

как регистры общего назначения и память); хранение программных данных реализовано таким образом что искажение любого бита или байта при адресации не приводит к наложению одних данных на другие.

Неодновременное проявление отказа позволяет обнаружить одиночный отказ до появления второго, это достигается за счёт диверсификации ПО ядер двух параллельных каналов. Программы ядра одного и второго каналов скомпилированы разными компиляторами. Внутренние циклы выполнения всех алгоритмов различны, что создаёт диверситет при выполнении одинаковых по внешним проявлениям алгоритмов СЦБ.

Для минимизации влияния структуры микропроцессора основные алгоритмические места программы ядра МПЦ написаны в ассемблерном коде, что транслируются в тривиальный байт-код, позволяющий свести выполнение алгоритма МПЦ к тривиальным логическим операциям AND, OR, EQU (и, или, равенство) и практически исключить операции условного перехода, в которых отказ одного бита приводит к необнаружимому неправильному функционированию.

Целостность и корректность передачи проверяется контрольной суммой и маркером владельца.

Защита информации внутри ядра МПЦ осуществляется следующим образом: все бинарные данные закодированы 32-битным словом с кодовым расстоянием не менее 16. При выполнении действий над ними, если результат не является верным, происходит выставление ошибки. Все адреса данных выровнены таким образом, что имеют кодовое расстояние не менее 4 и не имеют в себе более 8 подряд стоящих 0 или 1, в младшей, значащей части адреса, данные адресные пространства формируются различным образом для всех четырёх ядер каналов МПЦ.

Таким образом, в разработанной МПЦ «Ипуть» реализованы все современные методы как в аппаратной реализации, так и в ПО, позволяющие обеспечивать высший уровень SIL4 по требованиям международного стандарта МЭК 61508.

Опытная эксплуатация МПЦ на станции «Ипуть» и приёмочные испытания свидетельствуют о том, что данная концепция и заключённые в ней принципы обеспечения безопасности соответствуют лучшим западным аналогам. Разработанная МПЦ при её освоении на производстве и широком её внедрении позволит Белорусской железной дороге сэкономить десятки миллионов долларов по программе импортозамещения.

УДК 656.222.3: 656.21

СИНТЕЗ АВТОРЕГУЛЯТОРА СКОРОСТИ НА ОСНОВЕ НЕЧЕТКОЙ ЛОГИКИ В СИСТЕМЕ АВТОМАТИЧЕСКОГО РЕГУЛИРОВАНИЯ СКОРОСТИ СКАТЫВАЮЩИХСЯ С ГОРКИ ОТЦЕПОВ

Н. К. МОДИН, В. В. ВОЛЫНЕЦ

Белорусский государственный университет транспорта

Комплекс автоматического регулирования скорости (АРС) осуществляет интервальное и прицельное регулирование скорости скатывающихся вагонов на сортировочной горке, используя в качестве исполнительных органов вагонные замедлители. Недостатки алгоритмов функционирования существующих систем АРС, а также невозможность получения достоверной информации об объекте управления являются основными причинами опасных ситуаций, приводящих к нарушению безопасности функционирования комплекса АРС.

При необходимости реализовать заданную скорость выхода v_3 оттормаживание замедлителей должно начинаться в момент, когда отцеп имеет скорость $v_{от} = v_3 + \Delta v$, где Δv – опережение скорости оттормаживания. Упреждение по скорости при оттормаживании определяют с учетом инерционности Δt горочных замедлителей

$$\Delta v = a_{zi} \Delta t, \quad (1)$$

где a_{zi} – интенсивность торможения.

Для расчета фактической величины a_{ϕ} системы управления требуется определить функцию изменения интенсивности торможения $a_3(t)$ на временном интервале $[t_1; t_1 + \Delta t]$, причем значение a_{ϕ}

должно быть получено уже в момент времени t_1 . В реальных условиях составить такой прогноз аналитическими методами не представляется возможным.

В данной работе для решения задачи нахождения значения Δv по известному значению a_{zi} предлагается использовать алгоритмы на основе теории нечетких множеств. Последовательность операций при разработке алгоритма нечеткой модели управления можно представить следующим образом:

1 Определяем элементы нечеткого множества (термы) для лингвистической переменной «интенсивность торможения». Пусть множество X содержит следующие элементы:

$$X = \{\text{«очень небольшая»}, \text{«небольшая»}, \text{«средняя»}, \text{«большая»}, \text{«очень большая»}\}.$$

2 Задаем элементы терм-множества Y лингвистической переменной «упреждение по скорости»:

$$Y = \{\text{«очень небольшое»}, \text{«небольшое»}, \text{«среднее»}, \text{«большое»}, \text{«очень большое»}\}.$$

3 Для каждого термина нечетких множеств X и Y на основе экспертных оценок определяем функции принадлежности (субъективные оценки принадлежности отдельных значений a_{zi} и Δv к элементам множеств X и Y) $m_1(a_{zi}), m_i(a_{zi}), \dots, m_n(a_{zi}); y_1(\Delta v), y_2(\Delta v), \dots, y_i(\Delta v)$.

4 Формируем базу знаний нечеткой модели управления в виде совокупности n нечетких правил следующего вида:

$$\text{если } a_{zi} \text{ есть } X_i, \text{ то } \Delta v \text{ есть } Y_i,$$

где X_i, Y_i – элементы множеств X и Y для соответствующего правила базы знаний.

5 Производим процедуру нечеткого логического вывода в следующем порядке:

– *введение нечеткости (фаззификация)*. Для четко заданного входного параметра a_{zi} рассчитываются принадлежности к отдельным термам множества X ;

– *нечеткая импликация*. Находим функции принадлежности заключения каждого правила для конкретного значения a_{zi} :

$$\mu_i(\Delta v) = \min(m_i(a_{zi}), y_i(\Delta v));$$

– *нечеткая композиция*. Находим результирующую функцию принадлежности всего набора правил для заданного значения входного параметра;

$$\mu_{\Sigma}(\Delta v) = \max(\mu_1(\Delta v), \mu_2(\Delta v), \dots, \mu_n(\Delta v));$$

– *приведение к четкости (дефаззификация)*. Получение конкретных значений выходного параметра можно осуществить, используя центроидный метод (метод «центра тяжести»):

$$\Delta v = \frac{\int \Delta v \cdot \mu_{\Sigma}(\Delta v) d(\Delta v)}{\int \mu_{\Sigma}(\Delta v) d(\Delta v)}.$$

5 Получаем зависимость $\Delta v = f(a_{zi})$ выходного параметра односвязной нечеткой модели управления от входного параметра a_{zi} путем многократного выполнения п. 4.

Сравнить модель управления замедлителями, основанную на применении формулы (1), и нечеткую модель, использующую функцию $\Delta v = f(a_{zi})$, можно, прибегнув к имитационному моделированию процесса торможения отцепов на ТП.

Результаты моделирования показывают, что введение нечеткости (вычисление упреждения по скорости по формуле $\Delta v = f(a_{zi})$) приводит к снижению математического ожидания абсолютной погрешности реализации скорости для трех весовых категорий отцепов.

Таким образом, применение авторегулятора скорости на основе нечеткой логики в системе АРС позволит уменьшить среднюю величину отклонения фактической скорости выхода отцепов от требуемой, что в свою очередь позволит снизить такие показатели сортировочной работы на горке, как число повреждений вагонов на сортировочных путях и протяженность окон.