Experience in Adopting an Embedded Real-Time Component Model in Autonomous Mobile Robot Software Design

Dayang Norhayati Abang Jawawi and Safaai Deris
Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, 81310 Johor Bahru.
dayang@fsksm.utm.my

ABSTRACT
Recently, Component-Based Software Engineering (CBSE) has become a popular approach to increase software productivity, improve software quality, consistency and reliability, and at the same time decrease the costs of software development. To bring the advantages of CBSE to Autonomous Mobile Robot (AMR) software, special domain considerations must be taken into account such as embedded system issues, real-time software issues, control theories and artificial intelligence aspects. In CBSE, a component model is required to specify the standards and conventions imposed on developers of components. Industrial component models such as CORBA, COM and JavaBeans are too expensive in terms of processing power and memory, hence not suitable in the Embedded Real-Time (ERT) systems. Consequently, a number of component models suitable for CBSE of ERT software such as PBO, Koala, PECOS and ReFlex are introduced. All these ERT component models have their own unique strengths to support their nature of ERT problem domain. This paper describes the experience in adopting PECOS ERT component model in developing AMR software. Based on the assessment of this adoption some modifications to PECOS model on addressing the issues of ERT software for resource constraint AMR, predictable real-time behavior of the AMR and platform-independent AMR software component are suggested.

KEYWORDS
Component model, real-time systems, embedded systems, autonomous mobile robot.

1. Introduction
Developing software for Autonomous Mobile Robot (AMR) is difficult and required knowledge in embedded systems, real-time software issues, control theories and artificial intelligence aspects. To tackle the difficulty in developing software for AMR, many robotics research communities [1], [2], [3], proposed the use of Component Based Software Engineering (CBSE) approach. A component-based solution is expected to help robotic research groups in the following aspects [4]:

- exchange of software parts or components between robotics laboratories, allowing specialists to focus on their particular field.
- comparison of different solutions would be possible from the available components.
- startup in robot research can be accelerated using the available components.
- speed up the transfer of research laboratories works in robotics to commercial business application.

There have been some efforts on providing CBSE of mobile robot software [1], [5]. However, most of these works do not address the issues of ERT software for resource constraint AMR and predictable real-time behavior of the robot. Furthermore, only platform-dependent components were considered in their work.

In CBSE, a component model specifies the standards and conventions imposed on developers of components in order to achieve uniform composition, appropriate quality attributes and deployment of components and applications [6]. There are a number of component models suitable for ERT software such as ReFlex [7], PBO [8], Koala [9] and PECOS [10]. All these ERT component models have their own unique strengths to support their nature of ERT problem domain. Currently, none of these component models is specifically targeted directly for AMR software development.
Our research objective is to study CBSE approach for software development of AMR software, which emphasized on three requirements: facilitates predictable real-time performance of the components assembly, support for resource constraint embedded AMR systems and support high-degree of platform-independent component. This paper discusses the possibilities of using an ERT component model called PECOS (PErvasive COmponent Systems) in our AMR software and analyzes the consequences of the adoption on the above three requirements. In Section 2, the used of PECOS model in AMR software will be illustrated using a case study of AMR software. Modifications to PECOS in addressing AMR software problems are also suggested in Section 3. Finally, the conclusion is presented in Section 4.

2. Adopting PECOS Model

In traditional development, system architecture design is based on the system requirements, and the design process continues with a sequences of refinement from the initial assumptions to the final design goal. In contrast with traditional development, in component based development, many decision related to the system design will be a consequence of the component model selected. The component model defines the collaboration between the components, while component framework provides the infrastructure supporting this collaboration. In this research, the PECOS component model will be adopted as a core component model for our AMR CBSE due to its favorable strength identified previously in our evaluation of ERT component models [11].

Software design using components involves connecting sets of component to create a software system capable of performing some useful function. This is called assembling activity. This assembling activity consists of component integration and component composition [12]. The goal the design activity is to find the most appropriate and feasible combination of the component candidates.

2.1 Methodology for Assessing PECOS Component Model

Assessment of PECOS component model will be performed following these steps. First we illustrate the design process of an AMR software case study using PECOS model by performing the component integration and component composition. Based on this case study, the component integration, component composition and the implementation of the composition are performed. We then assess the PECOS model on the following aspects: 1. Facilities for predictable real-time performance of the components assembly; 2. Support for resource constraint AMR systems; and 3. Support of platform-independent implementation.

The AMR used in the case study is a differential drive wheeled robot, capable of traversing in a structured environment. The goal of the robot software is to control the movement the robot while avoiding obstacles in its environment. The AMR consists of a body and a pair of wheels. Each drive wheel is move by a direct-current (DC) motor. The speeds of the motors are sensed using shaft encoders and fed back to the embedded controller for computation of control signal to the DC motor every 100 milliseconds using the proportional-integral (PI) control algorithm. The embedded controller also monitors the robot environment using four infrared (IR) proximity sensors and communicates with human using Liquid Crystal Display (LCD) and switches.

2.2 PECOS Integration

PECOS model supports component integration with static structure and execution model. The static structure of the model is specified using component, ports and connectors. A component is a computational object with one of the following real-time behaviors: active behavior, with their own thread of control, passive behavior, without their own thread of control and events behavior, which triggered by events. A port represents data that is read and written by a component and the data-sharing relationship between the ports is described using connector.

Figure 1 shows the static structure integration of components to design the AMR software. The integration consists of four active components and eight passive components. Consider the active composite component named AvoidBehavior in Figure 1. The component is assembled using two passive subcomponents named BumperSensor and SetAvoidBehavior. The composite component has two outports desiredSpeedKiri and desiredSpeedKanan. Connection in PECOS model must follows some rules relating inports and outports and level of hierarchy.
The execution model describes actual runtime semantics and how data between components are synchronized. The PECOS synchronization and mutual exclusion are supported with Petri-nets using data places, control places and event places. Petri-net diagram in Figure 2 shows the synchronization and mutual exclusion between the inner ports and outer ports of data places (shown in darker color) using control places (shown in lighter color) of Petri-nets representation for AvoidBehavior component.

Based on this static structure and execution model, it can be concluded that component integration using PECOS model can explicitly describe the real-time behavior of a component using component types in static structure. Petri-nets diagram used in execution model provides synchronization models to enable prediction of timing properties.

2.3 PECOS Composition

Once the real-time behavior and data synchronization has been specified in the integration stage, the next stage in PECOS assembly process is to consider the reasoning of the integration in component composition activity. Component composition supports means of determining the properties of assemblies in order to predict their runtime compatibility [12]. The properties are specified using property bundle in PECOS model. This timing property can be predicted using Rate Monotonic Analysis (RMA) based on PECOS component composition approach [13].

RMA verification in PECOS is to check whether the entire component involved in the composition meet their deadlines. To use RMA to verify the component composition in Figure 1, a mapping is needed from the composed components to RMA tasks. Based on the PECOS mapping guidelines the mapping result of the MobileRobot components for this case study is tabulated in Table 1. The tasks in the table are ordered from highest to lowest priority. The result from the table is used as input for RMA analysis timing properties. The tasks worst-case execution time (WCET) is measured at runtime, and tasks period are obtain from the AMR requirements. WCET and period are examples of property bundle expressed in PECOS model.

Table 1: The AMR tasks.

<table>
<thead>
<tr>
<th>Task</th>
<th>Period (ms)</th>
<th>WCET (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MotorControl_exec</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>AvoidBehavior_sync</td>
<td>250</td>
<td>5</td>
</tr>
<tr>
<td>MotorControl_sync</td>
<td>250</td>
<td>5</td>
</tr>
<tr>
<td>HR1_sync</td>
<td>250</td>
<td>5</td>
</tr>
<tr>
<td>AvoidBehavior_exec</td>
<td>300</td>
<td>50</td>
</tr>
<tr>
<td>HR1_exec</td>
<td>500</td>
<td>125</td>
</tr>
</tbody>
</table>

We can then apply Theorem 1 from RMA to determine whether the deadline for each task in Table 1 can be met. We applied the theorem for value \( i \) ranging from 1 to 6, since there are six tasks derived from the AMR component.
composition. For each $i$ we found at least one possible pair of $(k, l)$ that satisfies the equation in the theorem. The results indicate that the design composition of components in Figure 1 is predicted to be schedulable according to RMA theory. In composition activity, PECOS approach shows how real-time theory such as RMA can be used to guarantee the AMR system predictability and performance.

**Theorem 1:** A set of $n$ independent periodic tasks will always meet its deadlines, for all task phasing, if and only if

$$\forall, 1 \leq i \leq n, \min_{(k,l) \in R_i} \sum_{j=1}^{k} \frac{C_j}{I_{T_i}} \left( \frac{I_{T_j}}{T_j} \right) \leq 1$$

where $C_i$ and $T_i$ are the WCET and period of task $i$, respectively, and

$$R_i = \{ (k,l) \mid 1 \leq k \leq i, l = 1, \ldots, \left( \frac{T_l}{T_i} \right) \} .$$

### 3. Assessment Results of the Adoption

PECOS integration, using static structure and execution model, supports static binding components at design-time composition. This allows optimization of the design and code toward resource constrain of AMR systems. PECOS design-time composition also supports platform-independent component technology, where the composition could be the instance of specific adaptation of components or codes toward specific microcontroller families and RTOS. We proposed to further enhance PECOS reusability by introducing configuration parameters to the components in their static structure. The configuration parameters are useful for AMR software to reconfigure components for use with specific hardware or application.

PECOS also use constraint solving approach to perform the schedule verification, since, the timing verification using RMA alone is claimed not enough when not all tasks run concurrently and hence schedule verification is needed to check the possibility to fit execution and synchronization behavior sequentially in each task. Based on our experiment with PECOS the used of RMA alone is enough to reason the component composition. This can be done by reviewing the mapping component behavior to tasks in PECOS and reconsidering the mapping of synchronization behavior. The process to fit execution and synchronization behaviors sequentially in each task need to be done before RMA can be performed on the tasks.

After experimenting the PECOS component model in our AMR domain and implementing it, we found that the assembling activities of component at design stage are simple but the implementation of the components is complex without the support from PECOS tools. PECOS component model is the foundation for the tools that used in PECOS CBSE approach. PECOS components require a runtime environment (RTE) that takes care of the communication between the application parts and the RTOS. PECOS generated code targets the RTE in order to achieve the platform-independent. However, this requires that the component-based development process must be supported with appropriate tools during implementation and testing. PECOS component model can be used effectively in our AMR development, provide that the support tools required in PECOS can be replaced by other existing tools such as real-time UML tools [14]. This mapping process of PECOS model and real-time UML model require further study and work.

### 4. Conclusion

This paper discusses the possibilities of using an ERT component model called PECOS in AMR software and analyzes the consequences of the adoption. Based on static structure and execution model, it can be concluded that component integration using PECOS model can explicitly describes the real-time behavior of a component using static structure and execution model. While the RMA theorems can be used to guarantee system predictability in component composition.

The major conclusion drawn from this work is that modification of PECOS is required to satisfy our AMR implementation issues. The immediate future work planned are to implement the proposed modification and conducting experiment and collecting timing measurement in real-time. These measurements will be analyzed to validate the accuracy of the prediction made using PECOS model.

### 5. Acknowledgements

The authors would like to thank Public Service Department of Malaysia for funding of this PhD work.
References


