Self-awareness in autonomous nano-technology swarm missions.

Self-Awareness in Autonomous Nano-Technology Swarm Missions

Emil Vassev
Lero–the Irish Software Engineering Research Centre
University of Limerick
Limerick, Ireland
emil.vassev@lero.ie

Mike Hinchey
Lero–the Irish Software Engineering Research Centre
University of Limerick
Limerick, Ireland
mike.hinchey@lero.ie

***Abstract*—NASA is currently exploring swarm-based technologies, targeting the development of prospective exploration missions to explore regions of space, where single large spacecraft would be impractical. Such systems are envisioned to operate autonomously and their success factor depends highly on self-awareness capabilities. This research emphasizes the development of algorithms and prototyping models for self-awareness in swarm-based space-exploration systems. This article tackles the self-initiation and self-healing properties of swarm-based space-exploration systems.**

Keywords –self-awareness; self-healing; ANTS; NASA.

I. INTRODUCTION

As a philosophical term, self-awareness is the explicit understanding that one exists. In the context of computing systems, self-awareness is used to denote novel classes of computing systems that show an exceptional degree of autonomous behavior. Agent technologies have been identified as a key enabler for engineering autonomous behavior in systems, in part due to their capability to provide self-governance (or autonomy). Nowadays, autonomy plays a crucial role in space exploration. Without risking human lives, robotic technology such as robotic missions, automatic probes and unmanned observatories allow for safe and efficient space exploration. However, unmanned space exploration poses numerous technological challenges. This is basically due to the fact that unmanned missions are intended to explore places where no man has gone before and thus, such missions must deal, often autonomously and with no human control, with unknown factors, risks, events and uncertainties. A nice example of such a challenging task is the exploration of the Asteroid Belt, a region in our solar system located between the planets Mars and Jupiter and probably containing millions of asteroids composed of metals and minerals [1]. Realizing that single and monolithic spacecraft are impractical to explore the Asteroid Belt, NASA is proposing new biologically-inspired swarm-based classes of space exploration missions called ANTS (Autonomous Nano-Technology Swarm), where potentially thousands of small spacecraft will work together to cooperatively explore asteroids [2], accomplishing missions through cooperative action by a group of autonomous individual spacecraft. Note that the high levels of autonomy of such intelligent swarm-based systems require high degrees of self-awareness that will help such systems organize and adapt to changes. We tackle the aspects of self-awareness that help ANTS self-initiate for team formation and self-heal. The self-awareness capability enables ANTS to self-initiate to react to changes in the swarm or the environment and to enable mitigation of adverse events during mission.

II. ANTS

The Autonomous Nano-Technology Swarm (ANTS) concept sub-mission PAM (Prospecting Asteroids Mission) is a novel approach to asteroid belt resource exploration (see Figure 1). ANTS necessitates extremely high levels of autonomy, minimal communication requirements with Earth, and a set of very small explorers with a few consumables [2]. These explorers that form the swarm are *pico-class*, *low-power* and *low-weight* spacecraft units, yet capable of operating as fully autonomous and adaptable agents. Ideally, the ANTS' explorers will be *automatically* assembled from reusable components by an ANTS space laboratory.

Each spacecraft is equipped with a solar sail and relies primarily on power from the sun, using only tiny thrusters to navigate independently. Moreover, each spacecraft also has onboard computation, artificial intelligence, and heuristics systems for control at the individual and team levels. The spacecraft forming a swarm are able to interact with each other and self-organize. In general, a swarm consists of several sub-swarms, which are temporal groups organized to perform a particular task. Each swarm group has a group leader (*ruler)*, one or more *messengers*, and a number of *workers* carrying a specialized instrument (see Figure 1). The messengers are needed to connect the team members when they cannot connect directly, due to long distances or a barrier. For ANTS exploration, individual autonomy is not crucial, but the mission cannot succeed unless each *team* has the following *autonomic properties*:

Self-configuration. ANTS must be fully reconfigurable to support concurrent exploration and examination of hundreds of asteroids or to adapt to changes in the system.

Self-healing. ANTS must be able to recover from errors or damage, including those caused by either a solar storm or a collision with an asteroid or another spacecraft.

Self-optimizing. ANTS must be able to improve performance on the fly, e.g., rulers can use experience to self-optimize by improving their ability to identify asteroids.

Self-protecting. ANTS must be able to anticipate and recover from intrusions or self-protect from solar storms.

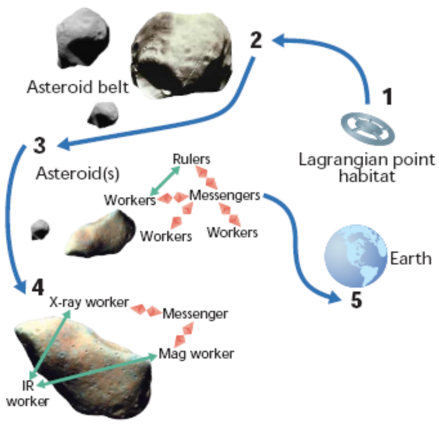

FIGURE 1. ANTS MISSION CONCEPT [2]

III. SELF-AWARENESS MODELS FOR ANTS

A. Self-Initiation for Team Formation

The awareness capability helps an idle spacecraft unit self-initiate to react to changes in the swarm or the environment. To help an ANTS spacecraft unit process its knowledge and become aware, a behavior model based on the so-called Partially Observable Markov Decision Processes (POMDP) [3] is considered. Note that this model is appropriate when there is uncertainty and lack of information necessary to determine the state of the entire swarm. For example, ANTS spacecraft might be idle, i.e., not actively participating in the swarm's activities, because they are not certain about the current swarm state. Thus, the POMDP model helps a spacecraft unit reason on the current swarm state (or that of the environment) and eventually self-initiate when an action should be performed. According to our POMDP-based model, an ANTS spacecraft unit takes as input *observable* situations, involving other ANTS spacecraft and the environment, and generates as output actions initiating spacecraft activity. Formally, this model is a tuple:

$$M = \langle S; A; T; R; Z; O \rangle \qquad (1)$$

where:

- S is a set of swarm states that are not observable.
- An initial belief state $s_0 \in S$ is based on p_0 $(s_0; s_0 \in S)$, which is a discrete probability distribution over the set of swarm states S representing for each state the unit's belief it is currently occupying that state.
- A is a finite set of actions that might be undertaken.
- $T: S \times A \rightarrow \Pi(S)$ is the state-transition function, giving for each state s and action a a probability distribution. Here, $T\,(s; a; s')$ computes the probability of ending in state s', given that the start state is s and the unit takes action a, $p\,(s' \mid s; a)$.
- $O: A \times S \rightarrow \Pi(Z)$ is the observation function giving for each swarm state s and action a, a probability distribution over observations Z. For example, $O\,(s'; a; z)$ is the probability of observing z, in state s' after taking action a, $p\,(z \mid s'; a)$.
- $R : S \times A \rightarrow R$ is a reward function, giving the expected immediate reward gained by the spacecraft unit for undertaking an action a in a state s, e.g., $R\,(s; a)$. The reward is a scalar value in the range [0..1] determining what action should be undertaken in compliance with the swarm goals.

Self-Initiation for Team Formation. To illustrate this model, let's assume that an ANTS swarm is currently occupying the state s = "*new asteroid is discovered, but no exploration team has been formed yet and still no ruler is self-initiated for team formation*". Let's assume there is at least one idle ruler in the swarm ready to undertake a few actions A, including the action a = "*self-initiation for team formation*". The ruler performs the following steps:

1) It computes its current belief state s_0, the state with the highest probability p_0 and eventually $s_0 = s$.
2) It computes the probability p_1 of the swarm occupying the state s' = "*new asteroid is discovered and a ruler is self-initiated for team formation*" if the action a is undertaken from state s_0.
3) It computes the probability $p_2\,(z \mid s'; a)$ of observation z = "*there is a sufficient number of idle workers and messengers to form a new team*".
4) It computes the reward $r\,(s_0; a)$ for taking the action a (*self-initiation for team formation*) in state s_0. If no other immediate actions are required (forced by other swarm goals), the reward r should be the highest possible.

Probability Computation. The POMDP model for self-initiation requires the computation of a few probability values. In this subsection, we present a *model for assessing probability* applicable to the computation of POMDP probability values such as probability of the swarm being in a state and probability of observation. In our approach, the *probability assessment* is an indicator of the number of possible execution paths a spacecraft unit may take, meaning the amount of certainty (excess entropy) in the swarm's behavior. To assess that behavior prior to the swarm implementation, it is important to understand the complex interactions among the units in an ANTS swarm. This can be achieved by modeling the behavior of individual reactive spacecraft units together with the swarm (or team) behavior as *Discrete Time Markov Chains* [4], and assessing the level of probability through calculating the probabilities of the state transitions in the corresponding models. We assume that the unit-swarm interaction is a stochastic process where the swarm events are not controlled by the spacecraft unit and thus, their probabilities are considered equal.

The theoretical foundation for our Probability Assessment Model is the property of Markov chains, which states that, *given the current state of the swarm, its future evolution is independent of its history*, which is also the main characteristic of a reactive and autonomic spacecraft unit.

An algebraic representation of a Markov chain is a matrix (called *transition matrix*) (see Table 1) where the rows and columns correspond to the states, and the entry p_{ij} in the i^{th} row, j^{th} column is the transition probability of being in state S_j at the stage following state S_i.

TABLE I. TRANSITION MATRIX P

	S_1	S_2	…	S_j	…	S_n
S_1	p_{11}	p_{12}	…	p_{1j}	…	p_{1n}
S_2	p_{21}	p_{22}	…	p_{2j}	…	p_{2n}
…	…	…	…	…	…	…
S_i	p_{i1}	p_{i2}	…	p_{ij}	…	p_{in}
…	…	…	…	…	…	…
S_n	p_{n1}	p_{n2}	…	p_{nj}	…	p_{nn}

The following property holds for the calculated probabilities:

$$\Sigma_j\, p_{ij} = 1 \qquad (2)$$

We contend that probability should be calculated from the *steady state* of the Markov chain. A steady state (or *equilibrium state*) is one in which the probability of being in a state before and after a transition is the same as time progresses. Here, we define probability for a swarm configuration composed of k units as the level of certainty quantified by the source excess entropy, as follows.

$$\textit{Probability (ANTS)} = \Sigma_{i=1,k} H_i - H \qquad (3)$$

$$H_i = -\Sigma_j\, p_{ij} \log_2 (p_{ij}) \qquad (4)$$

$$H = -\Sigma_i\, v_i\, \Sigma_j\, p_{ij} \log_2 (p_{ij}) \qquad (5)$$

Here,

- H is an entropy that quantifies the level of uncertainty in the Markov chain corresponding to an ANTS swarm;
- H_i is a level of uncertainty in a Markov chain corresponding to a spacecraft unit;
- v is a steady state distribution vector for the corresponding Markov chain;
- p_{ij} values are transition probabilities in the extended state machines modeling the behavior of the i^{th} unit.

Note that for a transition matrix P, the steady state distribution vector $\boldsymbol{v}$ satisfies the property $v*P = v$, and the sum of its components v_i is equal to 1.

Interpretation. The level of uncertainty H is exponentially related to the number of *statistically typical paths* in the Markov chain. Having an entropy value of 0 means that there is no level of uncertainty in a Markov system for a specific unit's behavior. A higher value of probability implies less uncertainty in the model.

B. Self-Healing

In addition to the specific control and notification messages, to facilitate the *proactive monitoring*, the individual spacecraft units exchange on a regular basis the so-called *pulsebeat messages* carrying useful information including the current *health status* of the sender. For example, each *worker* sends, on a regular basis, *pulsebeat messages* to the *ruler* of its group. This helps the ruler determine when a worker is not able to continue its operation, due to a failure.

Self-healing in ANTS is about finding the right *self-healing strategy* that will eventually help the swarm repair the faulty spacecraft units without decreasing the overall swarm performance or affecting the mission goals. Such a self-healing strategy, we term a "*smart self-healing strategy*". A self-healing strategy is determined by an *initial faulty state* $\boldsymbol{s_f}$, a *destination nominal state* $\boldsymbol{s_n}$ and a *set of self-healing actions* $\boldsymbol{A_{sh}}$ (repair plan) distributed among the spacecraft units participating in the self-healing process. Formally, a *self-healing strategy* $\boldsymbol{M}$ is a tuple

$$M = \langle s_f; s_n; A_{sh} \rangle \qquad (6)$$

In this approach, a computed self-healing strategy is a *smart strategy,* because it is computed by taking into consideration the global mission goals and policies. We assume that a smart self-healing strategy will be always applicable, i.e., without *additional* preconditions. Here, the challenges are 1) how to determine the faulty state $\boldsymbol{s_f}$; and 2) how to determine the destination nominal state $\boldsymbol{s_n}$ and the appropriate actions $\boldsymbol{A_{sh}}$ leading to that state in complement with the global mission goals and policies. We presume that a *faulty state* $\boldsymbol{s_f}$ is a deviation from a *nominal state* when a *fault* has occurred in the system. Therefore, to determine faulty states of a spacecraft unit, we consider an *initial nominal state* $\boldsymbol{s_0}$ and a set of possible basic faults $\boldsymbol{F}$, which are actually non-desirable (or unobservable) events. Formally, this can be presented as following:

$$R_{sf} : S_u \times F \rightarrow S_f \qquad (7)$$

Here, $\boldsymbol{R_{sf}}$ is a function computing the possible faulty states $\boldsymbol{S_f}$ for each nominal state $\boldsymbol{S_u}$ of the space unit and basic faults $\boldsymbol{F}$. For example,

$$s_f = R_{sf}(s_0; f_i) \qquad (8)$$

computes the *faulty state* $\boldsymbol{s_f}$, which is a deviation from the *nominal state* $\boldsymbol{s_0}$ ($\boldsymbol{s_0} \in \boldsymbol{S_u}$) when a *fault* $\boldsymbol{f_i}$ ($\boldsymbol{f_i} \in \boldsymbol{F}$) has occurred in the system. Note that $\boldsymbol{R_{sf}}$ may be more complex when there is uncertainty in the state evaluation. In such a case $\boldsymbol{R_{sf}}$ may be a function returning a set of possible faulty states $\boldsymbol{S_f}$ and a probability distribution $\boldsymbol{\Pi}$ over those states. Thus,

$$R_{sf} : S_u \times F \rightarrow \Pi(S_f) \qquad (9)$$

and $\boldsymbol{s_f = R_{sf}(s_0; f_i)}$ is the faulty state with highest probability.

As we have stated, the second challenge in this approach is to determine the *destination nominal state* $\boldsymbol{s_n}$ and the *set of self-healing actions* $\boldsymbol{A_{sh}}$ that will do the transition from $\boldsymbol{s_f}$ to $\boldsymbol{s_n}$. The problem is that a smart self-healing strategy copes with the mission goals and policies. For example, if the repair process significantly slows down the mission, which eventually can be accomplished without repairing the faulty units, the strategy might be to leave those units unrepaired.

Note that to compute both the nominal state $\boldsymbol{s_n}$ and the *set of self-healing actions* $\boldsymbol{A_{sh}}$ of a smart self-healing strategy, a spacecraft unit needs to know the goals, policies and current state of the entire swarm. The computational model is a probabilistic one because there is uncertainty and lack of information needed to determine the state of the entire swarm. The formal model for computing both the nominal state s_n and the *set of self-healing actions* A_{sh} is a tuple

$$M_n = < S; P; G; A; T; Z; O; R> \quad (10)$$

where:

- S is a finite set of global states of the swarm.
- An initial belief state $s_0 \in S$ is based on $p_0\ (s_0;\ s_0 \in S)$, which is a discrete probability distribution over the set of swarm states S, representing for each state the unit's belief that the swarm is currently occupying that state.
- P is a finite set of global policies of the swarm. A policy is a set of semantically related *rules* and *constraints*.
- G is a finite set of swarm goals. A goal g ($g \in G$) may be presented as a *desired transition* from a state to another, i.e.,

  $$g = (s \Rightarrow s') \quad (11)$$

  Note that a goal considers a *transitive transition* $\Rightarrow$, where in order to get to a state s' from a state s the system will go through a numerous intermediate state transitions, possibly at the unit level, i.e.,

  $$\Rightarrow = \{ s_1 \rightarrow\ s_2 \rightarrow, \ldots, \rightarrow s_n \} \quad (12)$$

- A ($A_{sh} \subset A$) is a finite set of actions that may be undertaken by the spacecraft units of the swarm.
- $T: S \times A \rightarrow \Pi(S)$ is the state transition function, giving for each swarm state S and spacecraft unit action A, a probability distribution over states. Here, $T\ (s;\ a;\ s')$ computes the probability of ending in state s', given that the start state is s and the unit takes action a, $p\ (s'|s;\ a)$.
- $O: A \times S \rightarrow \Pi(Z)$ is the observation function giving for each swarm state S and action A, a probability distribution over observations Z. For example, $O\ (s';\ a;\ z)$ is the probability of observing z, in state s' after taking action a, $p\ (z \mid s';\ a)$.
- $R: S \times A \times P \times G \rightarrow R$ is a reward function, giving the expected immediate reward gained by the unit for taking an action a from a state s, e.g.,

  $$r = R\ (s;\ a;\ P;\ G) \quad (13)$$

  The reward r is a scalar value in the range [0..1] determining, which action should be undertaken in compliance with the swarm goals and policies.

The model for smart self-healing strategy must:

1) pick up a *destination nominal state* s_n that will "move" the swarm closer to its current goal g, i.e., s_n shall be one of the possible intermediate states (see formula (12)) that is closest to the goal state s' (see formula (11));
2) based on the chosen nominal state, pick up the self-healing set of actions A_{sh} ($A_{sh} \subset A$) and assign performers (spacecraft units) to them.

Note that the repair of a spacecraft unit is usually a self-task performed by the faulty unit, but it may also involve other spacecraft units. In such a case, a ruler or an idle worker must drive the self-healing process and assign the self-healing actions A_{sh} to spacecraft units. Here, the planning and scheduling algorithms for ANTS might be borrowed from [5]. A smart self-healing strategy may decide that it is not worth repairing a faulty unit, but may decide to destroy it or transform it, e.g., from a worker to a ruler.

IV. Conclusions

This paper has presented our theoretical models for self-initiation for team formation and self-healing in swarm-based space exploration systems such as NASA ANTS. Both self-initiation and self-healing are possible, because ANTS incorporates self-awareness capabilities helping the swarm detect and react to changes. Self-initiation is the first step of the team formation process where an idle ruler (a special ANTS spacecraft) automatically determines the need of a new team and starts the team formation procedure. Our formal model for team formation is based on the Partially Observable Markov Decision Processes and Discrete Time Markov Chains where we do not consider any central controller, but complex algorithms working on state-action relationships and considering a variety of probability values.

Our theoretical model for self-healing in ANTS is based on a sort of "smart" self-healing strategy that is built by the swarm on-the-fly by taking into consideration the mission goals and policies. Thus, such a strategy may decide not to repair a faulty spacecraft unit, because, for example, the repair process might have a bad impact on the mission goals. Instead, the faulty unit might be transformed, destroyed, or left unrepaired. Future work is mainly concerned with implementation of our models and simulated experiments.

Acknowledgement

This work was supported by Science Foundation Ireland grant 10/CE/I1855 to Lero—the Irish Software Engineering Research Centre.

References

[1] Science Clarified ‚*How Humans Will Mine Asteroids and Comets*, Science Clarified, http://www.scienceclarified.com/scitech/Comets-and-Asteroids/How-Humans-Will-Mine-Asteroids-and-Comets.html

[2] W. Truszkowski, M. Hinchey, J. Rash and C. Rouff, NASA's swarm missions: The challenge of building autonomous software, *IT Professional*, vol. 6(5), pp. 47-52, 2004.

[3] M. L. Littman, Algorithms for Sequential Decision Making, PhD Thesis, Department of Computer Science, Brown University, 1996.

[4] W. J. Ewens and G. R. Grant, "Stochastic processes (i): poisson processes and Markov chains", Chapter in Statistical methods in Bioinformatics, 2nd edition, Springer, New York, 2005.

[5] E. Vassev, M. Hinchey and P. Nixon, A Formal Approach to Self-configurable Swarm-based Space-exploration Systems. In *Proceedings of the 2010 NASA/ESA Conference on Adaptive Hardware and Systems (AHS-2010)*, IEEE Computer Society, 2010, pp. 89-96.