Indian Journal of Science and Technology, Vol 8(19), DOI: 10.17485/ijst/2015/v8i19/76692, August 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Designing Goal Model for Autonomic Control Point of Cyber-Physical Systems (CPS)

Jeongmin Park[1]*, Seunghwa Lee[2] and Taebok Yoon[3]

[1]Department of Computer Engineering, Korea Polytechnic University, 237, Sangidaehak-ro, Siheung-si, Gyeonggi-do, South Korea; jmpark@kpu.ac.kr
[2]Division of Information and Communication, Baekseok University 76, Munam-ro, Dongnam-gu, Cheonan-si, Chungcheongnam-do, SouthKorea; sh.lee@bu.ac.kr
[3]Department of Computer Software, Seoil University Seoul, 131–702, South Korea; tbyoon@seoil.ac.kr

Abstract

Goal model provides essential knowledge determining autonomic control point and abstracting the complicated management problem. Most of the existing methods suffer from a common problem, in that the developers have to analyze the complex target CPS. Namely much time and effort is required to inspect the CPS. Thus, this paper offers a method to build a goal model for the autonomic control of Cyber-Physical Systems (CPSs). This approach attempts to define and apply the system knowledge required for the autonomic control from the goal model.

Keywords: Goal Based Autonomic Computing, Goal Modeling, Self-Healing

1. Introduction

Cyber-Physical Systems (CPS) is integrated systems that involve a network, software and physical system including sensors and actuators[1]. In general, CPSs can be divided into two parts, the physical part that manages the physical environment or objects required in the real world and the cyber space that retains the environment operated by software. Therefore, a CPS is an integrated system including 'Hardware' and 'Software'.

However, due to complexity and management cost on the rise for CPSs, it is hard to retain and operate CPSs in a complicated computing environment. Especially, it is difficult to recognize the diverse problems arising in CPSs and solve the problems that arise. Those people directly controlling CPSs should be professional standards, which overcomes limitations such as the increasing human resources and management cost[2,3].

As a way to deal with these problems, 'Self-Adaptation' research on high-reliability for CPSs has recently become an important issue. The 'Self-Adaptation' approach comprises 1. Goal modeling, 2. Monitoring, 3. Analysis, 4. Planning and 5.Execution. Of particular importance among these issues is 'Goal Modeling', because the 'Goal Model' provides essential knowledge facilitating 'Self-Adaption' and abstracting the complicated management problems.

A 'Goal Model' in this context is a model that describes the 'Goal' to be achieved in the Cyber-Physical System (CPS). That is, the goal model includes the core knowledge that enables high-reliability to be retained in the CPS. Thus, this paper presents a method of generating a 'Goal Model' for autonomic control that spontaneously offsets the problems in curring in the CPS. This approach consists of various steps, viz. 1. Goal extraction, 2. Goal interpretation, 3. Association between goal tree and designed model and 4. Linking leaf-goal and constraints. Through these activities, it is possible to help ensure that the 'Goal Model' is able to provide the key knowledge required for autonomic control in order to rectify the abnormal operations of the CPS.

*Author for correspondence

2. Preliminary Research and Problem Statement

2.1 From Goal Model to Autonomic Control System Knowledge

The 'Goal Model' promises to reduce the complexity and software maintenance costs of the CPS in an 'Autonomic Control System'[4].

Figure 1 shows the architecture of the 'Autonomic Control System' (This architecture is depicted by)[4]. An 'Autonomic Control System' is an architectural component called an Autonomic Manager that controls and manages elements such as resources, components, etc. In order to accomplish this, the 'Autonomic Control System' has to be endowed with the tools required for monitoring the elements being managed, examining the gathering data to decide whether these factors are executing as anticipated, establishing strategies to deal with problems detected, and executing the planning strategy, in order to accomplish the goals of the 'Autonomic Control System'.

2.2 Problem Statement

There are few related works and applicable cases for the autonomic control of CPSs, because this research is still in the initial stage. However, there are three key methods to build an 'Autonomic System'. The first approach is to make planning capabilities and self-configuration skills in order to reconfigure the task through external system components[5]. The second way is the artificial intelligence approach to autonomic systems[6]. The third way is to ensure the high-reliability of the CPS. Several approaches have been proposed to develop autonomic control systems in the CPS research community, including a 'Reliable Home Service Robot[7]', 'Vehicular Cyber-Physical System[8]', 'Self-Adaptive Robot[9]'. These approaches were developed to make the systems more reliable. On the one hand, most of the existing methods suffer from a common problem, in that the developers have to analyze the complex target CPS themselves. Namely, much time and effort is required to inspect the system. If it were possible to assign core knowledge to the autonomic control system using a 'Goal Model', it would be possible to reduce this time and effort. In this paper, we consider the collaboration between the designer and developer, and propose an approach to generating a 'Goal model' for autonomic control. Through the proposed factors, it is possible to recover from abnormal problems and to operate the system safely.

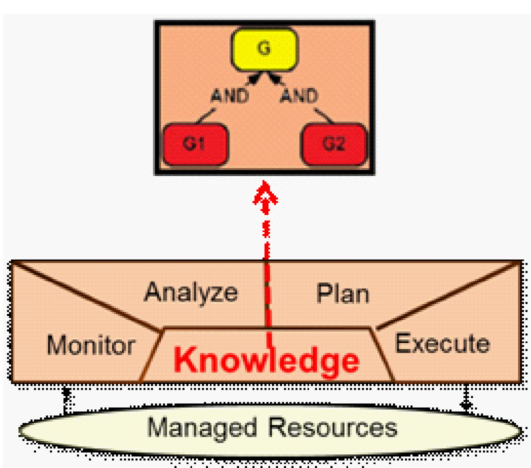

Figure 1. Architecture and goal model for autonomic control system.

3. Goal Modeling Requirements for Autonomic Control

In this section, we briefly describe the requirements of 'Goal Modeling' for 'Autonomic Control' that can adapt to changes in the external and internal environment. These requirements are demonstrated in Table 1.

In order to meet the goal to be achieved in CPS environment, as described in Table 1, there are four ways to model the 'Goal' in order to make an 'Autonomic Control System'. Firstly, step1 is to extract the 'goal' using natural language. Secondly, step 2 is to interpret the extracted goal by analyzing the 'Goal ID' and generate a 'goal tree' to for man appropriate data structure. Step 3 is to associate the goal tree with the designed model for mapping the goal tree and sub-modules generated at design time, and then to link the constraints to ensure the normal system state in the goal tree. Finally, step 4 is to set autonomic control points if the constraints are violated.

Table 1. Goal model requirements

Phase	Description
Step1	Goal Extraction
Step2	Goal Interpretation and Goal Tree Generation
Step3	Linking Constraints
Step4	Setting Autonomic Control Point

An 'Autonomic Control System' has capabilities such as self-configuration, self-healing and self-optimization by having the ability to manage and reduce the complexity and maintenance costs of the CPS. The 'Autonomic Control' analyzes and changes the behavior of the various systems atrun-time. Therefore, the 'Autonomic Control System' needs to have a 'Goal Model' to achieve the management purposes of the 'target system'. The 'Goal Model' is the core knowledge of the 'Autonomic Control System'[1].

4. CPS Environment

This section briefly introduces an 'Intelligent Service Robot (ISR)' described in previous research[10]. We apply this ISR (see Figure 2) to our 'Goal Model'.

4.1 Intelligent Service Robot for CPS

As presented in Figure 2, the ISR is a service robot that has various sensors and carries out the tasks it is commanded to do. A user application enables interaction with the ISR. Herein, we explain the role of the CPS. The User Application operates in 'Cyber Space' to manage the environment operated by software, while the ISR operates in the real world, together forming the CPS.

4.2 Scenario

The action scenario of the user application is presented in Figure 3 (ISR2.0 with Arm board)[7]. When the user draws the desired zone or room by pushing the requisite button, the user application creates the drawing path and sends the route to the ISR. The web camera sends the images around the ISR to the web server. Then, through the internet, the user application shows the images to the user.

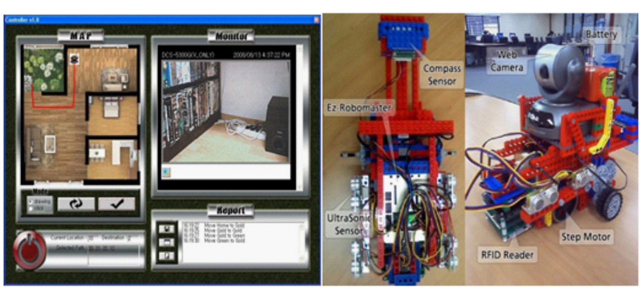

Figure 2. User application and 'Intelligent Service Robot (ISR)'.

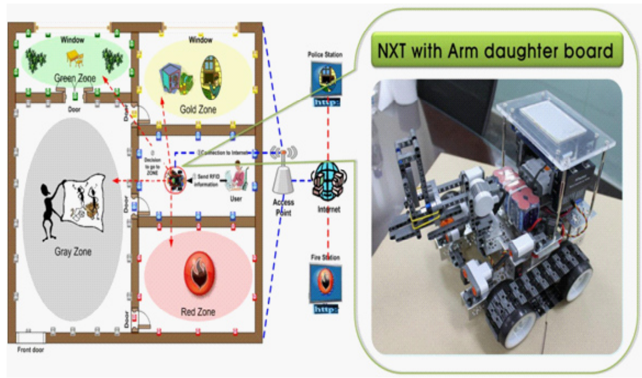

Figure 3. Scenario of user application.

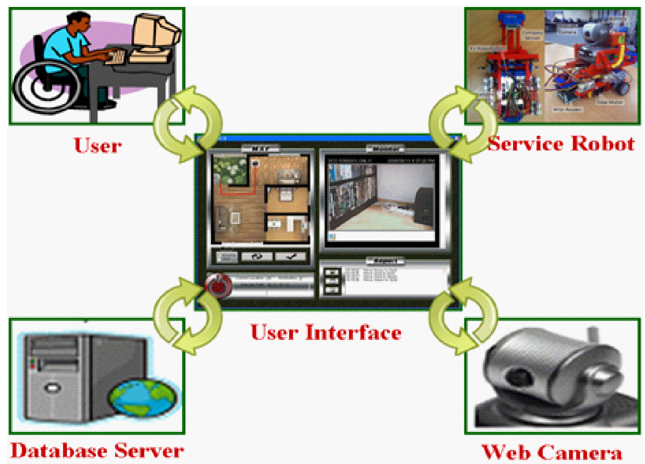

Figure 4. Architecture of CPS.

As presented in Figure 4, the CPS consists of the user, service robot, user interface, database server and web camera. The web server enables the system to be accessed from the outside and saves the log data using the DB server (see Figure 4).

The user interface sends the user's commands to the robot and shows the resulting information. It communicates with the robot via Bluetooth and is linked with the internet through a web server. The user interface has two modes: drawing mode, in which the user can directly draw the navigation zone, and click mode, in which the user can select the room that the robot is to navigate to by pushing the appropriate button.

Once navigation begins following the user's command, the route from the current location of the robot to the final destination is determined and sent to the robot. After the robot starts making its way to the destination, its location can be tracked using the ID number of the RFID Tag that is relayed back.

5. Applying Goal Model to CPS Environment

In this section, we briefly apply the 'Goal Model' to an 'Intelligent Service Robot (ISR)'. To illustrate the role of the goal model and what it can do for the design of an autonomic control system for an ISR, we examine an example in which the ISR has the goal of 'Driving straight'.

5.1 Step1: Goal Extraction

The Goal Extraction step is used to extract the goal in the ISR using natural language. Table 2 explains the process of Goal Extraction.

5.2 Step2: Goal Interpretation and Goal Tree Generation

The Goal Interpretation and Goal Tree Generation step interprets the extracted goal and generates the goal tree according to the dependencies of the goal IDs, such as G1, G2, etc. Figure 5 shows the goal tree generated by Step 1, which is Goal Extraction.

Table 2. Goal extraction

System's Goals	
G1	Driving Straight
G2	Angle variation of compass sensor is 0(zero)
G3	Arriving within 10 seconds
G4	The number of rotations has to be consistent
G5	…
G6	…
GN	…

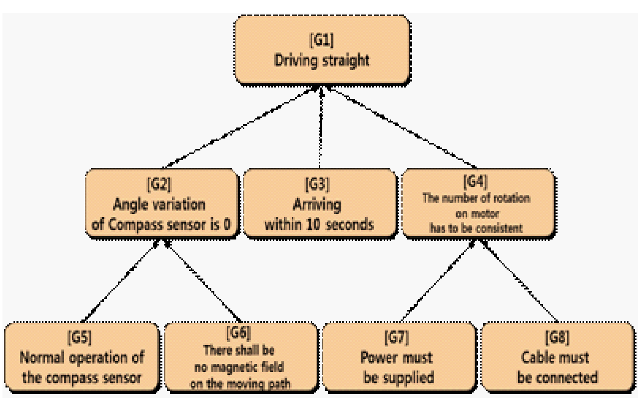

Figure 5. Goal tree generated based on Step1.

5.3 Step3: Linking Constraints

The Goal Linking Constraints step consists of two parts. The first step is to associate the goal tree with the designed model for the purpose of mapping the sub-modules generated at design time. It is depicted in Figure 6.

The second is to link the constraints to ensure the normal state in the goal tree. A leaf-node linked with the constraints is a sub–module/system-state or sensor name. Figure 7shows the link between the constraints and goal-tree.

For example, [G7]'s constraint, constraint5, means that 'Battery Power' must be supplied. Constraint6 explains that a cable must be connected.

5.4 Step4: Setting Autonomic Control Point

The purpose of the Setting Autonomic Control Point step is to identify the constraints that have been violated in order to establish an 'Autonomic Control Point (ACP). It is imperative to detect and identify ACPs, because this reduces the management complexity of the CPS

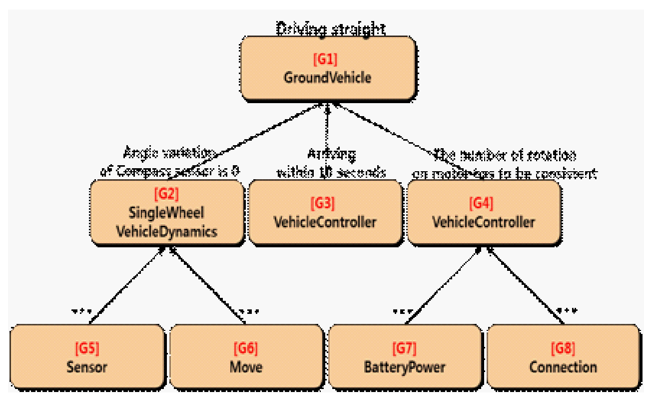

Figure 6. Mapping goal tree and sub-modules.

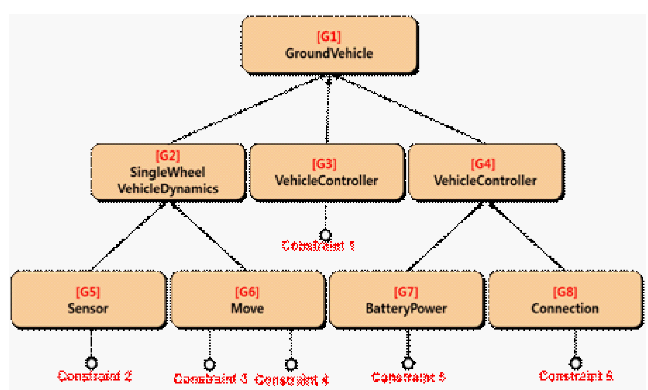

Figure 7. Linking constraints and goal tree.

and makes it possible to realize an 'Autonomic Control System'. Figure 8 shows an example of detecting violated constraints.

6. Evaluating Goal Achievement

In this section, we evaluate whether the goals applied to the 'Intelligent Service Robot (ISR)' are achieved. In order to evaluate the application of the 'Goal Model' to the ISR, as shown in Figure 9, we include the following information. Firstly, the Goal ID is the identifier (i.e. G1, G1.1, G1.2, …, G1.3.2). The Goal Name is the name of the goal to be achieved. The Weight (w) is the value to be accomplished for each of the goals established by the designer. Mandatory (M) means that (this step?) is necessary to accomplish the mission.

We implemented an 'ETRI-detector' as a prototype for evaluating the goals. It inspects the mapped constraints on the goal tree to determine whether there is an abnormal state or not. Figure 10 presents these activities. The violated constraint is constraint6. The weight of constraint6 is about 0.07 (that is 0.2 / 3).

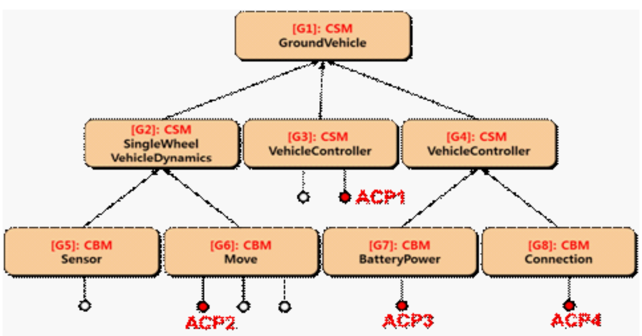

Figure 8. Setting Autonomic Control Point(ACP).

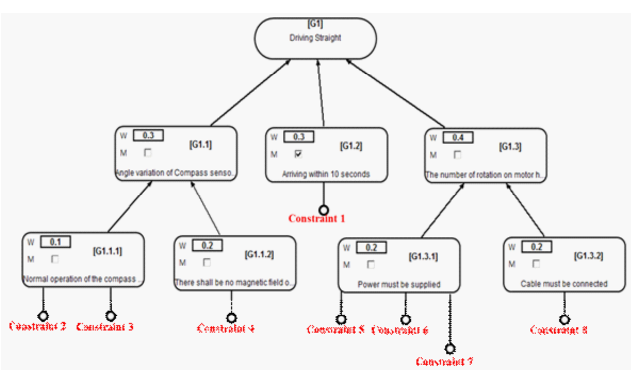

Figure 9. Goal Model applied to 'ISR'.

Using the function depicted in Figure 11, we compute the goal model's achievement by evaluating the success ratio of the constraints: Y = (0.3 + 0.05 + 0.05 + 0.2 + 0.07 + 0.07 + 0.2)* 100. Thus, the degree of achievement of the goal, Y, is about 94 percent. In Figure 12, the state monitor presents a red color as evidence of a problem arising in the CPS environment.

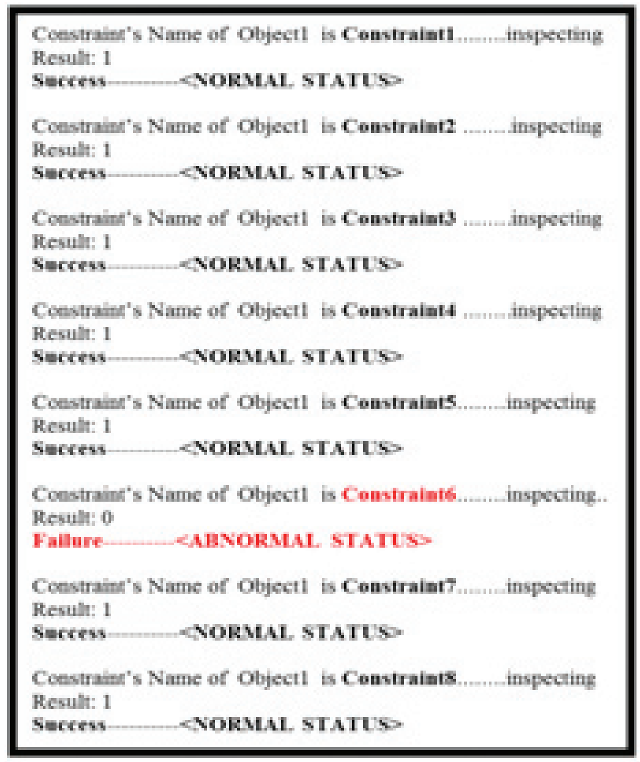

Figure 10. Setting autonomic control point.

$$X \longrightarrow \boxed{\sum_{i=1}^{n} Success\ Weight(i) * 100} \longrightarrow Y$$

Figure 11. Function calculating goal achievement ratio.

Figure 12. State monitor in User Application.

7. Conclusion

In this paper, we have proposed the following process for designing goal model based on autonomic control point for CPS: goal extraction, goal interpretation, association between goal tree and designed model, and linking leaf-goal and constraints. Through these activities, it was possible to help ensure that 'Goal Model' is able to become a key knowledge towards autonomic control which adjusts CPS's abnormal operation. However, in order to improve the goal achievement, designing extended goal model is elemental. Thus, it needs to link a goal model and fault event that occurs in the CPS. As way to deal with these problems, goal model based autonomic control method is required. The approach has the issues comprising 1. Goal modeling, 2. Monitoring, 3. Analysis, 4. Planning and 5. Execution. Therefore, in the future work, through these activities, the present research will be improved by focusing on analyzing and testing the goal of CPS.

8. References

1. Park JM, Kang SJ, Chun IG, Kim WT. An approach to generating goal model for cyber-physical systems. Proceedings of International Symposium on Embedded Technology; 2013 May; Daegu, Korea. p. 23–4.
2. Kephart JO, Chess DM. The vision of autonomic computing. IEEE Computer Society.2004 Jan; 36:41–50.
3. Rechard L. Creating robust software through self-adaptation. IEEE Intelligent Systems. 2000 May; 15:26–9.
4. Lapouchnian A, Yu Y, Liaskos S, Mylopoulos J. Requirements-driven design of autonomic application software. Proceedings of DEAS 2005; 2005 May; LA, USA. p. 20–6.
5. Lee JH, Park JM, Yoo GJ, Lee ES. Goal-based automated code generation in self-adaptive system. Journal of Computer Science and Technology. 2010 Nov; 25:1118–29.
6. Park JM, Youn HS, Lee ES. An automatic code generation for self-healing. Journal of Information Science and Engineering. 2009 Nov; 25:1753–81.
7. Chun IG, Lee HY, Kim WT, Lee ES. SMART-CPS: Self-Managed Reliable System Development Method for Cyber-Physical Systems. Communications in Computer and Information Science. 2011 Nov; 256:138–48.
8. Li X, Qiao C, Yu X, wagh A, Sudhaakar R, Addepalli S. Toward effective service scheduling for human drivers in vehicular cyber-physical systems. IEEE Transactions on Parallel and Distributed Systems. 2012 May; 23(9):1775–89.
9. Chun IG, Kim JM, Lee HY, Kim WT. Faults and adaptation policy modeling method for self-adaptive robots. Communications in Computer and Information Science. 2011 Sep; 150:156–64.
10. Park JM, Chun IG, Lee HY, Kim WT, Lee E. Intelligent service robot and application operating in cyber-physical environment. Lecture Notes in Electrical Engineering. 2011 Sep; 181:301–10.