Risk Management Systems Must Provide Automatic Decisions According to Crisis Computable Algebras

Olivier Bartheye
CREC St-Cyr
Écoles de Saint-Cyr Coëtquidan
F–56381 Guer Cedex, France
olivier.bartheye@st-cyr.terre-net.defense.gouv.fr

Laurent Chaudron
ONERA
BA 701
F–13661 Salon-de-Provence, France
laurent.chaudron@polytechnique.org

Abstract

Is it nowadays our responsibility to convince our contemporary citizens that AI devices as UGVs (Unmanned Ground Vehicles) and UAVs (Unmanned Aerial Vehicles) are crucial actors of today’s life in a dual domains, both civilian and military. In particular, the decision process is the main component of every military operation and is of high interest because of two main reasons: it is necessary designed to cope with conflict issues and it requires a very complex planning process to be successful. The difficulty to find a good plan is worse in urban areas because of the high uncertainty due to the topology of these areas, the presence of civilians, who can be hostile or friendly, and the unpredictable nature of enemies. The idea in that paper is to qualify what can be a valid computed plan in that context, i.e. well-designed for recovering of peace, rescue operations after a bombing event, hostage salvage, non-combatant evacuation operations, civil-military co-operation, . . . , in urban areas.

This planning process leads to associate actually four components, the representation of the tactical scheme, the implementation of the tactical scheme as the behaviour of special forces, military units or emergency squads, the proof process or the explanation process, and finally the handling of external factors depending on the current environment or the current context in which the operation takes place.

This paper uses a quaternary representation called the epistemological quadriptych, in order to highlight that the integration of UGVs or UAVs devices requires actually to understand the role of knowledge and behaviour and to provide secure and valid action plans, i.e. which can be explained and justified.

1 Introduction

Robotics and particularly UAVs and UGVs are considered by tactic specialists as very suspicious devices for which it seems hazardous to give some real executive power (CREC SAINT-CYR 2013). This is certainly due to an insufficient modelling work from these officers and engineers who are totally refractory to include fully these devices in the battlefield. A major reason is that robot introduction during a military operation and more generally AI introduction will necessary shake up their tactical status whereas their expertise is built upon the very delimited knowledge they have on the tactical domain. This refractory attitude is clearly comparable to commercial airline pilots decision to reject being replaced by AI despite the loss of all aboard several commercial airliners in recent years. Performing a transition from the current knowledge they have to this new era, in which AI will be a common today’s life component requires to consider the computational capabilities. This paper aims to show that the necessary and sufficient condition is to trust on the so-called planning calculus.

The sterile issue, very closed to the delirium around the existential threat posed by AI raised by Stephen Hawkins or Bill Gates, is always the following: “should one give them the ability to fire ?” whereas the answer is obviously yes once they become active support for special forces or emergency squads. In particular, the aim of this paper is to qualify AI devices action plans well-designed for recovering of peace, rescue operations after a bombing event, hostage salvage, non-combatant evacuation operations, civil-military co-operation, . . . , in urban areas (CREC SAINT-CYR et al. 2014).

In effect, take a very concrete example: urban warfare (GRIMOND and HENNET 2008). In urban warfare, military operations are always very critical since every elementary action is critical: setting postures according to the threat, choose the right street angle, pass through a door, come into a room and find a safe location. Performing some of these actions by UAVs and UGVs permits to gain control with a limited number of humans exposed to enemy fire. In other words, thanks to a relevant substitution of human soldiers with these devices, the random factor during a fight is straightly controlled.

To achieve this goal, one must investigate for a qualified understanding and a qualified implementation of chief leaders orders by AI devices and this paper focuses on the modelling process of these orders according to the epistemological quadriptych. Military orders always occur in a strategic context; strategy is an implementation of a political objective (strategy is the manner in which military power should be developed and applied to achieve national objectives or those of a group of nations). According to relative means of the two opponents and the importance of the issue, a strategic plan will be declined according to several models.

Tactic is an implementation of strategy (tactic is the science and art of organizing a military force, and the tech-
niques for using weapons or military units in combination for engaging and defeating an enemy in battle). It is important to note that officers claim that “the conduct of war is an art because it is based on the perception of events and the interpretation of them associated to a free and fruitful action based on scientific evidence.” Therefore the decision process can have irrecoverable consequences and however is based on very doubtful premises (BARTHEYE and CHAUDRON L. 2015b). That’s why, distinct headquarters services: intelligence, command plans and operations center, planning and policy, information and communication systems, exercises and retex, . . . , are required to plan and to monitor a military operation.

**Doctrines** consider the power of current forces and propose the solution which corresponds to the best use of these forces. Successive situations occur after applications of selected actions; actions are triggered according to valid conditions expressed by situations. Military actions agree with the following properties:

- **Sequencing** : every system in motion performs sequentially elementary actions in a fixed or a variable order.
- **Phasing** (planning step) : this step triggers actions with the best timing in order to gain the initiative.

**Initiative** is the ability to define the action terms all along the battle or the military operation, i.e. the ability to be an initiator when no clear statement holds or when the situation changes. Gaining the initiative means to control the “next” shot.

That clearly means that one must determine the best action plan to provide according to the knowledge, expressed as **tactical models**, chief leaders can have about tactical implementation details. According to the size of the tactical operation, there can be actually a “hole” depending on the warfare fog between the tactical knowledge and its implementation in terms of elementary orders performed by soldiers.

This disconnectedness issue should be expressed according to the epistemological quadruplych and must be reduced as much as possible according to formal modelling. The aim of formal modelling is to provide better action plans and the ideal modelling process is to converge towards a unique fix-point: the best optimal plan. Since the ideal is not yet obtained, one can be interested in computation of effective suitable actions plans.

## 2 Suitable action plans in urban areas

The selected example concerns the simulation of tactical motions of a combat squad (6 soldiers) inside a urban area using a video game support. Roughly speaking the aim is to plug into video games architecture usable software components such AI action planning, sandboxes, in order to provide a feasible prototype able to manage a very high degree of uncertainty.

This game architecture provide some results in certain tactical scenarios as follows:

1. Assume that firing conditions are the same for all soldiers,
2. assume that entities receive order to fire on sight on every entity identified as an enemy,
3. assume that enemies have same combat rules than soldiers.

In all cases (100%), the first soldier get shot by enemies before any other soldier in the squad has the opportunity to fire. That means that protection of the first soldier is to be reinforced in order to survive after a direct fight before passing through the wall corner.

Using the benefit of video game simulation, one can go further and one can encode the eleven French Army elementary soldier actions (Orient, Observe, Move, Protect, Hide, Estimate a distance, Show a target, Fire, Communicate, Report, Keep contact) using non player characters actions in video games (BARTHEYE 2012).

The purpose is to provide a large set of possible situations (or states) and to model user-defined macro actions as transitions between possible states. Reachability from one state to another is ensured by action combination (ordered action sequences). That way, the behaviour graph is dynamically computed according to the goal to achieve.

**PDDL** (Planning Domain Definition Language) is the standard encoding language for “classical” planning tasks. Components of a PDDL planning task are:

- **Objects**: things in the world that interest us,
- **Predicates**: properties of objects; can be true or false,
- **Initial state**: the state of the world that we start in,
- **Goal specification**: things that we want to be true,
- **Actions**: ways of changing the state of the world.

Actions and transitions between the initial state and the final state compatible with the goal to achieve are expressed using PDDL; that is PDDL data are separated into two files:

1. A domain file for predicates and actions,
2. A problem file for objects, initial state and goal specification.

When planning is required explicitly, the following operations are performed: *find the current objectives (the goal to achieve) if any, build automatically a planning PDDL problem, start the planning task using actions defined in the PDDL operator file (see Figure 1)* according to the **Sequencing** property and the **Phasing** property mentioned in previous section.

The main remark is that a plan execution architecture is required in order to handle emergency situations; in effect, in a video game, a game frame is computed every 50 milliseconds (that is, the game situation changes every 50 milliseconds leading to recompute action effects) and consequently, each plan becomes false very quickly and must eventually be recomputed. Either the current action can be modified without altering the plan by recomputing the current action with new game instructions or the whole plan is to be cancelled (see Figure 2).

In that context, actions plans are very short (less that 4 actions) and concern very simple combination of actions of the form Move, Hide, Fire, . . . . In the general case (i.e. outside...
of a video game architecture, very long plans of even infinite plans may naturally occur (infinite sequences are semantically perfectly valid). In effect, take any conjugated action pair of the form (pick, drop) performed by a robot hand (see next section); then it exists an infinity of identity action sequences. That means that the planning process is in the best case exponential and in the worse case undecidable.

One can assume that difficulties encountered when computing an action plan indicate that the conflict mode holds in order to solve that computation issue. Take models as favorable situations for a given entity; then counter-models for another entity (i.e. unwanted situations for that entity) is a way to locally simplify the planning computation. Therefore conflict is not restricted to military situations, but rather occurs once behaviours of local entities are computed.

3 Historical definitions of the AI planning calculus

In order to qualify computation issues, one can examine carefully the definition of the so-called AI Classical Planning Calculus, a quite old fashion concept imagined in the early days of Artificial Intelligence (NEWELL and SIMON H.A. 1959).

Definition 3.1 (AI Classical Planning Calculus) The AI Classical Planning Calculus is a process used in Artificial Intelligence which is intended given:

- a world description,
- a goal to achieve,
- a set of actions which can be brought into play in this world,

to find a correct sequence of actions to apply in this world in order to transform the initial state $\Sigma_i$ into a state $\Sigma_f$ compatible with the goal to achieve.

Definition 3.2 (STRIPS Formal Action) A STRIPS formal action (FIKES and NILSSON N.J. 1971) $\alpha$ in AI planning is a triple $\alpha = (\text{Pre}_\alpha, \text{Add}_\alpha, \text{Del}_\alpha)$ where

- $\text{Pre}_\alpha, \text{Add}_\alpha, \text{Del}_\alpha$ are conjunctions (or lists) of quantified positive literals $\text{at}_\alpha(y) \land \text{at}_\alpha(x,y) \land \text{free}_\alpha(z) \land \ldots$

with quantified variables $x, y, z, \ldots$

- $\text{Pre}_\alpha$ (preconditions) stands for the “requirements” of $\alpha$,
- $\text{Add}_\alpha$ (add lists) and $\text{Del}_\alpha$ (delete lists) are the “effects” of $\alpha$.

As an example of action, one can consider those from the robot Robby example (see Figure 3). Robby moves from one room to another and uses its grippers to pick and drop balls. The action system $\mathcal{A}$ consists of 3 quantified actions:

- $\text{pick}(x, y, z)$
  - $x : \text{ball}$
  - $y : \text{room}$
  - $z : \text{gripper}$

- $\text{drop}(x, y, z)$
  - $x : \text{ball}$
  - $y : \text{room}$
  - $z : \text{gripper}$

- $\text{move}(x, y)$
  - $x : \text{room}$
  - $y : \text{room}$

$\text{pick/drop}$ are two conjugate actions plus an action move forcing the root to move and preventing natural loops to occur. In effect, loops occur naturally because of the equalization

$$\text{Add}_{\text{pick}} = \text{Del}_{\text{drop}} \text{ and } \text{Add}_{\text{drop}} = \text{Del}_{\text{pick}}$$

(1)

One can define formally the planning problem which corresponds to the picture on the right in Figure 3 and where its associated plan is printed on the left.
Definition 3.3 (AI Classical Planning Problem) A classical planning problem $P$ is the sextuplet

$$P = (R, D, \mathcal{R}, \Sigma_i, \Sigma_f, \prec)$$

1. $R$ is the STRIPS formal action system,
2. $D = \{\text{rooma}, \text{roomb}, \text{roomc}, \text{roomd}, \text{ball1}, \text{ball2}, \text{ball3}, \text{ball4}, \text{left}, \text{right}\}$ is the finite domain of objects,
3. $\mathcal{R}$ is the set of ground operators obtained from $R$ wrt $D$ by substituting free quantified variables$^2$,
4. $\Sigma_i = \{\operatorname{at}((\text{ball1, rooma}), \operatorname{at}((\text{ball2, roomba}), \operatorname{at}((\text{ball3, rooma}), \operatorname{at}((\text{ball4, room}), \operatorname{at}((\text{roomba, left}), \operatorname{free}(\text{left})), \operatorname{free}(\text{right}))\}$ is called the initial state,
5. $\Sigma_f = \{\operatorname{at}((\text{ball1, rooma}), \operatorname{at}((\text{ball2, roomba}), \operatorname{at}((\text{ball3, room}), \operatorname{at}((\text{ball4, roomb})\}) \}$ is called the final state,
6. $\prec$ is the precedence condition $\Sigma_i \prec \Sigma_f$.

Definition 3.4 (State Formula) A planning state formula $\Psi$ is taken to be a non void conjunction of planning atoms,

$$\operatorname{at}((\text{ball1, roomba}) \land \operatorname{at}((\text{ball2, roomba}) \land \operatorname{free}(\text{left}) \land \ldots)$$

the set of planning states formulas $\mathcal{S}_\lambda$ is the free subalgebra of propositional conjunctive formulae $F_\lambda$ defined wrt to the domain $D$ and the set of predicates occurring in $\mathcal{R}$ minus the empty formula $\emptyset$. A planning transition is an element of $\mathcal{S}_\lambda \times \mathcal{S}_\lambda$.

If one wants to adapt proof theory to compute actions plans, one should be able to identify the null difference, or equivalently the compatibility between the final state $\Sigma_f$ and the $k_i$-current state $\Sigma_{k_i}$ computed from the initial state noted by the query $\Sigma_f \subseteq \Sigma_{k_i}$? Since the topology is Hausdorff, i.e. discrete and separable, one can compute as well state incompatibility $\Sigma_f \sqsubset \Sigma_{k_i}$.

$^2$For instance, $x, y, z, \ldots$ can be substituted with domain objects ball1, rooma, left according to the equations $x = \text{ball1}, y = \text{rooma}, z = \text{left}$, that way the operator $\operatorname{pick}(x, y, z)$ is mapped to $\operatorname{pick}(\text{ball1}, \text{room4}, \text{gripper})$.

$^3$A topological space $X$ is Hausdorff if for every $x, y \in X, x \neq y$, there are open neighbourhoods $O_x \ni x, O_y \ni y$ so that $O_x \cap O_y = \emptyset$.

In order to qualify better in that general case, tactical models, validity of action plans, and action plan computation, one has to provide formal action plans and one can use the epistemological quadriptych which provides a separation of knowledge according to four distinct basis.

### 4 Characterization of the dynamic using the epistemological quadriptych

More generally, the role of knowledge in any modelling process can be defined according to the epistemological quadriptych (CHAUDRON L. 2005) (see Figure 4). Epistemology concerns the classification of scientific methods, logical models and inference mechanisms as principles, designs and theories such that their scope and their relevance can be discussed.

According to that quadriptych, theories can be classified in four knowledge basis:

- the empirical basis: “real world”, data, \ldots,
- the conceptual basis: natural language, structures, boxology, \ldots,
- the formal basis: mathematics, theory, \ldots,
- the methodological basis: algorithms, programs, machines, \ldots.

According to self-representation structures and self-orientation processes (BARTHEYE and CHAUDRON L. 2015a), one should take into account the following question: what is the knowledge basis from which this kind of abstract classification functor can be defined? It seems obvious that the conceptual basis is the convenient one. Assume that the epistemological quadriptych can be classified according to a modulo-4 structure as in Figure 5.
Then it is clear that the knowledge basis concerns the upper layer at arithmetic index 3, the validity of plans the central layer at arithmetic index 2 and the implementation support the lower layer at arithmetic index 1. Therefore, the utilization of the epistemological quadrityprch as a conceptual diagram can be localized in the neighborhood of the upper layer at arithmetic index 3 and is not defined elsewhere. It is equivalent to set that the index 3 is a unique fix-point identifying the context in which a plan is to be defined.

One important point to be determined is the exact role of the empirical basis. The empirical basis provides enough entropy to represent any dynamic event. Consequently, it can be taken as the dynamic knowledge requirements for any intelligence device, artificial or not. The pair of layers (0, 4) stands for some open-closed embedding structure called the carrier or the support on which the dynamic can be defined. Namely, the carrier focuses locally the dynamic and can be represented as a clique, i.e. a maximally connected graph.

According to this, one can assume that this empirical basis acts as a reflection group on an operator structure encoding dynamicity and is nothing but qualification of dynamic validity. A reflection group can be taken to be an automorphism group $\text{Aut}(A)$ on an operator algebra $A$. In order to deal with some completeness property, one can assume that this automorphism group is totally defined inside the structure; such an automorphism is said to be inner.

In effect, in order to encode the empirical basis as the validator of the dynamic, the group of inner automorphisms (the 0-remainder modulo 4) acts as a double mirror effect (see Figure 6). That is, the empirical basis, represented by the 0 value modulo-4, is not implemented as a value but as an automorphism group acting on the dynamic which has necessary to be classified. That is, the left semi-product $\times$ in column 2 expresses left folding and can be set as the generator $\top$, i.e. a rising edge $\nearrow$ for 3 whereas the right semi-product $\times$ in column 4 expresses right unfolding and can be set as the annihilator $\perp$, i.e. a falling edge $\searrow$ for 3.

From the clique generator (0, 4), one can define a consistent arrow $\top : (0, 4) \rightarrow 3$ whose codomain is the knowledge-based edge. The order of the morphism $\|\top\|$ corresponds to a ratio 3/0 (generate 3 from 0) required in order to perform the modelling task as the adjoint process (abstraction, interpretation) in Figure 7. Two incompatible assumptions can be set:

1. $\|\top\| = \infty$ in order to shift at the “abstract level” 3 (one cannot model while staying at the concrete level 0 or equivalently “$\|\top\|$ maps outside the concrete category” or even “$\|\top\|$ is not an 0-endomorphism”),

2. the set of vertexes of the clique (0, 4) suggests to handle a family of generators $\|\top\|_i, i = 1 \ldots n, i \in \mathbb{N}$. That is, a single instance of $\top$ is not conceivable and consequently some finite integer $n$ characterizes the adjoint process (abstraction, interpretation) of the modelling task in Figure 7.

First note that inner automorphisms are endomorphisms; therefore the assumption $\|\top\| = \infty$ prevents classical group action theory to hold in the dynamic space. Furthermore, incompatibility of that kind generates conflict. That is, the knowledge basis is characterized by the conflict property for dynamic classification. In other words, one cannot define the dynamic space as a classical analytic space, i.e. as a delimited closed domain well defined in its center and more or less undetermined in its boundaries but exactly the other way round: the dynamic space is a chaos space or a bifurcation space in which an entropic factor holds and which is well determined in the neighborhood of full entropy. That is, entropy and conflict are natural dynamic features.

The geometrical meaning is as follows: the dynamic is determined in the neighborhood of a singularity $\perp$ rather than outside this neighborhood like in a classical analytic space. Moreover, in that conflict space, a planning solution means that dynamic must be separated; that is, one should classify among convenient dynamic (i.e. models) and unwanted dynamic (i.e. counter-models). Therefore the separate set

\[\begin{array}{cccc}
0 & 3 & 4 & 1 \\
2 & 2 \times & 2 & 2 \\
3 & 1 & 3 & 1 \\
0 & \end{array}\]

---

*In the mathematical area of graph theory, a clique is subset of vertexes of an undirected graph, such that its induced subgraph is complete; that is, every two distinct vertexes in the clique are adjacent.*

*In abstract algebra an inner automorphism is a function in which an initial operation is applied, then another operation, and then the initial operation is reversed. With letters to indicate the operations and thing-being-transformed: $f^{-1} \circ g \circ f(X)$. Sometimes the initial action and its subsequent reversal change the overall result (“raise umbrella, walk through rain, lower umbrella” has a different result from just “walk through rain”), and sometimes they do not (“take off left glove, take off right glove, put on left glove” has the same effect as only “take off right glove”). More formally an inner automorphism of a group $G$ is a function $f : G \rightarrow G$ defined for all $x \in G$ by $f(x) = a^{-1} xa$, where $a$ is a given fixed element of $G$, and where we deem the action of group elements to occur on the right.*

---

6A category $C$ consists of (i) a class $\text{Obj}(C)$ of objects (ii) a class $\text{Hom}(C)$ of morphisms, or arrows, or maps, between the objects. Each morphism $f$ has a unique source object $a$ and target object $b$ where $a, b \in \text{Obj}(C)^2$. We write $f : a \rightarrow b$, and we say “$f$ is a morphism from $a$ to $b$”. We write $\text{Hom}(C(a, c))$ to denote the Hom-class of all morphisms from $a$ to $b$ (iii) for every three objects $a, b$ and $c$, a binary operation $\text{Hom}(a, b) \times \text{Hom}(b, c) \rightarrow \text{Hom}(a, c)$ called composition of morphisms; the composition of $f : a \rightarrow b$ and $g : b \rightarrow c$ is written as $g \circ f$ or $gf$, such that the following axioms hold:

- **(associativity)** if $f : a \rightarrow b, g : b \rightarrow c$ and $h : c \rightarrow d$ then $h \circ (g \circ f) = (h \circ g) \circ f$, and (identity) for every object $x$, there exists a morphism $1_x : x \rightarrow x$ (some authors write $\text{id}_x$) called the identity morphism for $x$, such that for every morphism $f : a \rightarrow x$ and every morphism $g : x \rightarrow b$, we have $1_x \circ f = f$ and $g \circ 1_x = g$. From these axioms, one can prove that there is exactly one identity morphism for every object. Some authors use a slight variation of the definition in which each object is identified with the corresponding identity morphism.
rability property is the crucial property for planning computation.

Separability must be handled according to a phase structure: we can assume that the clique (0, 4) is endowed by that phase structure \( \Psi \) (i.e. a well-defined structure to represent signal functions); since the planning computation classifies the unwanted dynamic by the annihilator \((.)^\perp\), then the unwanted dynamic is the dynamic involved from the phase \( \Psi \), i.e. from a commutative representation of the signal according to a set of orthogonal characters as in Fourier transform. It is equivalent to set that the integrated dynamic from \( \|T_i\|, i = 1 \ldots n, i \in \mathbb{N} \) is actually the null dynamic; that is, the dynamic freezes and generates a deadlock. Therefore one should assume that deadlocks occur due to the direct product of signal functions.

5 Classification of the conflicting dynamic

In order to understand the planning computation as a process integrating various kind of dynamic on these four basis, one can express the dynamic by actions patterns definable actually on two very distinct layers:

- the macroscopic level on which the conceptual basis is defined,
- the microscopic level on which the methodological basis is defined.

That is, an algorithm expresses the dynamic in a more precise way, i.e. with a greater granularity\(^7\) but cannot be used to compute or to understand strategic issues. A more problematic issue occurs: in general, the dynamic defined according to these two sides is conflicting and when these two layers are merged, some entropy factor is generated. Take the microscopic level as the minimal level of dynamic which can be observed according to the empirical basis; in this level, dynamic is expressed according to elementary operators and as such cannot be ignored; therefore one can wonder why the macroscopic level becomes conflicting wrt the microscopic level. It is clear that if elementary microscopic operators are headed to vanish, then the dynamic stops; therefore they are persistent. But on the other hand, if macroscopic operators are persistent, then the dynamic stops as well due to the phase structure issue; therefore they have to vanish.

In order to mitigate conflict between layers, one can define the merging operation according to the knowledge basis in between, the formal basis which is understood as the proof process. The microscopic level contains relevant information which is not covered by the macroscopic knowledge and which has to be integrated. The macroscopic level provides some kind of division as a Fourier transform can do whereas the microscopic level characterizes signal residues, i.e. noise. Neither in physics nor in mathematics, it is expected that the noise should alter and eventually replace the signal characters, i.e. the remainder should alter and eventually replace the divisor. Therefore the merging operator

\[\parallel\top\parallel\]

full entropy.

Assume that knowledge basis identifies autonomous entities; then the modelling task is understood to be a normal activity for an autonomous entity. It is equivalent to set that a precedence holds; any AI entity behaviour is only defined in level 3, the macroscopic level, and this entity is intended to play the role of a self-autonomous entity in the spirit of (Chaudron et al. 2015)). Furthermore, due to conflict, every plan to compute is of torsion. Therefore the plan computation process is not a consistent function but rather a fork and can deliver either models (valid plans) or counter-models (invalid plans). In order to remove the torsion, one should compute the correspondence in level 2, the formal basis. The plan is the computation of the AI entity behaviour at level 2 by the full entropic action of the residues from the microscopic level 1 to the level 3 according to the merging representation by the group of inner automorphisms (the change dynamic in the continuous level) introduced in Figure 6. Moreover, this proof process, if any, must encode in a consistent way the merging operation.

6 Qualification of the formal modelling task

The modelling task (see Figure 7) is a central process in tactic, but any modelling task is abstract and obviously is not limited to military domains. The formal modelling task requires a simulation device in order to compare modelling results and experimental results. One obtains that way a representation of the experiment by a formal model. For instance, harmonic analysis is a formal model. According to type theory, logical formulae \( A \supset B, \neg C, \ldots \) (i.e. types) are taken as representative objects of the formal basis; elements of the methodological basis are proofs and terms \( t, u, v \) are proof items. Proof planning is based on the duality (terms, types); the proof process normalizes any term \( t \) of the form \( t: \Sigma \vdash \Sigma_f \) according to the knowledge contained in \( \Sigma \) and \( \Sigma_f \). Assume that a proof process is a solution of a given problem (the Russian mathematician Kolmogorov understood the logical constants as problems in 1932); computing the term \( t \) provides a representation of \( t \) as a solution by building a tree whose root is the query \( t: \Sigma \vdash \Sigma_f \), whose nodes are formal actions and and leaves are identity axioms as \( 1_{\text{free(}}\text{gripper)}: \text{free(}}\text{gripper)} \parallel \text{free(}}\text{gripper)}\).

Proof theory and AI actions plans diverge here but the salvation may come from group theory. In effect, identity axioms \( 1_f \) are the only deductive arrow where the premise \( f \vdash \) and the conclusion \( \vdash f \) freely commute and identity arrows

\[\parallel\top\parallel\]

Figure 7: The modelling task

should totally recompute the representation, i.e. generates full entropy.

\(^7\)Granularity is the level of detail considered in a model or decision making process. The greater the granularity is, the deeper is the level of detail. Granularity is usually used to characterize the scale or level of detail in a set of data.
are a family of trivial groups. A proof is an acyclic graph representation of a direct product of trivial groups. Groups and graphs are totally disconnected and deadlocks come from that disconnection. The problem comes from step 3 in definition 3.3: one cannot freely substitute variables by domain objects otherwise the dynamic is frozen by identity loops. Conflict is represented by mutual exclusion $\text{Add}_\alpha/\text{Del}_\beta$.

Solving that problem means that plan computation must be deterministic: find at random a single action sequence with a single substitution and stop. Any non-deterministic process trying another substitution by backtrack generates loops. That means that the generator $\top$ cannot be divided by backtrack. It is equivalent to set that any non-deterministic process is phase-dependent, i.e. generates noise which prevents to extract dynamic saliencies.

The geometric interpretation of the level 2 is as follows; edges $\nearrow$ and $\searrow$ in Figure 6 corresponds to the cusp identification as a singularity and the AI planning calculus corresponds to the hyperbolic resolution $\infty = 1/0$ by switching the positive side on the x axis (see Figure 8). This anti-commutative resolution process is performed in the neighborhood of zero and is actually the operator $\infty = 1/0$ classifying the dynamic according to the global unit dynamic 1.

The separable proof process can be set as follows: since the phase $\Psi$ holds as a counter-model, then a model is a anti-phase $\lambda$ where $\text{Support}(\Psi) \cap \text{Support}(\lambda) = \emptyset$ and which is denoted according to the arithmetic-geometric equation $\lambda = 1 - \Psi$. The planning computation corresponds to the inner consistent morphism $\Psi \rightarrow \lambda$ (see Figure 9) expressed according to the disjunctive law $\Psi^\perp + \lambda^\perp$ which is the non commutative weaker form of De Morgan laws and can be interpreted as “in order to gain $\lambda$, one should remove $\Psi$”.

7 Conclusion

This work sets that AI planning computational capabilities are necessary and sufficient conditions to trust on AI devices integration in today’s life and particularly for management of conflict issues.

Incomplete solutions according to video game architectures can be already defined in order to improve military doctrines in very critical situations. In any case, there is no reason to militate against this integration.

References


BARTHEYE, O., and CHAUDRON L. 2015b. Risk management systems must provide automatic decisions according to crisis computable algebras. AAAI SpringSymposia 15, Standford, USA.

BARTHEYE, O. 2012. Artificial Intelligence planning game software architectures are able to model certain very critical military operations. Deutsche Gesellschaft Für WehrTechnik, Bonn, DE.


