The CPSwarm Project: Swarm Modelling in Self-organization and Swarm Intelligence in Cyber Physical Systems

Alessandra Bagnato, Etienne Brosse, and Alexandre Beaufays

Softeam Research & Development, 21 avenue Victor Hugo, 75016 Paris, France
{alessandra.bagnato,etienne.brosse}@softeam.fr

Abstract. The paper describes the Swarm Modelling in the context of the CPSwarm project, The project proposes a new science of system integration and tools to support engineering of CPS swarms.CPSwarm tools will ease development and integration of complex herds of heterogeneous CPS that collaborate based on local policies and that exhibit a collective behavior capable of solving complex, industrial-driven, real-world problems.

Keywords: System modeling · Design exploration
Herds of heterogeneous CPS

1 Introduction

CPSwarm is a 36-months Research and Innovation Action (RIA) funded under H2020 call ICT-01-2016. The project aims at defining a complete toolchain, enabling the designer to:

- Set-up collaborative autonomous CPSs;
- Test the swarm performance with respect to the design goal and
- Massively deploy solutions of "reconfigurable" CPS devices and CPSoS.

Within the project Softeam developed a dedicated CPSwarm modelling language on top of System Modeling Language (SysML) standard. SysML is a general-purpose visual modelling language for systems engineering applications defined as a profile of the Unified Modeling Language (UML) standard. It supports the specification, analysis, design, verification and validation of a broad range of systems and systems-of-systems. These systems may include hardware, software, information, processes, personnel, and facilities.

Within the CPSwarm modelling language we used SysML concepts to fulfill the CPSwarm end user modeling needs. A dedicated CPSwarm UML2 Profile has been defined which includes UML/SysML customizations, and specific graphical notations [1].

The following paragraphs provide a quick description of CPSwarm project and also CPSwarm Modelling language definition.

Bagnato, A., Brosse, E. and Beaufays, A.
The CPSwarm Project - Swarm Modelling in Self-organization and Swarm Intelligence in Cyber Physical Systems.
DOI: 10.5220/0008862200910099
In *OPPORTUNITIES AND CHALLENGES for European Projects (EPS Portugal 2017/2018 2017)*, pages 91-99
ISBN: 978-989-758-361-2

2 The CPSwarm Project

2.1 CPSwarm Project Overview

Driven by industrial needs, which are expressed through industrial use cases cf. Section 2.2, the CPSwarm project aims to:

1. Drastically Improve support to design of complex, autonomous CPS
2. Provide a self-contained, yet extensible library of re-usable models for describing Cyber Physical Systems
3. Enabling a sensible reduction in complexity and time of CPS development workflow by automating deployment
4. Define a complete library of swarm and evolutionary algorithms for CPS design
5. Establish reference patterns and tools for integration of CPS artefacts
6. Address real industrial needs in CPS design, with a particular focus on the autonomous robotic vehicles, freight vehicles and smart

A set of requirements has been defined according to the project goals defined earlier. These requirements will be evaluated in the context of three case studies described below.

2.2 CPSwarm Case Studies

CPSwarm results will be applied and evaluated on the following case studies:

Swarm Drones
Heterogeneous swarms of ground robots/ rovers and UAVs to conduct certain missions in surveillance of critical infrastructures like, e.g., industrial or power plants. These missions includes:

- Intrusion detection (detection of unauthorized persons entering the plant area),
- Monitoring of actions of unauthorized persons in the plant areas.

Search and Rescue tasks can also be fulfilled by these heterogeneous swarms. These tacks might include:

- Generating a situation overview of the disaster scene in case of an industrial plant accident including real-time images (VIS, IR), toxic and explosive gas leakage detection,
- Finding of human casualties or persons trapped in the disaster area.

Automotive CPS
Applications for collective driving with a focus on autonomous driving vehicles intended for freight transportation independent vehicles could join or leave a swarm at any point during the journey. The leading vehicle has autonomous driving capability and prescribes the actions and decisions for the follow-up vehicles. The follow-up vehicles have also autonomous driving capability and follow, if possible, the leading vehicle’s actions.

Swarm Logistic Assistant

Focus on robots and rovers designed to assist humans in logistics domain, in this case the CPS swarm will be able to:

- Scan the entire area of the warehouse and share the acquired information;
- Collect information about the maps of the entire area;
- Collect additional information implicitly e.g. room temperature, presence of humans, detection of in-path obstacles etc.
- Join forces to move a heavy obstacle from one place to another.

3 CPSwarm Modelling

The CPSwarm modelling tool is built on top of Modelio open source modelling environment https://www.modelio.org/. CPS Swarm modelling activity can be succinctly described as the creation and population of several diagrams or views. Each of these view has a specific and particular purpose. The following sections describe how to model a simple but complete swarm.

3.1 CPSwarm Model Template

Modelling is always a difficult task especially when it is carried out from scratch. In this case, guidance is helpful. The main goal of the CPSwarm template generation is to help the Modeler to create a simple CPS swarm model with all minimum concepts. As depicted in the following figure (Figure 1), the model generation using the CPS Swarm template can be easily be done just by right clicking on the root model package.

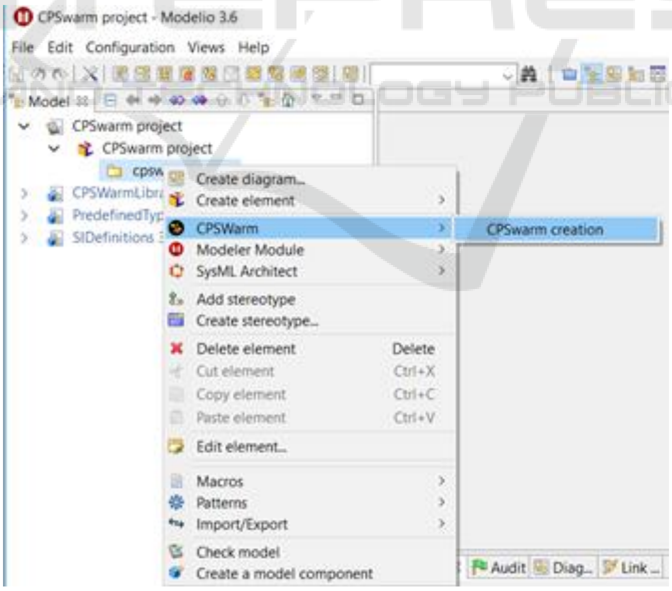

Fig. 1. Creating a new swarm modelling.

Figure 2 shows the result of the CPS swarm template generation in terms of created model elements.

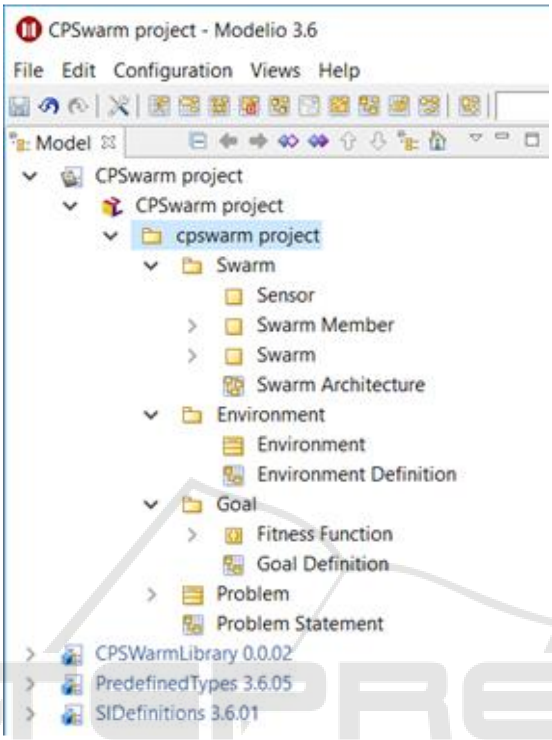

Fig. 2. New swarm result.

The swarm template generator produces a set of initial diagrams (as shown in Figure 3) that have been identified as necessary to completely model a CPS swarm.

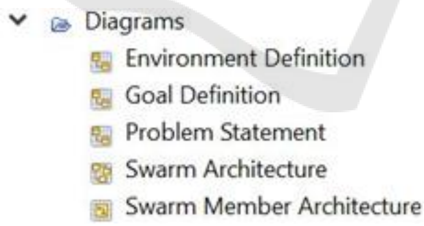

Fig. 3. CPSwarm predefined diagrams.

Then the CPSwarm modeler can modify the initial content following the needs of the specific case study he/she is modelling. The CPSwarm modeler will find on the right part of each of the diagrams (namely Environment Definition, Goal Definition, Problem Statement, Swarm Architecture, and Swarm member architecture) the predefined selection of the modelling elements he/she can specifically use for that specific diagram context.

Three main concepts compose a CPSwarm model:

1. CPS Swarm itself which is composed of several type ofCPS.
2. CPS architecture which is composed Hardware component as Sensor, Controller or Actuator.
3. CPS behavior which is described as a UML state machine.

A more detailed description of these three main concept are given in the following sections. Note that these notions not the only one available on a CPSwarm model. In CPSwarm project context, a CPS Swarm might evolve inside one or many physical environment for a specific goal. These two notion of Environment and Goal are also part of the CPSwarm model but are not detailed here.

3.2 CPSwarm Architecture Modelling

A swarm is composed of one to many type of CPS aka Swarm Member. To model this relation, you must use the UML composition relation from the Swarm block to one or many Swarm Member block. The multiplicity at the end of the relation indicates the number of Swarm Member instance. Figure 4 depicts a simple Swarm composed of one unique Swarm Member.

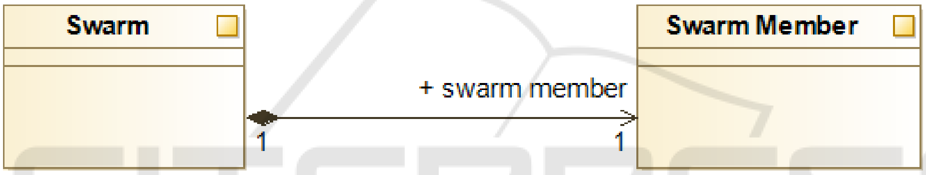

Fig. 4. Swarm Architecture Modelling Elements.

3.3 Swarm Member Architecture Modelling

Another aspect of Swarm Modelling is the specification of each Swarm Member internal architecture. This specification is made in two times. First, the list of internal components (which can be a controller, a sensor, or an actuator component) must be defined. Each of this internal component must expose the data it provides or requires. Figure 5 represents a simple component having two ports respectively named FlowPort and FlowPort1. FlowPort expresses the fact that the component provides a Boolean value at contrary FlowPort1 expresses the fact that the component requires a Boolean

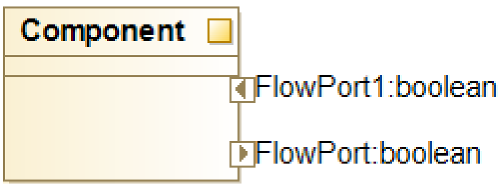

Fig. 5. Simple sub Component example.

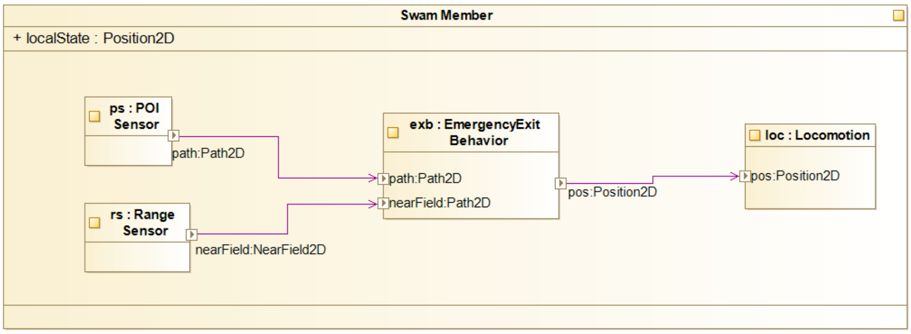

Fig. 6. Swarm Member Architecture Example.

The Figure 6 depicts the internal architecture of a Swarm Member. This diagram shows a “localState” attribute but also four internal components (parts) respectively named “ps”, “rs”, “exb”, and “loc”. These four parts are connected through specific point (port) named “path, nearfield, or “pos”. The data going through the connection is model thanks to interface (not shown in this diagram) but named “Path2D”, “Near-Field2D”, and “Position2D”.

3.4 Swarm Member Behaviour Modelling

The internal architecture of a Swarm Member is a key aspect of its definition. The second key aspect is its internal behavior. UML state machines are used to model Swarm Member behavior. Figure 7 depicts the simplest possible Swarm Member behavior. This latter is simply composed of a State named State. Both Initial and Final state are mandatory to all State Machine. The two transitions respectively connect the initial State to the State state and the State state to the Final state

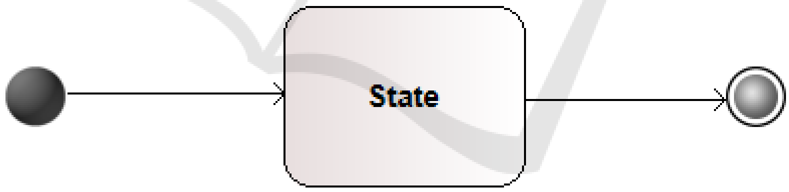

Fig. 7. Simple Swarm Member Behaviour.

Of course, a real behavior will be more complex. Figure 8 for example represents two states - aka State1 and State 2 – executed in parallel.

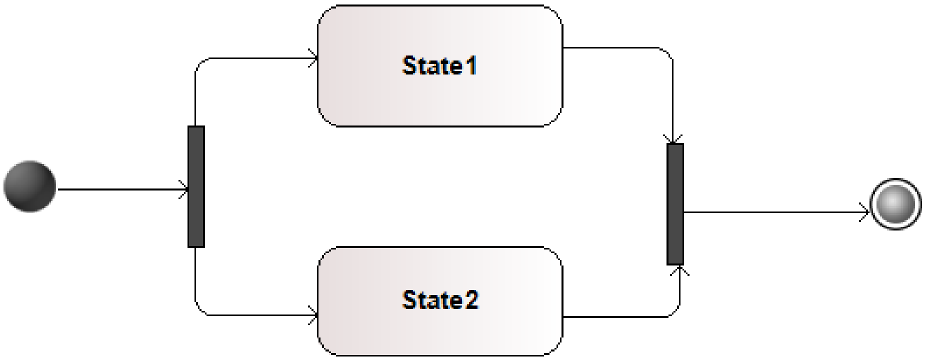

Fig. 8. Swarm Member Behaviour.

So the Swarm Member behaviour, shown in Figure 9, depicts with five states “Read sensor data”, “Update routing table”, “Wait w0”, “Wait wmax”, and “Move randomly”. The first state reached is the “Read sensor data”. According to the data collected the CPS/Swarm Member waits for a specified time w0, for another one, called wmax, or moves randomly. In any case, he updates its routing table and then reads again sensor data.

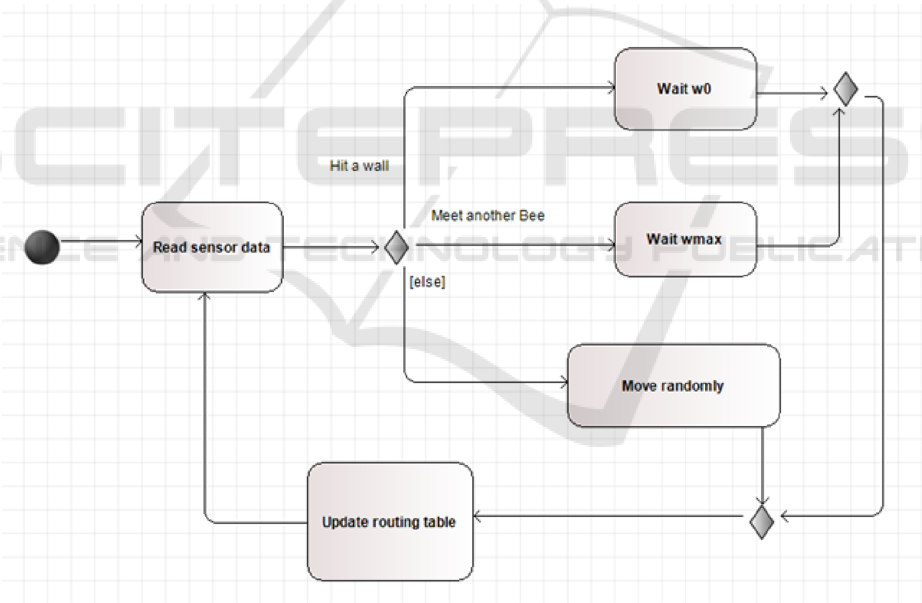

Fig. 9. CPS behaviour example.

To handle the complexity of these state machine it is possible to extract part of them inside another state machine and then refer this extracted content as a sub state machine. Figure 10 shows the call of a sub state machine by a particular State.

Fig. 10. Hierarchical State.

Figures 11 and 12 represent the behaviour depicted in Figure 9 but using a hierarchical state name "Analysis" in order to clarify and allow re usage of the CPS behavioural modelling.

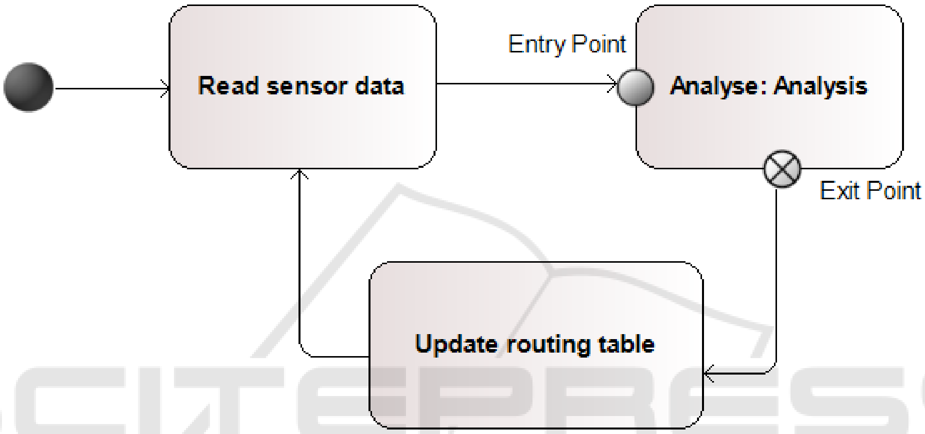

Fig. 11. Top state machine example

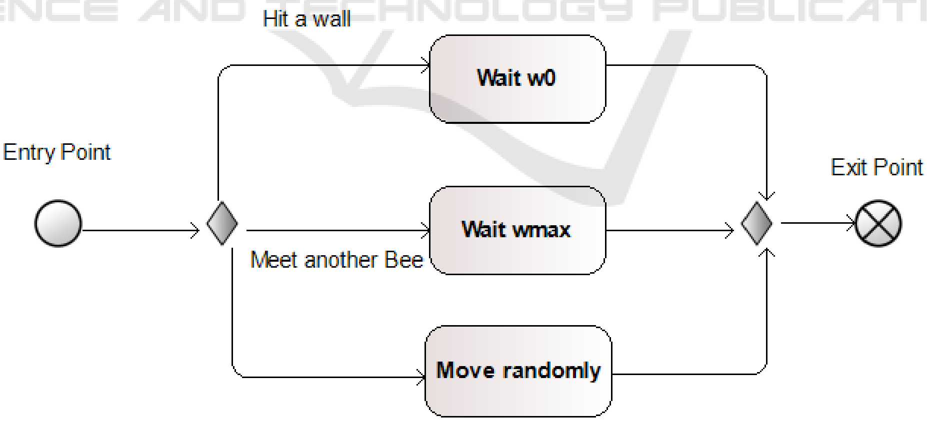

Fig. 12. Low state machine example.

4 Conclusions

The book chapter describes the CPSwarm project as it has been presented during the European Project Space Workshop within MODELSWARD 2018 in Madeira and concentrates on the swarm modeling part of the project managed by Softeam.

Acknowledgments. The research presented in this book chapter is funded by the H2020 Project CPSwarm started on 1st January 2017 and running till 31st December 2019.

References

1. Etienne, B, Beaufays, A and Bagnato, A. (2018). Modelio Forge CPSwarm Module. https://forge.modelio.org/projects/cpswarm. Last accessed on 2018-07-18.