A Genetic Programming Framework for Error Recovery in Robotic Assembly Systems

Cem M. Baydar
cbaydar@umich.edu

Kazuhiro Saitou
kazu@umich.edu

Department of Mechanical Engineering and Applied Mechanics
University of Michigan
Ann Arbor, MI 48109-2125, USA

Abstract

The advantages and performance of genetic programming in the use of error recovery planning in robotic assembly systems are presented. A framework is developed and coupled with a 3D robotic simulation software for the generation of error recovery logic in assembly systems to generate robust recovery porgrams in robot language itself. Performance of the system is evaluated with the simulations made on a three dimensionally modeled automated assembly line. The obtained results showed that the system is efficient of generating robust recovery plans for different error states.

1 DEVELOPED FRAMEWORK

Robotic assembly is a sophisticated industrial process, composed of complex tasks. Automated assembly lines are subject to unexpected failures, which can result in costly shutdowns. The recovery from these failures is done either manually by human experts or by programmable logic controller codes based on anticipated error recovery scenarios. However, generation of recovery plans for all of the error cases is a hard task to accomplish since it requires well understanding of the task domain, which is incorporated with the unexpected situations. Therefore these recovery plans cover only small portion of the possible error conditions anticipated by human programmer. In literature, several systems were developed to produce efficient recovery plans in manufacturing and assembly domains. However these systems require post-processing to compile the recovery algorithm into a code for a programmable logic controller.

In this study, a unique approach is presented for generating recovery plans in robotic assembly systems. The system uses genetic programming to generate efficient recovery plans in the robot language itself. Therefore no post-processing is required to convert the plans into controller codes. Besides, it is coupled with a robotic simulation software, which enables simulation of *unexpected* errors. Previously developed systems require an initial error state to develop a recovery plan and these initial states are anticipated by human experts. However, 3D robotic simulation provides the generation of these initial error states. The problem of generating error recovery logic for failures occurred during an assembly process is discussed in our previous work (Baydar and Saitou, 2000). With this approach, the generated programs can be directly embedded to the system by using the commercial software package's features.

In order to implement the GP to the problem, a computer program is developed and integrated with the commercial robotic simulation package called Workspace (Workspace, 1998). This program enables the user to generate programs for the problem and after that evaluate the outputs form the commercial software and proceed with the evolution process for the next generations. Two crossover operators are implemented to the system and evaluated simultaneously with the experimental studies. The results showed that the deterministic crossover operator improves the evolution of the plans.

2 CONCLUSIONS

Performance of the system is evaluated with the simulations and the results showed that the system is efficient to solve recovery problems and capable of generating robust recovery plans for multiple different error states . It is expected that this approach will require less time for the generation of error recovery logic.

References

Baydar C., Saitou K. (2000), Off-line error recovery logic synthesis in automated assembly lines by using genetic programming, *Proceedings of the 2000 Japan-USA Symposium on Flexible Automation*, Ann Arbor, MI, July 2000.

Workspace v.4 Educational User Guide Manual, (1998).