perceived by actor a_2 , DANA chooses the best possible tactics (as seen by a_1) for these actions. The difference is a good indicator for the *opposition* of a_1 against the preferred strategy of a_2 , as it measures how much a_1 stands to lose if this strategy were implemented.

Comparison of two perception graphs can also reveal the extent to which actor a_1 is dependent on the actions of actor a_2 to achieve his goals. To this purpose, the functions defined in Sections 4 and 5 allow DANA to compute a range of dependence measures (see [3] for an empirical example):

- The number of actions of a_2 that *may be opportune* for a_1 , i.e., when the expected utility is positive for at least one tactic in the action’s feasible action range.
- The number of opportune tactics divided by the total number of feasible tactics of a_2 . This ratio is inversely proportional to the *benign discretion* of a_2 , where 1 means that a_2 has no choice but to act in favor of a_1 .
- The total utility and satisfaction that a_1 may gain, and the total frustration that a_1 may avoid, through the actions of a_2 . These measures (which are relative to the ‘base case’ i.e., the overall strategy in which no action is taken by any actor), reflect the ‘size of the carrot’ that actor a_2 can offer to a_1 .
- The number of actions of a_2 that *may be threatening* for a_1 , i.e., when the expected utility is negative for at least one tactic in its feasible action range.
- The number of threatening tactics divided by the total number of feasible tactics of a_2 . This ratio is inversely proportional to the *malevolent discretion* of a_2 , where 1 means that a_2 has no choice but to act against a_1 .
- The total utility and satisfaction that a_1 may lose, and the total additional frustration that a_1 may experience, through the actions of a_2 . These measures (again relative to the ‘base case’), reflect the ‘size of the stick’ that actor a_2 can wield over a_1 .

Obviously, if the measures for disagreement and dependence can be computed for two actors, they can be computed for any actor pair (a_i, a_j) when a_i and a_j are stakeholders whose perception on some policy issue has been modeled as a perception graph.

When a dependence relation is computed for all n actors in a single policy arena, the result can be presented in an $n \times n$ table, but also as a graph in which the actors are the nodes. Cycles in this graph indicate possibilities for ‘deals’ in which a subgroup of two or more actors trade off their different interests.

In addition to these more sophisticated policy-relevant indicators, DANA can also compute indicators that are more commonly used in the analysis of causal maps: occurrence of factors and links, and factor centrality [14, 15]. Details on the analyses currently supported by can be found on the DANA web site [28].

7. Conclusion and outlook

Building on the assumption that policy analysts can benefit from analyzing and comparing the perceptions of the actors involved in a policy issue, Dynamic Actor Network Analysis (DANA) has been proposed as a method to support such analysis and comparison.

The focus of this paper has been the definition of perception graphs and the inference they afford. Although perception graphs have the desired property of allowing the analyst to represent the strength and uncertainty of causal links as two separate aspects, the inference mechanism may in some cases produce incorrect probabilities. The significance of these errors in practice needs to be investigated empirically. In addition, a comparative study in which the same causal maps are represented and interpreted using different formalisms can reveal whether perception graphs offer a real advantage.

References

- [1] Aguilar, J. (2005) A Survey about Fuzzy Cognitive Maps Papers. *Int. J. Computational Cognition* **3**, 27-33
- [2] Axelrod, R. (Ed.) (1976) *Structure of Decision – The Cognitive Maps of Political Elites*. Princeton, NJ: Princeton University Press.
- [3] Bots, P.W.G. (2007) Benchmarking in Dutch Urban Water Management: An Assessment. In Pahl-Wostl, C. (Ed.) *Proc. Int. Conf. on Adaptive & Integrated Water Management (CAIWA)*. Berlin: Springer.
- [4] Bots, P.W.G., J.R. van Duin, M.J.W. Van Twist (2000) Automatic Pattern Detection in Stakeholder Networks. *Proc. 33rd Hawaii Int. Conf. on System Sciences*. Los Alamitos: IEEE Press.
- [5] Chaib-draa, B., J. Desharnais (1998) A relational model of cognitive maps. *International Journal of Human-Computer Studies* **49**(2), 181-200.
- [6] Chen, S.M. (1995) Cognitive-map-based decision analysis based on NPN logics. *Fuzzy Sets and Systems* **71**(1), 155-163.
- [7] Cohen, M.D., J.G. March, J.P. Olson (1972) A Garbage Can Model of Organizational Choice. *Administrative Science Quarterly* **17**(1), 1-25.
- [8] De Bruijn, H. (2005) Roles for unilateral action in networks. *International Journal of Public Sector Management* **18**(4), 318-329.
- [9] Dunn, W.N. (1981) *Public Policy Analysis: An Introduction*. Englewood Cliffs, NJ: Prentice-Hall.
- [10] Eden, C., S. Jones, D. Sims (1983) *Messing About in Problems*. Oxford, UK: Pergamon Press.
- [11] Eden, C. (1988) Cognitive mapping: A review. *European Journal of Operational Research* **36**(1), 1-13.
- [12] Eden, C. (2004) Analyzing cognitive maps to help structure issues or problems. *European Journal of Operational Research* **159**(3), 673-686.
- [13] Huff, A.S. (Ed.) (1990) *Mapping Strategic Thought*. New York: Wiley.
- [14] Khan, M.S., M. Quaddus (2004) Group Decision Support Using Fuzzy Cognitive Maps for Causal Reasoning. *Group Decision and Negotiation* **13**(5), 463-480.
- [15] Kosko, B. (1986) Fuzzy cognitive maps. *International Journal of Man-Machines Studies* **25**(1), 65-75.
- [16] Langfield-Smith, K., A. Wirth (1992) Measuring Differences between Cognitive Maps. *The Journal of the Operational Research Society* **43**(12), 1135-1150.
- [17] Marsh, D. (1998) *Comparing Policy Networks*. Buckingham, PA: Open University Press.
- [18] Mayer, I.S., C.E. Van Daalen, P.W.G. Bots (2004) Perspectives on policy analyses: a framework for understanding and design. *International Journal of Technology, Policy and Management* **4** (2), 169-191.
- [19] Montibeller, G., V. Belton (2006) Causal maps and the evaluation of decision options: a review. *The Journal of the Operational Research Society* **57**(7), 779-791.
- [20] Nadkarni, S., P. Shenoy (2001) A Bayesian network approach to making inferences in causal maps. *European J. of Operational Research* **128**, 479-498.
- [21] Quade, E.S. (1975) *Analysis for Public Decisions*. New York: Elsevier.
- [22] Richards, D. (2001) Coordination and Shared Mental Models. *American J. of Political Science* **45**, 259-276.
- [23] Sabatier, P.A. (1988) An advocacy coalition framework of policy change and the role of policy-oriented learning therein. *Policy Sciences* **21**(2), 129-168.
- [24] Thomas, W.I., M. Janowitz (1966) *W.I. Thomas on social organization and social personality: selected papers*. Chicago, IL: Chicago University Press.
- [25] Walker, W. (1988) Generating and Screening Alternatives. In Miser, H.J., E.S. Quade (Eds) *Handbook of systems analysis; craft issues and procedural choices*. Chichester: Wiley.
- [26] Wellman, M.P. (1994) Inference in cognitive maps. *Mathematics and Computers in Simulation* **36**, 137-148.
- [27] Zhang, W., S. Chen, J.C. Bezdek (1989) Pool2: a generic system for cognitive map development and decision analysis. *IEEE Trans. Systems, Man and Cybernetics* **19**(1), 31-39.
- [28] <http://dana.actoranalysis.com>