

Разработка и реализация на базе ПЛК комплексных стратегий управления

ПАРАСАР КОДАТИ, ТОМ ЭРККИНЕН, АРКАДИЙ ТУРЕВСКИЙ, Softline, Matlab@sl-matlab.ru

Что лучше: настроить контур управления на реальном объекте или имитаторе либо провести предварительное моделирование? Ответ на этот вопрос зависит от многих обстоятельств: от опыта и навыков инженера, от сложности объекта регулирования и алгоритмов управления и т.д. Истина, скорее всего, посередине: необходимо предварительное моделирование с последующей доводкой системы управления на технологическом объекте.

Статья публикуется в рамках совместного проекта медиагруппы «Электроника» и Департамента MathWorks компании Softline.

Программируемые логические контроллеры (ПЛК, PLC) используются для решения различных задач управления — от сравнительно простых систем автоматического управления с одним входом и одним выходом до систем с несколькими связанными контурами и сложными алгоритмами управления. В случае простых задач управления, например, в системах с пропорционально-интегрально-дифференциальным регулированием (ПИД) инженеры могут ввести ПИД-регулятор и настроить усиление при работающей машине. В более сложных системах управления программирование и проверка логики управления на ПЛК требуют значительно больших усилий. Конструкторы должны определить значения всех параметров схемы управления и убедиться, что все составляющие алгоритма управления будут взаимодействовать так, как запланировано. Настройка сложной схемы управления на аппаратном прототипе или на реальном технологическом процессе не только отнимает много времени, но и характеризуется высокой степенью риска повреждения оборудования.

Альтернативным решением является использование моделирования, что позволяет создавать и проверять сложные стратегии управления на моделях. Затем такую же модель можно использовать для автоматического создания структурированного текста, соответствующего международному стандарту МЭК 61131 и предназначенного для программирования ПЛК при развертывании. В этой статье такой подход показан на примере использования в системе управления сталепрокатным станом.

ЗАДАЧИ СИСТЕМЫ УПРАВЛЕНИЯ СТАЛЕПРОКАТНЫМ СТАНОМ

На сталепрокатном стане из плоской стальной заготовки изготавливают лист заданной толщины. Обычно прокатный стан состоит из нескольких каскадов, на каждом из которых валки сдавливают проходящий через них стальной лист (см. рис. 1). Между каскадами валков с помощью



Рис. 1. Схема технологического процесса прокатки стали

каскада механизма подачи поддерживается натяжение листа, предотвращается образование разрывов и провисаний. Для исполнения модели многокаскадного процесса следует сначала смоделировать один каскад прокатных валков и систему управления ими. Более общую конфигурацию можно проанализировать, соединив несколько таких более простых установок.

Система управления рассматриваемым простым прокатным станом должна удовлетворять следующим требованиям:

- поддержание толщины производимого стального листа на выходе последнего валка на уровне $8 \text{ мм} \pm 0,1 \text{ мм}$;
- поддержание требуемой производительности на выходе последнего валка на уровне $1 \text{ м/с} \pm 0,1 \text{ м/с}$;
- поддержание давления на материал на уровне $1,75 \text{ Н/м}^2 \cdot 10^5$ по прошествии 100 с для каждого валка;
- обнаружение неисправностей датчиков и приводов и последующее приведение их в норму либо безопасная остановка.

СОЗДАНИЕ МОДЕЛИ ОБЪЕКТА

Моделирование начинается с создания модели Simulink® прокатного стана, которая используется для разработки и испытания рассматриваемого контроллера. Процесс моделируется в два этапа. Сначала создаются модели отдельных каскадов валков, затем — находящийся между ними каскад подачи. В каскадах валков для создания сдавливающего усилия валков на стальную ленту применяется гидропривод. Прикладываемый к валкам крутящий момент, который создается электроприводом, позволяет контролировать скорость прокатки. Используя SimMechanics™, Simscape™ и SimHydraulics®, можно моделировать, соответственно, механические, электрические и гидравлические элементы без необходимости выведения специальных уравнений. Для моделирования каскада подачи используется библиотека SimMechanics; отображается каскад подачи и стальная лента до и после объединения трех узлов в местах стыка. Затем модели каскадов валков и подачи объединяются в одной системной модели Simulink (см. рис. 2).

КОНСТРУИРОВАНИЕ И ПРОВЕРКА РЕГУЛЯТОРОВ

На следующем этапе для конструирования регуляторов используется модель объекта. На рисунке 3 показана многоконтурная архитектура типичной системы управления технологическим процессом многокаскадного прокатного

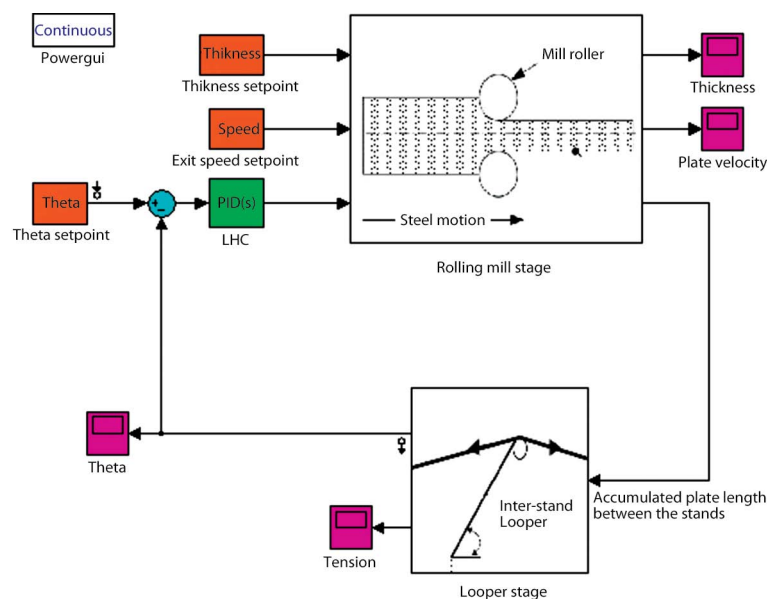


Рис. 2. Модель Simulink с каскадом валков и каскадом подачи. θ = угол подачи

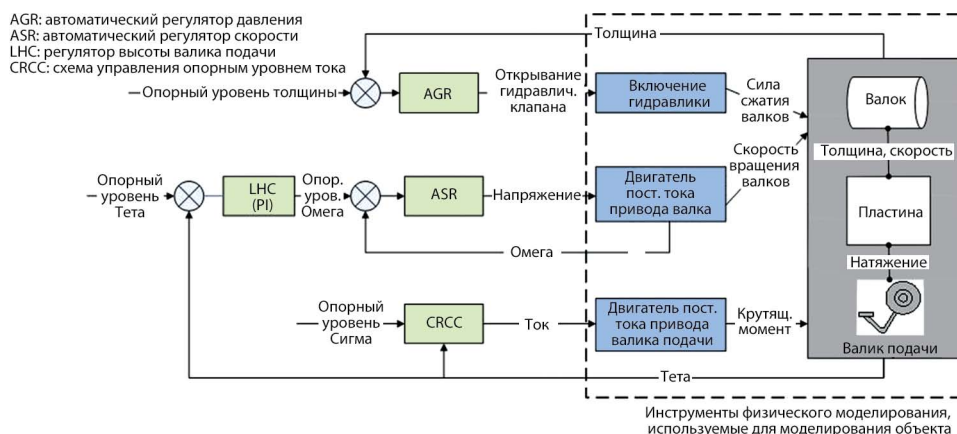


Рис. 3. Многоконтурная архитектура системы управления (тета = угол подачи; сигма = натяжение пластины; омега = скорость пластины)

стана. Система управления состоит из следующих регуляторов:

- **AGR** — управление открыванием гидравлического клапана, создающего сдвигающее усилие валков для регулировки толщины стальной ленты;
- **ASR** — управление напряжением двигателя постоянного тока, который создает требуемый крутящий момент валков и за счет этого обеспечивает контроль скорости стальной ленты;
- **LHC** — задание опорного значения частоты вращения валков для непрямого достижения требуемого натяжения материала (когда натяжение больше требуемого значения, увеличивается уставка скорости, что способствует поступлению дополнительного материала и снижению натяжения. Если натяжение ниже требуемого значения, провисания избегают снижением скорости прохождения стального листа);
- **CRCC** — регулировка тока двигателя валика подачи для позиционирования валика, поддерживающего натяжение материала.

Все контуры обратной связи связаны между собой. Например, гидропривод, управляемый регулятором AGR, влияет не только на толщину стального листа, но и на его

скорость. Оба регулятора LHC и ASR предназначены для поддержания требуемого натяжения и нужной скорости листа.

Сначала проектируются регуляторы, управляющие работой одного валка. С помощью Simulink Control Design™ линеаризуется нелинейная модель. Затем выполняется настройка устройства управления с помощью инструментов конструирования ПИД в Simulink Control Design, заключающаяся в подсчете коэффициента усиления данного устройства управления. Устройство настройки (см. рис. 4) автоматически подсчитывает коэффициенты усиления ПИД по требуемому времени реакции. С помощью Simulink Design Optimization™ выполняется точная настройка коэффициентов усиления устройства управления, чтобы система устойчиво работала при наличии нелинейностей. Работа конструкции в целом проверяется путем запуска нелинейного моделирования. Модель объекта используется для решения двух задач: во-первых, линеаризованная модель объекта, созданная с помощью Simulink Control Design, используется для настройки регуляторов, и, во-вторых, полная нелинейная модель применяется для проверки работы системы управления путем исполнения модели с замкнутыми контурами обратной связи.

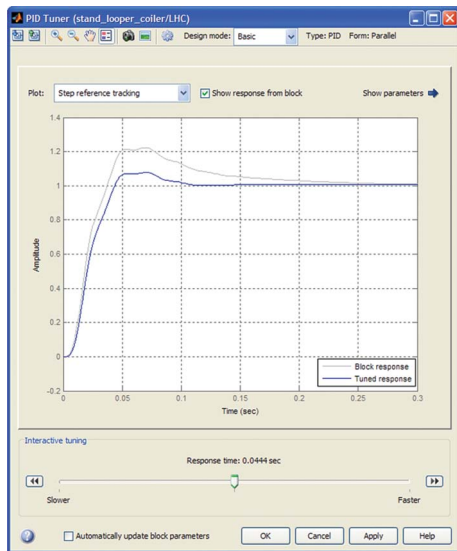


Рис. 4. Инструмент настройки ПИД. Точную настройку конструкции можно выполнить, перемещая ползунок

МОДЕЛИРОВАНИЕ И ИСПОЛНЕНИЕ МОДЕЛИ МНОГОКАСКАДНОГО ПРОЦЕССА

Используя блоки пользовательской библиотеки, повторно применяем и подсистемы каскада валков и каскада подачи в качестве компонентов многокаскадного технологического процесса (см. рис. 5). Для моделирования других аспектов процесса используются дополнительные подсистемы, такие как сохранение массы и передаточное запаздывание на разных каскадах валков. На рисунке 6 показаны переменные технологического процесса на каждом из трех его каскадов. Получены уставки толщины для

каждого каскада, чтобы получился стальной лист требуемой толщины. Достигнуто также эффективное подавление возмущений натяжения, вызываемых воздействием разных каскадов валков.

КОНСТРУИРОВАНИЕ И ПРОВЕРКА ЛОГИКИ ОБНАРУЖЕНИЯ ОТКАЗОВ

Помимо регуляторов обратной связи в состав устройств управления технологическими процессами должна входить и логика наблюдения, обнаружения отказов и восстановления, задача которой — следить в том числе и за состоянием датчиков и исполнительных механизмов системы. Сконцентрируем внимание на логике восстановления, которая обнаруживает отказы гидравлических клапанов и принимает меры по коррекции их последствий. Эта логика перераспределяет целевую уставку уменьшения толщины по индивидуальным уставкам для каждого каскада многокаскадного технологического процесса. Если происходит отказ системы гидравлического сжатия одного каскада, логика проверяет возможность компенсации функции отказавшего каскада за счет других каскадов. Если такая компенсация возможна, производится пересчет уставок работоспособных каскадов таким образом, чтобы достигалась целевая уставка снижения толщины листа для системы в целом.

Для разработки такой логики используется Stateflow® (см. рис. 7). Рассматриваемый пример логики представляет собой упрощенный вариант реальной системы управления технологическими процессами, т.к. при этом предполагается, что неисправный каскад не создает сжатия и в то же время пропускает обрабатываемый материал. В реальной системе управления необходимо использовать значительно более сложную логику, учитывающую большее количество вариантов и возможностей.

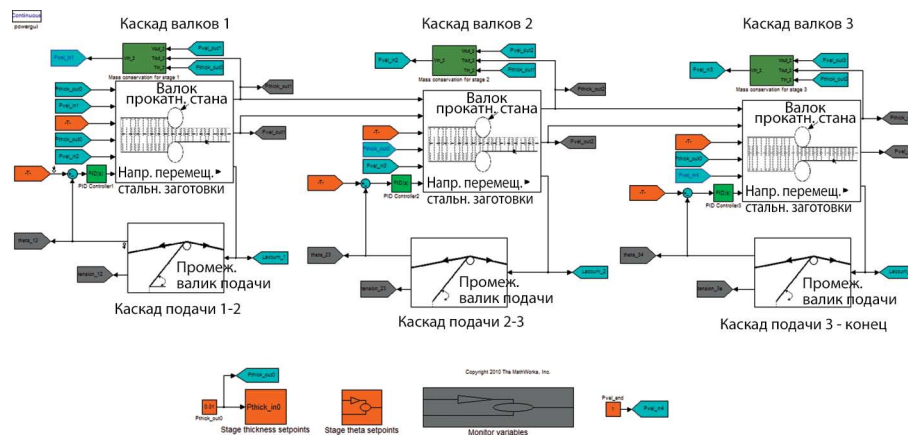


Рис. 5. Модель Simulink с несколькими каскадами прокатных валков и каскадами подачи в промежутке между ними

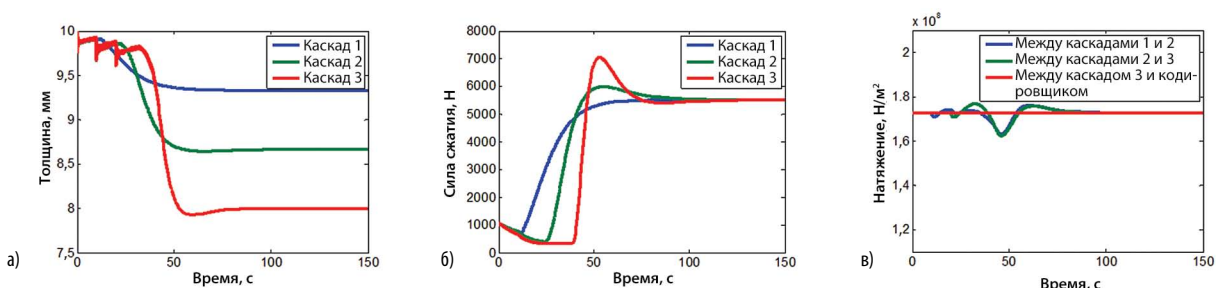


Рис. 6. Результаты моделирования для переменных технологического процесса: а) общее целевое уменьшение толщины, получаемое на выходе каскада 3; б) общее целевое уменьшение толщины, равномерно распределенное по трем каскадам; в) сокращение возмущений натяжения стального листа

Рассмотренную выше логику можно проверить в действии, искусственно введя отказы в модель Simulink. На рисунке 8 показаны результаты исполнения модели логики, повышающей отказоустойчивость. При отказе одного каскада логика наблюдения проверяет возможность распределения нагрузки на остальные работоспособные каскады. Если такая возможность имеется, на AGR отдельных каскадов поступают команды с новыми уставками уменьшения толщины. Если такой возможности нет, технологический процесс останавливается путем прекращения подачи стального листа.

ПРОГРАММИРОВАНИЕ ФУНКЦИИ УПРАВЛЕНИЯ В ПЛК

Для автоматической генерации структурированного текста, соответствующего международному стандарту МЭК 61131, использовался Simulink PLC Coder™. Затем такой транспортируемый структурированный текст можно импортировать в среду моделирования, используемую для работы с целевым ПЛК. На рисунке 9 показан структурированный текст, соответствующий международному стандарту МЭК 61131, сгенерированный с логики обнаружения отказов и коррекции. Сгенерированный структурированный текст в достаточной степени снабжен комментариями, поэтому его происхождение можно без труда отследить по модели.

Автоматическая генерация команд позволяет избежать ошибок, которые могут возникнуть при программировании вручную, и гарантирует, что окончательный структурированный текст обеспечит численные результаты в ПЛК, очень близкие к результатам, которые наблюдались при исполнении модели. Simulink PLC Coder выполняет проверку, которая позволяет сравнить результаты проверок с IDE и результаты исполнения оригинальной модели.

Если бы потребовалось провести тестирование полностью внедренного оборудования и программного обеспечения ПЛК с помощью аппаратно-программного моделирования, пришлось бы с помощью Simulink Coder™ сгенерировать код «С» из модели установки и запустить его в моделирующем устройстве, подключенном к системе управления на базе ПЛК, например, с операционной системой xPC Target™, в реальном масштабе времени.

Использовавшиеся программные продукты: Simulink®, SimHydraulics®, SimMechanics™, Simscape™, Simulink Control Design™, Simulink Design Optimization™, Simulink PLC Coder™, Stateflow®.

```

END_IF;
Setpoint_IN_STAGE_1_FAILED:
(* During 'STAGE_1_FAILED': '<S1>:119' *)
IF (stage3_sensor <= 0) OR (stage2_sensor <= 0) THEN
(* Transition: '<S1>:150' *)
(* Transition: '<S1>:152' *)
IF stage2_sensor > 0 THEN
(* Transition: '<S1>:155' *)
is_c2_Setpoint := Setpoint_IN_STAGES_1_3_FAILED;
(* Entry 'STAGES_1_3_FAILED': '<S1>:120' *)
rtb_stage1_setpoint := L0;
rtb_stage2_setpoint := L0 - overall_target;
distributed_target := rtb_stage2_setpoint;
ELSE
(* Transition: '<S1>:154' *)
IF stage3_sensor > 0 THEN
(* Transition: '<S1>:159' *)
is_c2_Setpoint := Setpoint_IN_STAGES_1_2_FAILED;
(* Entry 'STAGES_1_2_FAILED': '<S1>:121' *)
rtb_stage1_setpoint := L0;
rtb_stage2_setpoint := L0;
distributed_target := L0 - overall_target;
ELSE
guard_0 := TRUE;
END_IF;
END_IF;
ELSE
guard_0 := TRUE;
END_IF;

```

Рис. 9. Структурированный текст EC 61131, сгенерированный с использованием Simulink PLC Coder™ (комментарии со ссылкой на схему Stateflow выделены синим цветом)

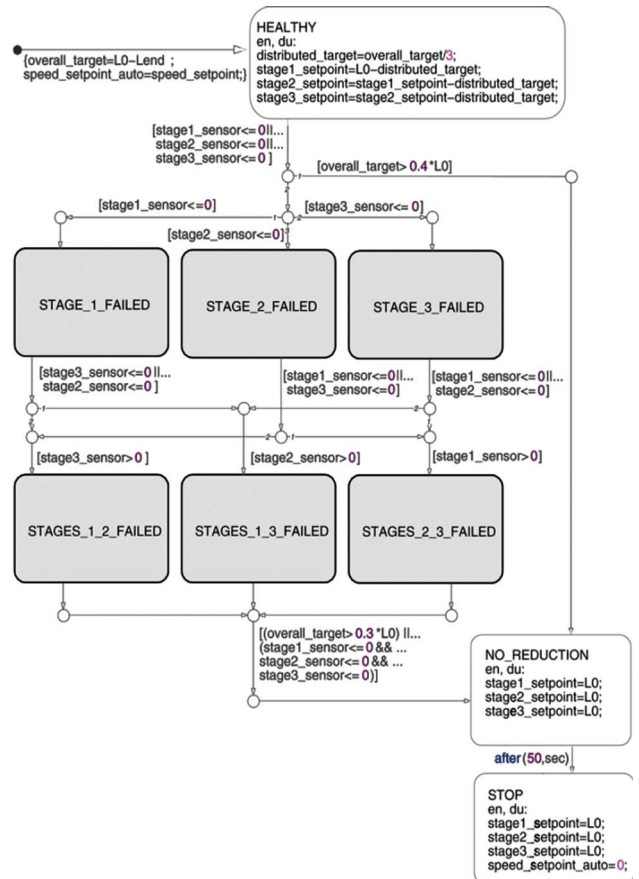


Рис. 7. Внедрение в Stateflow распределения уставок, что обеспечивает отказоустойчивую работу

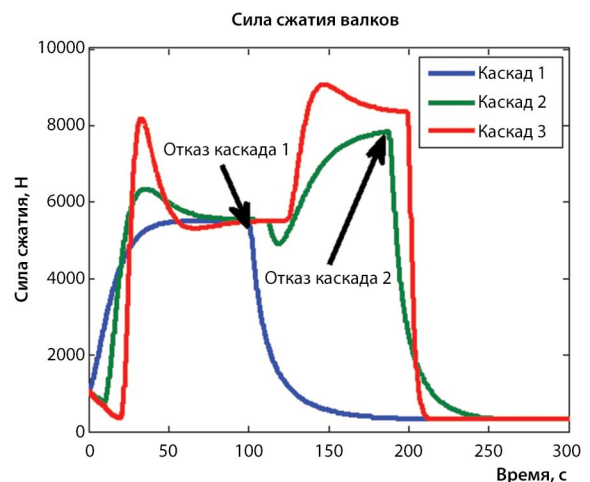


Рис. 8. Результаты исполнения модели восстановления системы при отказах каскадов. Компенсация каскадами 2 и 3 отказа каскада 1. При отказе наряду с каскадом 1 еще и каскада 2 была произведена остановка системы, поскольку оставшийся работоспособным каскад 3 в одиночку не мог обеспечить требуемого общего уменьшения толщины стального листа

Дополнительная информация: видеобзор Simulink PLC Coder™; разработка компанией AVL устройства динамического управления для системы контроля состояния двигателя, в которой используется генерация встроенного кода для ПЛК.

Ознакомиться с другими статьями и оформить подписку можно по ссылке: mathworks.com/newsletters.