
ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ

УДК 004.021, 004.89

ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА В ЗАДАЧЕ ОПТИМИЗАЦИИ ЗАМЕНЫ ЭЛЕМЕНТОВ СИСТЕМЫ

© 2022 г. О. И. Кос^{а,*}, В. Ю. Смирнов^а

^а МАИ (национальный исследовательский ун-т), Москва, Россия

*e-mail: kosoksana90@gmail.com

Поступила в редакцию 03.05.2022 г.

После доработки 21.05.2022 г.

Принята к публикации 30.05.2022 г.

Задача оптимизации замен элементов сложной технической системы решается с помощью генетического алгоритма. Проанализированы его различные операторы и произведена их модификация в соответствии с поставленной задачей. Создан алгоритм решения задачи оптимизации, на основе которого разработано программное обеспечение. Оно было применено для решения практической задачи оптимизации замен или ремонтов элементов. В результате решения задачи достигается максимальный экономический эффект при условии обеспечения заданного уровня надежности.

DOI: 10.31857/S0002338822050109

Введение. В эпоху всеобщей цифровизации применение алгоритмов искусственного интеллекта становится все более популярным. В настоящее время для решения многих практических задач используются методы искусственного интеллекта на базе эволюционных вычислений, которые являются стохастическими методами оптимизации, основанными на принципах моделирования биологической эволюции.

Разработка и применение алгоритмов искусственного интеллекта — интенсивно развивающееся направление. Благодаря универсальности вычислительной схемы, возможностям параллельной реализации, устойчивости к шуму алгоритмы искусственного интеллекта находят успешное практическое применение при решении многих сложных задач. В их число входят такие, как оптимизация функций; оптимизация запросов в базах данных; разнообразные задачи на графах (задача коммивояжера, раскраска, нахождение пар сочетаний); настройка и обучение искусственной нейронной сети; задачи компоновки; составление расписаний; игровые стратегии; теория приближений; искусственная жизнь; биоинформатика; синтез конечных автоматов. Наиболее успешно генетические алгоритмы искусственного интеллекта используются для решения сложных нелинейных многомерных задач оптимизации, в том числе задач управления эксплуатацией сложных технических систем, таких, как искусственные сооружения, например мостовые конструкции.

В работе генетические алгоритмы искусственного интеллекта применяются для решения задачи оптимального распределения времен замен или ремонтов элементов сложной технической системы за все время эксплуатации. Эта задача обладает рядом специфических особенностей, таких, как стохастический характер переменных величин, большое их количество, а также интервальные оценки их размещения на временной шкале. Для учета этих особенностей хорошо подходят методы группового учета аргументов, в том числе алгоритмы массовой селекции, которые являются одной из разновидностей алгоритмов искусственного интеллекта.

Под генетическим алгоритмом искусственного интеллекта будем понимать алгоритм поиска решения, используемый для решения задач оптимизации с помощью механизмов, аналогичных естественному отбору в природе. Данный генетический алгоритм искусственного интеллекта выступает разновидностью эволюционных вычислений, с помощью которых решаются оптимизационные задачи с применением методов естественной эволюции, таких, как селекция, кроссинговер и мутация, роль которых в алгоритмах искусственного интеллекта аналогична их роли в живой природе. Генетический алгоритм искусственного интеллекта хорошо масштабируется под сложность решаемой задачи.

1. Постановка задачи. Рассмотрим сложную техническую систему, состоящую из большого числа разных элементов. В ходе ее эксплуатации требуется эффективно ремонтировать и заменять исчерпавшие ресурс элементы для обеспечения безотказности работы этой сложной технической системы.

Первая часть задачи состоит в нахождении совокупности оптимальных времен замен (ремонтов) каждого из элементов сложной технической системы [1], т.е. поэлементной оптимизации. На основании математической модели расчета оптимального интервала предупредительных замен [2, 3] и рекуррентного метода расчета надежности сложной технической системы, такой, как искусственное сооружение [4–6], а также алгоритмов, построенных на их основе [7–10], находится оптимальное время замены (ремонта) для каждого отдельного элемента сложной технической системы. Будем считать, что эти оптимальные времена замен (ремонтов) элементов уже найдены, т.е. первая часть задачи решена. Рассматриваем решение второй части задачи эффективной замены или ремонта элементов сложной технической системы.

Вторая часть задачи состоит в оптимизации имеющейся совокупности оптимальных интервалов времен замен (ремонтов) элементов, полученных при решении первой части задачи, т.е. в результате поэлементной оптимизации. Во второй части задачи требуется оптимизировать имеющуюся совокупность интервалов и тем самым построить оптимальную схему эксплуатации сложной технической системы в целом. Задача оптимизации эксплуатации сложной технической системы за счет эффективной замены ее элементов решается с помощью генетического алгоритма искусственного интеллекта.

Множеством решений данной задачи является оптимальная, т.е. эффективная с точки зрения эксплуатации совокупность оптимальных времен замен (ремонтов) элементов сложной технической системы. Ограничением при решении задачи оптимизации будет требование на совокупность оптимальных интервалов, заключающееся в том, что эти интервалы могут быть уменьшены до ближайшего интервала. Но они не могут быть увеличены, так как в этом случае не будет обеспечена надежность сложной технической системы.

Критерий оптимальности решения задачи выражен целевой функцией, значения которой используют для сравнения решений. В нашем случае целевой функцией выступает сумма следующих затрат в процессе эксплуатации сложной технической системы:

стоимость замен (ремонтов) элементов сложной технической системы;

затраты на выезд специалистов для ремонта (замены) элементов;

стоимость времени работы элементов, которые были заменены раньше времени своей оптимальной замены.

В рассматриваемой задаче под оптимальным решением будем понимать оптимальный набор времен замен (ремонтов) элементов для всей сложной технической системы в целом, который доставляет минимум целевой функции и удовлетворяет заданному ограничению на совокупность оптимальных интервалов.

Для построения оптимального набора времен замен (ремонтов) элементов для всей сложной технической системы, т.е. оптимальной схемы ее эксплуатации, предлагаются следующие методы.

1. Для достижения экономического эффекта можно использовать синхронизацию. Данный метод применяется, когда оптимизация, с точки зрения затрат на эксплуатацию всей сложной технической системы в целом, может быть произведена за счет синхронизации замен (ремонтов) его элементов без их сдвигов. При этом интервалы, кратные друг другу, синхронизируются. В результате ремонт каждого элемента, исчерпавшего свой ресурс, будет происходить одновременно с ремонтом элемента, имеющего самый маленький ресурс. Таким образом, уменьшается количество замен (ремонтов) элементов для всей сложной технической системы в целом и, следовательно, уменьшаются экономические затраты на ее эксплуатацию. Это происходит в результате того, что выезд специалистов на объект, транспортные затраты, применение специальной техники, приостановка использования сложной технической системы по назначению и другие факторы, связанные с ремонтом или заменой, влекут за собой большие материальные издержки. При этом обеспечивается заданный уровень надежности, так как оптимальные интервалы замены (ремонта) для каждого элемента рассчитаны с учетом заданного уровня надежности.

2. С целью достижения более высокого уровня экономической эффективности по сравнению с синхронизацией можно использовать генетический алгоритм искусственного интеллекта, в котором сочетаются преимущества детерминированного и стохастического подходов к оптимизации. Сложную техническую систему необходимо эксплуатировать по принципу: “максимум

надежности при минимуме экономических затрат” [7], который означает минимизацию экономических затрат при обеспечении заданного уровня надежности.

В силу своей специфики генетический алгоритм искусственного интеллекта хорошо подходит для решения нашей задачи, т.е. достижения оптимального уровня экономических затрат на эксплуатацию сложной технической системы при условии обеспечения заданного уровня надежности. Основным способом достижения экономического эффекта, так же, как и при применении метода синхронизации, будет уменьшение количества выездов специалистов на объект. Времена замен (ремонтов) элементов могут быть в небольших пределах изменены в сторону уменьшения в соответствии с ограничением данной задачи на совокупность оптимальных интервалов, и тем самым обеспечивается безопасность эксплуатации сложной технической системы. Эти сдвиги должны быть достаточно малыми, чтобы экономические издержки на эксплуатацию не были слишком большими. Данные сдвиги времен замен (ремонтов) элементов позволяют уменьшить экономические затраты на эксплуатацию всей сложной технической системы, т.е. оптимизация каждого отдельного элемента частично приносится в жертву ради оптимизации всей системы в целом.

3. Достигается наиболее значительный экономический эффект при последовательном применении генетического алгоритма искусственного интеллекта и алгоритма синхронизации. Согласно третьему методу, сначала проводится оптимизация за счет сдвигов времен замен (ремонтов) путем применения генетического алгоритма искусственного интеллекта, что приводит к уменьшению количества выходов на объект и, следовательно, к сокращению издержек. Затем выполняется синхронизация времен замен (ремонтов) элементов без сдвигов по времени с целью дальнейшего сокращения количества выходов на объект.

2. Краткий обзор и анализ генетических алгоритмов. Генетический алгоритм искусственного интеллекта работает следующим образом.

1. Генерируется начальная популяция решений, например, следующими способами:

1.1. производится инициализация, т.е. формирование начальной популяции, которое заключается в случайном выборе заданного количества членов популяции, представляемых последовательностями фиксированной длины [11];

1.2. начальная популяция генерируется случайным образом с использованием полного дерева с ограниченной максимальной глубиной [12];

1.3. начальная популяция формируется на основании детерминированного алгоритма.

В генетическом алгоритме искусственного интеллекта для решения задачи достижения оптимального уровня экономических эксплуатационных затрат на сложную техническую систему для формирования начальной популяции применяется детерминированный алгоритм.

2. Популяция оценивается с помощью специальной фитнес-функции (функции приспособленности), которая является целевой функцией оптимизации.

Фитнес-функция — специальная функция, служащая для оценки популяции по заданному критерию. Оценивание приспособленности всей популяции состоит в расчете функции приспособленности для каждого члена этой популяции. Чем больше значение этой функции, тем выше “качество” члена популяции.

Вид функции приспособленности зависит от характера решаемой задачи. Предполагается, что функция приспособленности всегда принимает неотрицательные значения, возможно, с заданной точностью и, кроме того, что для решения оптимизационной задачи требуется максимизировать функцию. Если исходная функция приспособленности не удовлетворяет этим условиям, то выполняется соответствующее преобразование (например, задачу минимизации функции можно легко свести к задаче максимизации) [11].

В генетическом алгоритме искусственного интеллекта для решения задачи достижения оптимального уровня экономических эксплуатационных затрат на сложную техническую систему фитнес-функция используется при оценивании начальной популяции.

3. Выполняется селекция (отбор) с помощью генетических операторов, в качестве которых используются, например, следующие.

3.1. Метод “рулетки” или отбор пропорциональной пригодности, описанный в [11–13], получил название по аналогии с азартной игрой. Каждому члену популяции может быть сопоставлен сектор колеса рулетки, величина которого устанавливается пропорционально значению функции приспособленности данного члена популяции, поэтому чем больше значение функции приспособленности, тем больше сектор на колесе рулетки. Все колесо рулетки соответствует сумме

значений функции приспособленности всех членов рассматриваемой популяции. Вероятность селекции каждого члена популяции или коэффициент участия данного члена популяции при выборе родителей вычисляется как отношение фитнес-функции этого члена популяции к сумме значений фитнес-функции всех членов рассматриваемой популяции, выраженное в процентах.

Селекция членов популяции может быть представлена как результат поворота колеса рулетки, поскольку “выигравший” (т.е. выбранный) член популяции относится к выпавшему сектору колеса. Очевидно, что чем больше сектор, тем больше вероятность “победы” соответствующего члена популяции. Поэтому вероятность выбора данного члена популяции оказывается пропорциональной значению его функции приспособленности. Если всю окружность колеса рулетки представить в виде цифрового промежутка $[0, 100]$, то выбор члена популяции можно отождествить с выбором числа из интервала $[a, b]$, где a и b обозначают соответственно начало и окончание фрагмента окружности, соответствующего этому сектору колеса; очевидно, что $0 \leq a < b \leq 100$. В этом случае выбор с помощью колеса рулетки сводится к выбору числа из интервала $[0, 100]$, которое соответствует конкретной точке на окружности колеса. В результате процесса селекции создается родительская популяция, также называемая родительским пулом (mating pool) с численностью N , равной численности текущей популяции.

3.2. “Турнирный отбор” (tournament selection), описанный в [12, 13], при котором в популяции, содержащей N членов, выбирается случайным образом t членов. Лучший из них, т.е. имеющий большее значение функции приспособленности, записывается в промежуточный массив. Эта операция повторяется N раз. Члены популяции из промежуточного массива затем используются для скрещивания (также случайным образом). Количество случайно выбираемых членов популяции t часто равно двум. В этом случае говорят о двоичном (парном) турнире, а t называют численностью турнира.

3.3. Панмиксия представляет собой свободное скрещивание членов популяции в пределах популяции или какой-либо группы внутри популяции [13]. О панмиксии можно говорить лишь тогда, когда каждый член популяции имеет равновероятностные возможности скрещивания с любым другим членом популяции.

3.4. Инбридингом называется метод, когда первый родитель выбирается случайным образом, а второй родитель — член популяции, ближайший к первому [13]. Под ближайшим членом популяции понимается наиболее близкий член в смысле минимального расстояния Хэмминга (в случае, если члены популяции представлены бинарными последовательностями фиксированной длины) или наиболее близкий член в смысле евклидова расстояния между двумя вещественными векторами (в случае, если члены популяции представлены вещественными числами). Так инбридинг можно охарактеризовать свойством концентрации поиска в локальных узлах, что фактически приводит к разбиению популяции на отдельные локальные группы вокруг подозрительных на экстремум мест.

3.5. Аутбридинг использует понятие схожести членов популяции [13]. В этом методе родители, выбираемые для образования потомков, формируются из максимально далеких членов популяции. Расстояния вычисляются теми же способами, что и в инбридинге. Но, в отличие от инбридинга, аутбридинг по-другому влияет на поведение генетического алгоритма искусственного интеллекта, т.е. направлен на противодействие сходимости алгоритма к уже найденным решениям, заставляя алгоритм просматривать новые неисследованные области.

3.6. Отбор усечением не имеет аналогов в естественной эволюции и обычно применяется для больших популяций, где число членов популяции $N > 100$ [13]. В этом методе используется параметр — порог отсечения T , также определяемый как интенсивность отбора, показывающий долю, т.е. часть популяции, которая отбирается в качестве родителей. Обычно эта доля составляет от 10 до 50%. При помощи этого метода сначала отбираемые члены популяции упорядочиваются, согласно значениям их целевой функции. Затем в качестве родителей выбираются только лучшие члены популяции с учетом порога отсечения. Далее, с равной вероятностью, среди этих отобранных членов популяции случайным образом выбирают пары, которые производят потомков.

3.7. Элитарный отбор представляет собой метод, при котором в новое поколение включается заданное количество лучших членов популяции предыдущего поколения [13]. Все члены популяции оцениваются, и в новую популяцию отбирается заданное количество (например, 10%) самых лучших членов популяции, которые называются “элитными”. Остальные члены популяции (например, 90%), не вошедшие в элиту, становятся родителями и порождают потомков. Таким образом создается комбинированная популяция, которая включает в себя как “элитных” членов популяции, так и потомков других членов популяции. Применение стратегии элитарности ока-

зывается весьма полезным для эффективности генетического алгоритма искусственного интеллекта, так как не допускает потерю лучших решений. Например, если не использовать элитарный отбор, то возможно возникновение следующей ситуации. Популяция находится в области локального минимума, а мутация вывела один из членов популяции в область глобального минимума, но этот член популяции, находящийся в области глобального минимума, с большей вероятностью будет потерян, так как будет скрещиваться с членами популяции, находящимися в области локального минимума. Если же используется стратегия элитарности, то полученный самый лучший член популяции останется в популяции до тех пор, пока он не будет вытеснен еще более лучшими членами популяции из элитарного набора.

3.8. Отбор вытеснением представляет собой метод, когда в данном отборе выбор членов популяции в новую популяцию зависит как от величины ее фитнес-функции, так и от того, есть ли уже в сформированной популяции член популяции с аналогичным набором участков [13]. При проведении отбора из всех членов популяции родителей с одинаковой приспособленностью предпочтение сначала отдается членам популяции с разными наборами участков. Таким образом в популяции постоянно поддерживается разнообразие. Отбор вытеснением формирует новую популяцию скорее из удаленных членов популяции, вместо членов популяции, группирующихся около текущего найденного решения. Данный метод наиболее пригоден для многоэкстремальных задач, при этом помимо определения глобальных экстремумов появляется возможность выделить и те локальные максимумы, значения которых близки к глобальным.

Проанализируем рассмотренные методы селекции с точки зрения того, насколько они подходят к решаемой задаче достижения оптимального уровня экономических эксплуатационных затрат на сложные технические системы.

При панмиксии каждый член популяции имеет равные возможности стать родителем. Данный метод не подходит, так как он не позволяет сделать так, что родителями должны стать лучшие члены популяции. При инбридинге первый родитель выбирается случайным образом, а вторым родителем является член популяции, ближайший к первому. Таким образом количественная оценка членов популяции с помощью фитнес-функции не производится. Данный метод не подходит, так как один из родителей выбирается случайным образом. При аутбридинге используется понятие схожести членов популяции. В этом методе родители, порождающие потомков, выбираются из максимально далеких членов популяции. Следовательно, формируются максимально отличающиеся новые члены популяции. При этом с точки зрения фитнес-функции эти члены не будут являться лучшими в качестве родителей. При турнирном отборе анализируются не все члены популяции, а случайно выбранные группы. В этом случае не каждый член популяции оценивается вероятностью селекции члена популяции.

Таким образом предпочтение отдается методу “рулетки”, в котором каждый член популяции оценивается вероятностью селекции члена популяции. Для выполнения операции селекции в рассматриваемой задаче достижения оптимального уровня экономических эксплуатационных затрат на сложные технические системы разработан собственный гибридный метод селекции или метод “выборной рулетки”, включающий помимо метода “рулетки” элитарный отбор и отбор усечением, которые его дополняют.

Сначала все члены популяции оцениваются с помощью фитнес-функции. Далее применяется метод элитарного отбора и в новую популяцию отбирается заданное количество (например, 10%) самых лучших членов популяции, которые называются “элитными”. Затем используется метод отбора усечением. Из данного метода используется параметр — порог отсечения T , также определяемый как интенсивность отбора, показывающий долю, т.е. часть популяции, которая отбирается в качестве родителей. В данном случае порог отсечения T принимается равным 90%, следовательно, 10% членов популяции с наихудшими значениями фитнес-функции отбрасываются. После этого к оставшимся после элитарного отбора и отбора усечением 80% членов популяции применяется метод “рулетки”.

Для каждого члена популяции рассчитывается вероятность селекции члена популяции, обозначаемого $p(ch_i)$, где ch_i — i -й член популяции, для $i = \overline{1, N}$ (где N — численность популяции), которой соответствует параметр $\vartheta(ch_i)$, выраженный в процентах, согласно формуле:

$$\vartheta(ch_i) = p(ch_i)100\%, \quad (2.1)$$

где

$$p(ch_i) = \frac{F(ch_i)}{\sum_{i=1}^N F(ch_i)}, \quad (2.2)$$

причем $p(ch_i)$ – вероятность селекции члена популяции ch_i , $F(ch_i)$ – значение функции приспособленности члена популяции ch_i .

В итоге 10% членов популяции отобрано с помощью элитарного отбора, еще 10% убраны из родителей с помощью порога отсеечения. Таким образом родителями становятся 80% членов популяции, каждый из которых будет родителем с определенной вероятностью селекции члена популяции $p(ch_i)$, а 10% элитарных членов просто переносятся в следующую популяцию. Из тех членов популяции, которые стали родителями случайным образом, выбираются пары для создания потомков.

4. Выполняется оператор кроссинговера (скрещивания), который из каждой пары родителей, выбранных в результате селекции, создает потомка, например, следующими способами.

4.1. Классический кроссинговер (classic crossover) или односточный кроссинговер (single point crossover), описанный в [11, 13], действует следующим образом. На первом этапе для каждой пары родителей определяется точка скрещивания, которая представляет собой случайное число в диапазоне от 2 до $L-1$, где L – это количество участков, из которых состоит каждый из родителей. На втором этапе формируется потомок, у которого участки от 1 до l_k берутся от первого родителя, а участки от $l_k + 1$ до L – от второго родителя:

$$\begin{array}{cc} \text{Родители} & \text{Потомки} \\ x_1x_2x_3 \dots x_{n-1}x_n \dots x_m & \rightarrow x_1x_2x_3 \dots x_{n-1}y_n \dots y_m, \end{array} \quad (2.3)$$

$$y_1y_2y_3 \dots y_{n-1}y_n \dots y_m \rightarrow y_1y_2y_3 \dots y_{n-1}x_n \dots x_m. \quad (2.4)$$

4.2. Двухточечный кроссинговер, описанный в [13], действует следующим образом. На первом этапе для каждой пары родителей определяется две точки скрещивания, которые представляют собой случайные числа k_1, k_2 в диапазоне от 1 до $L-1$, где L – количество участков, из которых состоит каждый из родителей. На втором этапе возможны два варианта. Если вторая точка скрещивания больше первой точки, то формируется потомок, у которого участки с номерами от 1 до k_1 берутся от первого родителя, далее участки от $k_1 + 1$ до k_2 – от второго родителя, затем участки от $k_2 + 1$ до L – опять от первого родителя. Если вторая точка скрещивания меньше первой точки, то формируется потомок, у которого участки от 1 до k_1 берутся от второго родителя, далее участки от $k_1 + 1$ до k_2 – от первого родителя, затем участки от $k_2 + 1$ до L – от второго родителя.

4.3. Многоточечный кроссинговер (multi-point crossover), описанный в [13], представляет собой метод, при котором выбирается m точек скрещивания $k_i \in \overline{1, L-1}$, $i \in \overline{1, m}$, где L – количество участков, из которых состоит каждый из родителей. Точки скрещивания выбираются случайно без повторений и сортируются в порядке возрастания. Эти m точек разбивают каждого из родителей на $m + 1$ групп участков. При многоточечном кроссинговере происходит последовательный выбор групп участков родителей с чередованием, например, группы участков с нечетными номерами групп берутся от одного родителя, а группы участков с четными номерами групп – от другого родителя. Из этих групп участков, выбранных таким образом, формируется первый потомок. В случае, если чередование групп участков берется в противоположном порядке, то формируется второй потомок.

Продemonстрируем применение многоточечного кроссинговера на примере двух родителей, состоящих из 11 участков ($L = 11$):

$$\text{родитель 1: } 01110011010, \quad (2.5)$$

$$\text{родитель 2: } 10101100101. \quad (2.6)$$

Выберем три точки скрещивания ($m = 3$):

$$2, 6, 10. \quad (2.7)$$

Создадим потомков:

$$\text{потомок 1: } 01101111011, \quad (2.8)$$

$$\text{потомок 2: } 10110000101. \quad (2.9)$$

Метод многоточечного кроссинговера применяется для скрещивания родителей с наименьшими значениями фитнес-функции.

4.4. Триадный кроссинговер (triadic crossover), описанный в [13], представляет собой метод, в котором после выбора пары родителей, из остальных членов популяции случайным образом выбирается такой, который в дальнейшем используется в качестве маски. Далее определенная часть участков маски, например 10%, меняется. Затем участки первого родителя сравниваются с участками маски: если участки одинаковы, то они передаются первому потомку, в противном случае на соответствующие позиции участков потомка переходят участки второго родителя. Вторым потомком отличается от первого тем, что на тех позициях, где у первого потомка стоят участки первого родителя, у второго потомка стоят участки второго родителя и наоборот.

4.5. Перетасовочный кроссинговер представляет собой метод, в котором члены популяции, отобранные в качестве родителей, случайным образом обмениваются участками [13]. Затем выбирают точку для скрещивания и формируют потомков с помощью одноточечного кроссинговера. Таким образом, на каждой итерации перетасовочного кроссинговера создаются не только новые потомки, но и модифицируются родители, так как старые родители удаляются. Это позволяет сократить число операций по сравнению с одноточечным кроссинговером [13].

4.6. Кроссинговер с уменьшением замены действует таким образом, что для создания потомков выбираются только отличающиеся участки родителей [13]. Это осуществляется за счет ограничения на выбор точек скрещивания, которые должны появляться только там, где участки родителей различаются.

4.7. Кроссинговер на древовидных структурах действует следующим образом [12]. Члены популяции представляются в древообразной форме. Применяется три разновидности оператора кроссинговер на древовидных структурах:

- узловой оператор кроссинговера;
- кроссинговер поддеревьев;
- смешанный кроссинговер.

4.7.1. В узловом операторе кроссинговера выбираются два родителя и по одному узлу в каждом из них. Узлы в деревьях могут быть разного типа. Поэтому сначала проверяется, что выбранные узлы у родителей — взаимозаменяемы. Если узлы не являются взаимозаменяемыми, то у одного из родителей выбирается другой узел, так чтобы он был взаимозаменяем. Далее производится обмен узлов. Поддеревья узлов остаются на месте.

4.7.2. В кроссинговере поддеревьев родители обмениваются не узлами, а определяемыми ими поддеревьями. После выбора родителей и узлов в них проверяется, что выбранные узлы взаимозаменяемы, т.е. принадлежат одному типу. Иначе, как и в предыдущем случае, в одном из деревьев выбирается другой узел с последующей проверкой взаимозаменяемости. Затем производится обмен поддеревьев, которые определены этими узлами.

Далее вычисляется размер (сложность потомка) для ожидаемых потомков. При этом под размером поддерева понимается либо его высота, либо число его вершин. Если ожидаемый размер не соответствует заданному порогу, то этот обмен не происходит и выбираются другие поддеревья для обмена.

Кроссинговер поддеревьев является основным оператором кроссинговера на древовидных структурах.

4.7.3. При смешанном операторе кроссинговера для одних узлов выполняется узловой оператор кроссинговера, а для других — кроссинговер поддеревьев.

Для выполнения операции кроссинговера в рассматриваемой задаче достижения оптимального уровня экономических эксплуатационных затрат на сложные технические системы разработан собственный гибридный метод, включающий помимо метода одноточечного кроссинговера и метода кроссинговера с уменьшением замены. Сначала применяется метод кроссинговера с уменьшением замены, т.е. для создания потомка выбираются только отличающиеся участки родителей. Это осуществляется за счет ограничения на выбор точек скрещивания, которые должны появляться только там, где участки родителей различаются. Таким образом, для каждой пары родителей определяется точка скрещивания, которая представляет собой случайное число

из выбранных предполагаемых точек скрещивания в диапазоне от 2 до $L-1$, где L — количество предполагаемых точек скрещивания, из которых состоит каждый из родителей.

Кроссинговер формирует новых членов популяции на основе двух имеющихся родителей, комбинируя их части. Поэтому число различных новых членов популяции, которые могут быть получены кроссинговером при использовании одной и той же пары родителей, ограничено. Соответственно пространство, которое генетический алгоритм искусственного интеллекта может покрыть, используя лишь кроссинговер, жестко зависит от разнообразия участков популяции. Чем разнообразнее участки родителей, тем больше пространство покрытия. При обнаружении локального экстремума фитнес-функции соответствующий ему член популяции будет стремиться занять лидирующую позицию в популяции при выборе родителей и алгоритм может сойтись к локальному экстремуму. Поэтому в генетическом алгоритме искусственного интеллекта важную роль играют мутации. Существуют несколько способов мутации членов популяции.

5. Выполняется оператор мутации (mutation), который изменяет участки членов популяции. Это делается следующими способами.

5.1. Оператор мутации с вероятностью, напоминающий рассмотренный в [11, 13, 14], можно более полно описать так. Для каждого члена популяции который будет подвержен мутации, выполняются следующие действия. Для каждого i -го участка члена популяции вычисляется случайное число p_{mi} , расположенное на отрезке $[0; 1]$, которое интерпретируется как вероятность мутации этого участка p_i . Далее каждое такое число сравнивается с заранее заданным пороговым значением p_0 , близким к единице, т.е. проверяется условие:

$$p_{mi} \geq p_0, \quad i = \overline{1, N}, \quad (2.10)$$

где N — количество участков члена популяции.

Если это условие выполнено, то значение i -го участка члена популяции меняется с 0 на 1 или с 1 на 0 соответственно. Например, если в члене популяции $[100110101010]$ мутации подвергается участок на позиции семь, то его значение, равное 1, изменится на 0, что приводит к образованию члена популяции $[100110001010]$.

За счет выбора порогового значения p_0 можно варьировать степень мутации члена популяции. Чем ближе пороговое значение к 1, тем вероятность того, что произойдет мутация хотя бы в одном участке этого члена популяции, будет меньше. В качестве порогового значения можно выбрать

$$p_0 = 1 - \frac{1}{N}. \quad (2.11)$$

Вместо выбора порогового значения p_0 можно задавать детерминированно или случайно количество членов популяции, которые будут подвергнуты мутации. Например, можно задать, что мутации будет подвержен только один участок члена популяции с наибольшей вероятностью p_{mi} или заданное количество участков члена популяции с наибольшими вероятностями p_{mi} . Количество участков члена популяции, которые будут подвергнуты мутации, также можно выбирать случайно. Так же, как и кроссинговер, мутации могут проводиться не только по одной случайной точке. Можно выбирать для изменения несколько точек в члене популяции, причем их число также может быть случайным. Используют и мутации с изменением сразу некоторой группы подряд идущих точек.

Вероятность мутации p_m , как правило $p_m \ll 1$, может являться или фиксированным случайным числом на отрезке $[0, 1]$, или функцией от какой-либо характеристики решаемой задачи. Например, можно положить вероятность мутирования участков, обратно пропорциональную числу всех участков в члене популяции. Оператор мутации с вероятностью может также применяться для членов популяции, участки которых заданы вещественными числами. При этом мутирующий участок заменяется на случайное вещественное число в заранее заданном диапазоне, который является одинаковым для всей популяции и соответствует рассматриваемой задаче.

5.2. Мутация с изменением количества участков члена популяции, описана в [13, 14]. Пусть член популяции t представлен следующей последовательностью участков: $t = t_1, \dots, t_N$, где N — количество участков члена популяции. Мутация происходит одним из следующих способов.

5.2.1. В процессе мутации случайный участок или набор случайных участков s длиной k из совокупности всевозможных значений участков присоединяется к концу последовательности, т.е. член популяции превращается в последовательность $t \rightarrow t_1, \dots, t_N, s$.

5.2.2. В процессе мутации производится вставка случайного участка или набора случайных участков s длиной k из совокупности всевозможных значений участков в случайно выбранную позицию в последовательности, т.е. член популяции превращается в последовательность $t \rightarrow t_1, \dots, t_{i-1}, s, t_i, \dots, t_N$.

5.2.3. В процессе мутации удаляется случайно выбранный участок или набор случайных участков s длиной k из последовательности, т.е. член популяции превращается в последовательность $t \rightarrow t_1, \dots, t_i, t_{i+k+1}, \dots, t_N$.

5.2.4. В процессе мутации происходит обмен случайно выбранными участками или случайно выбранными наборами участков p и q длиной k и m соответственно в последовательности, т.е. член популяции превращается в последовательность $t_1, \dots, t_i, p, t_{i+k+1}, \dots, t_j, q, t_{j+m+1}, \dots, t_N \rightarrow t_1, \dots, t_i, q, t_{i+k+1}, \dots, t_j, p, t_{j+m+1}, \dots, t_N$.

5.3. Оператор двоичной мутации (binary mutation), описанный в [13, 14], действует следующим образом. Для членов популяции, представленных в виде двоичного кода, мутация заключается в случайной инверсии участка (0 заменяется 1 или наоборот). Эффект мутации зависит от того, в каком из этих представлений записан член популяции. Так в одних задачах при мутации наилучший эффект достигается в случае, когда члены популяции представлены в виде кода Грея, а в других — с помощью двоичного кода.

5.4. Оператор мутации, где главную роль играет процесс дубликации, т.е. удвоения, выполняется следующим образом [13, 14]. В члене популяции случайным образом выбирается участок, который дублируется в этом члене популяции. Пусть член популяции t представлен последовательностью участков $t = t_1, \dots, t_k, \dots, t_N$, где N — количество участков члена популяции. Тогда при применении оператора дубликации к участку t_k мы получим следующий член популяции: $t = t_1, \dots, t_k, t_{k+1}, \dots, t_N$.

В случае многократного дублирования определенного участка в члене популяции данный оператор называется амплификацией.

5.5. Мутация поддерева (subtree mutation) является методом генетического алгоритма искусственного интеллекта, в котором члены популяции представлены в древовидной форме [12, 14]. В таком случае мутация поддерева происходит следующим образом. В дереве случайным образом выбирается точка мутации и корневое поддерево заменяется на случайно сгенерированное поддерево.

5.6. Точечная мутация, рассмотренная в [12], выполняется таким образом, что для каждого члена популяции определяется точка мутации, которая представляет собой случайное число в диапазоне от 1 до L . Здесь L — количество участков, из которых состоит каждый из членов популяции. В данной точке происходит случайная мутация участка члена популяции. Пусть член популяции t представлен следующей последовательностью участков $t = t_1, \dots, t_k, \dots, t_N$, где N — количество участков члена популяции. Тогда при применении оператора точечной мутации к участку t_k , который заменяется на участок t_m , получается член популяции $t = t_1, \dots, t_m, \dots, t_N$.

Для выполнения операции мутации в рассматриваемой задаче достижения оптимального уровня экономических эксплуатационных затрат на сложные технические системы разработан собственный гибридный метод, включающий помимо метода точечной мутации метод мутации с вероятностью.

Оператор мутации с вероятностью, напоминающий упомянутый в [11, 13], можно более полно описать так. Для каждого члена популяции, который будет подвержен мутации, выполняются следующие действия. Для каждого i -го участка члена популяции вычисляется случайное число, расположенное на отрезке $[0; 1]$, которое интерпретируется как вероятность мутации этого участка p_i . Далее каждое p_{mi} сравнивается с заранее заданным пороговым значением p_0 , близким к единице, т.е. проверяется условие

$$p_{mi} \geq p_0, \quad i = \overline{1, N}, \quad (2.12)$$

где N — количество участков члена популяции.

Затем из выбранных точек определяется точка мутации, которая представляет собой случайное число в диапазоне от 1 до L , где L — количество предполагаемых точек мутации. В данной точке происходит случайная мутация участка члена популяции.

В процессе поиска решения необходимо соблюдать баланс между “эксплуатацией” полученных на текущий момент лучших решений и расширением пространства поиска, чтобы избежать

Таблица. Результат процесса слияния для создания начальной популяции решений

Оптимальные интервалы	Оптимальные интервалы, слитые по два	Оптимальные интервалы, слитые по три	Оптимальные интервалы, слитые через один	Оптимальные интервалы, слитые через два
1	2	3	4	5
τ_1	τ_1, τ_2	τ_1, τ_2, τ_3	τ_1, τ_3	τ_1, τ_4
τ_2			τ_2, τ_4	τ_2, τ_5
τ_3	τ_3, τ_4			τ_3, τ_6
τ_4		τ_4, τ_5, τ_6		
τ_5	τ_5, τ_6		τ_5, τ_7	
τ_6			τ_6, τ_8	
τ_7	τ_7, τ_8	τ_7, τ_8, τ_9		τ_7, τ_{10}
τ_8				τ_8, τ_9
1	2	3	4	5
τ_9	τ_9, τ_{10}	τ_{10}	τ_9, τ_{10}	
τ_{10}				

генетического вырождения. При этом операторы селекции и кроссинговера делают поиск направленным, а оператор мутации обеспечивает широту поиска, т.е. расширение пространства поиска на всю популяцию (множество решений).

В отличие от других методов оптимизации генетический алгоритм искусственного интеллекта оптимизирует различные области пространства решений одновременно и более приспособлен к нахождению новых областей с лучшими значениями целевой функции за счет объединения квазиоптимальных решений из разных популяций. Поэтому генетический алгоритм искусственного интеллекта хорошо подходит для сокращения количества замен или ремонтов элементов сложной технической системы путем объединения оптимальных интервалов замен или ремонтов. Тем самым уменьшится количество выходов на объект эксплуатации и, следовательно, сократятся экономические затраты на осуществление замен или ремонтов, а значит, повысится эффективность эксплуатации всей сложной технической системы в целом.

3. Решение задачи оптимизации замен (ремонтов). Покажем, как генетический алгоритм искусственного интеллекта позволяет сократить количество замен и ремонтов элементов сложной технической системы, т.е. минимизировать экономические затраты на ее эксплуатацию при условии обеспечения заданного уровня надежности. Для этого рассмотрим небольшой тестовый пример.

Для создания начальной популяции при применении генетического алгоритма искусственного интеллекта необходимо сформировать наборы начальных решений. В нашем примере начальная популяция будет состоять из пяти наборов таких решений.

Первым набором решений начальной популяции генетического алгоритма искусственного интеллекта будем считать начальный набор оптимальных интервалов замен (ремонтов) элементов, состоящий из 10 интервалов и полученный на основании расчетов с помощью математической модели расчета оптимального интервала предупредительных замен [1–3] и рекуррентного метода расчета надежности сложной технической системы [4–6].

Для построения следующих (в данном случае четырех) наборов решений начальной популяции необходимо выполнить процесс сдвига по времени интервалов замен (ремонтов) нескольких различных элементов. Сдвиг по времени интервалов необходимо произвести таким образом, чтобы время замены данного элемента равнялось времени замены другого элемента с наименьшим интервалом или было кратно интервалу замены такого элемента. Для реализации таких сдвигов нужно осуществить процесс слияния. Под слиянием будем понимать выполнение одновременной замены (ремонта) двух и более элементов, при этом для одного элемента это будет оптимальное время, а для остальных – близкое к оптимальному.

Результат процедуры слияния для создания четырех дополнительных наборов решений из 10 оптимальных интервалов замен (ремонтов) элементов представлен в таблице. В первом столб-

це таблицы приведен начальный набор оптимальных интервалов, а в столбцах со второго по пятый — четыре набора оптимальных интервалов, созданные с помощью процедуры слияния. Таким образом в пяти столбцах представлены пять наборов оптимальных интервалов, образующих начальную популяцию.

Далее построенная популяция оценивается с использованием фитнес-функции. В данном случае фитнес-функция — это среднее значение суммы следующих затрат по всем пяти наборам:

стоимость замен (ремонтов) элементов сложной технической системы;

затраты на выезд специалистов для ремонта (замены) элементов;

стоимость времени работы элементов, которые были заменены раньше времени своей оптимальной замены.

Фитнес-функция является целевой функцией в данной задаче оптимизации, а оптимизация состоит в достижении наименьшего значения этой целевой функции. Для всех наборов решений осуществляется операция селекции (отбора) с помощью гибридного метода или метода “выборной рулетки”, включающий помимо метода “рулетки” элитарный отбор и отбор усечением.

Сначала все члены популяции оцениваются с помощью фитнес-функции. Далее применяется метод элитарного отбора и в новую популяцию отбирается 10% элитных членов популяции. Затем используется метод отбора усечением, при котором порог отсечения T принимается равным 90%. К оставшимся после элитарного отбора и отбора усечением 80% членов популяции применяется метод “рулетки”. С помощью полученного значения целевой функции можно построить следующую популяцию решений.

Для каждого решения, т.е. набора оптимальных интервалов, определяется своя сумма затрат. Далее для каждого i -го набора вычисляется по формуле (3.1) коэффициент участия K_i , т.е. отношение обратной величины затрат этого набора к сумме обратных величин затрат по всем наборам:

$$K_i = \frac{\frac{1}{S_i}}{\sum_{i=1}^n \frac{1}{S_i}}, \quad (3.1)$$

где $\frac{1}{S_i}$ — обратная величина затрат этого набора, $\sum_{i=1}^n \frac{1}{S_i}$ — сумма обратных величин затрат по всем наборам, n — количество всех наборов (в рассматриваемом примере $n = 5$).

Количество наилучших решений в качестве родителей для создания последующего набора решений задается таким образом: частота появления i -го набора решений в качестве родителя для создания дочерней популяции приблизительно равна коэффициенту участия K_i этого набора. Следовательно, наборы решений с большим коэффициентом участия K_i появляются в качестве родителей для создания дочерней популяции решений пропорционально большему числу раз, чем наборы с меньшим коэффициентом участия. Таким образом, из пяти наборов решений родительской популяции образуется 10 наборов решений для создания дочерней популяции, где каждый набор решений родительской популяции встречается такое количество раз, которое соответствует его коэффициенту участия.

Для построения дочерней популяции используется кроссинговер, включающий помимо метода одноточечного кроссинговера и метод кроссинговера с уменьшением замены. В данной операции применяется гибридный метод кроссинговера. Сначала из 10 родительских наборов случайным образом выбирается пять пар. Затем каждая пара родителей случайным образом порождает набор из 10 оптимальных интервалов, который является дочерним. Таким образом, образуется пять новых наборов оптимальных интервалов. Далее для полученной новой популяции рассчитывается фитнес-функция. Здесь возможны три варианта дальнейших действий.

Первый вариант: фитнес-функция дочерней популяции оказалась меньше, т.е. лучше, родительской. В этом случае продолжается поиск лучшего решения, т.е. повторяются операции селекция и кроссинговер и формируется новый дочерний набор.

Второй вариант: фитнес-функция дочерней популяции оказалась больше, т.е. хуже, родительской. Тогда применяется гибридный метод мутации, включающий помимо метода точечной

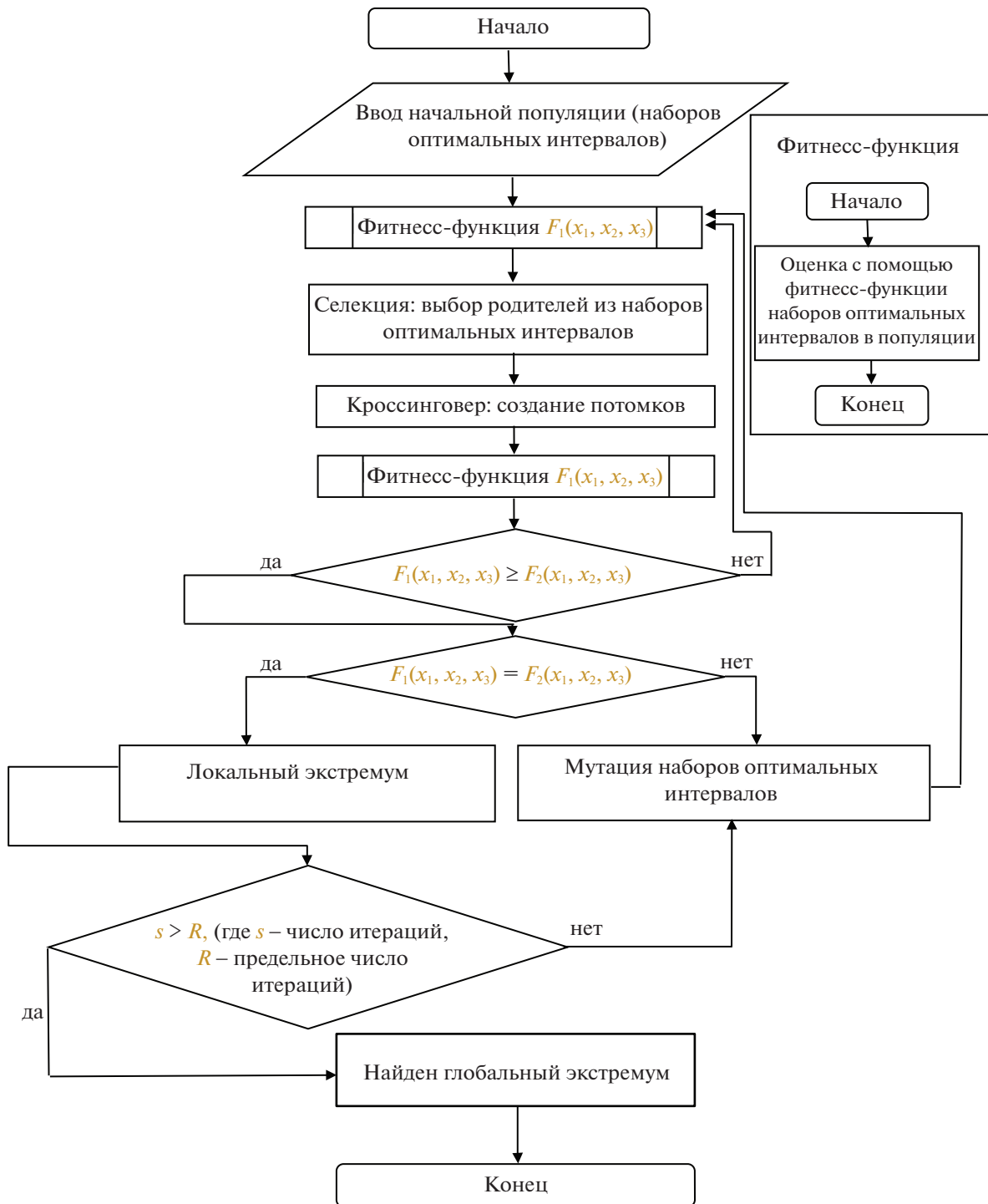


Рисунок. Блок-схема алгоритма решения задачи оптимизации

мутации метод мутации с вероятностью, т.е. случайное изменение набора из 10 оптимальных интервалов для того, чтобы заново начать поиск наилучшего решения.

Третий вариант: фитнес-функция дочерней популяции оказалась слишком близка к родительской. При этом возникает состояние схождения, т.е. такое состояние, когда фитнес-функция “замораживается” и перестает меняться. В такой ситуации кроссинговер практически никак не изменяет популяции, так как создаваемые при нем потомки представляют собой копии наборов из родительской популяции. В этом случае также происходит операция мутации, с помощью

гибридного метода мутации, который включает помимо метода точечной мутации метод мутации с вероятностью, и начинается новый поиск наилучшего решения, чтобы уйти от локального экстремума фитнес-функции и найти глобальный.

Поиск наилучшего решения прекращается, если заданное количество раз возникло состояние схождения с очень близкими значениями фитнес-функции и в результате выполнения мутаций происходит возврат в то же самое состояние. Это означает, что найдено минимальное значение фитнес-функции, т.е. наилучшее решение.

Блок-схема алгоритма решения поставленной задачи оптимизации замен элементов сложной технической системы с использованием разработанного гибридного генетического алгоритма представлена на рисунке. На основе этой блок-схемы разработано обеспечение для решения практических задач оптимизации замен элементов сложных технических систем.

4. Применение генетического алгоритма к практической задаче оптимизации замен элементов.

Разработанное программное обеспечение для решения задач оптимизации замен элементов сложных технических систем на основе генетического алгоритма искусственного интеллекта было применено для оптимизации замен или ремонтов элементов искусственного сооружения, являющегося сложной технической системой, а именно моста, расположенного через реку Нерусса 478 км II пути участка Брянск—Суземка. За 30 лет эксплуатации по регламенту планировалось осуществить 50 выездов на объект для замены или ремонтов элементов. За счет применения генетического алгоритма искусственного интеллекта для этого искусственного сооружения количество выездов на объект можно сократить до 20, а за счет синхронизации — уменьшить еще на два, что в общем итоге сократит количество выездов до 18, т.е. на 64%. Это приведет к снижению издержек на эксплуатацию данной сложной технической системы на 24%, т.е. принесет существенный экономический эффект при условии обеспечения заданного уровня надежности.

Заключение. Поставленная задача оптимизации замен элементов сложной технической системы в процессе ее эксплуатации с использованием разработанного генетического алгоритма искусственного интеллекта решена следующим образом. Первая часть задачи выполнена за счет построения совокупности интервалов замен (ремонтов) элементов для всей сложной технической системы. Это описано в предыдущих статьях авторов, посвященных математической модели расчета оптимального интервала предупредительных замен [2], рекуррентному методу расчета надежности сложных технических систем [4], а также алгоритмам, построенным на их основе [7–9]. Вторая часть задачи оптимизации осуществлена с помощью генетического алгоритма искусственного интеллекта.

Результатом решения задачи является оптимальная совокупность интервалов замен (ремонтов) элементов сложной технической системы в процессе ее эксплуатации, которая обеспечивает минимум целевой функции, определяемой экономическими затратами в процессе эксплуатации, с учетом ограничений на совокупность оптимальных интервалов замен (ремонтов), которые обеспечивают заданный уровень надежности. Таким образом достигается максимальный экономический эффект при заданном уровне надежности.

Построенный метод оптимизации замен (ремонтов) элементов можно применить для ряда сложных технических систем, в качестве которых рассматриваются искусственные сооружения, например мостовые конструкции, находящиеся в непосредственной близости друг от друга. В этом случае будут происходить оптимальные замены или ремонты сразу нескольких элементов на различных искусственных сооружениях, что даст дополнительную экономию. В качестве еще одного метода оптимизации можно использовать синхронизацию замены или ремонта с другими элементами, у которых подходит срок замены или ремонта путем небольшого продления их ресурса за счет снижения эксплуатационной нагрузки.

Модели эксплуатации сложных технических систем, которые могут быть рассчитаны с помощью алгоритмов параллельных вычислений [9], по своей сути, имеют вероятностный характер, так как все показатели надежности сложных технических систем определяются с помощью вероятностных оценок. В рассматриваемых более сложных стохастических моделях эксплуатации времени замен или ремонтов имеют не точечный, а интервальный характер [8].

Оптимальные интервалы замен (ремонтов) элементов сложной технической системы располагаются случайным образом и могут накладываться друг на друга, в том числе и многократно, что сильно усложняет задачу оптимизации и приводит к тому, что становится практически невозможным применение стандартных методов поиска оптимального решения. Это является первым существенным аргументом в пользу использования алгоритмов искусственного интеллекта при решении задач оптимизации эксплуатации сложных технических систем. Кроме того, сложные технические системы могут содержать очень большое число элементов, т.е. сотни ты-

сяч и более. При таком огромном количестве элементов стандартные методы поиска оптимального решения сталкиваются с большими трудностями как теоретическими, так и вычислительными. Это является вторым существенным аргументом для применения генетических алгоритмов искусственного интеллекта, например, методов группового учета аргументов, в том числе алгоритмов массовой селекции. Таким образом, в данной постановке задачи оптимизации времен замен (ремонтов) элементов сложных технических систем за все время эксплуатации использование генетических алгоритмов искусственного интеллекта может оказаться в некоторых случаях единственно возможным, а в большинстве случаев — наиболее эффективным решением для построения оптимальной схемы эксплуатации таких систем с целью достижения максимального экономического эффекта при условии обеспечения заданного уровня надежности.

СПИСОК ЛИТЕРАТУРЫ

1. Барзилович Е.Ю., Каишанов В.А. Некоторые математические вопросы теории обслуживания сложных систем. М.: Сов. радио, 1971. 340 с.
2. Кос О.И., Смирнов В.Ю. Оптимальный интервал предупредительных замен для искусственных сооружений железных дорог. Мир транспорта. М.: МИИТ, 2013. С. 152–155.
3. Гнеденко Б.В. Математические вопросы теории надежности / Под ред. Б.В. Гнеденко. М.: Радио и связь, 1983. 440 с.
4. Кос О.И. Рекуррентный алгоритм расчета и прогнозирования вероятности безотказной работы искусственных сооружений. Транспортное строительство. М.: ООО “Трансстройиздат”, 2014. С. 30–32.
5. Васильев А.И. Основы надежности транспортных сооружений: учебное пособие. М.: МАДИ, 46 с.
6. Вентцель Е.С. Теория вероятностей. 4-е изд. Перераб. и доп. М.: Наука, 1969. 576 с.
7. Кос О.И. Схема управления техническим состоянием искусственных сооружений. Мир транспорта. М.: МИИТ, 2016.
8. Smirnov V.U., Kos O.I. Program Module for Calculating the Optimal Interval of Preventive Substitutions // Intern. Conf. “Quality management, Transport and Information. Security Information Technologies” (IT&QM&IS). St.Petersburg, Russia, 2017. <https://doi.org/10.1109/ITMQIS.2017.8085811>
9. Kos O.I., Smirnov V.U., Eseva E.A., Adaptation of Reliability Calculation Software Packages for High Performance Distributed Computing Systems // Proc. IEEE Intern. Conf. “Quality Management, Transport and Information Security, Information Technologies” (IT&QM&IS). St.Petersburg, Russia: St. Petersburg Electrotechnical University “LETI”, 2018. P. 219–221. <https://doi.org/10.1109/ITMQIS.2018.8525095>.
10. Smirnov V.U., Yeseva E.A., Kos O.I. Rules and Regulations of Potential Impact of Acoustic Factors from High-Speed Railway Lines on Environment and Human Body. During Construction of New Facilities // Chaotic Modeling & Simulation Web Conf., Springer Proceedings in Complexity. Springer Nature Switzerland AG, 2021. P. 1–12. https://doi.org/10.1007/978-3-030-70795-8_61
11. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. М.: Телеком, 2006. 452 с.
12. Poli R., Langdon W.B., McPhee N.F., Koza J.R. A Field Guide to Genetic Programming // Lulu Enterprises, UK Ltd, 2008. 252 p.
13. Павченко Т.В. Генетические алгоритмы: учебно-методическое пособие / Под ред. Тарасевича Ю.Ю. Астрахань: Издательский дом “Астраханский университет”, 2007. 87 с.
14. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы / Под ред. Курейчика В.М. 2-е изд. Испр. и доп. М.: Физматлит, 2010. 368 с. ISBN 978-5-9221-0510-1.