

МЕТОД УЧЕТА ОГРАНИЧЕНИЙ ПРИ ОПТИМИЗАЦИИ АРХИТЕКТУРЫ АППАРАТНО-ПРОГРАММНОГО КОМПЛЕКСА СИСТЕМЫ УПРАВЛЕНИЯ РЕАЛЬНОГО ВРЕМЕНИ

В статье предложен метод учета ограничений при оптимизации архитектуры аппаратно-программного комплекса системы управления реальным временем с помощью генетических алгоритмов. Оптимизация архитектуры аппаратно-программного комплекса производится с учетом критериев надежности и стоимости.

Ключевые слова: автоматизированная система оперативного управления перевозками, аппаратно-программный комплекс



Е.М. Лыткина



С.Н. Ефимов

С целью повышения качества оперативно-диспетчерского контроля перевозочным процессом на сети железных дорог России предусмотрена автоматизированная система оперативного управления перевозками (АСОУП). Система позволяет поддерживать в реальном времени информационную модель перевозочного процесса, прогнозировать и планировать эксплуатационную работу линейных подразделений, выполняющих те или иные операции перевозочного процесса. Таким образом, АСОУП обеспечивает единство управления всех уровней, начиная с систем на станциях и других линейных предприятий и заканчивая системами верхнего уровня. В основе системы лежит актуальная информация о поездах, тяговом и нетяговом подвижном составе, их местона-

хождении и состоянии, что делает систему динамичной. Динамическая модель перевозочного процесса позволяет осуществлять полный надзор за выполнением технических норм комплектования составов, решать возникающие задачи, такие как автоматизация сбора информации, отслеживание и отображения поездной ситуации, автоматизация задания маршрутов следования поездов и т.д. [1].

Постановка задачи

Таким образом, АСОУП является, по сути, системой управления реальным временем, то есть системой управления, способной реагировать на изменения в объекте управления и внешней среде в рамках требуемых временных ограничений.

Лыткина Екатерина Михайловна, кандидат технических наук, доцент кафедры «Эксплуатация железных дорог» Красноярского института железнодорожного транспорта – филиала Иркутского государственного университета путей сообщения. Область научных интересов: зональная система повышения надежности тягового подвижного состава. Автор 67 научных работ, в том числе одной монографии. Имеет шесть патентов РФ.

Ефимов Сергей Николаевич, кандидат технических наук, доцент кафедры информационно-управляющих систем Сибирского государственного университета науки и технологий имени академика М.Ф. Решетнева. Область научных интересов: реконфигурируемые вычислительные системы обработки информации и управления. Автор 89 научных работ, в том числе четырех монографий. Имеет 13 патентов РФ.

Терсков Виталий Анатольевич, доктор технических наук, профессор кафедры информационно-управляющих систем Сибирского государственного университета науки и технологий имени академика М.Ф. Решетнева. Область научных интересов: моделирование и оптимизация архитектуры аппаратно-программных комплексов систем управления, работающих в реальном времени. Автор 127 научных работ, в том числе семи монографий. Имеет 16 патентов РФ.

Ярков Константин Владимирович, аспирант кафедры «Управление персоналом» Красноярского института железнодорожного транспорта – филиала Иркутского государственного университета путей сообщения. Область научных интересов: информатика и вычислительная техника. Автор 11 научных работ. Имеет два патента РФ.

Подобные системы управления представляют собой аппаратно-программные комплексы (АПК), то есть совокупность аппаратных средств и программного обеспечения (ПО), совместно функционирующих для выполнения поставленной задачи.

Одним из основных параметров, определяющих функционирование подобных систем, является надежность [2].

Надежность АПК системы управления реального времени, то есть вероятность успешно выработать корректное управляющее воздействие за определенное время, является критически важным параметром. Невозможность выработки корректного управляющего воздействия может быть вызвана проявлением ошибок ПО или выходом из строя компонентов аппаратного обеспечения [3].

Оценка надежности вычислительных систем путем опытной эксплуатации изготовленных образцов, как правило, затруднительна вследствие достаточно большого времени наработки на отказ. Поэтому представляется целесообразным создание математических моделей, которые позволят выбирать оптимальные структуры АПК.

Для исследования влияния отказов элементов программного обеспечения АПК применяется подход, связанный с дублированием версий ПО, которые разрабатываются независимо друг от друга [4], за счет этого можно увеличивать надежность программного компонента АПК до необходимого уровня.

Модели надежности аппаратного обеспечения многопроцессорных вычислительных систем могут быть построены с применением математического аппарата и методологии теории массового обслуживания [5;6].

Целесообразным является рассмотрение АПК как единого целого с точки зрения моделирования параметров надежности. Подобная модель может быть построена также с использованием математического аппарата теории массового обслуживания.

Процесс функционирования вычислительной системы представляется замкнутой системой массового обслуживания (СМО) с ограниченной очередью, которая состоит из любого количества разнородных процессоров, соединенных с оперативной памятью посредством шин. При этом потоки отказов процессоров и шин интерфейса независимы, а интегральный поток отказов в системе можно рассматривать как простейший пуассоновский.

Состояния, в которых может находиться рассматриваемая система, обозначим как $a_{j_1, j_2, \dots, j_N, j_{N+1}}^{r_1, r_2, \dots, r_K}$ — это состояние, в котором $(m_1 - j_1)$ процессоров первого типа исправны и участвуют в вычислительном про-

цессе, а j_1 неисправны и восстанавливаются; $(m_2 - j_2)$ процессоров второго типа исправны и участвуют в вычислительном процессе, а j_2 неисправны и восстанавливаются; ...; $(m_N - j_N)$ процессоров N -го типа исправны и участвуют в вычислительном процессе, а j_N неисправны и восстанавливаются; $(n - j_{N+1})$ шин интерфейса исправны и участвуют в вычислительном процессе, а j_{N+1} неисправны и восстанавливаются. Индекс безошибочности (величина, равная единице, если соответствующая версия программного обеспечения работает без ошибок, и нулю — с ошибками) первой версии равен r_1 , второй версии равен r_2 , ..., K -й версии равен r_K . Индексы j_i изменяются от 0 до m_i , а индексы безошибочности могут принимать значения 0 или 1.

Переход системы в состояние, когда при определенной архитектуре аппаратной части АПК одна из версий критичного по надежности модуля ПО выдает неверный результат, определяется вероятностью безотказной работы данной версии p_k .

Составляя систему дифференциальных уравнений Чепмена-Колмогорова [7], и приравняв к нулю производные для стационарного режима, получим систему уравнений, которая однозначно разрешается с учетом условия нормировки.

В результате получим выражение для предельных вероятностей, в которых может находиться система:

$$P_{j_1, j_2, \dots, j_{N+1}}^{r_1, r_2, \dots, r_K} = \frac{\prod_{i=1}^{N+1} p_i^{j_i}}{\sum_{\substack{j_1=0, m_1 \\ j_2=0, m_2 \\ \dots \\ j_{N+1}=0, n}} \prod_{i=1}^{N+1} p_i^{j_i} \left[1 + \sum_{k=1}^K (1 - p_k) \right]}, \quad (1)$$

где N — количество типов процессоров;

m_i — количество процессоров i -го типа;

n — количество шин интерфейса;

K — количество критичных по надежности версий ПО;

p_i — интенсивность загрузки канала обслуживания.

Имея аналитическую модель для вычисления производительности АПК с определенной архитектурой, например [8], а также используя выражение (1) для определения предельных вероятностей состояний вычислительной системы, можно оценивать надежность АПК, то есть находить вероятность работы системы с заданной производительностью.

При решении задачи оптимизации архитектуры АПК системы управления реального времени имеем две группы критериев:

- критерий надежности, который должен быть максимизирован (вероятность нахождения в состоянии,

в котором производительность достаточна для выработки управляющего воздействия);

- критерии стоимости, которые должны быть минимизированы (стоимость системы, стоимость разработки системы, стоимость эксплуатации, стоимость ремонта и т.д.).

При этом на переменные задачи будут наложены ограничения, например, по энергопотреблению, скорости и т.д. Для упрощения задачи критерии стоимости можно перевести в ограничения, так как для всех стоимостных характеристик системы, как правило, имеются верхние границы, заданные заказчиком системы управления. Тогда получаем задачу однокритериальной условной оптимизации с набором существенных ограничений.

Формальная запись задачи оптимизации структуры АПК может выглядеть следующим образом:

$$S_0(m_1, \dots, m_N, n, k) \rightarrow \max,$$

при условиях

$$C_l(m_1, \dots, m_N, n, k) \leq C_l^0, l=1, \dots, L_C,$$

$$m_i^- \leq m_i \leq m_i^+, i=1, \dots, N,$$

$$n^- \leq n \leq n^+,$$

$$1 \leq k \leq K.$$

В данной задаче приняты следующие обозначения:

S_0 — критерий оценки надежности;

C_l — критерии оценки стоимости;

C_l^0 — предельные допустимые уровни критериев, переведенных в ограничения.

Если приблизительно оценить мощность пространства оптимизации, то получим общее число возможных конфигураций более $1,6 \cdot 10^{20}$. Кроме того, задача характеризуется большой размерностью, разнотипностью переменных, многоэкстремальностью целевой функции (1).

Это означает, что для решения поставленной задачи могут быть использованы эвристические алгоритмы усеченного перебора.

При решении задач оптимизации положительно зарекомендовали себя алгоритмы эволюционной оптимизации, например, алгоритм генетического программирования [9], вероятностный генетический алгоритм [10] или асимптотический вероятностный генетический алгоритм [11].

Метод учета ограничений

Генетический алгоритм (ГА) это стохастический метод псевдобулевой оптимизации (то есть оптими-

зации функций с булевыми аргументами и вещественными значениями), имитирующий эволюционные процессы, происходящие в популяциях живых организмов [10]. Среди рассматриваемых решений производится отбор (селекция) перспективных особей в промежуточную популяцию, которая затем используется для порождения особей новой популяции.

Укрупненно генетический алгоритм можно описать следующим образом:

1. Создать и оценить начальную популяцию с равномерным распределением генов.
2. Если выполнен критерий останова, прекращаем работу.
3. Сформировать промежуточную популяцию с помощью выбранного метода селекции.
4. Создать новую популяцию на основе промежуточной с использованием выбранного оператора скрещивания и мутации.
5. Перейти к шагу 2.

Генетические алгоритмы оптимизации имеют большое количество настраиваемых параметров, в частности, необходимо выбирать конкретный вид генетических операторов в зависимости от конкретной задачи оптимизации. Чтобы исключить этап настройки параметров, который требует, как правило, привлечение специалиста в области эволюционных алгоритмов оптимизации, был предложен метод самоконфигурации [12]. Основная идея этого метода заключается в том, что одновременно используются различные генетические операторы. Частота использования каждого оператора меняется в ходе работы алгоритма в зависимости от качества получаемых с помощью этого оператора решений.

Поставленная задача оптимизации имеет существенные ограничения, то есть является задачей условной оптимизации. Одной из основных проблем задач условной оптимизации является вопрос, что делать с недопустимыми (не удовлетворяющими хотя бы одному ограничению) решениями в процессе поиска, то есть какой метод учета ограничений использовать.

Один из методов учета ограничений — полностью игнорировать недопустимые решения и продолжать процесс поиска только с допустимыми решениями. Тем не менее, недопустимые решения могут содержать полезную информацию. Если полностью игнорировать недопустимые решения, разнообразие может быть потеряно на ранних стадиях процесса поиска, что может завести поиск в локальный оптимум.

В работе [13] предлагается следующая классификация методов учета ограничений:

- сохранение допустимости решений;
- штрафные функции;

- разделение допустимых и недопустимых решений;
- гибридные методы.

Разнообразие методов учета ограничений приводит к необходимости выбора одного из них, что, как правило, не может быть сделано без экспериментирования и привлечения специалиста в области эволюционных методов оптимизации.

Для того, чтобы упростить работу ГА при проектировании многопроцессорных АПК с мультиверсионным программным обеспечением, целесообразно реализовать самонастройку выбора метода учета ограничений. При этом в ходе оптимизации используется несколько методов учета ограничений. В начале оптимизации каждый из них работает с популяцией, размер которой составляет $1/z$ общего размера популяции, где z число используемых методов учета ограничений. На каждом поколении вычисляется доля допустимых решений, порождаемых с помощью каждого метода, а также определяется метод учета ограничений, у которого эта доля является наибольшей. Доля в популяции лучшего метода учета ограничений на следующем поколении увеличивается на

$$Cn \frac{z-1}{zN},$$

где Cn — константа, определяющая скорость адаптации;

N — номер поколения.

Доли в общей популяции остальных методов учета ограничений уменьшаются на величину

$$Cn \frac{1}{zN}.$$

При этом существует минимальный порог доли, если какой-либо метод достигает этого порога, то он перестает «отдавать» долю, а лучший на данном

поколении метод учета ограничений перестает ее от него получать. Если на каком-то этапе оптимизации все порождаемые решения становятся допустимыми, то доли не изменяются вообще.

Самонастройка метода учета ограничений будет характеризоваться большим «давлением» на методы учета ограничений, порождающие плохие решения, чем при использовании нескольких методов учета ограничений, когда вычислительные ресурсы для всех конкурирующих методов учета ограничений одинаковы и не перераспределяются, так как выделяемые им ресурсы будут уменьшаться.

Описанный подход самонастройки метода учета ограничений не вводит дополнительных настраиваемых параметров, если уже используется самонастройка генетических операторов, то есть сложность ее использования не изменяется.

Заключение

Задача проектирования многопроцессорного АПК системы управления реального времени с мультиверсионным ПО может быть формализована как задача условной оптимизации на дискретной решетке.

Подобные задачи в силу свойств целевой функции и ограничений, а также объема пространства поиска, могут быть эффективно решены только с использованием эвристических алгоритмов оптимизации, например, эволюционных алгоритмов.

Применение эволюционных алгоритмов для решения задач условной оптимизации требует выбора метода учета ограничений, причем эффективность методов учета ограничений зависит от конкретной решаемой задачи.

Для автоматизации выбора метода учета ограничений в эволюционных алгоритмах был предложен алгоритм самонастройки учета ограничений, который основан на перераспределении долей в популяции в пользу лучших методов учета ограничений. 

Литература

1. Васильев, В.А. Системы реального времени и области их применения / В.А. Васильев, К.Е. Легков, И.В. Левко // Информатика и космос. — 2016. — №3. — С. 68—70.
2. Павский, В.А. Вычисление показателей надежности вычислительных систем [Текст] / В.А. Павский, В.Г. Хорошевский // 7-я Международная научно-техническая конференция «Искусственный интеллект. Интеллектуальные и многопроцессорные системы». — Таганрог: ТРТУ, 2006. — Т2. — С. 17—21.
3. Buttazzo, G. Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications. — New York, NY, Springer. — 2011.

4. Ковалёв, И.В. К вопросу реализации мультиверсионной среды исполнения бортового программного обеспечения автономных беспилотных объектов средствами операционной системы реального времени / И.В. Ковалёв, В.В. Лосев, М.В. Сарамуд [и др.] // Вестник СибГАУ. — 2017. — №1(18). — С. 58–61.
5. Efimov, S.N. Methods of Assessing the Characteristics of the Multiprocessor Computer System Adaptation Unit / Sergey.N. Efimov, Valeriy N. Tyapkin, Dmitry D. Dmitriev, Vitaly A. Terskov // Journal of Siberian Federal University. Mathematics & Physics 2016, 9(3). — P. 288–295.
6. Ефимов, С.Н. Реконфигурируемые вычислительные системы обработки информации и управления / С.Н. Ефимов, В.А. Терсков. — Красноярск: КРИЖТ ИрГУПС, 2013. — 249 с.
7. Ross, S. M. Introduction to Probability Models (11th ed.). 2014. P 187. — ISBN: 978-0-12-375686-2.
8. Galushin, P.V. A performance model of a multiprocessor computer appliance of a real-time control system / P.V. Galushin, O.Yu. Serikova, V.A. Terskov, K.V. Yarkov // IOP Conference Series: Materials Science and Engineering. Krasnoyarsk Science and Technology City Hall of the Russian Union of Scientific and Engineering Associations. Krasnoyarsk, Russia, 2020. — С. 12103.
9. Ефимов, С.Н. Алгоритм генетического программирования с автоматически определяемыми функциями для выбора спецпроцессоров МВС интеллектуального анализа данных в режиме реального времени / С.Н. Ефимов, А.С. Егоров, Е.С. Семенкин // Вестник Томского государственного университета. Серия «Математика. Кибернетика. Информатика». — 2006. — № 19. — С. 223–225.
10. Goldberg, D 1989 Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison-Wesley Professional.
11. Sopov, E The convergence prediction method for genetic and PBIL-like algorithms with binary representation / E. Sopov, S. Sopov. — // IEEE International Siberian Conference on Control and Communications (SIBCON 2011). — p. 2036.
12. Семёнкин, Е.С. Самоконфигурируемые эволюционные алгоритмы моделирования и оптимизации: монография / Е.С. Семёнкин, М.Е. Семёнкина. МДП. — Магнитогорск, 2014. — 312 с.
13. Michalewicz, Z. Evolutionary algorithms for constrained parameter optimization problems / Z. Michalewicz, M. Schoenauer // Evolutionary Computation, vol. 4, 1996. — P. 1–32.