



ОЦЕНКА ЭФФЕКТИВНОСТИ МЕТОДОВ ПОСТРОЕНИЯ ОТКАЗОУСТОЙЧИВЫХ СТРУКТУР В ПРОГРАММНОМ КОМПЛЕКСЕ SANDS

А. А. Семенов, В. Л. Оленев

Санкт-Петербургский государственный университет аэрокосмического приборостроения

В статье исследуется эффективность алгоритмов автоматизированного построения отказоустойчивых структур в программном комплексе SANDS, предназначенном для проектирования бортовых сетей стандарта SpaceWire. Анализируются методы повышения отказоустойчивости, включая резервирование, избыточность связей и автоматическое восстановление. На примере трех схем различной сложности демонстрируются преимущества и недостатки встроенных алгоритмов SANDS, в частности, избыточное использование коммутаторов. Предложена новая формула для расчета оптимального количества коммутаторов, позволяющая сократить их число без снижения надежности системы. Проведено сравнение результатов автоматизированного и экспертного подходов, подтвердившее эффективность предложенного метода. Обсуждаются ограничения метода и возможные пути его улучшения, включая учет исходной структуры сети. Практическая значимость работы заключается в оптимизации проектирования отказоустойчивых систем для аэрокосмической отрасли. Результаты исследования могут быть использованы для совершенствования алгоритмов SANDS и разработки более гибких инструментов автоматизированного проектирования.

Ключевые слова: отказоустойчивость, SpaceWire, SANDS, автоматизированное проектирование, резервирование, оптимизация сетей.

Для цитирования:

Семенов, А. А. Оценка эффективности методов построения отказоустойчивых структур в программном комплексе SANDS / А. А. Семенов, В. Л. Оленев // Системный анализ и логистика. – 2025. – № 2(45). – с. 29-41. DOI: 10.31799/2077-5687-2025-2-29-41.

Введение

Одним из самых важных свойств систем, работающих в условиях возможных сбоев, является отказоустойчивость. Атомная, космическая и авиационная, робототехническая области, где оборудование подвержено жестким условиям эксплуатации, являются тому подтверждением. Длительная работа в экстремальных условиях неизбежно приводит к возникновению отказа какого-либо элемента системы. Одно из возможных решений данной проблемы – создание отказоустойчивых структур.

В настоящее время большинство систем автоматического проектирования, поддерживающие SpaceWire, могут только моделировать системы – они не могут проектировать и автоматически строить отказоустойчивые структуры [1]. Однако SANDS – SpaceWire Automated Network Design and Simulation – позволяет не только вручную строить схемы и моделировать поведение данной схемы, но и автоматически дорабатывать схемы до требуемой отказоустойчивости благодаря встроенным алгоритмам доработки и построения сети.

Встроенные в SANDS алгоритмы автоматического построения отказоустойчивых структур, безусловно, имеют плюсы, один из них – экономия времени при проектировании отказоустойчивой схемы, однако при использовании мы должны быть уверены в их эффективности.

Поэтому цели данной статьи – проанализировать эффективность алгоритмов построения сети в программном комплексе SANDS и выявить возможные недостатки.

1. Отказоустойчивость

Отказоустойчивость позволяет системе сохранять работоспособность при отказе отдельных её компонентов [2]. В современных программных и аппаратных системах решения по созданию отказоустойчивых структур представляют собой в большинстве случаев введение



избыточности в сети, как, например, в работах [3, 4, 5]. Это не единственный способ достичь отказоустойчивости, так как она включает в себя несколько принципов:

- Резервирование — дублирование критически важных компонентов системы для обеспечения их бесперебойной работы [6];
- Избыточность связей — создание альтернативных путей передачи данных между компонентами системы [7];
- Автоматическое восстановление — переключение на резервные компоненты или пути передачи данных при обнаружении отказа.

Для оценки уровня отказоустойчивости используются следующие критерии:

- Время восстановления — период, необходимый для возврата системы в работоспособное состояние после отказа;
- Глубина отказоустойчивости — количество последовательных отказов, которые система может выдержать без потери работоспособности [8].

Для оценки отказоустойчивости в данной работе мы будем использовать критерий глубины.

2. Система автоматизированного проектирования и моделирования SANDS

Система SANDS обладает расширенным функционалом, предназначенным для проектирования, разработки и моделирования систем стандарта SpaceWire, одного из самых популярных стандартов для проектирования бортовых сетей [9]. При этом осуществляется детальное описание технических параметров и структурных особенностей сети, распределение потоков данных, а также анализ работы сети с учётом таких факторов, как задержки передачи, вероятность возникновения ошибок в каналах связи и других параметров, влияющих на производительность [10].

Архитектура SANDS включает четыре основных компонента [10]:

- Компонент 1 – проектирование физической структуры и оценка характеристик сети;
- Компонент 2 – прокладка маршрутов передачи данных в сети;
- Компонент 3 – генерация таблиц планирования транспортного протокола СТП-ИСС для передачи данных;
- Компонент 4 – моделирование работы сети.

В контексте данной статьи мы будем взаимодействовать с компонентом 1.

3. Алгоритмы SANDS

SANDS строит отказоустойчивые структуры в два этапа:

- 1) Создание сети с требуемым уровнем отказоустойчивости;
- 2) Итерационная обработка сети.

На первом этапе требуемый уровень отказоустойчивости сети достигается при помощи трёх дополнительных алгоритмов:

- 1) Достаивание комплектов и портов в терминальных узлах;
- 2) Достаивание коммутаторов в сети;
- 3) Построение связи между коммутаторами.

Каждый из этих алгоритмов и этапов был подробно рассмотрен в работах [10, 11]. В данной статье для наглядности представим блок-схему, демонстрирующую порядок создания отказоустойчивой структуры (Рис. 1).

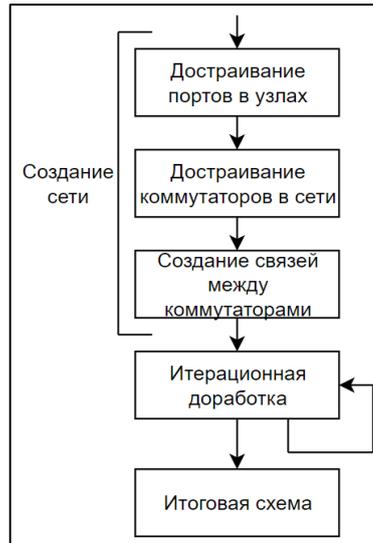


Рис. 1. Блок-схема алгоритмов

4. Демонстрация работы алгоритмов

Анализ эффективности алгоритмов SANDS проведем на 3-х разных по сложности схемах. Для этого вручную создадим три схемы, которые не соответствуют требованиям отказоустойчивости из-за наличия узких мест и отличаются друг от друга количеством узлов, изначальных коммутаторов и топологией. Схемы, готовые к преобразованию, показаны на Рис. 2, Рис. 3, Рис. 4 соответственно.

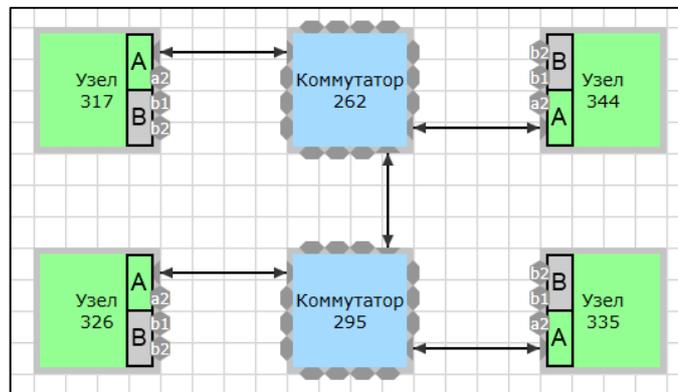


Рис. 2. Схема №1

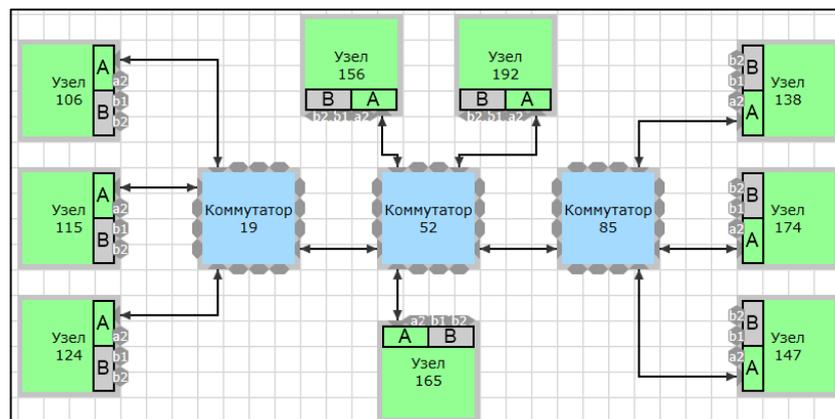


Рис. 3. Схема №2

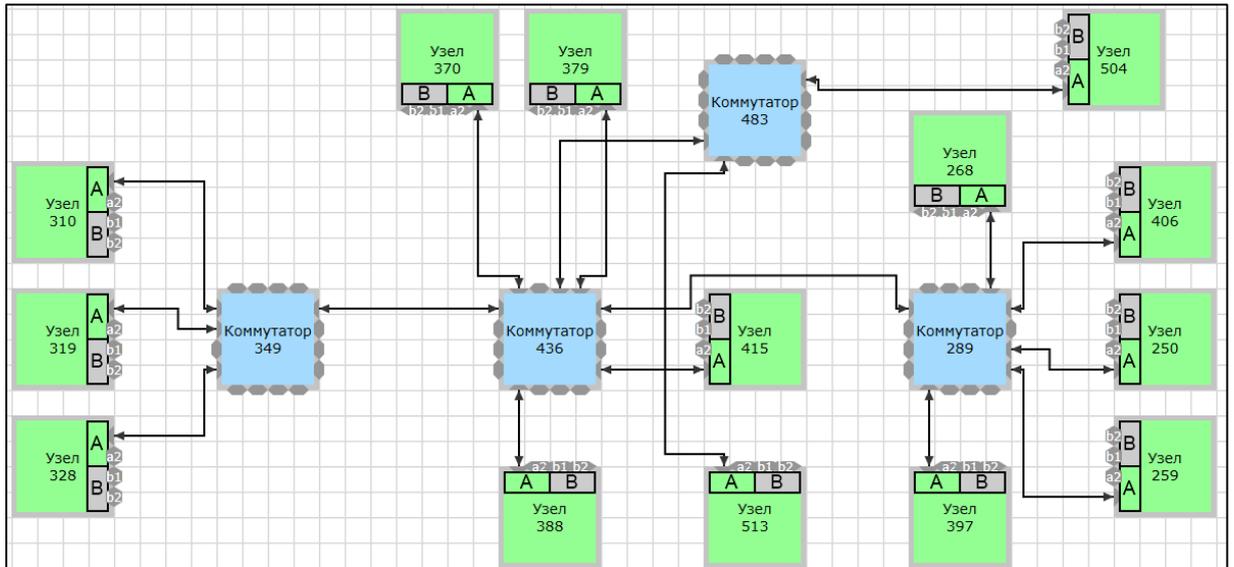


Рис. 4. Схема №3

Теперь последовательно для каждой схемы зададим требуемый уровень отказоустойчивости 3. После подтверждения изменений и ожидания вычислений получаем преобразованные алгоритмом схемы (Рис. 5, Рис. 7, Рис. 9). Их свойства – на Рис. 6, Рис. 8, Рис. 10.

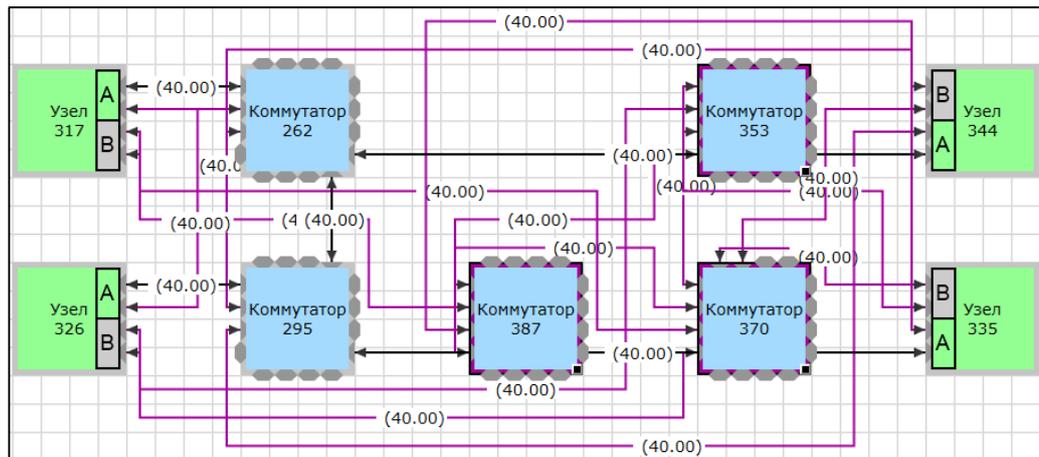


Рис. 5. Преобразованная схема №1

Общие свойства:
 - Общее количество коммутаторов в сети - 5
 - Общее количество узлов (основных комплектов) - 4
 - Общее количество резервных комплектов в сети - 4
 - Вес - 4,99 кг
 - Диаметр сети - 1 хопов
 - Максимальная скорость передачи данных - 50,00 Мбит/с
 - Минимальная скорость передачи данных - 50,00 Мбит/с
 - Отказоустойчивость всей сети - 3

Рис. 6. Свойства преобразованной схемы №1

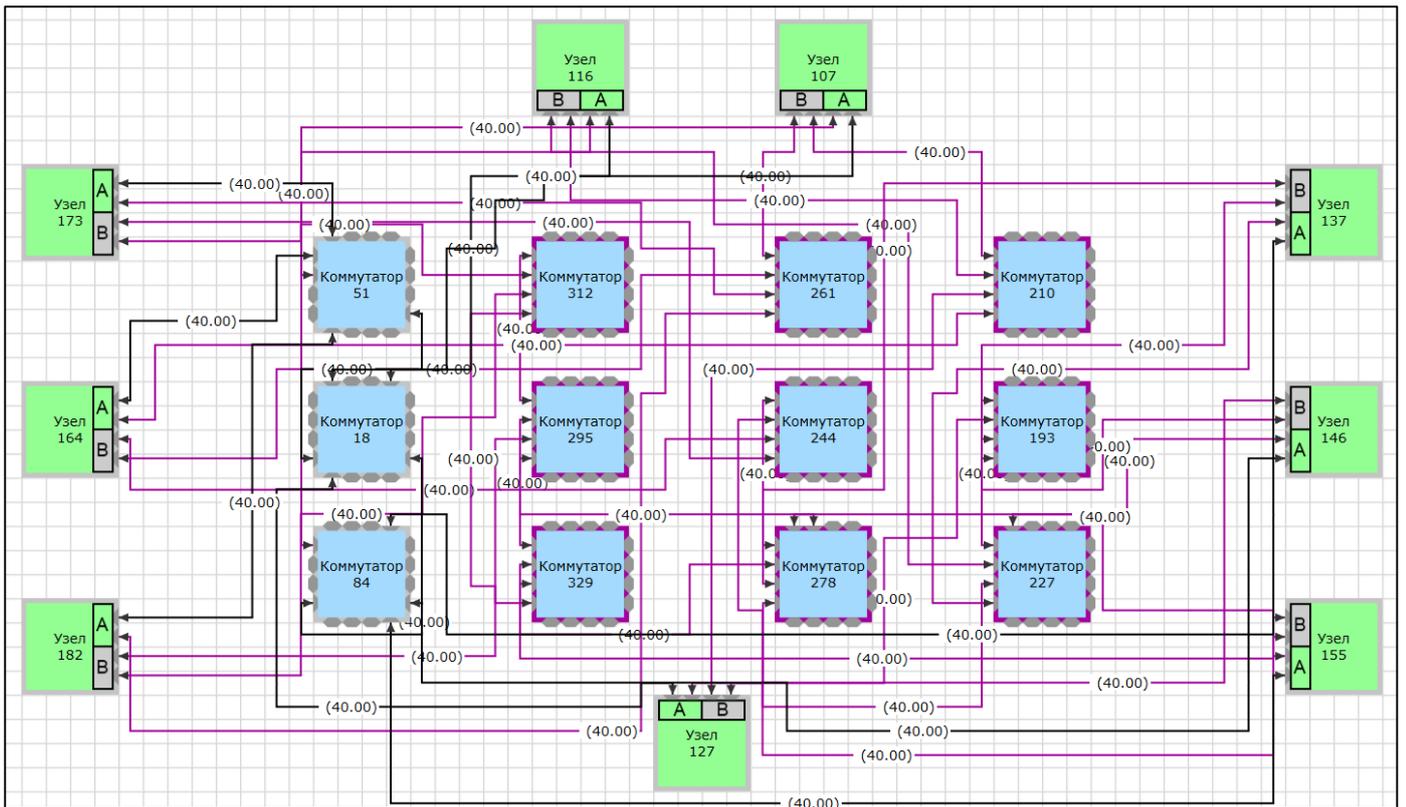


Рис. 7. Преобразованная схема №2

Общие свойства:
 Общее количество коммутаторов в сети - 12
 Общее количество узлов (основных комплектов) - 9
 Общее количество резервных комплектов в сети - 9
 Вес - 10.62 кг
 Диаметр сети - 3 хопов
 Максимальная скорость передачи данных - 50.00 Мбит/с
 Минимальная скорость передачи данных - 50.00 Мбит/с
 Отказоустойчивость всей сети - 3

Рис. 8. Свойства преобразованной схемы №2

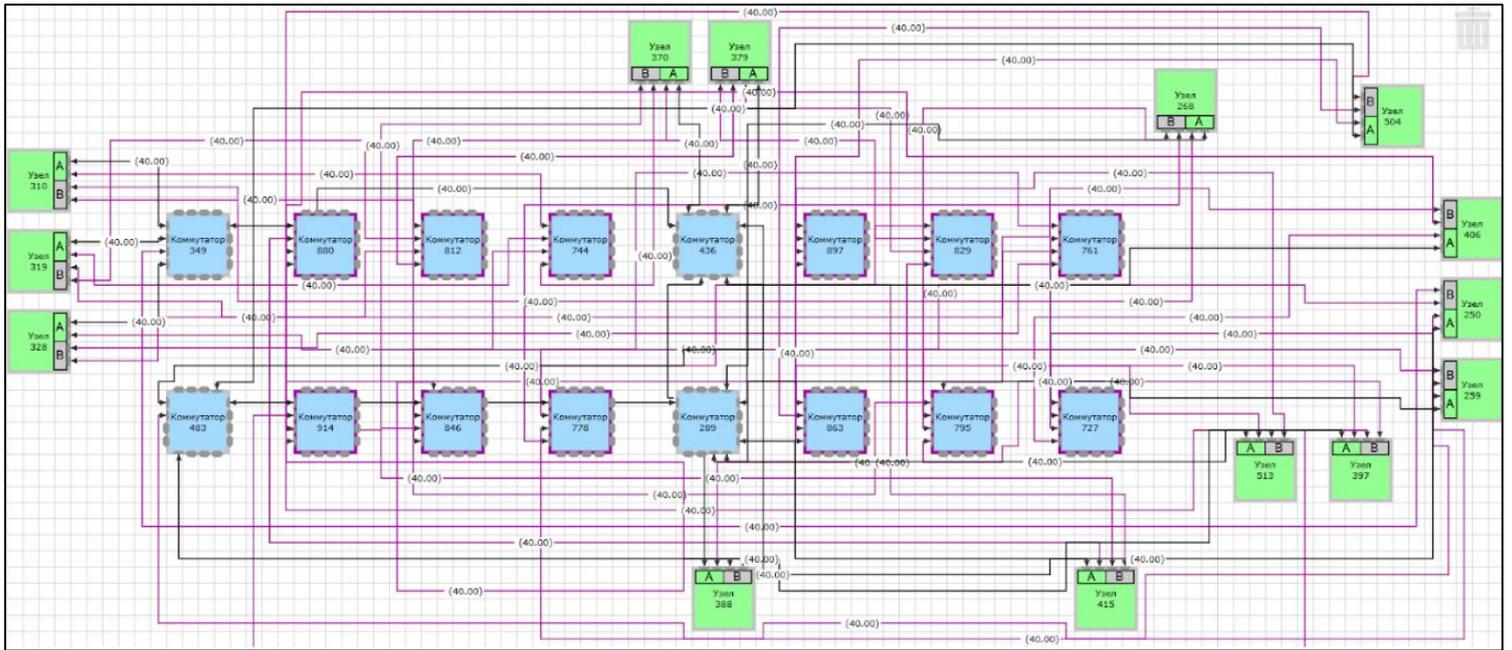


Рис. 9. Преобразованная схема №3

Общие свойства:	
Общее количество коммутаторов в сети -	16
Общее количество узлов (основных комплектов) -	14
Общее количество резервных комплектов в сети -	14
Вес -	59.20 кг
Диаметр сети -	4 хопов
Максимальная скорость передачи данных -	50.00 Мбит/с
Минимальная скорость передачи данных -	50.00 Мбит/с
Отказоустойчивость всей сети -	3

Рис. 10. Свойства преобразованной схемы №3

Проанализируем полученные схемы. Для всех 3-х случаев можно выделить общий подход SANDS к поставленной задаче: сделать число терминальных узлов и коммутаторов пропорциональным, чтобы одному узлу соответствовал \pm один коммутатор. И если данный подход выглядит вполне уместным для малых и средних схем, то для сложных схем данный метод может привести к чрезмерному количеству коммутаторов, что ведет к большей массе, повышенному энергопотреблению и прочему.

5. Метод расчёта оптимального количества коммутаторов

Очевидно, что данный метод создания отказоустойчивой структуры имеет недостаток в большом количестве коммутирующих элементов. Поэтому мы предлагаем новый метод расчёта количества коммутаторов. Его можно сформулировать так: “Необходимое минимальное количество коммутаторов для схемы рассчитывается по формуле:

$$C_A = \left\lceil \frac{N_N \times P_N}{P_C} \right\rceil + 1 \quad (1)$$

где: C_A (Commutator_All) – нужное количество коммутаторов, P_N (Ports_Node) – количество портов у одного узла, P_C (Ports_Commutator) – количество портов у одного коммутатора, N_N (Number_nodes) – число терминальных узлов на схеме”.



Данная формула предусматривает узлы и коммутаторы с любым количеством портов – 4, 6, 8 и так далее – а её использование позволяет сильно уменьшить итоговое количество коммутаторов на схеме при том же уровне отказоустойчивости. Так, по нашей формуле для первой схемы мы можем обойтись 3-мя коммутаторами, для второй схемы – 7-мью, для третьей – 8-мью, так как получаем:

$$C_{A1} = \left\lceil \frac{N_N \times P_N}{P_C} \right\rceil + 1 \rightarrow C_{A1} = \left\lceil \frac{4 \times 4}{8} \right\rceil + 1 = 3 \quad (2)$$

$$C_{A2} = \left\lceil \frac{N_N \times P_N}{P_C} \right\rceil + 1 \rightarrow C_{A2} = \left\lceil \frac{9 \times 4}{8} \right\rceil + 1 = 6 \quad (3)$$

$$C_{A3} = \left\lceil \frac{N_N \times P_N}{P_C} \right\rceil + 1 \rightarrow C_{A3} = \left\lceil \frac{14 \times 4}{8} \right\rceil + 1 = 8 \quad (4)$$

На основе вычислений данной формулы создадим обновлённые схемы (Рис. 11, Рис. 13, Рис. 15). Их свойства – на Рис. 12, Рис. 14, Рис. 16.

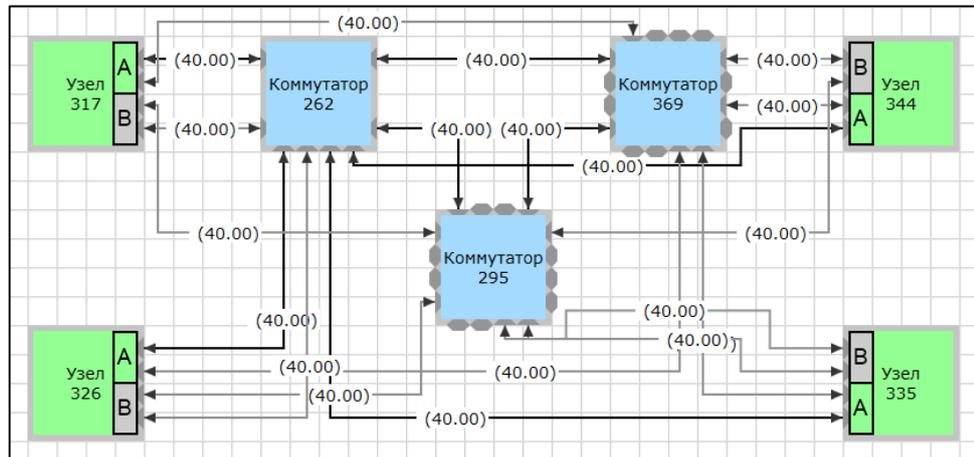


Рис. 11. Схема №1, преобразованная при помощи формулы

Общие свойства:	
Общее количество коммутаторов в сети -	3
Общее количество узлов (основных комплектов) -	4
Общее количество резервных комплектов в сети -	4
Вес -	17.05 кг
Диаметр сети -	1 хопов
Максимальная скорость передачи данных -	50.00 Мбит/с
Минимальная скорость передачи данных -	50.00 Мбит/с
Отказоустойчивость всей сети -	2

Рис. 12. Свойства схемы №1, преобразованной при помощи формулы

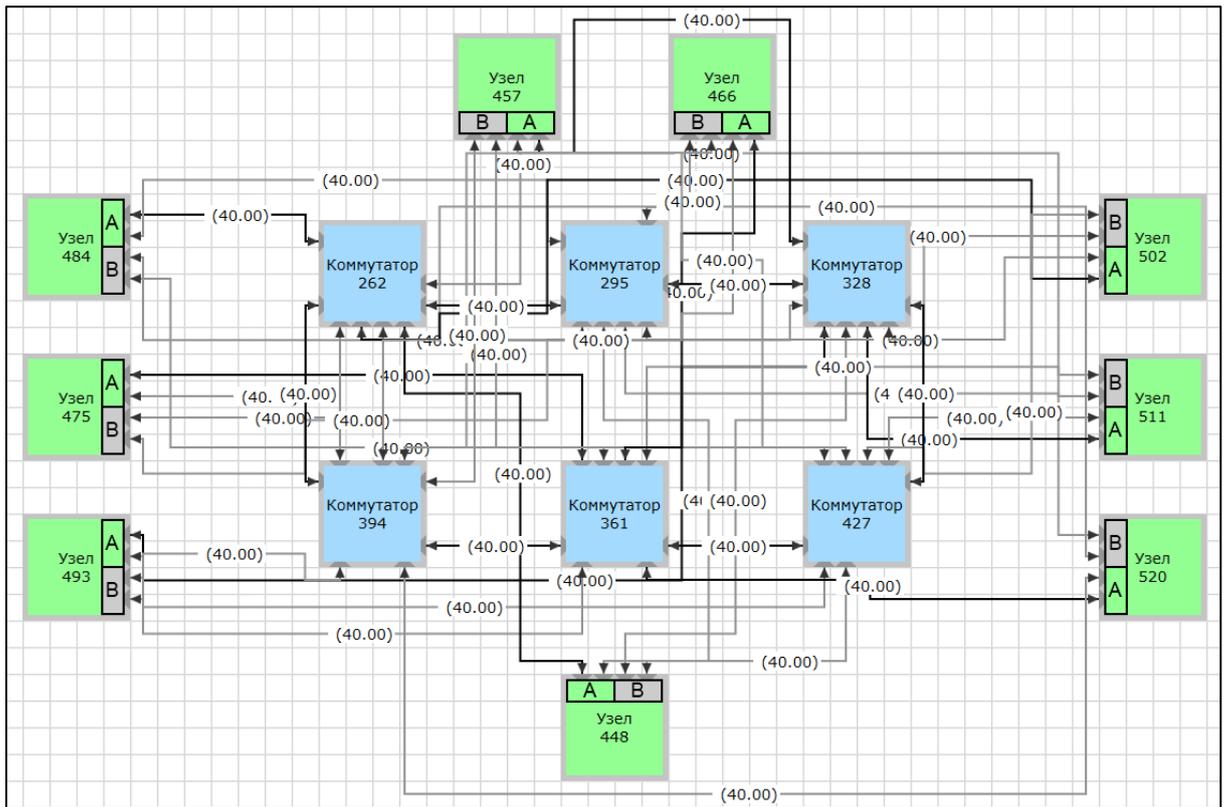


Рис. 13. Схема №2, преобразованная при помощи формулы

Общие свойства:	
Общее количество коммутаторов в сети -	6
Общее количество узлов (основных комплектов) -	9
Общее количество резервных комплектов в сети -	9
Вес -	37.50 кг
Диаметр сети -	1 хопов
Максимальная скорость передачи данных -	50.00 Мбит/с
Минимальная скорость передачи данных -	50.00 Мбит/с
Отказоустойчивость всей сети -	3

Рис. 14. Свойства схемы №2, преобразованной при помощи формулы

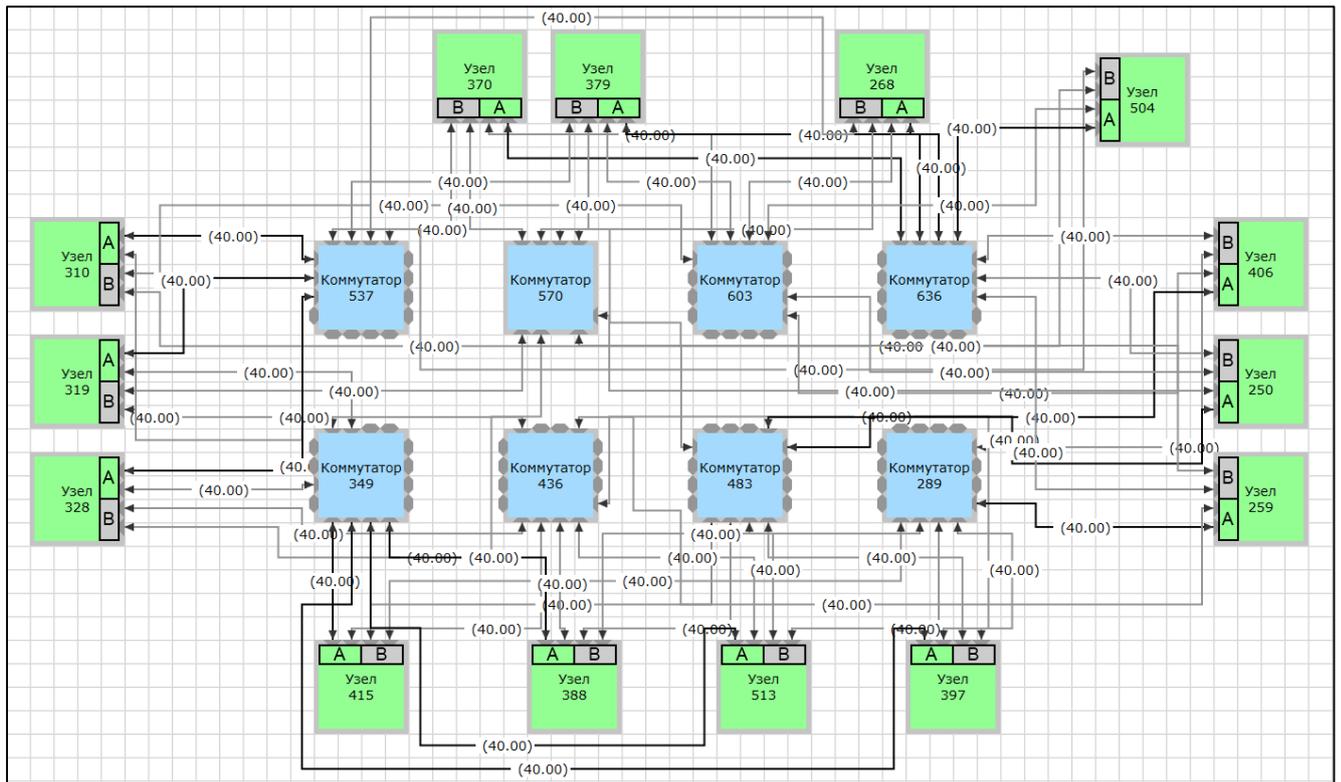


Рис. 15. Схема №3, преобразованная при помощи формулы

<p>Общие свойства:</p> <p>Общее количество коммутаторов в сети - 8</p> <p>Общее количество узлов (основных комплектов) - 14</p> <p>Общее количество резервных комплектов в сети - 14</p> <p>Вес - 50.00 кг</p> <p>Диаметр сети - 3 хопов</p> <p>Максимальная скорость передачи данных - 50.00 Мбит/с</p> <p>Минимальная скорость передачи данных - 50.00 Мбит/с</p> <p>Отказоустойчивость всей сети - 3</p>

Рис. 16. Свойства схемы №3, преобразованной при помощи формулы

Исходя из рисунков и свойств схем можно сделать вывод, что формула действительно позволяет уменьшить количество коммутирующих элементов. Однако данный метод имеет два недостатка:

- Нельзя выбрать требуемый уровень отказоустойчивости;
- Изначальная сетевая структура схемы нарушается.

Если первый недостаток можно аргументировать тем, что формула дает максимальный результат, что ведёт к максимально возможному уровню отказоустойчивости при минимальном изменении схемы, то второй недостаток можно частично решить, если изменить формулу так, чтобы она учитывала изначальное количество коммутаторов:

$$C_A = \left\lceil \frac{N_N \times P_N}{P_C} \right\rceil + 1 + \left\lceil \frac{FP_N + FP_C + 1}{P_C} \right\rceil \quad (5)$$

где: C_A (Commutator_All) – нужное количество коммутаторов, P_N (Ports_Node) – количество портов у одного узла, P_C (Ports_Commutator) – количество портов у одного



коммутатора, N_N (Number_nodes) – число терминальных узлов на схеме, FP_N – количество свободных портов у узлов, FP_C – количество свободных портов у коммутаторов.

Теперь формула будет учитывать количество изначальных коммутаторов при условии, что они образуют единую сетевую структуру.

6. Программная реализация

Когда мы взаимодействуем с компонентом 1 – оценивая сеть или автоматически её достраивая – SANDS создает конфигурационный файл типа «.xml», в котором содержится информация о структуре сети: количество коммутаторов, узлов, их портов и так далее. Поэтому на данном этапе новый метод можно реализовать в виде сторонней программы на C++, которая на вход получает конфигурационный файл схемы, которая требует доработки, а на выходе пользователь получает обновленный «.xml» файл с новым количеством коммутаторов. Пример работы программы для схемы №3 приведен на Рис. 17.

```
-----  
Введите имя входного XML файла (например: input.xml): C:\Users\Arseniy\Desktop\Работы\SANDS\Scheme_3  
_For_Program_export_parameters.xml  
Введите имя выходного XML файла (например: output.xml): Result.xml  
  
Результаты анализа:  
-----  
Количество узлов: 14  
Портов на одном узле: 4  
Портов на одном коммутаторе: 8  
Найдено изначальных коммутаторов: 4  
-----  
Требуется добавить коммутаторов: 4  
-----  
Успешно создан новый XML файл: Result.xml  
Добавлено коммутаторов: 4  
Использован базовый ID: 148
```

Рис. 17. Пример выполнения программы

7. Сравнение количественных метрик решений

Чтобы наглядно продемонстрировать эффективность нового метода, отразим на графиках результаты анализа при разном количестве узлов (Рис. 18, Рис. 19).

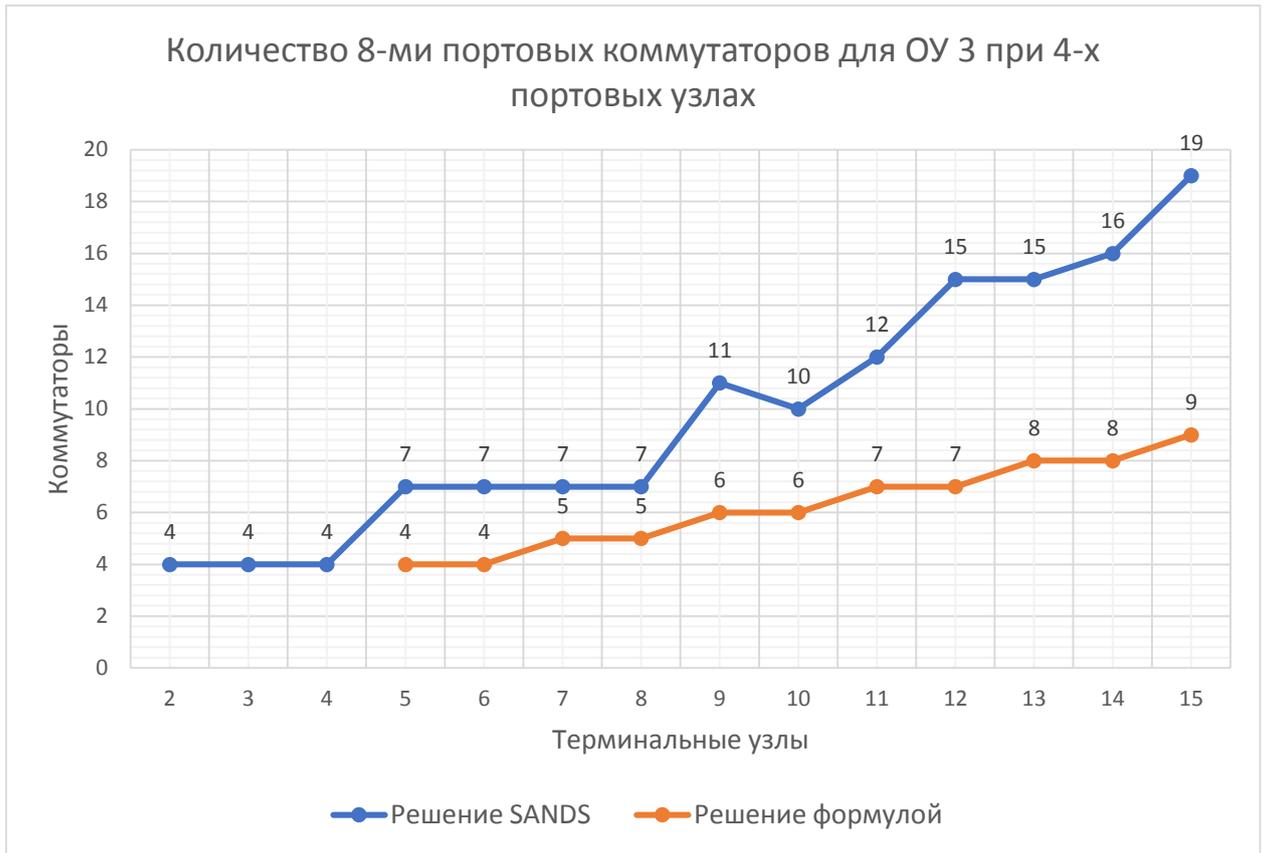


Рис. 18. Визуализация анализа при 4-х портовых узлах

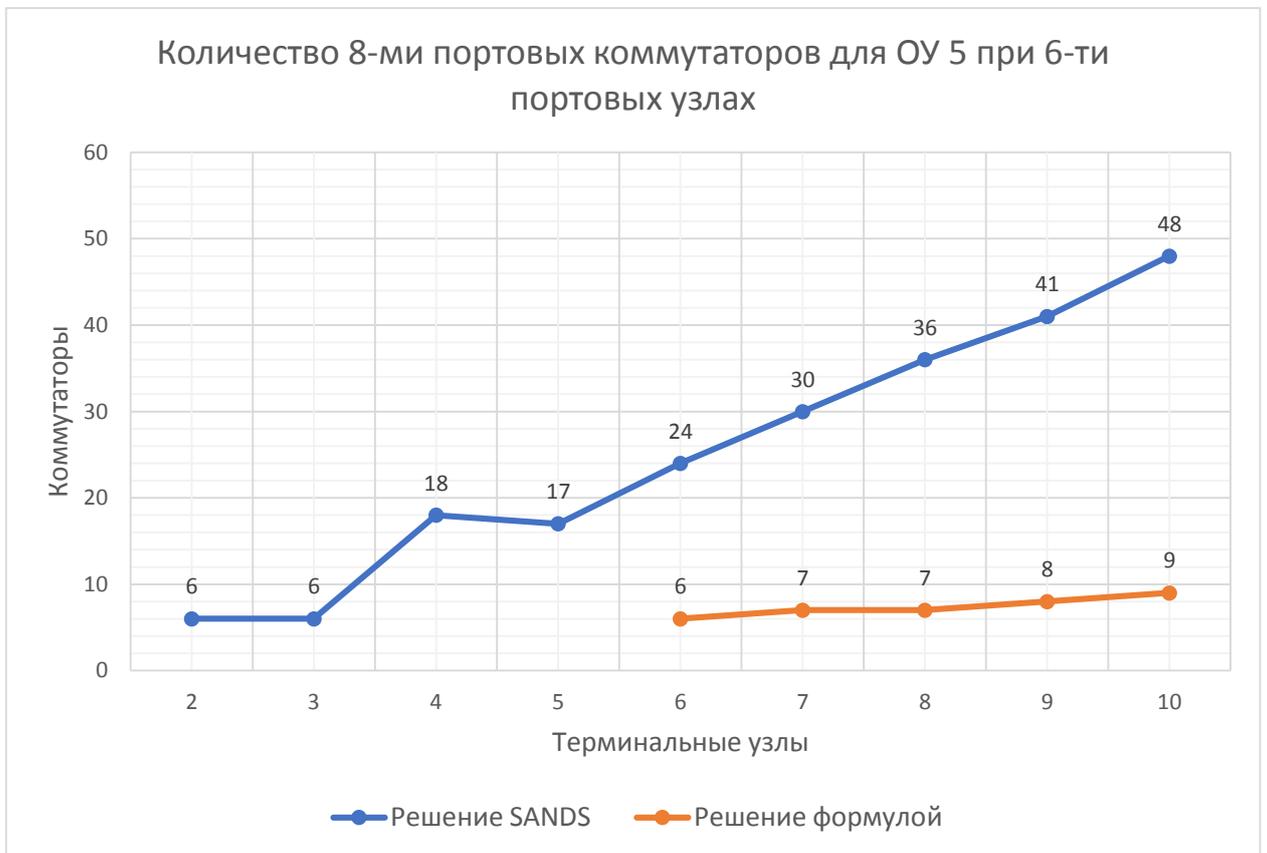


Рис. 19. Визуализация анализа при 6-ти портовых узлах



Заключение

Работа выполнена при финансовой поддержке Министерства науки и высшего образования Российской Федерации, соглашение № FSRF-2023-0003, "Фундаментальные основы построения помехозащищенных систем космической и спутниковой связи, относительной навигации, технического зрения и аэрокосмического мониторинга".

Проведенное исследование позволило оценить эффективность алгоритмов автоматизированного построения отказоустойчивых структур в программном комплексе SANDS. Результаты показали, что, несмотря на успешное достижение заданного уровня отказоустойчивости, алгоритмы демонстрируют избыточность в использовании коммутаторов, особенно для сложных схем, что увеличивает ресурсозатратность проектируемых систем.

Предложенный метод расчета оптимального количества коммутаторов на основе аналитической формулы подтвердил свою эффективность, значительно сокращая число коммутирующих элементов при сохранении требуемой надежности. Однако выявлены ограничения метода, такие как невозможность выбора уровня отказоустойчивости и нарушение исходной структуры сети. Модификация формулы с учетом начальной конфигурации схемы частично решает эти проблемы.

Практическая значимость работы заключается в возможности оптимизации проектирования отказоустойчивых систем, что особенно актуально для аэрокосмической отрасли. Дальнейшие исследования целесообразно направить на разработку адаптивных алгоритмов, сочетающих автоматизацию с экспертными решениями, а также на интеграцию предложенных методов в функционал SANDS для повышения его гибкости и эффективности.

Работа вносит вклад в развитие методов автоматизированного проектирования, предлагая конкретные пути совершенствования инструментов для создания сложных отказоустойчивых структур.

СПИСОК ЛИТЕРАТУРЫ

1. *Sheynin Y., Olenev V., Lavrovskaya I., Korobkov I., Kochura S., Shkolniy V., Dymov D.* Computer-Aided Design System for On-board SpaceWire Networks Simulation and Design // Proc. of the 20th Conference of Open Innovations Association FRUCT. – St. Petersburg, 2017. – P. 398-405.
2. *Xu J.* Topological structure and analysis of interconnection networks / J. Xu. — Dordrecht: Kluwer Academic Publishers, 2001. — 350 p.
3. *Суворова Е. А., Шейнин Ю. Е.* Проектирование бортовых сетей SpaceFibre с пространственным резервированием // XLIV Академические чтения по космонавтике, посвященные памяти академика С. П. Королёва и других выдающихся отечественных ученых – пионеров освоения космического пространства: сб. тезисов. – Москва, 2020. – Т. 2. – С. 226–228.
4. *Глухих М.И., Моисеев М.Ю., Егоров И.В., Крикун Т.С.* Автоматизация анализа надежности невосстанавливаемых информационно-управляющих систем / М. И. Глухих, М. Ю. Моисеев, И. В. Егоров, Т. С. Крикун // Информатика, телекоммуникации и управление. – 2012. – №2. – С. 81-90
5. *Maurer A., Balagurin O., Greiner T., Herbst T., Kaiser T., Kayal H., Riegler C., Schwarz T.* Hardware and Software Redundancy Concepts on-board of SONATE-2 // Proceedings of 73rd International Astronautical Congress (IAC 2022). — Paris, 2022. — P. 5–8.
6. *Olenev V. L.* Analysis of requirements for modern spacecraft onboard network protocols. / V. L. Olenev // Информационно-управляющие системы. – 2021. – №1. – С. 8-16.
7. *Blokydyk G.* Network Redundancy. A Complete Guide / G. Blokydyk. — USA: 5STARCOoks, 2022. — 307 p.
8. *Гаврилов М. А.* Надежность дискретных систем / М. А. Гаврилов, В. М. Остиану,



- А. И. Потехин // Итоги науки. Серия «Теория вероятностей. Математическая статистика. Теоретическая кибернетика, 1969». — Москва, 1970. — С. 7-104.
9. ECSS-E-ST-50-52C. SpaceWire — Remote memory access protocol. – Noordwijk: ESA Publications Division ESTEC, 2010. – 5 Feb. – 109 с.
10. *Оленев В. Л.* Автоматическое построение отказоустойчивых бортовых сетей / В. Л. Оленев // Научные технологии в космических исследованиях Земли. – 2023. – Т. 15. – № 4. – С. 4-13.
11. *Лавровская И. Я.* Трансформация сети для повышения отказоустойчивости в бортовых сетях SpaceWire / И. Я. Лавровская, В. Л. Оленев // Научная сессия ГУАП: сб. докладов. В 3-х частях. – Санкт-Петербург, 2018. – С. 157-167.

ИНФОРМАЦИЯ ОБ АВТОРАХ

Семенов Арсений Алексеевич

Студент

Санкт-Петербургский государственный университет аэрокосмического приборостроения

Россия, 190000, Санкт-Петербург, ул. Большая Морская, д.67, лит. А

E-mail: qquestions7725@gmail.com

Оленев Валентин Леонидович

канд. техн. наук, доцент

Санкт-Петербургский государственный университет аэрокосмического приборостроения

Россия, 190000, Санкт-Петербург, ул. Большая Морская, д.67, лит. А

E-mail: valentin.olenev@guap.ru

Дата поступления: 08.04.2025

Дата принятия: 01.07.2025