

0.1. *Пермяшкин Д.А.* Анализ методов анализа программ на конфликты за общие ресурсы для процесс-ориентированных программ

В современном мире любое производство включает в себя автоматизацию большей части производственных процессов. Связан данный факт с тем, что к данному моменту управление данными процессами и станками возможно только в автоматическом режиме. Человек же обычно является оператором, чья функция сводится к включению и выключению производства в штатном режиме, и к управлению установкой в гипервизорном режиме в случаях сбоев в производственных процессах по какой-либо причине.

В добавок к данному факту уже 10 лет существует Индустрия 4.0. Одна из её целей — исключить человека как управляющее звено. Для достижения этого требуются на этапе разработки проработать как можно больше возможных сценариев отказа. Для достижения этого требуются специализированные средства с возможностью потенциальной интеграции в интегрированные среды разработки ПО для микроконтроллеров.

Для целей поддержки задачи технологической поддержки создаются различные языки программирования. Например, процесс-ориентированная парадигма, рассматриваемая в данном докладе. Данная парадигма была разработана в ИАиЭ СО РАН для целей описания производственных киберфизических систем, и была успешно апробирована на реальных системах. Основным преимуществом данной парадигмы является описание программы как большое число параллельно исполняющихся процессов, что идеально отображает реальные производственные процессы в системе. Но это увеличивает шанс на конфликты из-за большого числа активных исполнителей, взаимодействующих с одними и теми же ресурсами реального мира.

Тема конфликтов за общие ресурсы в параллельных программах изучается с 1966 года и является хорошо изученной в общем случае. При этом некоторыми исследователями [1] отмечается недостаток работ в области промышленной автоматизации и разговоров о применимости методов. В связи с этим, данная работа попытается дать сравнительный анализ существующих методов, применимо к процесс-ориентированной парадигме. Поскольку работ, применимых конкретно к процесс-ориентированной парадигме не было обнаружено, то был выбран фокус на работах, применимых к языкам стандарта IEC-61131, поскольку существуют трансляторы для процесс-ориентированной программы в данные языки.

Были выделены следующие критерии для сравнения готовых реализаций:

- Поддерживаемые языки

- Удобство использования
- Число находимых сценариев
- Скорость работы на разных программах

Результатом анализа стал факт, что на данный момент не существует открытых готовых реализаций, подходящих для языков стандарта IEC-61131. Существуют либо научные работы [1], которые приводят результаты своих реализаций, но сам анализатор перестал быть доступным. Либо существуют программы наподобие plcVerif [2], которые требуют написания списка своих правил, которые будут привязаны к конкретной программе.

Выводом из данной работы является необходимость разработки программных комплексов для целей разработки промышленной автоматизации для проработки как можно больше возможных сценариев отказа на этапе написания управляющей программы.

Научный руководитель — д.т.н. Зюбин В. Е.

Список литературы

- [1] PRENOFER H. ET AL. Static code analysis of IEC 61131-3 programs: Comprehensive tool support and experiences from large-scale industrial application //IEEE Transactions on Industrial Informatics. 2016. Vol. 13. N. 1. P. 37–47. <https://doi.org/10.1109/tii.2016.2604760>
- [2] DARVAS D., BLANCO E., FERNANDEZ ADIEGO B. Transforming PLC programs into formal models for verification purposes. 2013. №. CERN-ACC-NOTE-2013-0040. <https://cds.cern.ch/record/1629275>