STRUCTURAL PROCEDURES OF S-A (SITUATION ASSESSMENT) FOR AUTONOMOUS REASONING MODULE

Adam M. Gadomski
Indep. consultant

c/o INN-RIN
C.R.E. Casaccia
ENEA

November 1993

STRUCTURAL PROCEDURES OF S-A
(SITUATION ASSESSMENT)
FOR AUTONOMOUS REASONING MODULE

Adam M. Gadomski

Contains

At least, everything is based on or lead to a consensus.
(B.D. Ridman)

Part I Conceptualization Background

1 Qualitative definition of *Situation Assessment*

Contrary to physical systems which behave according to externally observed initial state and physical laws, the behavior of an intelligent agent also depends on its unobserved and variable "mental" states represented by such abstract concepts as: information, goal, preference, and knowledge.

According to the TOGA theory , they form a conceptual web which can be interpreted as the *architecture* of Abstract Intelligent Agent (AIA).[1]

The goal-oriented intelligent reasoning of AIA , includes many activities and functions such as conceptualization, recognition, decision-making, learning, discovery, situation assessment and planning. These activities, applied recursively and repeatedly are necessary generic elements of AIA *behavior*.

The subject of this paper is a *situation assessment*, one of the fundamental "mental" functions of intelligent system interacting with its environment.

Situation

According The Concise Oxford Dictionary, *situation* is "set of circumstances and position in which one finds oneself". Using more formal terminology , we can say that

Situation is a set of properties of the predefined domain selected by IA (Intelligent Agent). This domain (S-A domain, s-a-d) must include the IA itself, its hypotetical domain of activity, and domain of influence.

AIA views every situation in the context of its goal-oriented activity.

<u>Definition 1</u>

Situation is a property of agent domain of activity represented from the point of view of the current agent intervention goal (a subjective property of AIA) under the constrains of agent abilities and possibilities of action, assumed as invariant in a given moment or time interval.

Situation script is a product of the conceptualization of a situation which always depends on the agent abilities and possibilities of reasoning.

Assessment

1 Most important terms are written in Italic if their textual contexts explain their notion, or in order to stress that their formal meaning in frame of the presented theory . For example: *Dog* is an abstract animal. A dog came here yesterday. Rex is a *dog*.

<u>Definition 2</u>

Assessment is an autonomous "mental" process carry of by an agent which uses currently available, frequently uncertain, and qualitative:

- information from a certain domain,
- methods or rules ,

in order to obtain the data considered necessary for the next anticipate mental operation/process.

For example, it can be an ad hoc estimation of the commercial value of an object, but also an evaluation of the utility of this object to imprecisely defined purposes.

Situation assessment

Situation assessment (S-A) definition is composed of the previous both definitions and it is a repetitive and recursive phase of AIA reasoning.

S-A produces a *situation script*, it directly activates a procedure which leads to the goal achieving, if such is available.
If the procedure does not exist then *situation script*, is an initial data for the planning phase.

Situation script is usually "less precise" than a *situation model*, it includes heuristic information on highest generalization levels and represents situation from specific points of view.

Therefore it can includes meta-informations , such as 'the situation is too difficult' and ' situation X is good'.

Situation model is the product of identification process, and includes only all initial data for the next anticipate mental operation/process.

Situation script is constructed top-down

S-A may also be viewed as a problem specification before problem solving.

Summarizing, S-A algorithms should operate on the agent's:

- image of the real environment, i.e. on an abstract domain of activity,
- possibilities, i.e. knowledge and available physical means,
- preferences, i.e. axiology, current motivation and priority rules, and goals.

These data are qualitative, incomplete, and uncertain.

S-A produce *situation script* as an abstract "mental" image of the arbitrary preselected domain of agent observation.

The subject of this paper regards the framework of S-A valid in different end-domains of intelligent agent activity, as well as on different levels of abstraction.**(** The subject of the work regards a *situation assessment* (S-A) on different abstraction levels, domain independent, and

which may be applied in different domains of intelligent agent activities.)
We should notice that from the external observer point of view, a situation assessment can also be defined as a property of not intelligent agent. For example, it can be performed in a fixed domain of activity by fixed goal and by one computer procedure.

In order to illustrate the domains of S-A we can mention , for example:

- S-A in learning, discovery, and in autonomous robot task execution,
- S-A in multi-agent worlds; organization management, cooperation, negotiation, tutoring,
- S-A in plant supervision and control, in emergency management, under risk and stress,

----

From the point of view of a certain initially assumed goal of AIA we distinguish:
Global S-A, its propagation depends on assumed structural properties of preselected AIA , and refers to all AIA physical and abstract domains of activity which can influence the achievement of initially accepted intervention goal.
Local S-A, it refers to one preselected domain of activity (d-o-a) with one predefined goal.

S-A can refer to static or dynamic d-o-a, in the last case it requires information or knowledge acquisition functions.
In general, S-A describes *which* and *how* the properties of AIA and its world can influence AIA decisions during :

- goal-oriented reasoning,
- physical interventions , and
- communications with other agents (intelligent or not intelligent, according to the TOGA definitions).

The next AIA activity, the intervention planning phase, is based on the assessed situation, available operational domain knowledge (ODK) and metalevel knowledge (i.e. strategic knowledge) which are presented on the Fig. 3, and will be explained in the next paragraph. .

In any cases, specification of S-A mental activity strongly depends on initially assumed conceptualization system, i.e. assumed paradigms, AIA architectures, and frameworks of the AIA goal-oriented activity.

2 Conceptualization system TOGA

The conceptualization system employed in this work is called TOGA [ ].

The TOGA theory is composed of three basic conceptual systems:

- **TAO**, **Theory of Abstract Objects**, it is a domain-independent *first level conceptualization* tool; it is the Y1 subtheory (The subtheories Y1,..Y4 are defined in [15]);
- **KNOCS**, **Knowledge Conceptualization System**, it is the *second level conceptualization* system with axiomatic assumptions on the real-world .

Let it be mentioned that every real problem can have many parallel particular problem oriented conceptualizations that constitute, according to the adopted hierarchy, the *third* *conceptualization level,*

KNOCS contains three following fundamental frameworks :

* models of a real and an abstract intelligent agent,
* conceptualization of the domain of activity of AIA,
* conceptualization of goal-oriented AIA activity.

KNOCS integrates Y2 and Y3 systems ;

- **MRUS**, **Methodological Rules System,** it is the system of rules for the specification of complex problem under incomplete and not ordered knowledge. It is TOGA realization of the Y4 system. The fundamental MRUS strategies refer to the problem identification, problem specification, and methodology of problem solving in the contexts of TAO and KNOCS conceptualizations.

MRUS is based on two mechanisms: *goal-driven mechanism* and *top-down mechanism.* They are also employed in the bottom-up strategies of knowledge acquisition, production, and construction of new preferences.

Let us note that the same entity can be in parallel conceptualized in each of the three above mentioned conceptualization levels. For example, the entity such as a *factory* if we use a third conceptualization level, is conceptualized as an *system* in KNOCS and as an *object* in TAO. Therefore in frame of TOGA, the used terms indicate uniquely the conceptualization level to be currently employed.

2.1 Basic assumptions of the theory and AIA model

A1. Every product of the human reasoning activity can be conceptualized and transformed in frames of the Theory of Abstract Objects.

TAO is founded on the following abstract concepts :

- *Objects*, specified by their *attributes* , *values*, and *value domains*.
- Relational isolated networks of objects, called *worlds-of-objects* (w-o-o).
  Each of them is arbitrarily decomposable on a system and its environment.
- *Universes*, they are ws-o-o structured in different abstraction hierarchies and according to preselected perspectives.

TAO includes the set of operations and rules, which enable creation and modification of these concepts in any distinguished set. TAO is a conceptualization system.

Def.

Conceptualization system is a frames system, and the operations set which is defined on these frames.

We can notice, that TAO has an algebra property. TAO can also be seen as a generalization and extension of three existing approaches: the entity-relationship approach, the object-oriented programming/design, and the frame system.

The TAO concepts are called *somethings* if they represent not specified elements or properties of agent domain of activity.

A2. ***Agent***, a system which has a deterministic capability of interactions on symbolic level (i.e. it has ability to communication),

Artificial agent is always goal-oriented.

An *agent* is composed of two parts (in abstraction hierarchy):

CrA - physical system which is a **Carrier** of an Abstract Agent (AA) and

AA - Abstract Agent which is a complex property of the dynamics of the structure of CrA.

Both are autonomous, but interact according to some rules.

A3. ***Abstract Agent***, AA is the trial system composed of complex abstract objects called:

domain-of-activity, knowledge system, preference system.

The model of AA is based on information processing and choice processes . The definitions of the TOGA primary terms are linked together in conceptual clusters. In the AA model (in KNOCS) the primary cluster is composed of:

domain-of-activity(d-o-a), information, knowledge, preference, goal.

Every d-o-a is always considered an information source.

Information is a conceptualization either of the states of the d-o-a itself, or of a state of other world of objects which are symbolically represented in this d-o-a.

Domain-of-activity of the agent is the reference domain of its/his ***knowledge system*** and, from the point of view of an external observer it can be called *knowledge reference domain.*

Knowledge

Knowledge has two components:

descriptive knowledge (passive) which describes possible situations/states in the d-o-a, and

operational knowledge (active) which conceptualize/describes possible actions/ operations on the d-o-a.

An example:

The expression ' T=5 °C ' is an ***information*** if :

- was obtained from d-o-a ,
- is included in *information buffer* ,
- refers to the previously localized object O[T], and
- the value domain of attribute T is known for the receiver.

It can be a fact if this information will be confirmed by verification in the domain of activity.

The expression ' T=5 °C ' is element of ***descriptive knowledge*** if :

- was chosen in *information buffer* for a certain goal-oriented description of d-o-a .
- is the argument a_i of rule : *if* a_1 *then* a_2 in the agent knowledge system.

In other words, the expression ' T=5 °C ' has different roles (and names) in different context.

Def.

Reference relation is the relation between a knowledge and its reference domain, for ex., between the knowledge and d-o-a of abstract agent.

A4. ***Reasoning process*** is a dynamic property of the agent *knowledge system*. Goal-oriented reasoning starts from an activation of the agent *knowledge system* by some states of its *preference system,* which finally changes the state of the *d-o-a* .

Therefore knowledge is the *carrier* of goal-oriented reasoning .
Preference system activates the knowledge only if an *intervention-goal* is established .

Def.
Intervention-goal is the specification of some attributes of the state or process in the environment of the abstract agent established by the preference system with maximal priority.

World of Agent and their elements are representable by means of *object-relation* networks, and their elements are conceptualized by reference to the *observer* .

These objects networks can be graphically well represented by means of GRAF. It enables, in a easy way, the top pictorial conceptualization of problems' domains. In the Fig. 1 the basic graphic symbols of GRAF are presented.

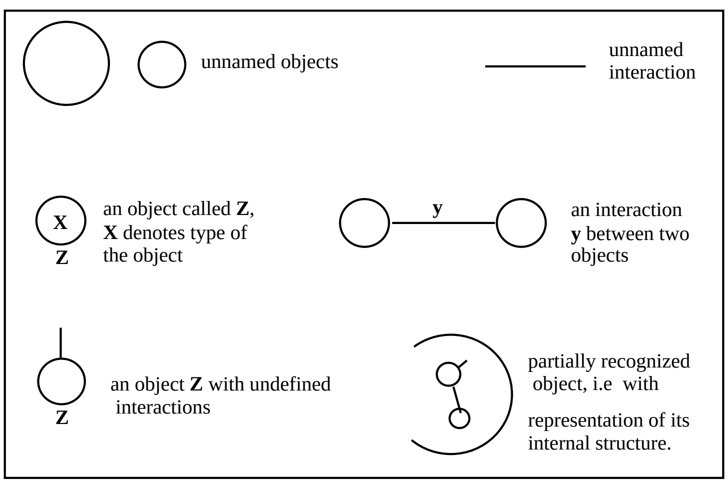

Fig. 1 . The basic graphic symbols of GRAF.

The Fig. 2 illustrates basic processes realized by the AA module .

In TOGA, knowledge is structured in two basic hierarchies: the *generalization hierarchy* and the *metaknowledge* hierarchy.

A domain-knowledge, i.e. referred to a defined d-o-a can be divided into generalization-levels (GL) , In this hierarchy, attributes of an object established on one GL are decomposable on the lower GL. All the definitions established on higher GLs are mandatory on all lower GLs.

The metaknowledge levels obey the following rule:

n-th metaknowledge level is the reference domain for the (n+1)th metaknowledge, where n=0,1,2...

One d-o-a can be conceptualized from different *points-of-view (p-o-v)* depending on the assumed activity goal.
The Real-World (RW) is an infinite source of direct information, hence, also human domain-knowledges.

A4. The information/knowledge of an IA is structured on three *object levels*, OL:

object-relation level (OL1),
attributes-behavior (interaction) ***level*** (OL2),
values-procedure (algorithm) ***level*** (OL3).

OL1 - On the first level, objects and relations with pre-chosen names exist or do not exist in w-o-o.
OL2 - On the second level, IA has an information/knowledge on some of object/relation attributes and it can change their names,
OL3 - On the third level, the values of the attributes and a structure of the changes are established.

If the knowledge about a *something X* is defined or currently available on OL2 then it is denoted as follows X^{OL2}, for example, if X is an object then it is recognized by known attributes names but not by their values.

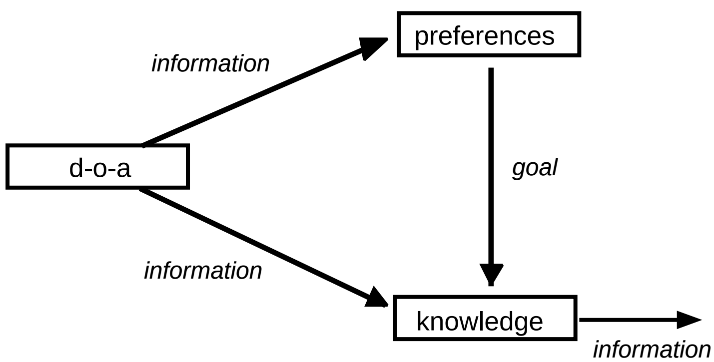

Fig. 2 Basic functional module of a reasoning process of Abstract Agent [17].

A5. The AA with metaknowledge levels which enable it to operate on the knowledge and preferences is called *Abstract Intelligent Agent* (AIA).
In such conceptualization, AIA is composed of the hierarchy of AAs. On the Fig.3 the architecture of AIA is schematically shown .

Let us remark that in the AIA model the concepts of information, knowledge, and goal have relative, AA dependent functional notion.
In order to modify/ acquist knowledge or preference related to a fixed d-o-a , the AIA must change its d-o-a. For ex. a reasoning string which includes a learning process can be repre-

sented as follows:

information -> *knowledge* -> *meta-knowledge* -> *knowledge* -> *information.*

A6. Every process A relying on the behavior of physically realized AIA (X) in its physical environment, or on the change of a state of its knowledge or preferences is *conscious* if X has such conceptual system where the process A is describable and he/it can perform it .
In this case, we can say that the process A is observable for X.

A7. A goal-oriented conscious activity of a human-agent can be conceptualized by its observer (another human-agent) as the activity of an AIA.

2.2 Representation of physical d-o-a

The first conscious representation of the RW in the form of symbols is the zero-level abstract d-o-a of a man.
It includes information about the agent physical d-o-a.
D-o-a is divided on two parts. The first is an *information buffer* which includes all conceptualized information obtained from human "sensorial perception". The second contains goal-oriented *descriptions* of the reality. These descriptions are frequently called static knowledge or domain knowledge.

In the TOGA methodological part (MRUS), any system from the initial domain of activity of AIA can either be specified using the *engineering design paradigm* or/and identified using the *scientific research paradigm*. The basic formal "interface" between these two attempts is constructed by the definition of the couple system [12]:

function - *carrier relation* - *process*

Here, *'function'* is the name of every goal-oriented property of a pre-identified process or system.

From the designer perspective , when a function is defined, its *carrier process* is a realization of this function in an assumed context (conceptualized as a certain *world-of-objects*) of AIA tools/possibilites.

A8. Any arbitrarily selected artificial object from the real d-o-a of human agent, can be conceptualized by the decomposition of the interrelation between a system and its goal , Goal-System Interrelation (GSI) [22].
The decomposition frame is an object network divided into the following layers:

system- goal layer, *functions* layer, *processes* layer, and *system* layer.

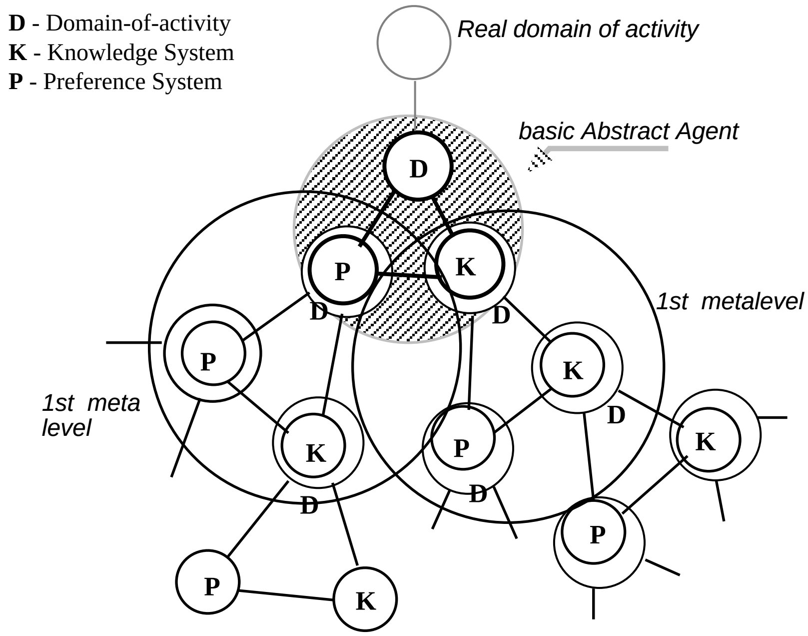

Fig. 3 The TOGA architecture framework of Abstract Intelligent Agent .

Def.
The *System-goal* of a system X in an environment En is the specification of some changes or properties of En required by the user or the creator of X, which should be/are obtained by interactions between X and En.

For this reason, the system-goal must be expressed only in terms of the environment descriptions, and can only be established by the system creator or its user.
The system -goal is also called *design-goal.*
The relations between the system-goal and functions are *cause/consequence, c/c* relations. The relations between functions and processes or systems are called *property/ carrier, p/c* relations.
The GSI conceptual frame allows the decomposition of the relation between a designed, modified or identified system, and its goal. Of course, any "natural" object can have an infinite number of functions and goals, i.e. they depend on the application of an analyzed object by the human-agent.

A9. If the d-o-a X of an AIA includes another AIA Y then X's knowledge related to a selected goal-oriented activity of Y may be conceptualized in terms of formally defined : *intervention-goals, tasks* , and *actions* referred to the Y's d-o-a .

Def.
The *task* is the property of an *action* oriented on predefined *intervention-goal*, it is expressed in terms of d-o-a description, and describes which changes must be introduced in the AIA d-o-a

for achieving an *intervention-goal*.

Def.
The a*ction* is a specification what AIA is to do in d-o-a for the realization of tasks, i.e. in order to achieve the predefined goal .

For such reason, an *action* must depend on executor possibilities, and one *task* can be performed by the execution of different alternative *actions*.
From the identification point of view, one selected *action* can be recognized as the *carrier* of different *tasks.*
Tasks System (task scenario) is executor independent and it depends on goal constrains .

A10. From the p-o-v of the specified goal, an unknown ignorance does not exist, i.e. any ignorance must have attributes because they define relations for a closure of any w-o-o.
Human reasoning referred to a certain d-o-a, is based on many conceptual systems and associative processes. A mixed, not verified in "real time" change of the conceptualization context, frequently leads to false conclusions and intuitive convictions , i.e. to the construction of false or "fiction" domain-knowledge which no longer has reference to any physical or abstract d-o-a of the human agent. We can notice that the human mind is full of such types of constructions. The above situations can be omitted in the design of an artificial AIA.

A11. From the perspective of the AIA knowledge, there are three types of objects/systems/subdomains in the d-o-a:
Dt1. driven only by physical principes; hipothetical goal is unknown for the agent,
Dt2. fixed goal-driven, their design-goal, is known for the agent,
Dt3. driven by temporal intervention-goals.
In the Dt1 domain, the agent can navigate and manipulate.
The Dt2 domain is supervised and controlled by the agent according to its intervention-goals.
The Dt3 domain can also be the domain of cooperations and other social interactions.

A12. According to the TOGA conceptualization, every perceived/conceptualized entity is represented as an object in three-dimensional discrete space (DDS) [4,5]).
- The first dimension called GSI (Goal-System Interrelation), is divided into four layers: *goal, function, process,* and *system.*
- The second dimension gives the possibility of setting up the model on different *generalization levels* (GL), which can be organized arbitrarily (from the initially assumed model definition up to the details level, when the model implementation will be possible).
- The third dimension , called Subject Layers, is used for a set up of the system description in the hierarchy of abstraction, from structures of directly measured (physical) attributes to highly abstract conceptualizations, for ex. material, structural, dynamic and information layers, or hardware, software and model layers.

The problem specification starts from the user knowledge collected in different conceptual systems and the initially specified goal which he intends to achieve.
The pyramidal, top-down problem structuralization requires bottom-up goal-driven structuring of available information and evaluation of knowledge.
The requested completeness of a problem description on any GL level in the DDS space, makes the *identification* of problem-solver *ignorance* (temporal *black objects*) necessary, and indi-

cates the information/knowledge which should be acquired.
The problem structuralization in is performed top-down in the generalization hierarchy by the *identification* of carrier systems and neighborhood processes, and the *specification* of functions necessary for the achievement of the assumed goal.

The Fig. 4 illustrates the general TOGA framework for the top-down specification of the problems related to abstract or real goal-oriented systems.
The conceptualizations of the problem specification , on every GL level is verified by its confrontation with the existent data (i.e. external information related to the currently constructed GL) and with problem attributes specified on the higher GL level.

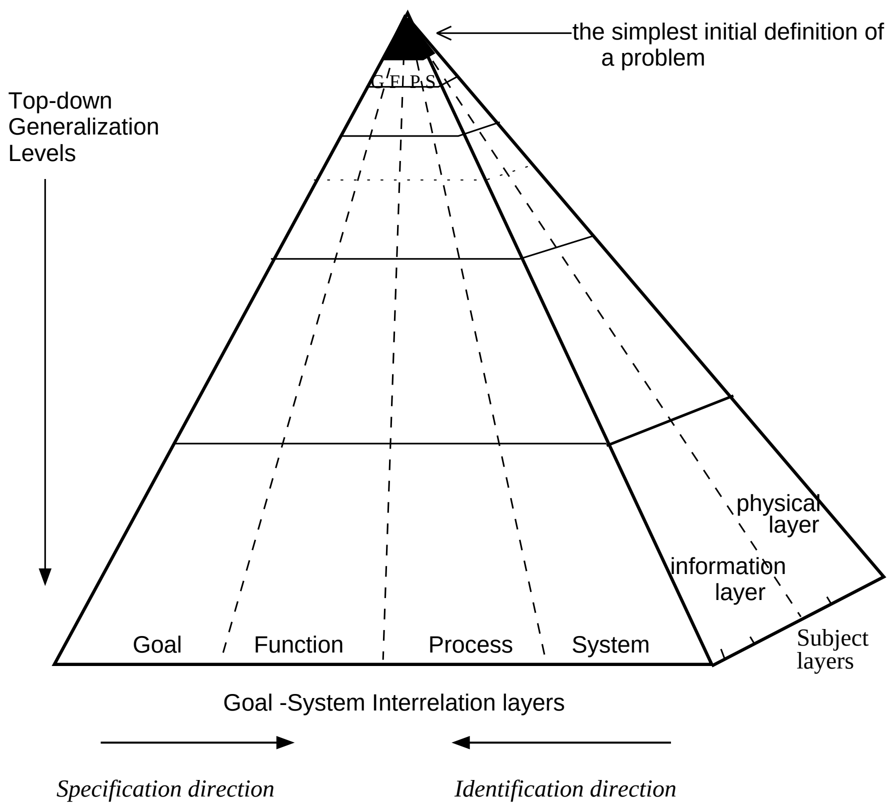

Fig. 4 The TOGA framework for the representation of problems related to goal-oriented systems.

3 S-A in TAO

3.1 Formal elements of the TAO framework

Let $\boldsymbol{Z}$ is a set called *dictionary*.

***Abstract object* concept**

Any form composed of an object name *'Q'* and the set of ordered couples is called *'abstract object'*, AO :

$$Q: ('Q', \boldsymbol{A}) \quad or \quad Q(\boldsymbol{A}),$$

where $\boldsymbol{A}: \{('a', v)\}, \quad a: ('a', v),$

and $'Q', 'a' \in \boldsymbol{Z}$,
v denotes value of attribute a in a certain measured space.

Let a formula $B =_U u$ denotes that *something* called B has a value u in a space U .

In consequence any abstract object can be also represented as follows:

$Q =_Z 'Q'$, where $'Q'$ is the name of Q , and $'Q' \in \boldsymbol{Z}$,

$Q =_\Phi \boldsymbol{A}$, where $\boldsymbol{A} \in \Phi$, and Φ is an element of the set of subsets of $\boldsymbol{Z}$,

$\boldsymbol{A} =_{\boldsymbol{V}} \underline{v}$ where $\underline{v} \in V_1 \times V_2 \times \dots V_n$

and V_j is any strongly or weakly ordered space.

$\boldsymbol{A} =_\alpha \{'a'\}$, where $\alpha \subset \boldsymbol{Z}$.

In this manner AIA can represent every abstract object on 3 before noticed levels, OL, as follows:
OL1- *object level*, Here, AIA knows only that an object with name *'Q'* exists:

$$Q =_\Psi 'Q' , \quad \text{where } \Psi \text{ is a subset of } \boldsymbol{Z} .$$

OL2 - *attribute level*, $Q =_\alpha \{'a'\}$,

OL3 - *value level*, $Q =_{\alpha \times \boldsymbol{V}} \{'a', v\}$,

here also $a =_V v$.

Usually , in the case of the attribute value we write simpler : a = v .

O-equivalence concept

Every concept which was recognized as AO , must be representable in $\{ 'Q' \}_{\times A}$.

Two objects can be *o-equivalent* on object , attribute, or value levels .

For example, on attribute level:

$$[\ Q_1(A) ==_\alpha Q_2(B)\ ] \text{ ----> } [\ \{'a'\} =_\alpha \{'b'\}\ ] \ == \ [A =_Z B],$$

where $==_\alpha$ denotes o-equivalence in a α set.

$==$ denotes equivalence on the formal representation level in TAO, i.e. the same information represented in different manner.

.

In general, two objects can be *o-equivalent* on all OLs.

Abstract relation concept

Every abstract object must be (not exist without) an abstract relation with another abstract object.

Axiom 1

Elementary unit of TAO is a couple of AOs:

$[\ Q_1,Q_2]$: $[Q1(A),\ Q2(B)]$

linked with a mathematical (logical) expression *R(a,b),*
where *'a' (- A, 'b' (- B,*
and *'R'* is *'name of relation of R(a,b)'.*

Axiom 2

Domain of TAO operations is a set of abstract objects linked by abstract relations, AR.

Axiom 3

The TAO user creates and modifies AOs and ARs constructing *worlds of objects, W*

$W: \quad [\{Q\},\{R\}].$

Axiom 4

a is an attribute of Q in W iff $\bigvee_{R\,(-\,\{R\}_W} R(a, .\,)$.

Definition of *system*

Every arbitrarily chosen set of AO, *{Q}*, where all its elements are linked by *{R}* is called *system* . It is denoted on the OL1 by

$S: \ [\{Q\},\{R\}\,]$ or $S(\{Q\},\{R\}\,),$

where *{R}* is called *structure.*

If some attributes of ***A*** of an object *Q , {A'},* are linked together by some relations *{R'}*

then the system $S: \ [\{A'\},\{R'\}]$ is called *property* of the *Q*.

For example, for $Q\,(a_1, a_2, a_3, a_4)$, if $a_4 =_V f(a_1, a_2, a_3)$ then a_4 is a *property* of Q, and

$\{a_1, a_2, a_3\}$ are a *base* of *Q properties*.

Developing this approach we can define *o-similarity* concept which depends on assumed *p-o-v,* and *o-regularity***.**
O-regularity is defined for internal properties of objects and for interrelations between objects.

3.2 General S-A procedure in TAO

S-A can be performed on the different deepness of *object levels* (OL) of currently available information about w-o-o according to arbitrarily established *influence filter* index , $\boldsymbol{I_f}$.

Influence filter index , it has the values 1 or 0, and it is established outside of the analyzed w-o-o.

1. The OL1 level is the object-relation-change level,

$W: \quad (\{O\},\{R\},\{C\}),$ with $\boldsymbol{I_f}(o_i),\ \boldsymbol{I_f}(r_k),\ \boldsymbol{I_f}(c_l)$.

2. The OL2 level refers to the existence of object attributes,

$O_i \ \{A\},$ with $\boldsymbol{I_f}(a_{i,j})$.

3. The OL3 level refers to the attributes values.

Here $I_f([v_a,v_b]_{i,j}$, where $[v_a,v_b]_j$ is the interval of j-th attribute value.

Formal objects (*f-objects*) defined on a lower OL have values on the next higher OL

For example the value of f-object O_i is the set of its attributes name

$$O_i =_Z \{'a'\}.$$

S-A is performed from the different of points-of view defined as an element of w-o-o architecture.
S-A is performed on all OL levels.

S-A is shifted down in OL hierarchy if the i/k on the previous one was not sufficient for *stop criterium.*

In this manner, all elements of d-o-a with $I_f = 1$ give new w-o-o which represent situation, it is a *script* in the object conceptualization..

Part II S-A Procedural Model

1 S - A goals and S-A functions

In this paper we talk on only goal-oriented activity of IA , therefore, for simplicity, the term *activity* used in this text means always *a goal-oriented activity of an intelligent agent.*

1.1 Why S-A ?

Every AIA must monitor state of its world from 3 basic *perspectives (multi-object p-o-v)*:

S_g - safety (min. of risk , hazards and losses for agents & environment),

P_g - production (realization of intervention goals),

E_g - economics (min. of effort/ energy, materials, time, tools consuming, etc.).

Therefore these three group of criteria applied to the agent knowledge and information are basic goals of S-A (top-frame design goals), see the Fig. 5 .

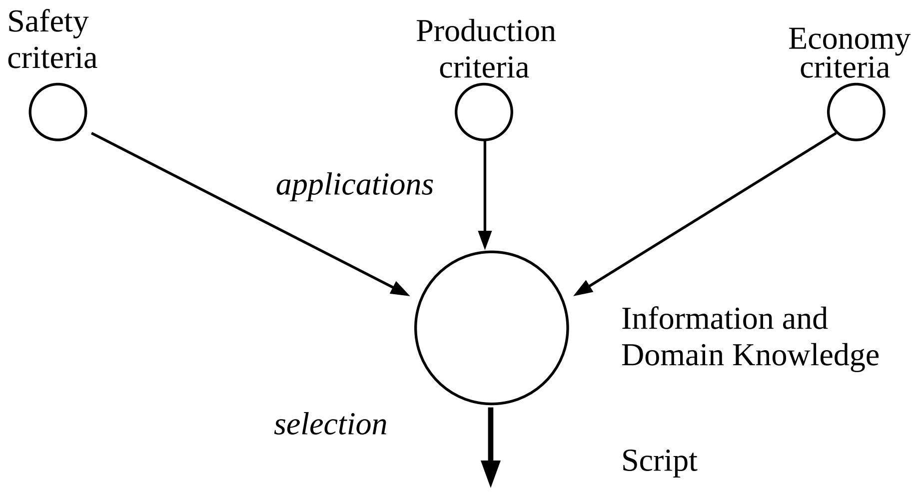

Fig. 5 Generation of the goal-dependent script**.**

For any particular case these three goals must always be established.

During intervention planning, from the one of the above perspectives other two goals are sources of constrains.

The goal P_g under constrains S_g and E_g is called *Global Goal,* GG.

In general , any specific S-A goal must be specify according to the goal frame [Gadomski, 88, 92].
Specification of the S-A goal can be represented as a w-o-o in terms of the d-o-a specification and constrains external to it .

The d-o-a of S-A-maker (D_{s-a}) includes :

$D1_{s-a}$ - information source, i.e. AIA physical d-o-a , D_{Ph},

$D2_{s-a}$- AIA abstract d-o-a, where *script* is constructed, D_A,

$D3_{s-a}$- AIA preferences which it can manipulate, P.

$D4_{s-a}$ - AIA current descriptive knowledge, Kd, which it can manipulate.

$D5_{s-a}$ - AIA current operational knowledge, Ko, which it can manipulate.

All invariants and not modificable preferences and knowledge are called *AIA firmware*.

S-A should contain methodology and strategies for hierarchical access to information, knowledge and preferences, necessary for the selection data and script construction . This access should be managed according to the p-o-v of temporal risk evaluation. In other words**,** AIA have to have criteria for the establishing deepness (detailness) of script related to the current states of D_{Ph} according to the following priorities:

1.- recognize possible extreme negative consequences of the s(D_{Ph}); S_g driven
2.- recognize other risk sources ; S_g driven
3.- recognize goal connected elements of of D_{Ph}: P_g driven ,
4.- recognize the elements of D_{Ph} connected with goal quality index; E_g driven ,

1.2 S-A Functions

S-A is one of top internal functions of reasoning of autonomous agents.

There are three such functions: 1- goal establishment, G-E,
2- situation assessment, S-A
3- intervention planning. PLAN.

All of them can be divided on two subfunctions (a and b):

1a. conceptualization of "what is required" for AIA;
1b. choice of a domain of activity where a particular state will cause "what is required". This state is called *intervention goal*;

2a. identification of current state of the domain of activity;
2b. recognition of possible operations, acts and actions in the context of constructed image of the domain of activity;

3a. *task* planning, it means, a specification which actions are needed/necessary for goal achieving (in terms of d-o-a description performed before).
3b. *action* planning, it also takes under consideration the results of 2b.

Whole reasoning process requires activation of these functions top-down on different generalization levels, and in a different order.

The S-A function prepare script which must include data :
a) for activation of the procedure of the action existed in the agent operational KB (knowledge system) and sufficient for direct achieving of the Global Goal, or
b) for the next S-A in anther d-o-a, or
c) for tasks and actions planning.

Unfortunately, the specification of a domain independent planning function not yet exist in the subject matter literature.

S-A can be performed before and after goal establishment. The second case is illustrated on the Fig. 6.

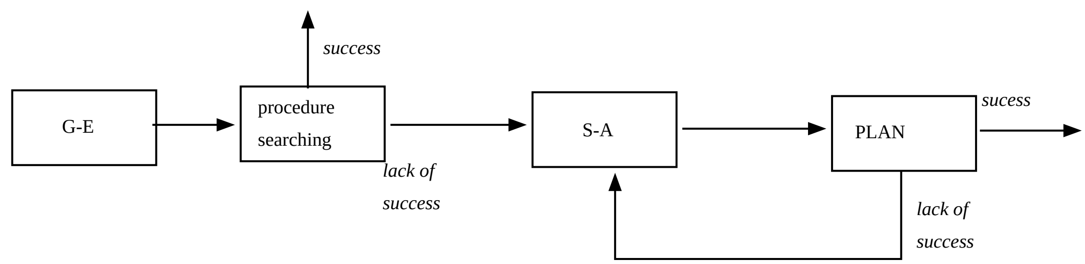

Fig. 6

Functional Procedures

SAF - S-A function , can be presented as a sequence of activated procedures.
It is repetitive and recursive.

SAF must be realizable by *S-A Functional Procedure,* SFP, which is composed of the following partial functional procedures :

IAP - activation and information acquisition procedure ,
IFP - information filtering procedure, realized from GG point of view and and according to planning possibilities (procedures and strategies), Ko..
KBP - Kd building procedure for the script,
SCP - stop criterium procedure,

SFP includes the following operations:
- choice (searching),
- algorithms ,
- rules,

and use all are conceptualized in TAO.

SFP is realized by the following sequential structure :

$$SFP_i \text{ -> } (SFP_{i+1} \textit{ or } PLANP), \quad \text{for } i=1, 2,3,... \qquad (1)$$

where : PLANP is the planning functional procedure, and

$$SFP : \quad IAP(D1_{s\text{-}a}) \text{ -> } IFP(\textit{inf}, D2_{s\text{-}a}, GG, PKo) \text{ -> } KBP(D2_{s\text{-}a}) \text{ -> } SCP \text{ (indexes)}, \qquad (2)$$

and : *inf* - one information portion,
PKo - available planning rules and procedures,
indexes - integrated attributes of *script*.

Decomposition of D_{s-a} on subdomains according to a *decomposition criterium* , $Crit_d$, which preserve the D_{s-a} basic structure, causes recursive application of SFP inside of the functional procedure where the *decomposition criterium* was applied.

In this manner SPF of the higher level depends on locally performed SPF:

$$SFP_{i+1}(SFP^1, ...,SFP^n). \qquad (3)$$

Every partial functional procedure is governed by four indexes: status index, performance index, quality index , and efficacy index.
The quality index depends on the choice of processes which are carriers of SAF function..

1. Allocation of S-A function

Allocation of the S-A function depends on the assumed architecture of intelligent agent, and on specific application domains of intelligent system. In a centralized intelligence architecture, a intelligent reasoning module (IRM) governs all, " less intelligent" functional modules.

On the next two figures, 7, 8 this idea is illustrated in the cases of a mobile autonomous robot, and menagerial decision support system.

2 S-A process

According to TOGA, from the specification perspective, a process is a realization of a function in a concret domain . The S-A process is performed by AIA and depends on its possibilities and constrains.

Which states of IA activate S-A process?

S-A is activated by the AIA carrier system, CIA [Gadomski,93a] when:

- an *intervention goal*, G_I was generated by *preference system* in the triangle of *Abstract Agent*, and
- the *knowledge system* requires "facts" (information) for its activations.

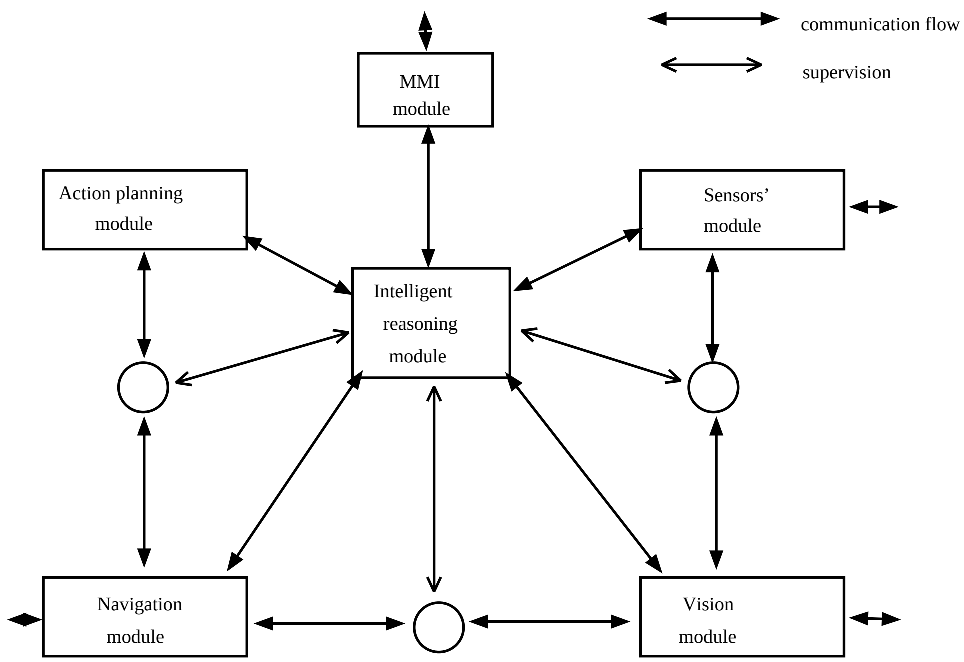

Fig. 7 An example of the central functional architecture of a mobile intelligent robot.

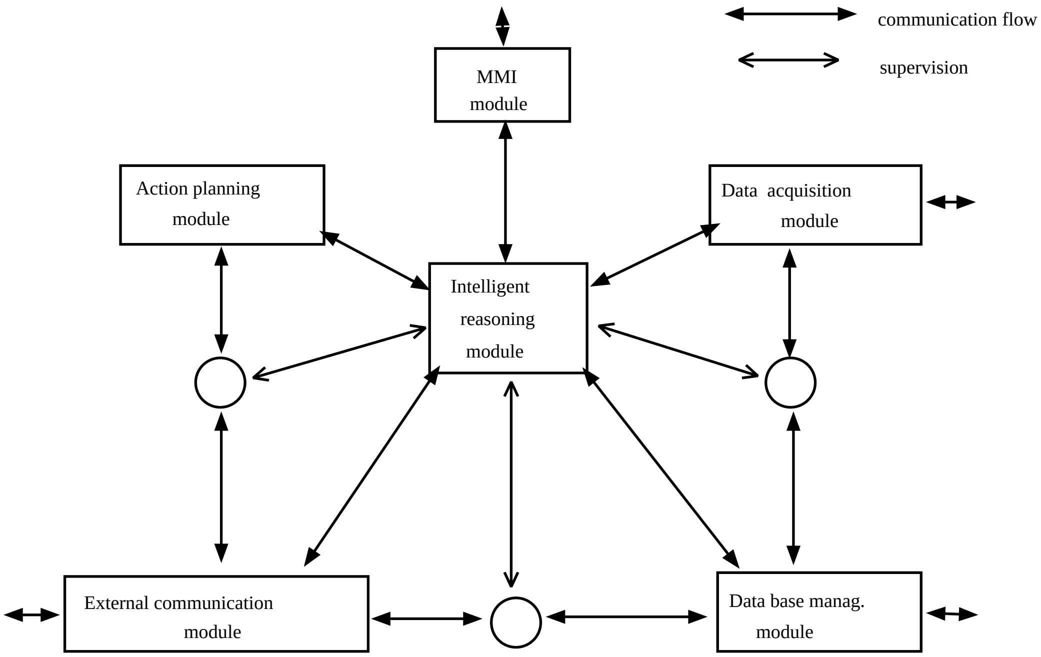

Fig. 8 An example of the central functional architecture of a managerial decision-support system.

2.1 Top-procedure of S-A

$$st \;=\; \underline{\mathbf{ASS}}\; \mathbf{[}\; \underline{\mathbf{CR}},\; G_{s\text{-}a},\; P_{s\text{-}a}, K_{s\text{-}a}\; \mathbf{]}\; \{\, G_I\, ,\; inf\,(\, D_{s\text{-}a})\; \mathbf{,}\; P,\, Kd,\; Ko\}, \qquad (4)$$

where : *st* is a *script* necessary for a GG,

ASS is a S-A operator in $D_{s\text{-}a}$,

,

CR is a conceptualization operator realized by the CONCEPTUALIZER system,

$G_{s\text{-}a}$ is a initial meta-goal of S-A,

G_I is a specification of $\mathbf{s(D_A)}$ which is required to be achived by AIA (it can be an external task for the agent),

s(**D**) is a specification of the state of the AIA domain of activity, **D**,

$K_{s\text{-}a}$ is a S-A knowledge in the form of the set of triples : **(pn, s*i* , s*j*)** , **pn** is a name of the procedure **,** and **s*i*, s*j*** are possible states of **D** .

$P_{s\text{-}a}$ is the preference subsystem necessary for S-A.

The top-procedure of S-A is a mandatory frame for the next decompositions which have to include functional requirements frames (1), (2), (3).

The next decomposition must take under consideration temporal, and architectional (structural) constrains of AIA.

S-A can be continued up to a situation identification , i.e. up to predictive model building. (situation recognition is sub-class of sit-ident.)

3. Situation Assessment Scenario

Let us denote: $\mathbf{s(P_{ij}, t)}$ - a state of the preference system $\mathbf{P_{ij}}$, in a moment **t,**
i - a number of metalevel in AIA architecture, **j-** a number of **P** on i-level.

$\mathbf{s(K_{ij}, t)}$ - a state of the knowledge system $\mathbf{K_{ij}}$ in moment **t,**
$\mathbf{s(D_{ij}, t)}$ - a state of the domain of activity $\mathbf{D_{ij}}$ in moment **t.**

Taking under consideration the repetitive character S-A and above introduced symbols, the operaror **ASS** can be decomposed on the iterative inference procedures .

I Phase of S-A

1- Acquisition of informations about objects from $D1_{s-a}$ i.e. object-oriented acquisition.
a) registrated variables(attributes) ---> recognition of objects ----> recognitions of the relations between objects (according to the available procedures or knowledge base systems) ---> goal-oriented filtering procedure,
and if exist,
b) registrated image ---> recognition of forms (pattern matching) --->
---> goal-oriented filtering procedure , as presented in the paragraph 3.2.
---> in $D2_{s-a}$, construction of the *domain script (D-script)*;

2 - "What if " analysis, i.e. S-A in time. This step includes simulation, and risk assessment procedures. Risk factor is calculated for extrapolated behaviour of objects and relations in D-script.
Constrains for strategies for Sit-Ass:
1. Risk sources (objects or relations or status, o/r/s) ---> losses o/r/s
2. Hazard sources (objects or relations or status, o/r/s) ---> losses o/r/s
3. Benefits, the monitoring of
production status,
economy status.

3 - Diagnostic procedures , if an abnormal state was dedected/discovered.
Data: abnormal variables in D-script.
Procedure: Model-based discovery of invariant abnormal attributes
----> their allocation to D-script objects.

4 - If no abnormal/dangerous state is signalized then
a) Reasoning resources assessment procedure;
searching of the goal related procedural and rule based knowledge
----> creation *knowledge-script ,K-script,* i .e. the knowledge base selected for a next phase of reasoning.
b) Criteria assessment procedure;
searching of the goal related preferences ---> construction *preference-script, P-script.*

II Phase of S-A

1 - Checking of the adequatness and congruence between D-script, K-script, and P-script , specification of common links between their ws-o-o.
Searching incomplet or black-box objects.
For example, searching of the states where common objects exist but they are represented ; on the different OLs.
If OK then exit from S-A.
2 - Choice of a new strategy for information acquisition.
3 - Go to information acquisition.

The Figures 9, 10 illustrate the data-procedure schemes.

S-A of an agent can be direct and indirect.

Direct S-A is when the domain of activity of the agent is its end-intervention domain , for ex. a simple (procedural) robot.

Indirect S-A is when the d-o-a of agent is conceptually linked by cause/concequences relation with the end-intervention domain.

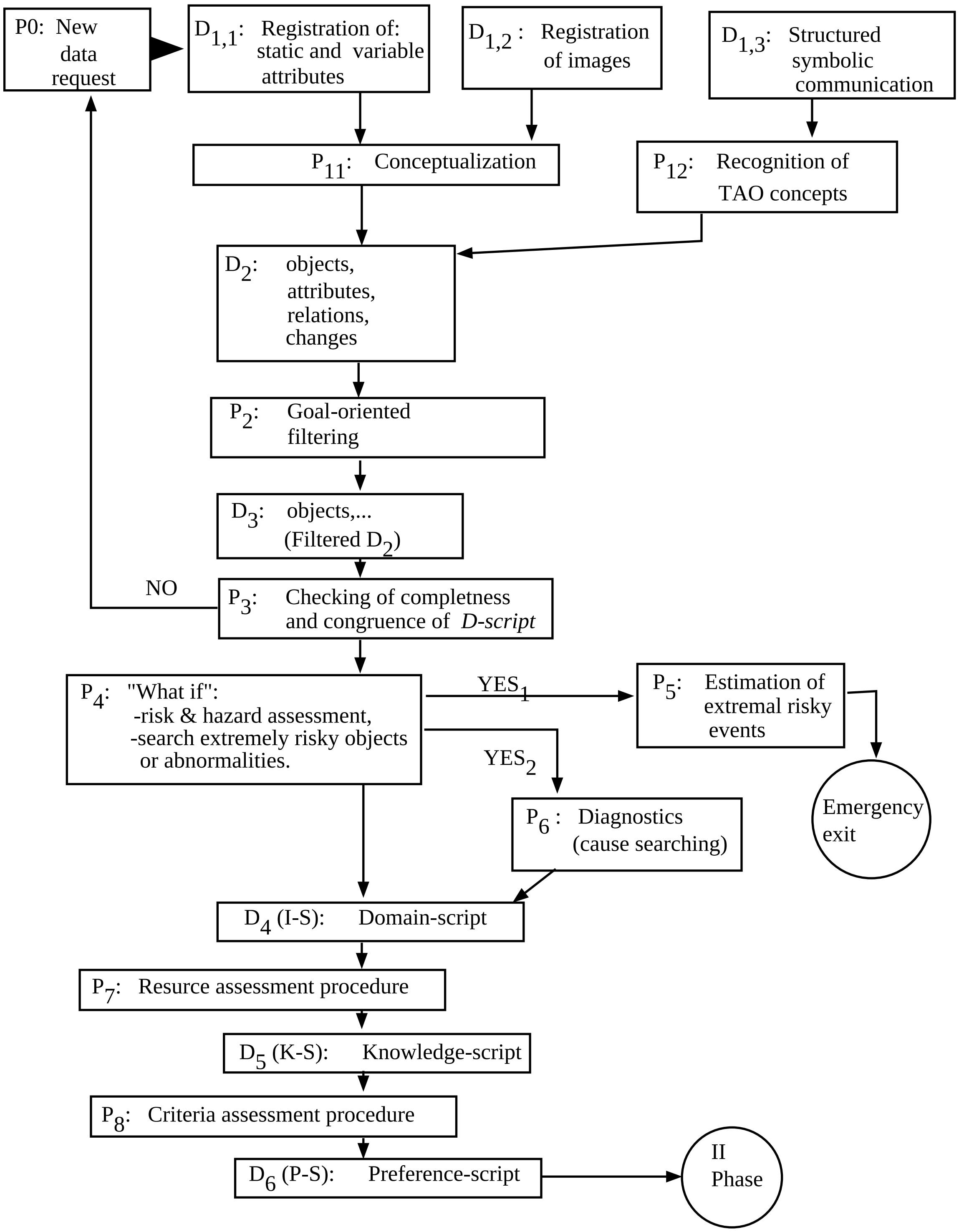

Fig. 8 I Phase of S-A

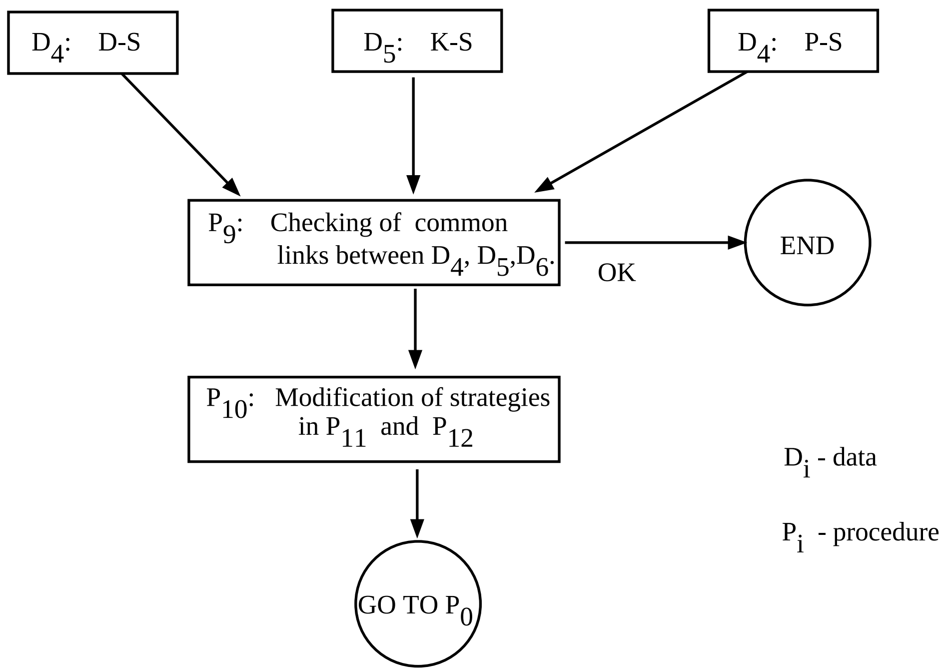

Fig. 9 II Phase of S-A

For ex.: - a task planning module is an intermediate system which prepares data for the action planning module (another agent).

The next (deeper) specification step requires the definition of the data for S-A.

4. Data for S-A

Invariants of S-A :
initial data - a. goal of end-intervention
b. domain-knowledge (knowledge on the end-interv-domain)
c. agent intervention-tools in its own interv-domain.

input data perception abilities
- information acquisition (on/without request) according to the map of the domain of activity.

Def.
A *map* is a domain-dependent conceptual frame which enables localisation of information sources.
It is an abstract network which only in particular cases is a geographical map.

Goal : $s_g(D)$

It can be specified on different level of generalization (GL) and different OL.

Initial state of D : $s_I (D, t)$ */ unknown /*

Knowledge
Primitives of D : {E} , {R} ; basic elements of D and possible interrelations,
Maps of D : {M} ; some invariant elements and substructures of D
Changes on D : {CH } ; possible changes/events in D.
Operations on D: {o} ; possible operations on elements and structures of D .

Preferences
a. They includes relations to safety, production and economy
b. Which elements are more important.
c. Which act/action is better.
Ad. b.
Absolute preferences are highly abstract and are unconditional.
.
We distinguish:
Frame-dependent preferences ,
Goal-dependent preferences,
Context-dependent preferences: its refers to a particular situation in D.

Preferences related to the D:
- Hot situations : specific objects , states or situations which lead (without A. intervention) to the Emergency situation.
- Emergency situation: a losses generation process (i.e. reduction of agent operationability)

Preferences related to the agent are the following:

1) - to achieve $s_g(D)$,
2) - time limit: after time $t < T_{max}$
3) - effort limit: $E < E_{max}$
4) - to avoid emergency states (unconditional)
5) - to minimalise effort (if not against 1),2),3))
6) - to minimalise time (if not against 1),2),3), 4))

Knowledge structure which enable reasoning in frame of assumed conceptual network.
Reasoning (multistep process) is a "navigation" over conceptual networks.
Reasoning steps are done by : - ***association*** (max assoc-index in Ass -Matrix)

c_1 -------------> c_2

- ***inferencing***

if c_1 then c_2

procedural or *rule based* when adequated rules are in KB

- ***searching & choice***

i.e. to take x from X , and to check maching attribute.

Examples: If the *varianble 'fuel level' is less than 1* then *parameter 'red light' is on.*

If the *last action was A* then *do B.*

If *a=1 and b=2* then *execute procedure X(a,b, result)* .

5 S-A in physical d-o-a

The environment of an IA is classified relatively to the possibility of symbolic information processing or management, and according to the presence of other Intelligent Agents.
In this way, we distinguish four disjointed types of environments:

- ***No Agent En,*** (E/NA); environments without any agents;
- ***No Intelligent Agent En,*** (E/NIA);
- ***Only Intelligent Agents En***, (E/OIA); without organizations of IAs;
- ***Multi-Intelligent En***, (E/MI).

The above classification enables the following top-configuration patterns for the conceptualization of the environments [gadomski]:

IA interactions: / En type	Observation	Intervention		
		Modification	Creation	Commu
E/NA	+	+	+	-
E/NIA	+	+	+	+
E/OIA	+	+	+	+
E/MI	+	+	+	+
E/(NA, NIA, ...)	+	+	+	+

The Fig. 10 illustrates some simple representation patterns.

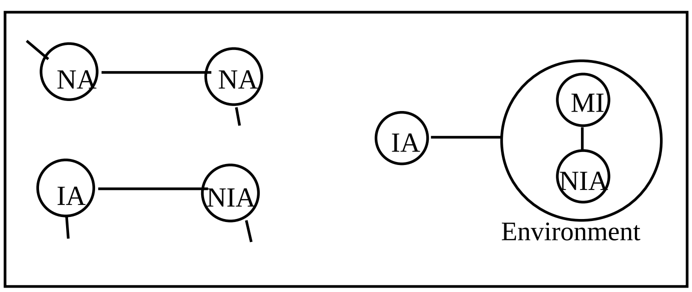

Fig. 10 Examples of the IAW representation patterns.

Examples referred to man-machine systems, car driver teaching, and to an intelligent robot in human and other intelligent robots environment are presented in Fig. 11. Of course, different problems can be formulated in the same domain of objects represented by one GRAF picture.

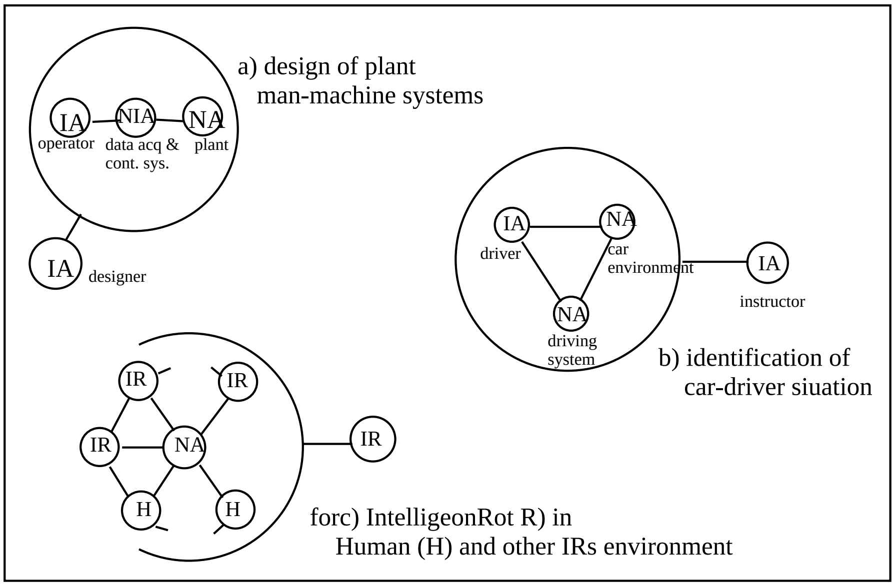

Fig. 11. Examples of some applications of the GRAF representation.

- a management requires S-A on different domains of activity of different direct task executors (agents).

In general, S-A is performed in an autonomous way by preselected agent in its (direct) d-o-a which can be related to (it can be called "control domain" of) another domain end-intervention domain where the intervention goal is localized.

S-A requires a reasoning mechanisms about all objects in an interactions and which are linked with a.,b.,c. .

Simple Example

Intervention-goal of the robot R with d-o-a D

Assumption: X is fixed element into D(OL1).

(in unformalized context) To enter into a room called X,

(in formal context)

pR denotes the position of R ,
pR is a common attribute of in the relation .r. with X, i.e. in a symbolic form

pR .r. X

.r. can be into two states (1) " to_be_outside"
(2) " to_be_inside"

6 Examples of specific applications

6.1 Elementary S-A procedure for reactive agents

1st step.
Repetitive recognition of *something* (attribute, attribute-value, object, change, ..) in d-o-a. This recognition can be *passive*, for ex. a registration of the value of the attribute fixed by agent input channel, or *active* , for ex. performed by dynamic perceptors' system of a robot.

2nd step.
Searching of a direct response in the form of intervention procedure or rule.
If such response not exists then
  if the agent is *autonomous* no output is appear,
  if agent is *not autonomous* then it inform its user that a recognition without response was done.
else intervention is executed.
An agent only with 1 and 2 step can be called *skill-reactive agent.*

Skill-reactive agent has only one absolute preference, it must react..

3th step.
is executed by *rule-reactive agent**,** RRA**,*** if in the *skill knowledge system* the response not exist.

Here the preference system can be activated if more then two somethings are recognized.

We have now to distinguish *RRA autonomous* and *RRA not autonomous.*

Autonomous RRA has a long term memory which can include:
M1. previously registrated somethings,
M2. *general knowledge (frame knowledge)* in the knowledge system
M3. information on previous actions, i.e. precedent executed interventions,
M4; Some combination of the M1, M2, M3.
Not autonomous RRA has case-oriented temporal memory, TM.
Not autonomous needs the repetion of the 1st step for activation of the preference system.

Here, the influence filter must be established in sequence:

M1 -> M2 -> M3.

6.2 Robot navigation with map
One of the most important elements in S-A is a specification of the S-A criteria.
In this paragraph we present an example related to the robot navigation in unstructured environment with descriptive knowledge in the form of geographical map.

<u>Intervention Goal</u>: to find the trajectory from A to B.

Possibilities : p1. available criteria for ’trajectory intervale’ construction

p2. available procedure for discovery of unovercome terrains

p3. available criteria for recognizing inacceptable high risk zone.

Goal constrains (for trajectory construction)

c1. required minimalization of the energy used, E

c2. required minimalization of time T of the navigation

c1. required minimalization of the risk in time, R.

First step: procedure schema of the modification of $D2_{s-a}$ is presented on the Fig.12 .

Second step: Searching of the criteria for c1,c2, c3. and the validation of K-script and P-script is illustrated on the Fig. 13.

The thirt step (equivalent of IIphase of S-A) is related to the confrontation if the attributes of objects in $D2_{s-a}$ are adequated to the attributes of rules, procedures and preferences in $D2_{s-a}$, $D3_{s-a}$.

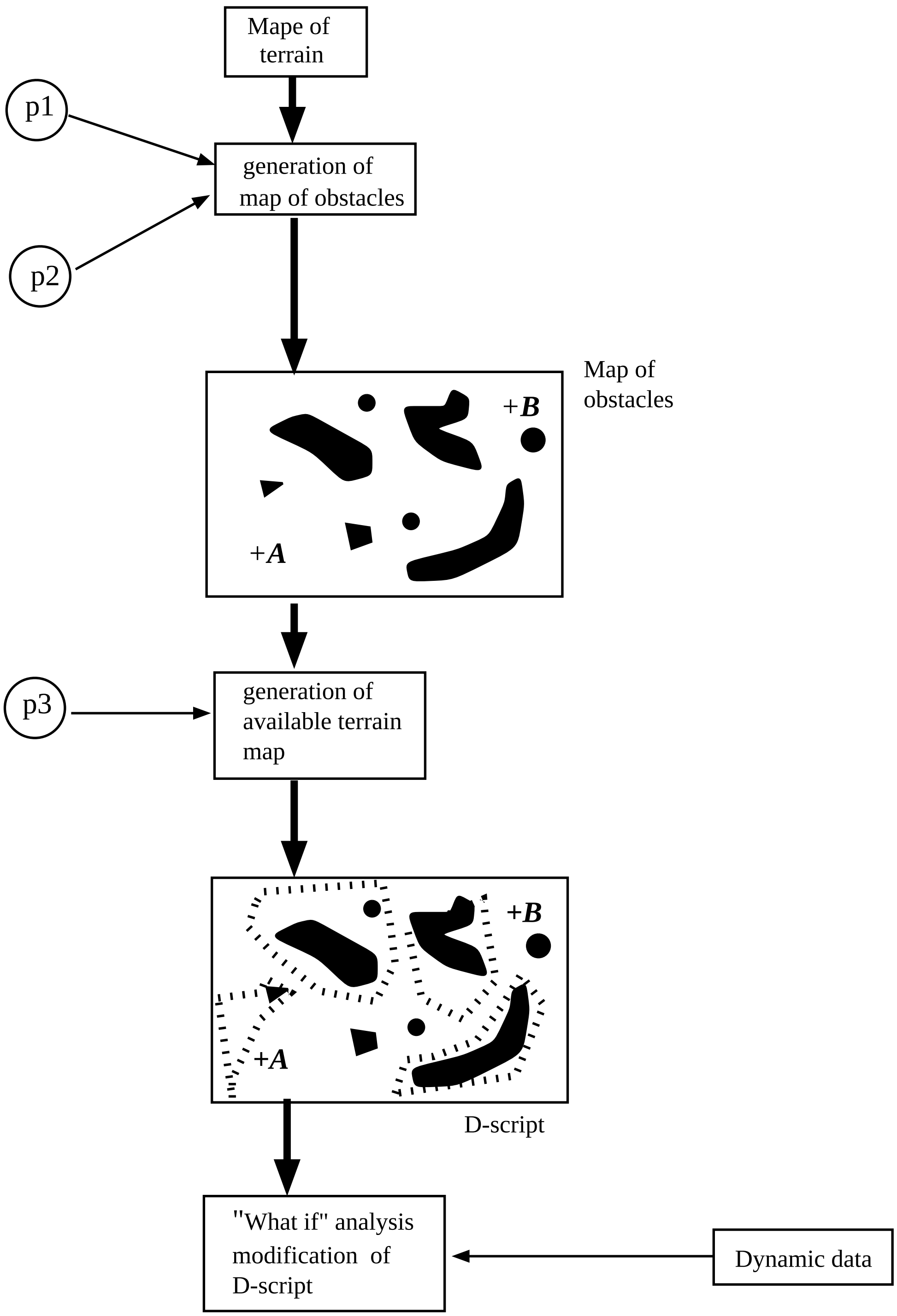

Fig 12 Recognition of the available terrain

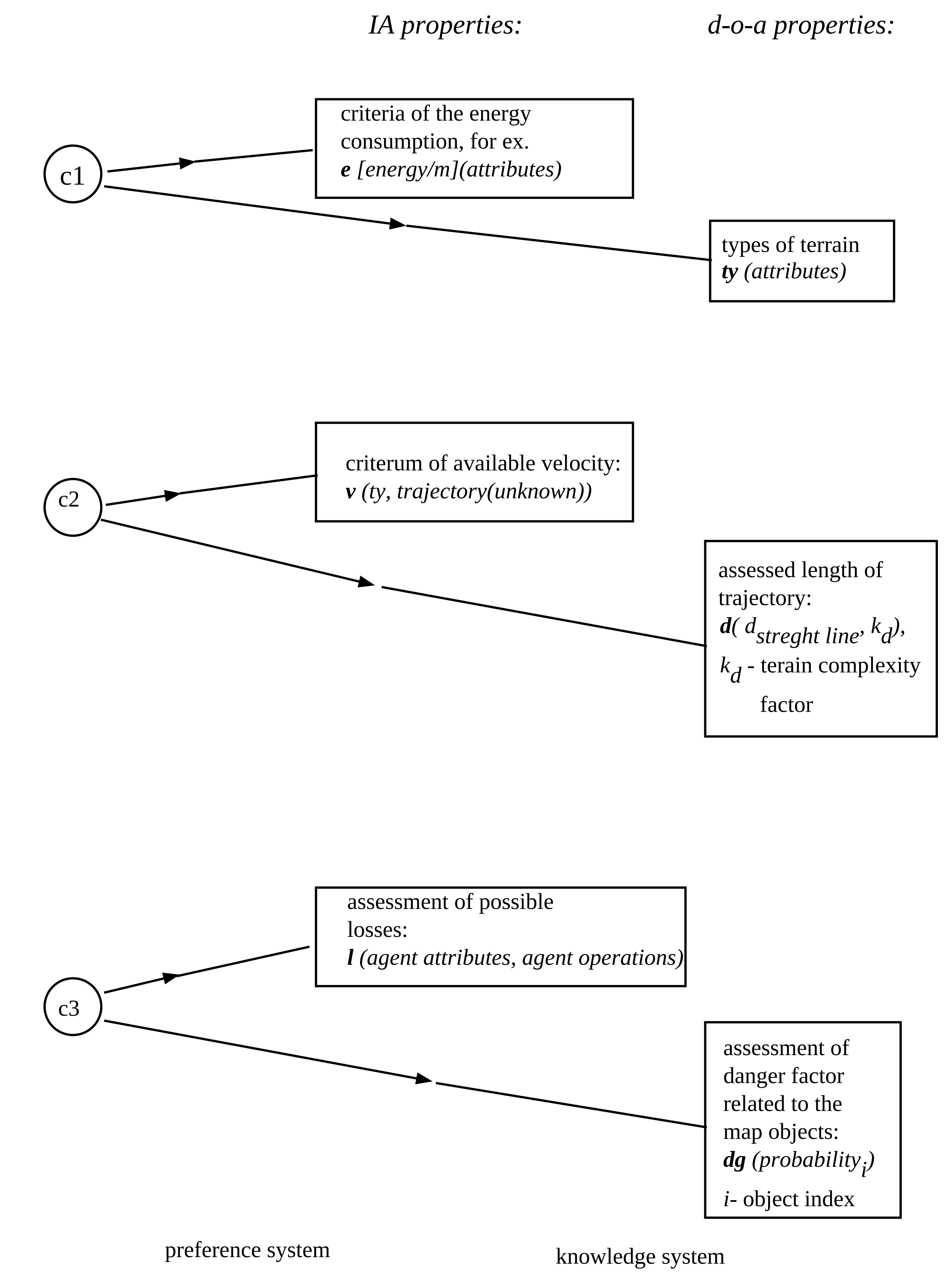

Fig.13 Criteria searching, validation of preferences and knowledge. ——▶— denotes *is necessary*.

References

1. A.M. Gadomski: Information about System-Process-Goal approach to diagnostics and supervisory of nuclear reactor systems, presented on " 19th Informal meeting on Reactor Noise", ENEA,Rome, (1986).
2. A.M. Gadomski: An application of System-Process-Goal approach to the TRIGA RC-1 reactor system description. In : Proceedings of 9th European TRIGA Users Conf., Rome, (1986). Also the ENEA's Report RT/TIB/88/2 (1988).
3. A.M.Gadomski, V.Nanni: Design criteria of TRIGA RC-1 supervisory system: System-Process-Goal (SPG) Approach, the ENEA Report, RTI/TIB (88) 12, (1988).
4. A.M. Gadomski, C. Moriconi: Knowledge representation and some methodological aspects of functional specification of intelligent mobile modular robots. ENEA Internal Report , RIA/TIB/88/01 (1988) .
5. A.M.Gadomski: From Know-How to How-to-Know : An approach to knowledge ordering for Specification of complex problems (TOGA methodology). Materials of The International Symposium on Computational Intelligence. Milan,Italy (1989).
6. A.M. Gadomski, M.Gadomska: Environmental and emergency communication and decision: confrontation of shallow models from the point of view of computer-support designing. Materials of The Second SRA-Europe Conference, IIASA, Laxemburg, Austria (1990).
7. A.M.Gadomski: A model of action-oriented decision making process: Methodological approach. In: Proceedings of Ninth European Annual Conference on Human Decision-Making and Manual Control. Varese, Italy (1990).
8. A.M. Gadomski, V.Nanni: Problems of knowledge about knowledge: An approach to the understanding of knowledge conceptualization, Proceedings of The World Congress on Expert Systems, Orlando, Florida (1991).
9. A.M.Gadomski: Intelligent agents and their worlds. (The document of ENEA/INN Informal series "Research notes on Abstract IntelligentAgent", N.3). RTI/INN(92)5, (1992).
10. A.M. Gadomski, V.Nanni: Intelligent computer aid for operators: TOGA based conceptual framework. In: H.Adeli (ed.), Proceedings of ICARCV'92 "Second International Conference On Automation, Robotics, and Computer Vision", Singapore (1992). Submitted for publication in Journal of ICAE.
11. A.M.Gadomski, V.Nanni: Architectures of intelligent agents' worlds: Top-down Object-based Goal-oriented Approach, Proceedings of Workshop AIIA on:"Intelligenza Artificiale Distribuita", Roma (1992).
12. A.M.Gadomski, V.Nanni: La ripresentazione integrata di sistemo artificiali complessi: L'approccio System-Process-Goal (SPG). The ENEA report RTI/INN(92)10, 1992.
13. A.M.Gadomski (ed.): Proceedings of AIA93: First International Round Table on Abstract Intelligent Agent. Rome: Edited by ENEA , In printing .
14. M. Gadomska: Technology and society: Impact and adjustment perspectives. Rome: Edited by ENEA Direzione Studi, Qs. N;17 (1993).
15. A.M.Gadomski, V.Nanni, S.Taglienti: Some Theoretical and Practical Aspects of Modeling of abstract Intelligent agent: ENEA 's Experiences, In: Proseedings of the second symposium on intelligent information systems, in Augustow, Poland, June (1993).