Trajectory tracking module, application to space systems

Fabien Perrot and **Louise Travé-Massuyès**
{fperrot,louise}@laas.fr
0033561336364 0033561336302
LAAS-CNRS
Université de Toulouse
7, avenue du Colonel Roche,
31077 Toulouse Cedex 4
France

Problematics

Increasing autonomy of space systems is important in order to achieve faraway exploration missions and to reduce ground segment load. In particular these systems should face up to defaults while accomplishing their mission.

Two major modules are pervasive in proposed architectures for autonomy: trajectory tracking module and decision module. Trajectory tracking module uses available observations of the system over time to compute a belief of the past trajectory of the system. Decision module uses this belief to compute the future trajectory of the system that achieve some goal.

This work covers the trajectory tracking module which mainly states and solves trajectory tracking models using the Constraint Satisfaction Problems (CSP) framework. This work extensively uses concepts and techniques from CSP and Model-Based Diagnosis (MBD) fields.

Goals

- Formulation of trajectory tracking models in the CSP framework.
- Modification of existing CSP algorithms to solve trajectory tracking models.

Context

This work takes part of the AGATA project of development of a prototype of an autonomous satellite. Its aim is to define an architecture for autonomous space systems and to define a mean for ensuring consistency of models used in this architecture.

These new applications (such as planning, trajectory tracking, or execution control) are based on a knowledge of the behavior of the system. This knowledge is incorporated into models: dynamic behavior, physics characteristics, ...

The different reasoning mechanisms are based on different kinds of models. Those models are written in different languages and contain different information. A model for planning contains the list of admissible activities. To each activity is associated its length, its needs in ressources, its preconditions to apply the activity, its effects, ... A model for diagnosis describes admissible nominal states of components, fault states, nominal dynamic behavior and some behavior under faults. These models share common information with different way of representation.

A same reasoning mechanism can be used at different levels of abstraction. Information at one level are abstractions of information at another level.

In the context of an autonomous space system, several mecanisms of reasoning must cooperate at different levels of abstraction. A way for ensuring consistency of information should be found. Different works are conjointly started by CNES, LAAS and ONERA to:

- identify needs for modeling the system at different levels of decision,
- define the suited formalisms to model this knowledge,
- find a way for ensuring global consistency of models.

State of current work

Bibliographical work

First part of the work was bibliography. It concerned principles of model-based diagnosis in AI, planning, model-checking, theories of abstraction and CSP.

Specific work for AGATA project

Study of the state of art of architectures for autonomous systems and models in these architectures led to a report (Py *et al.* 2006). Modeling of a satellite whose mission is the observation of Earth and fire detection has been done (Lemai *et al.* 2006).

Trajectory tracking work

Systems we are interested in modeling for trajectory tracking consists of a set of controllable components, that is commands could be sent to components. Observations of the system can be passively acquired. For example, observations could be the value of a voltage given by a sensor, or a command sent to a component.

Formulation of the trajectory tracking problem has been analysed in details. The trajectory tracking problem cannot be formulated as a classic CSP problem. Its incremental nature has led us to consider extensions of the CSP framework, such as Dynamic CSP.

A trajectory tracking problem informally consists of finding a belief of the past trajectory of the system in terms of a sequence of its states. This belief should be consistent with the model and observations of the system. The model of the system consists of the constraints describing behavior of the components, the constraints describing transitions between states when commands arrive, and the contraints describing the observations.

This trajectory tracking problem can be divided into subproblems. We can define a subproblem as soon as the trajectory tracking module recepts an observation. We call a subproblem a trajectory search problem. Constraints are static in trajectory search problems. Between two consecutive trajectory search problems, constraints corresponding to the new observations and behavior of the system are added. More problematically, if we do not want to keep the history of the whole life of the system, constraints must be removed. A trajectory search problem can be formulated in the classic CSP framework.

Solving trajectory search problems does not consist of the satisfaction task, as classic CSP solving does. Actually, only assignments of a subset of variables satisfying a property must be found. The variables in this subset are called the decision variables. Solving a trajectory search problem can be formulated as finding the globally consistent assignments of the decision variables. For this to be clear, we formally define CSP, trajectory search problem and their respective semantics.

Language

Definition 1 *A constraint satisfaction model or CSM* $\mathcal{M} = (X, D, C)$ *is defined by:*

- *a set* $X = \{x_1, \ldots, x_n\}$ *of* n *variables;*
- *a set* $D = \{d_1, \ldots, d_n\}$ *of* n *finite domains for the variables of* X*. A domain* d_i *is associated to the variable* x_i*.*
- *a set* $C = \{c_1, \ldots, c_m\}$ *of* m *constraints. Each constraint* c_i *is defined by a pair* (v_i, r_i)*.*
  - v_i *is a set of variables* $\{x_{i_1}, \ldots, x_{i_{n_i}}\} \subseteq X$ *on which the constraint is defined. Arity of* c_i *is the length of the sequence* v_i*.*
  - r_i *is a relation, defined by a subset of the cartesian product* $d_{i_1} \times \cdots \times d_{in_i}$ *of the domains associated to the variables of* v_i*. It explicitely represents the authorized tuples of values for these variables.*

Semantics We define the concept of assignment which is to be related to the concept of interpretation of propositionnal logic.

Definition 2 (Assignment) *Given a CSM* $\mathcal{M} = (X, D, C)$*, an assignment* $\mathcal{A}$ *of* $Y = \{x_{y_1}, \ldots, x_{y_{|Y|}}\} \subseteq X$ *is a function which associates to each variable* $x_{y_i} \in Y$ *a value* $\mathcal{A}(x_{y_i}) \in d_{y_i}$ *(the domain of* x_{y_i}*).*

In the following, we refer to an assignment A by a function or by the graph of this function (the set of pairs $(x_{y_i}, \mathcal{A}(x_{y_i}))$) which we denote:

$$\mathcal{A} = \{x_{y_1} \to \mathcal{A}(x_{y_1}), \ldots, x_{y_{|Y|}} \to \mathcal{A}(x_{y_{|Y|}})\}$$

or by a sequence of values $(\mathcal{A}(x_{y_1}), \ldots, \mathcal{A}(x_{y_{|Y|}}))$, the order of variables of Y being implicit. We say that an assignment whose domain is X to be complete, partial otherwise.

We can apply an assignment $\mathcal{A}$ of Y to a set $V \subseteq Y$ of variables. The result of this application is the set of the images of the elements of V by $\mathcal{A}$.

Definition 3 (Constraint satisfaction) *Given a CSM* $\mathcal{M} = (X, D, C)$*, an assignment* $\mathcal{A}$ *of* Y *satisfies the constraint* $c_i = (v_i, r_i)$ *of* C *(denoted* $\mathcal{A} \models c_i$*) if and only if* $v_i \subseteq Y$ *and* $\mathcal{A}(v_i) \in r_i$*. On the contrary, we say that an assignment* $\mathcal{A}$ *of* Y *violates* c_i *if and only if* $v_i \subseteq Y$ *and* $\mathcal{A}(v_i) \notin r_i$*.*

Definition 4 (Consistent assignment) *Given a CSM* $\mathcal{M} = (X, D, C)$*, an assignment* $\mathcal{A}$ *of the variables of* $Y \subseteq X$ *is said to be consistent if and only if:*

$$\forall c_i = (v_i, r_i) \in C \text{ such that } v_i \subseteq Y, \mathcal{A} \models c_i$$

An assignment is consistent if it does not violate any constraint of the CSM. Checking that an assignment on Y is consistent is a polynomial problem. It consists of checking that each constraint c_i is not violated by this assignment.

We define the concept of solution of a CSM, to be related to the concept of model in propositional logic.

Definition 5 (Solution of a CSM) *A solution* $\mathcal{S}$ *of* $\mathcal{M} = (X, D, C)$ *is a consistent assignment of the variable of* X*. We then say that the assignment* $\mathcal{S}$ *satisfies* $\mathcal{M}$ *(denoted* $\mathcal{S} \models \mathcal{M}$*). The set of solutions of* $\mathcal{M}$ *is denoted* $S_{\mathcal{M}}$*.*

Checking that a complete assignment is a solution is a polynomial problem. Finding a solution of a CSM or showing that none exists is a problem called Constraint Satisfaction Problem. It is an NP-hard problem.

Definition 6 (Consistency of a CSM) *A CSM* $\mathcal{M} = (X, D, C)$ *is said to be consistent if and only if* $S_{\mathcal{M}} \neq \emptyset$*.*

Checking if a CSM is consistent is an NP-complete decision problem. This problem is called satisfiability of a CSM. We can define a stronger property than consistency on assignments.

Definition 7 (Globally consistent assignment) *Given a CSM* $\mathcal{M} = (X, D, C)$*, an assignment* $\mathcal{A}$ *of* $Y \subseteq X$ *is said to be globally consistent if and only if* $\exists \mathcal{S} \in S_{\mathcal{M}}$ *such that* $\mathcal{A} \subseteq \mathcal{S}$*.*

If an assignment is not globally consistent, it is not possible to extend it in a solution. Smaller the assignment is, harder is to check this property. Global consistency of the empty assignment is equivalent to consistency of the CSM.

We can now define a trajectory search model and a trajectory search problem.

Definition 8 (Trajectory search model) *A trajectory search model is a CSM* $\mathcal{M}_{\mathcal{TS}} = (X, D, C)$ *where:*

- $X = \{w_1, \ldots, w_{n_W}, nd_1, \ldots, nd_{n-n_W}\})$.
  $W = \{w_1, \ldots, w_{n_W}\}$ *is a set of* n_W *decision variables.* $ND = \{nd_1, \ldots, nd_{n-n_W}\}$ *is a set of* $n - n_W$ *non decision variables.*

Generally when modeling a system to perform trajectory search, the obtained trajectory search model has always the same form. n_W represents the number of components in the system and each variable in W represents the status of a component. Variables in ND represent the state variables, the command variables and the observable variables of the components of the system. The domain of a variable of W represents the possible nominal modes and some fault modes of a component. Constraints of C represent the behavior of components in each mode and observations as unary constraint on non decison variables.

Given a trajectory search model, we are interested in finding globally consistent assignments of the decision variables. That is why we define the concept of solution trajectory.

Definition 9 (Solution trajectory) *A solution trajectory $\mathcal{S}_\mathcal{T}$ of a trajectory search model $\mathcal{M}_{\mathcal{TS}} = (X, D, C)$ is a globally consistent assignment of the variables of W. We then say that the assignment $\mathcal{ST}$ globally satisfies $\mathcal{M}_{\mathcal{TS}}$ (denoted $\mathcal{S}_\mathcal{T}\sqrt{}\mathcal{M}_{\mathcal{TS}}$). The set of solutions of $\mathcal{M}_{\mathcal{TS}}$ is denoted $S_{T\mathcal{M}_{\mathcal{TS}}}$.*

Definition 10 (Trajectory search problem) *A trajectory search problem consists of finding a solution trajectory of a trajectory search model or show that none exists.*

We also can be interested in finding all the solution trajectories of a trajectory tracking model or even can imagine other problems.

In a first approach, we defined a trajectory finding model as a classic CSM where variables are divided into decision variables and non decision variables. More information can be incorporated into the model. Actually, preference between trajectories can be used to refine the belief of the real trajectory of the system. The formulation of the trajectory search problem defined above can be changed toward a constraint optimization formulation. Extensions of the CSP framework such as Partial CSP, Valued CSP or Optimal CSP have been proposed in the literature to capture this concept of preference between solutions. Informally, we frame the trajectory search model as an Optimal CSM and the trajectory search problem as finding a globally consistent assignment of the mode variables which has best cost. Other problems can be defined given a trajectory search model like finding all the globally consistent assignements which have best cost or finding the k first ones, ...

Current work consists of implementing algorithms for solving trajectory search problems, more precisely in the machinery to interleave optimization and satisfaction.

Future work

Future work will consist of:

- finishing the implementation of an algorithm for solving trajectory search problem,
- consolidating the theoretical work,
- defining and implementing one or several algorithms according to the state of art of Dynamic CSP for solving trajectory tracking problems,
- making benchmarks with several algorithms for solving trajectory search and tracking problems, with or without cost additional information.

References

Lemai, S.; Py, F.; Perrot, F.; Orlandini, A.; and Verfaillie, G. 2006. Exemple de satellite d'observation de la terre simplifié - entrées pour l'élaboration de modèles de planification, contrôle d'exécution et diagnostic. Technical report, LAAS/CNRS.

Py, F.; Perrot, F.; Pencolé, Y.; Orlandini, A.; Travé-Massuyès, L.; and Ingrand, F. 2006. Projet agata. rapport de synthèse sur l'activité : études et définitions de modèles. Technical report, LAAS/CNRS.