

ИЗЫСКАНИЕ, ПРОЕКТИРОВАНИЕ,
СТРОИТЕЛЬСТВО И МОНТАЖ
ТЕХНОЛОГИЧЕСКОГО ОБОРУДОВАНИЯ

УДК 004.75

ВЕРОЯТНОСТНЫЙ МЕТОД И МОДЕЛЬ ДЛЯ ОЦЕНКИ
ПАРАМЕТРОВ МЕЖМОДУЛЬНОГО ВЗАИМОДЕЙСТВИЯ В
СИСТЕМЕ УПРАВЛЕНИЯ ЗАЩИТОЙ ТУРБОАГРЕГАТА ПО
ВИБРАЦИИ

© 2021 Д.А. Плотников

*Южно-Российский государственный политехнический университет (НПИ)
имени М.И. Платова, Новочеркасск, Ростовская обл., Россия*

В работе кратко рассмотрена структура модульной системы управления защитой турбоагрегата по вибрации (СУЗВ), сформулированы задачи оценки параметров межмодульного взаимодействия, возникающие на ранних этапах проектирования СУЗВ. На основе анализа алгоритмов работы различных служб протокола *CANopen* предложен вероятностный метод определения основных параметров коммуникационных процессов в СУЗВ с использованием элементов теории массового обслуживания. Рассмотрена математическая модель СУЗВ, использующая предложенный метод, описаны её входные параметры, алгоритм функционирования и выходные данные. Описана методика исследования реальной СУЗВ с помощью модели, приведены результаты моделирования, выполнен их анализ, на базе которого предложены способы оптимизации некоторых параметров межмодульного взаимодействия.

Ключевые слова: защита по вибрации, вибромониторинг, модульная система, межмодульное взаимодействие, система массового обслуживания, моделирование *CANopen*, шина *CAN*.

Поступила в редакцию 05.07.2021

После доработки 08.07.2021

Принята к печати 19.07.2021

Введение

Возникновение и развитие дефектов в различных частях роторной машины в большинстве случаев сопровождается изменением параметров вибрации её элементов, а в некоторых ситуациях вибрация является не только индикатором, но и причиной развития дефектов. Поэтому одной из наиболее универсальных и совершенных методик диагностики роторных машин, к которым относятся и турбоагрегаты (ТА) электростанций, является вибродиагностика, т.е. выявление дефектов на основе анализа параметров вибрации элементов машины. Следовательно, разработка и внедрение высоконадёжных средств вибромониторинга, вибродиагностики и защиты по вибрации позволит продлить срок службы ТА, сократить сроки ремонта за счёт обнаружения дефектов на самых ранних стадиях их возникновения, а также предотвратить аварии с катастрофическими последствиями путём своевременной остановки ТА в случае недопустимого повышения уровня вибрации.

Современные системы управления защитой ТА по вибрации (СУЗВ) проектируются в соответствии с многоуровневой структурой. На нижнем уровне находятся первичные и вторичные преобразователи измеряемых параметров: ортогональных составляющих вибрации опор и шеек роторов, осевого расширения и сдвига ротора и некоторые другие. Информация с них поступает на средний уровень, где в результате её комплексной обработки формируются управляющие воздействия, инициирующие при необходимости аварийный останов ТА или включение

сигнализации. Наконец, на верхнем уровне на основе углублённого анализа текущих и ретроспективных данных с использованием прогнозных моделей формируются рекомендации по срокам профилактических работ, по проведению технического обслуживания, по замене элементов агрегата.

Из сказанного следует, что наиболее ответственные решения по автоматическому управлению защитой ТА принимаются на среднем уровне СУЗВ: безосновательный нештатный останов агрегата влечёт за собой существенные материальные затраты на его повторный ввод в эксплуатацию. К ещё большим проблемам, вплоть до разрушения ТА, может привести отсутствие сигнала на аварийный останов ТА при недопустимом повышении вибрации. Таким образом, разработке надёжных алгоритмов обработки, передачи и хранения данных при проектировании среднего уровня СУЗВ должно уделяться особое внимание.

Средний уровень СУЗВ обычно имеет модульную структуру, показанную на рисунке 1. В неё входят модули M различных типов, количество модулей каждого типа – m_k ($k = \{1, 2, \dots, M\}$). Модули взаимодействуют друг с другом по общей сети передачи данных (СПД). Описанная структура обусловлена тем, что для ТА электрических станций характерна необходимость измерения однотипных групп параметров в нескольких точках, количество которых зависит от сложности ТА. Примерами таких групп могут служить ортогональные составляющие вибрации опор, шеек роторов и некоторые другие. Каждый модуль СУЗВ обслуживает одну группу параметров, а количество модулей определяется сложностью ТА. Такой подход улучшает ремонтпригодность СУЗВ и упрощает масштабирование системы, однако требует организации надёжного и эффективного межмодульного взаимодействия.

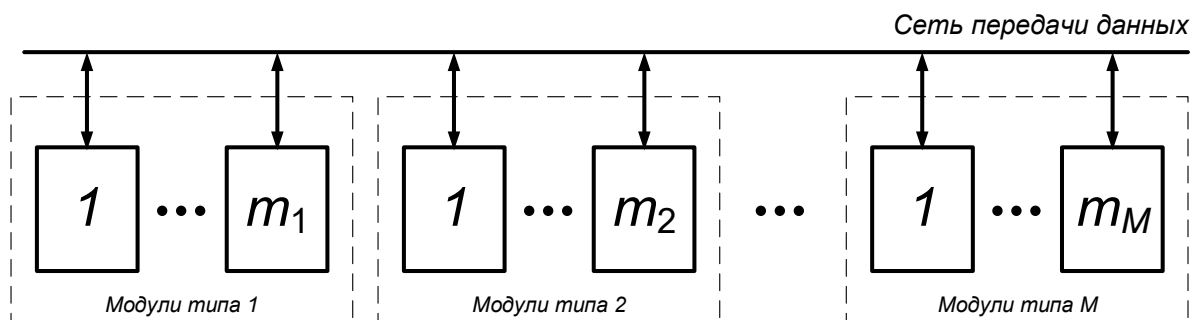


Рисунок 1 – Структура модульной СУЗВ (M – количество типов модулей, m_k ($k = \{1, 2, \dots, M\}$) – количество модулей k -го типа) [The structure of the modular VPCS (M is the number of types of modules, m_k ($k = \{1, 2, \dots, M\}$) – the number of the k -th type modules)]

Очевидно, что интенсивность потока данных, передаваемых модулями по СПД, определяется как количеством модулей, так и параметрами их настройки. С другой стороны, пропускная способность СПД остаётся постоянной и не зависит от конфигурации системы. Следовательно, на самых ранних этапах проектирования СУЗВ для конкретного ТА необходимо получить ответы на следующие вопросы:

- достаточно ли пропускной способности СПД для гарантированной и своевременной передачи высокоприоритетных данных от всех модулей?
- какова средняя скорость передачи низкоприоритетных данных?
- можно ли изменить параметры настройки модулей с целью получения удовлетворительных ответов на первые два вопроса и, если да, то как?

В некоторых источниках предлагаются методы и средства решения описанных задач путём имитационного моделирования коммуникационных процессов в системах на базе протокола *CANopen* [1, 2, 3]. Однако, обеспечивая воспроизведение отдельных реализаций этих процессов с высокой степенью достоверности, имитационные методы не позволяют оперативно оценить всю совокупность возможных реализаций: для этого

необходимо длительное моделирование с последующей статистической обработкой полученных результатов.

В данной статье предлагается вероятностный метод, позволяющий получить ответы на перечисленные вопросы с использованием математического аппарата теории массового обслуживания. Метод разработан для оценки параметров межмодульного взаимодействия в созданной с участием автора СУЗВ [4], использующей в качестве СПД шину *CAN* и протокол обмена данными *CANopen*, однако может применяться и в других модульных системах, работающих на базе этого протокола.

Разработка метода оценки параметров СУЗВ

СУЗВ (рис. 1) можно представить в виде системы массового обслуживания (СМО), схема которой приведена на рисунке 2.

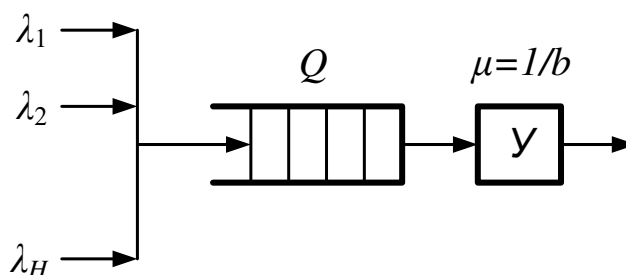


Рисунок 2 – Модель СУЗВ: λ_h – интенсивность h -го потока заявок, H – число классов заявок, Q – накопитель, Y – обрабатывающее устройство, μ – интенсивность обслуживания заявок, b – время обслуживания заявки [VPCS model: λ_h is the intensity of the h -th flow of requests, H – number of requests classes, Q – storage device, Y – processing device, μ is the intensity of requests processing, b is the request processing time]

От модулей СУЗВ в систему поступают потоки заявок на обслуживание (сообщений) с интенсивностями $\lambda_1, \lambda_2, \dots, \lambda_H$, где H – число классов заявок. Каждый модуль может формировать заявки различных классов, причем, чем меньше номер класса, тем выше приоритет заявки.

Заявки поступают в накопитель Q , где ожидают обслуживания. Под накопителем подразумевается некая виртуальная очередь, которую образуют заявки, сформированные модулями СУЗВ и ожидающие передачи в линию связи (в шину *CAN*). Размер накопителя в системе, работающей без перегрузки, можно считать неограниченным.

Обслуживающим устройством Y является линия связи, в которую поступают заявки из очереди Q в соответствии с их приоритетами (классами). Алгоритм функционирования интерфейса *CAN* [5] определяет дисциплину обслуживания заявок с относительными приоритетами, то есть поступление заявки с более высоким приоритетом не прерывает обслуживание текущей заявки. Обслуживание заявки заключается в передаче сообщения по линии связи, поэтому интенсивность обслуживания μ связана с временем передачи сообщения b выражением $\mu = 1 / b$.

Описанная система является СМО с неоднородным потоком заявок. Для таких СМО определяются две группы характеристик:

- характеристики по каждому потоку заявок одного класса;
- характеристики суммарного потока заявок.

С целью определения этих характеристик рассмотрим более подробно взаимодействие модулей СУЗВ по протоколу *CANopen* [6]. В нормальном режиме оно осуществляется в основном с помощью служб *SYNC* (синхронизация), *PDO* (передача / приём текущих данных модулей), *Heartbeat* (проверка состояния модулей), *SDO* (передача / приём служебных данных). Первые три службы функционируют в синхронном режиме, четвёртая – асинхронно. Взаимодействие организовано

следующим образом. В начале каждого цикла синхронизации длительностью T_{SYNC} единственный ведущий модуль отправляет другим модулям сообщение $SYNC$. В ответ на это каждый модуль может выдать в линию одно или несколько сообщений PDO , причём в одном сообщении находятся несколько текущих значений измеряемых или вычисляемых величин. Передача выполняется в каждом B -м цикле синхронизации, а также может произойти досрочно в том случае, если одна из величин, передаваемых в PDO , изменила своё значение по сравнению с отправленным в предыдущем сообщении. Кроме того, один раз за несколько циклов синхронизации каждый модуль выдаёт в линию сообщение $HBEAT$, информирующее систему о наличии и текущем состоянии этого модуля. Наконец, без привязки к циклам синхронизации модули могут обмениваться сообщениями SDO , предназначенными для передачи информации, к времени доставки, которой не предъявляются жёсткие требования. Здесь намеренно не рассматриваются другие службы $CANopen$: NMT , $TIME$, $EMCY$, поскольку в рабочем режиме СУЗВ количество генерируемых ими сообщений пренебрежимо мало по сравнению с описанными выше службами. Тем не менее, при необходимости приведённые далее рассуждения можно применить и к ним. Также не рассмотрена обработка ошибок передачи / приёма, приводящая к отправке дополнительных сообщений, поскольку в правильно спроектированной СПД на базе интерфейса CAN ошибки возникают крайне редко [7].

Каждое сообщение CAN имеет собственный приоритет, определяемый уникальным идентификатором сообщения, однако при исследовании протокола $CANopen$ удобнее работать с классами приоритетов. К первому, наиболее приоритетному классу относится сообщение $SYNC$. В следующие несколько классов [2 ... $H-2$] входят сообщения PDO , причём если модуль формирует несколько разных PDO , то они относятся к различным классам. Разные модули могут формировать PDO одного класса. К классу $H-1$ относятся сообщения $HBEAT$ всех модулей и, наконец, низшим приоритетом H обладают сообщения SDO .

Из приведённого описания следует, что поток сообщений $SYNC$ является детерминированным, его интенсивность определяется выражением:

$$\lambda_1 = 1/T_{SYNC} \cdot \quad (1)$$

Интенсивность потока сообщений $HBEAT$ можно вычислить как:

$$\lambda_{H-1} = \frac{1}{T_{SYNC}} \sum_{k=1}^M \frac{m_k}{B_k}, \quad (2)$$

где M – количество типов модулей в СУЗВ;

m_k – количество модулей k -го типа;

B_k – период передачи сообщений $HBEAT$ модулями k -го типа, выраженный в циклах синхронизации.

Наиболее сложным является поток PDO . Рассмотрим часть этого потока $\lambda_{k,h}$, состоящую из заявок класса h , формируемых одним модулем k -го типа. Как следует из приведённого выше описания, поток можно представить состоящим из двух частей: детерминированной $\lambda_{k,h}^d$ и случайной $\lambda_{k,h}^r$. Интенсивность детерминированной составляющей определяется выражением:

$$\lambda_{k,h}^d = (B_{k,h} T_{SYNC})^{-1},$$

где $B_{k,h}$ – период передачи сообщений класса h модулем k -го типа, выраженный в циклах синхронизации.

Интенсивность случайной составляющей зависит от того, насколько часто возникают заявки, вызванные изменением хотя бы одного из $v_{k,h}$ значений, передаваемых в рассматриваемом PDO . Пусть интервалы времени между изменениями одного такого значения описываются случайной величиной $\xi_{h,k,i}$ ($i = \{1, \dots, v_{k,h}\}$), распределённой по экспоненциальному закону с параметром $\alpha_{k,h,i}$ (это допущение основано на физической природе величин, измеряемых СУЗВ, и впоследствии будет проверено экспериментально). Тогда величина $\xi_{k,h} = \min(\xi_{k,h,i})$, определяющая интервалы времени между заявками, также будет иметь экспоненциальное распределение с параметром:

$$\alpha_{k,h} = \sum_{i=1}^{v_{k,h}} \alpha_{k,h,i} . \quad (3)$$

Вероятность того, что заявка будет сформирована досрочно, то есть вследствие изменения одного из передаваемых значений, определяется выражением:

$$p_{k,h}^r = P(t < B_{k,h} T_{SYNC}) = 1 - e^{-\alpha_{k,h} B_{k,h} T_{SYNC}} ,$$

средняя длительность интервала между досрочными заявками:

$$\overline{\xi_{k,h}} = \frac{1}{p_{k,h}^r} \int_0^{B_{k,h} T_{SYNC}} t \alpha_{k,h} e^{-\alpha_{k,h} t} dt ,$$

интенсивность потока досрочных заявок:

$$\lambda_{k,h}^r = 1 / \overline{\xi_{k,h}} .$$

Тогда общая интенсивность потока заявок, соответствующего передаче PDO класса h одним модулем k -го типа, может быть описана формулой:

$$\lambda_{k,h} = p_{k,h}^r \lambda_{k,h}^r + (1 - p_{k,h}^r) \lambda_{k,h}^d .$$

Подставляя в неё приведённые выше выражения для p^r , λ^r и λ^d , а также учитывая, что с учетом особенностей организации взаимодействия модулей интенсивность потока $\lambda_{k,h}$ не может превышать $1 / T_{SYNC}$, получим:

$$\lambda_{k,h} = \min \left(\frac{1}{T_{SYNC}} , \alpha_{k,h} \frac{\left(1 - e^{-\alpha_{k,h} B_{k,h} T_{SYNC}}\right)^2}{1 - \left(1 + \alpha_{k,h} B_{k,h} T_{SYNC}\right) e^{-\alpha_{k,h} B_{k,h} T_{SYNC}}} + \frac{e^{-\alpha_{k,h} B_{k,h} T_{SYNC}}}{B_{k,h} T_{SYNC}} \right) . \quad (4)$$

Суммарная интенсивность потока заявок класса h ($h \in \{1, \dots, H-1\}$) от всех модулей системы вычисляется как:

$$\lambda_h = \sum_{k=1}^M m_k \lambda_{k,h} , \quad (5)$$

где M – количество типов модулей в СУЗВ;

m_k – количество модулей k -го типа.

Тогда с учётом (1), (2), (5) общая интенсивность потока заявок, формируемых всеми модулями СУЗВ (за исключением заявок SDO класса H), может быть описана так:

$$\lambda_S = \sum_{h=1}^{H-1} \lambda_h, \quad (6)$$

где H – количество классов заявок.

Заявки SDO (класса H) обслуживаются в последнюю очередь, причем с учетом специфики протокола $CANopen$ можно считать, что они формируются лишь в периоды простоя СПД. Вследствие этой особенности в данный момент поток λ_H можно не рассматривать.

После получения выражений, описывающих интенсивности потоков заявок, определим интенсивность обслуживания заявок, то есть интенсивность их передачи по линии связи. Она зависит от времени передачи сообщения, которое определяется скоростью передачи данных по шине CAN , а также форматом и содержанием сообщений. Скорость C шины CAN является величиной постоянной, задаётся в битах в секунду и связана с временем передачи одного бита T_0 соотношением: $T_0 = 1 / C$. Минимальное время передачи сообщения стандартного формата $CAN 2.0a$ [5], как показано автором в [8], можно вычислить с использованием выражения:

$$t^{min} = T_0 (44 + 8 \cdot d),$$

где d – значение поля DLC сообщения (от 0 до 8).

В процессе передачи в сообщение могут быть автоматически вставлены дополнительные биты [5, 8, 9]. В наименее благоприятном случае, когда число автоматически добавленных дополнительных бит максимально, время передачи определяется выражением:

$$t^{max} = T_0 (49 + 8 \cdot d + \lceil 8 \cdot d / 5 \rceil + A),$$

где $A=1$ при $d \in \{0, 1, 3, 8\}$; иначе $A=0$. Аналогичные зависимости приведены в [8] и для сообщений расширенного формата $CAN 2.0b$.

Количество автоматически добавленных бит является случайной величиной, закон распределения которой существенно зависит от специфики передаваемых данных и от их конкретных значений [10, 11], поэтому учесть его в описываемой модели СУЗВ не представляется возможным. Некоторые исследователи [1] вводят специальный параметр модели – константу, задающую количество дополнительных бит в каждом сообщении. Этот подход оправдан в случае постоянной длины сообщений, но не соответствует реальной ситуации при переменной длине. По мнению автора, более корректно будет ввести коэффициент, определяющий степень увеличения времени передачи сообщения, поскольку, чем длиннее сообщение, тем больше дополнительных бит может быть в него добавлено:

$$t^{bs} = t^{min} + \eta (t^{max} - t^{min}), \quad (7)$$

где t^{bs} – время передачи сообщения с учетом дополнительных бит;

$\eta \in [0, 1]$ – коэффициент увеличения времени передачи.

Значения поля DLC могут отличаться у сообщений разных классов, формируемых различными модулями, поэтому введём обозначение $t_{k,h}$, соответствующее времени передачи t^{bs} сообщения класса h , формируемого модулем k -го типа. Время b обслуживания заявки для системы, работающей без перегрузки, можно с учетом (6) получить из выражения:

$$b = \frac{\lambda_1 T_{1,1} + \sum_{k=1}^M \left(m_k \sum_{h=2}^{H-1} \lambda_{k,h} t_{k,h} \right)}{\lambda_S}, \quad (8)$$

Если при вычислении $t_{k,h}$ задать $\eta = 0$, то получим время обслуживания b^{min} в наиболее благоприятном случае, при $\eta = 1$ – время обслуживания b^{max} в наименее благоприятном случае. Интенсивность обслуживания заявок μ вычисляется как $\mu = 1 / b$, а нагрузка системы $y = \lambda_S / \mu$ или $y = \lambda_S b$.

Нагрузка y является наиболее важной характеристикой, определяющей качество функционирования системы. При $y \leq 1$ все поступающие заявки обслуживаются, а система работает без перегрузки. При $y > 1$ возникает перегрузка, вследствие которой часть заявок остаётся необслуженной или, в случае неограниченной ёмкости накопителя, очередь заявок бесконечно растёт. В рассматриваемом случае все заявки класса $h \in [1, H-1]$ должны быть обслужены (то есть все соответствующие сообщения должны быть доставлены), поэтому условие $y \leq 1$ является необходимым для нормального функционирования СУЗВ. Более того, поскольку по шине CAN дополнительно должны передаваться низкоприоритетные сообщения класса H , неравенство следует записать как строгое: $y < 1$.

Далее проанализируем возможности системы по обслуживанию потока низкоприоритетных заявок класса H . Уменьшение интенсивности обслуживания таких заявок и даже их потеря не оказывает влияния на общую работоспособность системы, но увеличивает время выполнения некоторых операций, например, считывания протокола событий или осциллограмм сигналов. Определим коэффициент простоя $\rho = 1 - y$, показывающий в нашем случае, какую часть времени система свободна от обслуживания высокоприоритетных заявок и, следовательно, может обслуживать заявки класса H . Время обслуживания этих заявок b_H можно получить из выражения (7), учитывая, что поле DLC для них всегда равно 8. Интенсивность обслуживания заявок класса H вычисляется как: $\mu_H = \rho / b_H$.

Получим выражение, позволяющее определить среднюю скорость передачи низкоприоритетных данных по известной интенсивности обслуживания μ_H . В рассматриваемой СУЗВ эти данные передаются службой SDO в так называемом блочном режиме. Согласно [6], в этом режиме вначале происходит установление связи путём отправки запроса и получения подтверждения (2 сообщения), затем передаются блоки данных, содержащие до 127 сообщений по 7 байтов в каждом. Получение каждого блока подтверждается сообщением от приёмника, после чего может передаваться следующий блок. По окончании передачи всех блоков выполняется процедура завершения связи (2 сообщения). Таким образом, общее число сообщений K , необходимых для передачи N байтов, можно получить по формуле:

$$K = 4 + \left\lceil \frac{N}{7} \right\rceil + \left\lceil \frac{\lceil N/7 \rceil}{127} \right\rceil.$$

Они будут переданы за время $T_K = K / \mu_H$, при этом скорость передачи в байтах в секунду составит:

$$Q_H = \mu_H N / \left(4 + \left\lceil \frac{N}{7} \right\rceil + \left\lceil \frac{\lceil N/7 \rceil}{127} \right\rceil \right). \quad (9)$$

Разработка модели СУЗВ

Описанный метод оценки параметров был реализован автором в математической модели, предназначенной для исследования различных конфигураций СУЗВ и параметров настройки её модулей. Исходными данными для модели являются:

C – скорость шины CAN, бит/с;

T_{SYNC} – длительность цикла синхронизации, с;

η – коэффициент увеличения времени передачи за счёт автоматической вставки бит;

α – параметр экспоненциального распределения времени между изменениями одного измеряемого или вычисляемого значения, передаваемого в PDO;

H – количество классов заявок;

M – количество типов модулей в СУЗВ;

$M = [m_1 \ m_2 \ \dots \ m_M]^T$ – столбец количества модулей разного типа;

B – матрица периодичности отправки заявок (периодичность задаётся в циклах синхронизации);

D – матрица длин сообщений (содержит значения DLC от 0 до 8 или -1, если модуль не передаёт сообщения данного класса):

$$B = (b_{k,h})_{\substack{k=1,\overline{M} \\ h=1,H-1}}, \quad D = (d_{k,h})_{\substack{k=1,\overline{M} \\ h=1,H-1}};$$

V – матрица, содержащая количество отдельных значений, передаваемых в каждом сообщении PDO.

LM – матрица режима вычисления интенсивностей потоков заявок; содержит «0» для заявок, отправляемых с заданной периодичностью, «1» – для заявок, отправляемых периодически с возможностью досрочной отправки, «2» – для заявок со специальным режимом вычисления интенсивности потока:

$$V = (v_{k,h})_{\substack{k=1,\overline{M} \\ h=1,H-1}}, \quad LM = (lm_{k,h})_{\substack{k=1,\overline{M} \\ h=1,H-1}}.$$

Необходимость использования матрицы LM объясняется тем, что закон распределения интервалов времени между изменениями значений некоторых параметров СУЗВ определить либо затруднительно, либо невозможно, поскольку он существенно зависит от текущего режима работы и степени износа турбоагрегата. Например, если значение какой-либо составляющей виброскорости не превышает 1...2 мм/с, то соответствующий дискретный флаг (признак) превышения предупредительного уровня вибрации 4.5 мм/с не изменяется и всегда равен нулю. С другой стороны, если вибрация достигнет уровня 4.4...4.6 мм/с, то этот же флаг начнёт изменяться достаточно часто за счёт незначительных колебаний вибрации около значения 4.5 мм/с. В СУЗВ имеются и более сложные зависимости. Для таких величин в матрице LM введён специальный режим вычисления «2», в котором интенсивность потока заявок зависит от дополнительного параметра модели $StressMode$. Если этот параметр равен нулю, то используется минимальная интенсивность потока, что в подавляющем большинстве случаев соответствует действительности. Если же задать $StressMode$ равным единице, то интенсивность потока таких заявок принимается максимально возможной, позволяя моделировать предельные режимы работы СПД.

Модель работает следующим образом. На первом этапе формируется матрица параметров экспоненциального распределения интервалов времени между заявками каждого класса от каждого типа модулей путём умножения параметра α на количество отдельных значений, передаваемых в сообщении:

$$A = (\alpha_{k,h}) = \alpha V.$$

При этом для простоты считаем, что α одинаков для всех значений, передаваемых в сообщениях. Если такое допущение некорректно, то для вычисления элементов матрицы A следует воспользоваться выражением (3).

Далее формируется матрица длительностей сообщений:

$$T = (t_{k,h})_{\substack{k=\overline{1,M} \\ h=\overline{1,H-1}}} : t_{k,h} = F_1(d_{k,h}, \eta),$$

где $F_1(d_{k,h}, \eta)$ – функция, определяемая выражением (7).

После этого создается матрица интенсивностей потоков заявок каждого класса, формируемых одним модулем каждого типа:

$$L = (l_{k,h})_{\substack{k=\overline{1,M} \\ h=\overline{1,H-1}}} : l_{k,h} = \begin{cases} 0 & \text{при } b_{k,h} = 0 \\ (T_{SYNC} b_{k,h})^{-1} & \text{при } lm_{k,h} = 0 \\ F_2(T_{SYNC}, b_{k,h}, \alpha_{k,h}) & \text{при } lm_{k,h} = 1 \\ (T_{SYNC} b_{k,h})^{-1} & \text{при } lm_{k,h} = 2 \text{ и } StressMode = 0 \\ T_{SYNC}^{-1} & \text{при } lm_{k,h} = 2 \text{ и } StressMode = 1 \end{cases},$$

где функция $F_2()$ вычисляется в соответствии с выражением (4).

На следующем шаге каждый столбец матрицы L поэлементно умножается на столбец M с целью получения интенсивностей потоков заявок каждого класса, формируемых всеми модулями каждого типа:

$$A = (\lambda_{k,h})_{\substack{k=\overline{1,M} \\ h=\overline{1,H-1}}} = L \circ (E \otimes M),$$

где E – единичный вектор длиной $H-1$.

Общая интенсивность потока заявок вычисляется как сумма элементов матрицы A :

$$\lambda_S = \sum_{k,h} \lambda_{k,h}.$$

Среднее время обслуживания заявки b вычисляется в соответствии с (8):

$$X = (x_{k,h})_{\substack{k=\overline{1,M} \\ h=\overline{1,H-1}}} = T \circ A, \quad b = \lambda_S^{-1} \sum_{k,h} x_{k,h}.$$

Нагрузка системы y , коэффициент простоя ρ и интенсивность обслуживания низкоприоритетных заявок μ_H вычисляются по приведённым ранее формулам: $y = \lambda_S b$,

$\rho = 1 - \mu$, $\mu_H = \rho / b_H$, скорость передачи низкоприоритетных данных Q_H – по формуле (9).

Применение модели для исследования межмодульного взаимодействия

Описанная модель была использована автором для исследования межмодульного взаимодействия в проектируемой СУЗВ с целью формирования рекомендаций по выбору некоторых параметров настройки модулей. В состав рассматриваемой СУЗВ входят модули следующих типов:

- системный контроллер (тип 1) – 1 шт.;
- модуль измерения вибрации опор (тип 2) – 8 шт.;
- модуль измерения относительной вибрации (тип 3) – 4 шт.
- модуль измерения дополнительных параметров (тип 4) – 2 шт.

Системный контроллер с периодичностью $T_{SYNC} = 0.1$ с формирует высокоприоритетные сообщения *SYNC* (заявки класса 1). Остальные модули передают от двух до четырёх сообщений *PDO* (заявки классов 2...5), которые содержат от одного до четырёх отдельных значений. Часть этих значений является измеренными и вычисленными параметрами вибрации, другая часть – дискретными флагами состояния. Передача каждого *PDO* выполняется безусловно один раз в секунду и может произойти досрочно при изменении хотя бы одного из передаваемых значений. Сообщения *HBEAT* (заявки класса 6) формируются каждым модулем раз в секунду. Кроме того, системный контроллер обменивается служебной информацией с другими модулями по протоколу *SDO* (заявки класса 7). Скорость шины *CAN* предварительно выбрана максимально возможной: 1000 кбит/с.

С использованием этой информации были подготовлены следующие исходные данные для модели:

- количество классов заявок $H = 7$;
- количество типов модулей $M = 4$;

– столбец количества модулей разного типа $M = [1 \ 8 \ 4 \ 2]^T$.

– матрицы периодичности отправки заявок B , длин сообщений D , количества отдельных значений в сообщении V и режима вычисления интенсивностей LM :

$$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 10 \\ 0 & 10 & 10 & 10 & 10 & 10 \\ 0 & 10 & 10 & 0 & 10 & 10 \\ 0 & 10 & 10 & 0 & 0 & 10 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & -1 & -1 & -1 & -1 & 1 \\ -1 & 6 & 6 & 7 & 4 & 1 \\ -1 & 8 & 5 & -1 & 4 & 1 \\ -1 & 6 & 7 & -1 & -1 & 1 \end{bmatrix},$$

$$V = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 3 & 4 & 1 & 0 \\ 0 & 4 & 3 & 0 & 1 & 0 \\ 0 & 3 & 4 & 0 & 0 & 0 \end{bmatrix}, \quad LM = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 0 & 1 & 0 \\ 0 & 1 & 2 & 0 & 0 & 0 \end{bmatrix}.$$

При формировании матрицы D дополнительно использовалась информация о размере передаваемых значений в байтах и о разметке *PDO* [6], описание которой выходит за рамки этой статьи.

Значение коэффициента увеличения времени передачи за счёт автоматической вставки бит было выбрано на основании сведений, приведённых в [10, 12]. В результате анализа множества сообщений в реальной системе на базе интерфейса *CAN* было установлено, что при максимальной длине поля данных к сообщению добавляется в среднем 7.89 дополнительных бит из восемнадцати возможных. С учетом этих сведений значение η было принято равным 0.438.

При выборе параметра α , характеризующего вероятность изменения измеряемых и вычисляемых значений в течение некоторого времени, было принято во внимание, что результат аналого-цифрового преобразования некоторого сигнала может меняться как минимум на единицу младшего разряда при каждом измерении, вследствие чего окажется неэффективным механизм увеличения интервалов между сообщениями при отсутствии изменений. Для решения проблемы была введена зона нечувствительности δ , а изменение фиксировалось лишь при выполнении условия $|x_0 - x_i| > \delta$, где x_0 – последнее переданное значение, x_i – текущий результат измерения (вычисления).

Для определения α автором обработаны более 50 графиков изменения вибрации опор и шеек роторов длиной от нескольких часов до нескольких суток с целью определения вероятности изменения сигнала более, чем на заданную величину, за некоторое время. Полученные экспериментально функции распределения для нескольких сигналов при $\delta = 0.5\%$, а также результат их аппроксимации функцией экспоненциального распределения (сплошная линия) показаны на рисунке 3а.

Очевидно, что частота регистрации изменений уровня сигнала зависит от ширины зоны нечувствительности, поэтому аналогичные исследования были выполнены и для других значений δ в диапазоне $0.1...2.5\%$; для каждого случая определялся усреднённый параметр α . Полученная экспериментально зависимость α от ширины зоны нечувствительности δ приведена на рисунке 3б. Для удобства дальнейшего использования она была аппроксимирована функцией (10):

$$\alpha = 3.123 \cdot e^{-2.12\delta} + 0.0714. \quad (10)$$

Таким образом, в модель СУЗВ добавился ещё один параметр, влияющий на случайную составляющую потока сообщений *PDO*: ширина зоны нечувствительности δ .

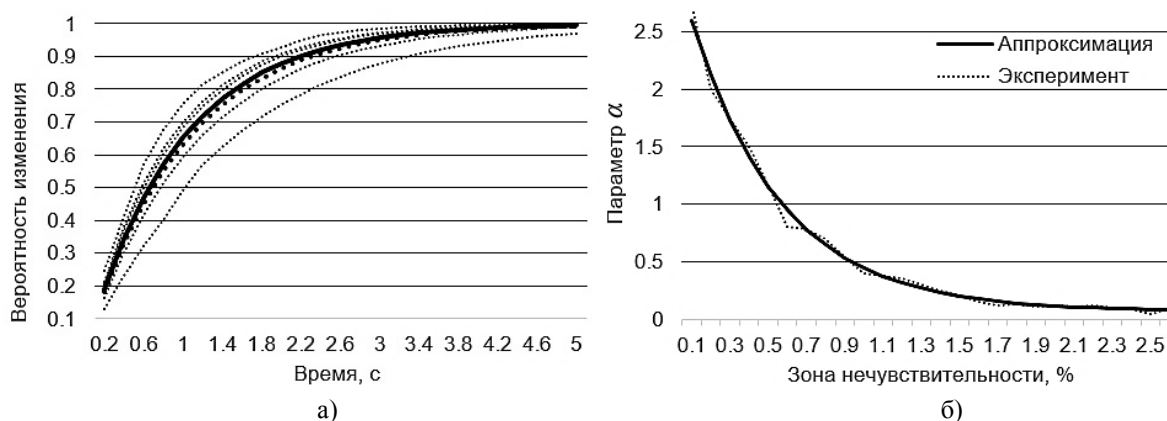


Рисунок 3 – Распределение вероятности изменения сигнала (а) и зависимость усреднённого параметра экспоненциального распределения от ширины зоны нечувствительности (б) [Probability distribution of the signal changes (а) and the dependence of the averaged parameter of the exponential distribution on the width of the dead zone (б)]

На первом этапе исследования СУЗВ была получена зависимость нагрузки СПД от ширины зоны нечувствительности (рис. 4 а). График обычной нагрузки соответствует режиму *StressMode* = 0, когда значения дискретных флагов изменяются сравнительно редко и не приводят к досрочной передаче соответствующих сообщений. График максимальной нагрузки получен в режиме *StressMode* = 1, то есть исходя из предположения, что изменения дискретных флагов происходят в каждом цикле синхронизации.

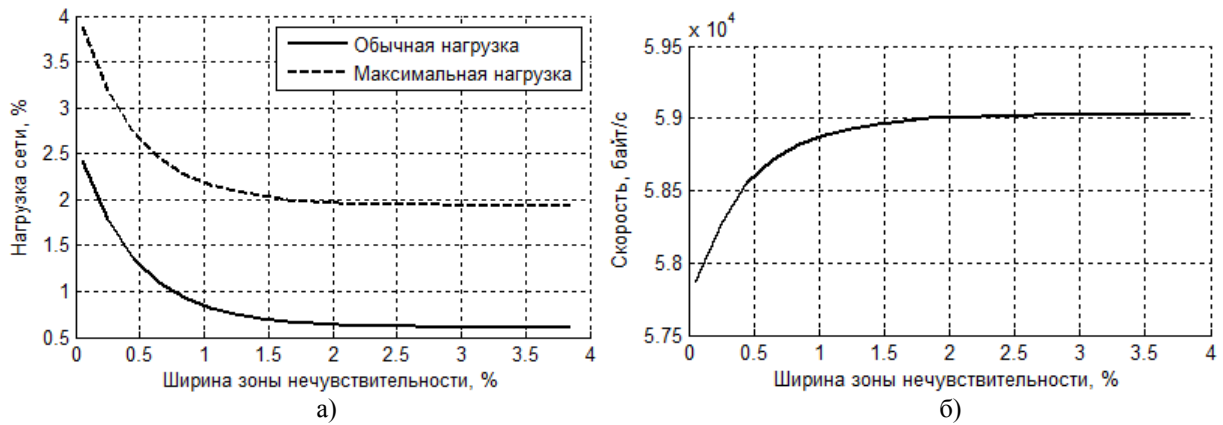


Рисунок 4 – Зависимость параметров СПД от ширины зоны нечувствительности δ : а – нагрузка y ; б – скорость передачи низкоприоритетных данных Q_H [Dependence of the data transmission network parameters on the dead zone width δ : а – load y ; б – low-priority data transmission rate Q_H]

Из графика видно, что при $\delta > 2\%$ влияние этого параметра на нагрузку сети практически отсутствует, а при $\delta < 1\%$ – наоборот, весьма заметно. Это согласуется с результатами наблюдений за параметрами вибрации ТА: изменения более чем на 2...5% обычно возникают при изменении нагрузки агрегата, что происходит сравнительно редко. С другой стороны, колебания значений на уровне менее 0.5...1% могут быть обусловлены шумами, погрешностями измерений и вычислений, поэтому выявляются значительно чаще. При дальнейшем уменьшении δ возрастание нагрузки, согласно (4), должно прекратиться, но значения $\delta < 0.1\%$ находятся на уровне погрешности измерения и практического интереса не представляют, вследствие чего исследования в этой области не выполнялись.

Кроме описанных зависимостей была вычислена предельно возможная нагрузка СПД y_{max} , возникающая в ситуации, когда все сообщения каждого модуля в силу стечения обстоятельств передаются в одном цикле синхронизации. Эта ситуация встречается крайне редко, но выполнение условия $y_{max} < 100\%$ гарантирует, что все приоритетные сообщения в любом случае будут переданы своевременно, то есть в пределах одного периода синхронизации T_{SYNC} . Для заданных параметров СУЗВ величина y_{max} оказалась равной 5.5%.

На рисунке 4б показана полученная в результате моделирования зависимость скорости передачи низкоприоритетных данных Q_H от заданной ширины зоны нечувствительности в режиме обычной нагрузки. В качестве примера были взяты наиболее часто передаваемые данные – осциллограммы сигналов виброскорости и виброускорения. Они имеют длину 4000 байт каждая и, как видно из графика, при $\delta = 1\%$ передаются на скорости около 58.8 кбайт/с. При этой скорости одна осциллограмма будет передана примерно за 68 мс, а 24 осциллограммы с восьми контролируемых опор – за 1.6 с.

Анализ результатов моделирования показал, что оптимальным значением ширины зоны нечувствительности δ можно считать 1%: в этом случае при незначительном увеличении нагрузки на СПД (по сравнению с минимальной) обеспечивается оперативное информирование модулей системы о каких-либо значимых изменениях вибрации. Увеличение δ не даёт существенного снижения нагрузки, но уже заметно снижает чувствительность СУЗВ к изменениям вибрации, а уменьшение δ в разы повышает нагрузку на СПД без какой-либо практической пользы.

Второй вывод, сделанный по результатам моделирования, заключается в том, что при заданных параметрах системы и $\delta = 1\%$ коэффициент простоя СПД превышает 99%. Это положительно сказывается на скорости передачи низкоприоритетных данных, но такие данные передаются очень редко, в основном по запросу оператора

или при возникновении нештатных ситуаций. С другой стороны, работа шины CAN на максимальной скорости снижает помехоустойчивость интерфейса и повышает тепловыделение модулей СУЗВ. Поэтому на следующем этапе исследований были получены и исследованы зависимости показателей СПД от скорости передачи данных по шине CAN при $\delta = 1\%$ (рис. 5).

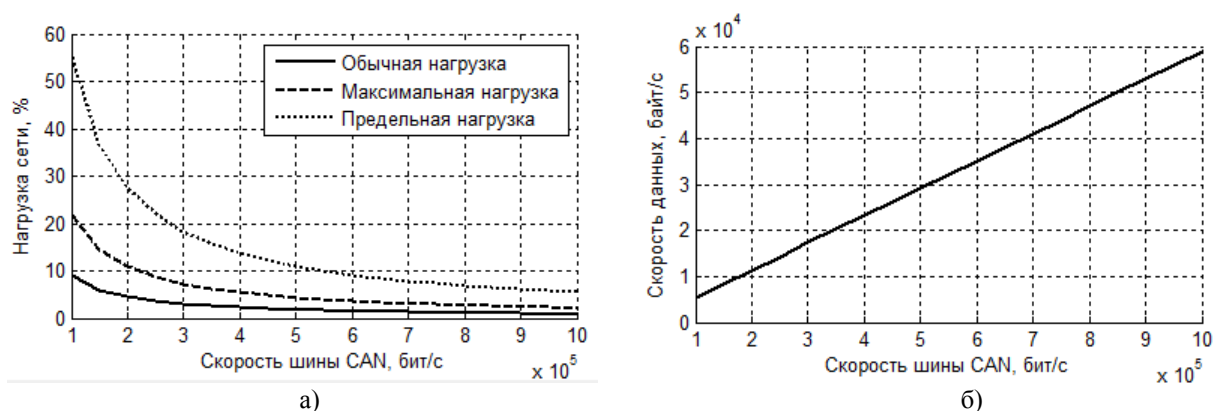


Рисунок 5 – Зависимость параметров СПД от скорости шины CAN: а – нагрузка; б – скорость передачи низкоприоритетных данных) [Dependence of the data transmission network parameters on the CAN bus speed: а – load y ; б – low-priority data transmission rate Q_H]

На рисунке 5а видно, что даже десятикратное снижение скорости шины CAN (до 100 000 бит/с) не приводит к возникновению перегрузки СПД при передаче приоритетных сообщений. Более того, предельная нагрузка в этом случае не превышает 60 %, что, безусловно, гарантирует своевременную доставку всех приоритетных сообщений. С другой стороны, скорость передачи низкоприоритетных данных (рис. 5 б) практически линейно зависит от скорости шины и при её минимальном значении уменьшается с 58.8 до 5.4 кбайт/с. Такое снижение скорости несущественно при автоматической передаче накопленных данных с целью архивирования, поскольку модули имеют буферную память достаточного объёма, но создаёт неудобства при получении данных по команде оператора: в этом случае, например, окончания передачи всех осциллограмм после запроса придётся ждать уже не 1.6, а почти 18 с. Исходя из этих рассуждений скорость шины CAN для исследуемой конфигурации СУЗВ была понижена в два раза до 500 кбит/с, что позволило так же в 2 раза уменьшить тактовую частоту приёмопередатчиков CAN во всех модулях СУЗВ, снизив за счёт этого их тепловыделение при сохранении приемлемой скорости передачи низкоприоритетных данных.

Сравнение результатов моделирования с данными, полученными на действующем макете проектируемой СУЗВ описанной конфигурации, показало их хорошее совпадение: реальная скорость передачи низкоприоритетных данных в нескольких опытах находилась в диапазоне 25...30 кбит/с. Некоторое уменьшение средней скорости по сравнению с вычисленной (29 кбит/с) можно объяснить дополнительными задержками, связанными с обработкой данных системным контроллером и с передачей их на рабочую станцию оператора.

Заключение

Описанный метод и реализованная с его использованием математическая модель позволяют оценить параметры межмодульного взаимодействия в СУЗВ ТА на самых ранних этапах проектирования, исключая возможность возникновения грубых ошибок, связанных с перегрузкой сети передачи данных, а также выполнить более тонкую настройку параметров взаимодействия с учётом количества модулей разных типов в конкретной конфигурации СУЗВ. Модель может быть использована как при адаптации СУЗВ к определённому типу ТА, так и при модернизации программного обеспечения

СУЗВ – для оптимизации количества передаваемых модулями параметров, распределения их по сообщениям, выбора длительности цикла синхронизации, частоты передачи сообщений, а также для решения иных коммуникационных задач. Планируемая интеграция модели в разработанный с участием автора программный комплекс имитационного моделирования СУЗВ [3] позволит существенно упростить подготовку исходных данных для неё за счёт автоматического извлечения этих данных из EDS-файлов описания конкретной СУЗВ. Предложенный метод является универсальным и позволяет при необходимости моделировать работу других модульных систем на базе протокола *CANopen*, а после модификации метода оценки времени передачи сообщений [8] – и более быстрого протокола *CANopen FD*.

СПИСОК ЛИТЕРАТУРЫ

1. *Lekidis Alexios, Bozga Marius, Bensalem Saddek*. Rigorous Modeling and Validation of CANopen Systems. 2014. doi: 10.13140/2.1.1742.1129.
2. *N. Sußmann, A. Meroth*. Model based development and verification of CANopen components 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2017, pp. 1-5, doi: 10.1109/ETFA.2017.8247745.
3. *Плотников, Д.А.* Разработка модели межмодульного взаимодействия в системах контроля вибрации на базе протокола CANopen / Д.А. Плотников, В.И. Лачин, В.К.М. Алджиязна, К.Ю. Соломенцев // Известия вузов. Электромеханика. – 2020. – Т. 63, № 2-3. – С. 68-75. – DOI: 10.17213/0136-3360-2020-2-3-68-75.
4. *Плотников, Д.А.* Цифровые технологии в системах вибромониторинга турбоагрегатов / Д.А. Плотников, В.И. Лачин, В.К.М. Алджиязна // Безопасность ядерной энергетики: тезисы докладов XVII Международной научно-практической конференции, 26-28 мая 2021 г. / НИЯУ МИФИ [и др.]. – Волгоград : ВИТИ НИЯУ МИФИ, 2021. – С. 16-19.
5. CAN Specification. Version 2.0 / ROBERT BOSCH GmbH, 1991. – 72 p.
6. CiA 301. CANopen application layer and communication profile. Version: 4.2.0. © CAN in Automation (CiA) 2011. URL: <https://www.can-cia.org> (дата обращения: 23.06.2021)
7. *K. Iehira, H. Inoue, K. Ishida*. Spoofing attack using bus-off attacks against a specific ECU of the CAN bus 2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC), 2018, pp. 1-4, doi: 10.1109/CCNC.2018.8319180.
8. *Плотников, Д.А.* Оценка времени отклика элементов в модульных информационно-измерительных и управляющих системах, использующих интерфейс CAN / Д.А. Плотников // Известия вузов. Сев.-Кавк. регион. Техн. науки. – 2017. – № 1. – С. 13-18. DOI: 10.17213/0321-2653-2017-1-13-18.
9. *Medhat Maha*. Bit stuffing techniques Analysis and a Novel bit stuffing algorithm for Controller Area Network (CAN). International Journal of Computer System (ISSN: 2394-1065), Volume 02– Issue 03, March, 2015
10. *Alaei Reza, Moallem Payman, Bohlooli Ali*. Statistical based algorithm for reducing bit stuffing in the Controller Area Networks. Microelectronics Journal. 101. 104794. – 2020. doi: 10.1016/j.mejo.2020.104794.
11. *Liu Yu, Yin Yue*. Statistical Analysis of Improved Bit-stuffing Method Based on CAN Bus. DEStech Transactions on Computer Science and Engineering. – 2018. doi: 10.12783/dtsc/ceic2018/24562..
12. *Nolte Thomas, Hansson Hans, Norström Christer, Punnekkat Sasikumar*. Using bit-stuffing distributions in CAN analysis. IEEE Real-Time Embedded Systems Workshop. – 2009. URL: https://www.researchgate.net/publication/255672829_Using_bit-stuffing_distributions_in_CAN_analysis (дата обращения: 15.06.2021).

REFERENCES

- [1] Lekidis Alexios, Bozga Marius, Bensalem Saddek. Rigorous Modeling and Validation of CANopen Systems. 2014. doi: 10.13140/2.1.1742.1129 (in English).
- [2] N. Sußmann, A. Meroth. Model Based Development and Verification of CANopen Components 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). 2017. P. 1-5, doi: 10.1109/ETFA.2017.8247745 (in English).
- [3] Plotnikov D.A., Lachin V.I., Aldzhiyazna V.K.M., Solomencev K.Yu. Razrabotka modeli mezhmodul'nogo vzaimodejstviya v sistemah kontrolya vibracii na baze protokola CANopen [Development of a Model of Intermodular Interaction in Vibration Control Systems Based on the

- CANopen Protocol]. *Izvestiya vuzov. Elektromekhanika [University News. Electromechanics]*. 2020. Vol.63. №2-3. P.68-75. DOI: 10.17213/0136-3360-2020-2-3-68-75 (in Russian).
- [4] Plotnikov D.A., Lachin V.I., Aldzhiazna V.K.M. Cifrovye tekhnologii v sistemah vibromonitoringa turboagregatov [Digital Technologies in Vibration Monitoring Systems of Turbine Units]. *Bezopasnost' yadernoj energetiki: tezisy dokladov XVII Mezhdunarodnoj nauchno-prakticheskoy konferencii, 26-28 maya 2021 g. / NIYAU MIFI [i dr.] [Safety of Nuclear Energy: Abstracts of the XVII International Scientific and Practical Conference, May 26-28, 2021 / NRNU MEPhI [and others]. Volgodonsk: VITI NIYAU MIFI [Volgodonsk: VITI NRNU MEPhI]*. 2021. P.16-19 (in Russian).
- [5] CAN Specification. Version 2.0 / ROBERT BOSCH GmbH, 1991. 72 p. (in English).
- [6] CiA 301. CANopen application layer and communication profile. Version: 4.2.0. © CAN in Automation (CiA) 2011. URL: <https://www.can-cia.org> (accessed: 23.06.2021) (in English).
- [7] K. Iehira, H. Inoue, K. Ishida. Spoofing attack using bus-off attacks against a specific ECU of the CAN bus 2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC). 2018. P.1-4. DOI: 10.1109/CCNC.2018.8319180 (in English).
- [8] Plotnikov D.A. Ocenka vremeni otklika elementov v modul'nyh informacionno-izmeritel'nyh i upravlyayushchih sistemah, ispol'zuyushchih interfejs CAN [Evaluation of the Response Time of Elements in Modular Information-Measuring and Control Systems Using the CAN Interface]. *Izvestiya vuzov. Sev.-Kavk. region. Tekhn. Nauki [University News. North Caucasus. region. Tech. science]*. 2017. №1. P.13-18. DOI: 10.17213/0321-2653-2017-1-13-18 (in Russian).
- [9] Medhat Maha. Bit Stuffing Techniques Analysis and a Novel Bit Stuffing Algorithm for Controller Area Network (CAN). *International Journal of Computer System (ISSN: 2394-1065), Volume 02. Issue 03. March, 2015* (in English).
- [10] Alaei Reza, Moallem Payman, Bohlooli Ali. Statistical Based Algorithm for Reducing Bit Stuffing in the Controller Area Networks. *Microelectronics Journal*. 101. 104794. 2020. DOI: 10.1016/j.mejo.2020.104794 (in English).
- [11] Liu Yu, Yin Yue. Statistical Analysis of Improved Bit-stuffing Method Based on CAN Bus. *DEStech Transactions on Computer Science and Engineering*. 2018. DOI: 10.12783/dtcse/ceic2018/24562 (in English).
- [12] Nolte Thomas, Hansson Hans, Norström Christer, Punnekkat Sasikumar. Using Bit-Stuffing Distributions in CAN Analysis. *IEEE Real-Time Embedded Systems Workshop. – 2009*. URL: https://www.researchgate.net/publication/255672829_Using_bit-stuffing_distributions_in_CAN_analysis (accessed 15.06.2021) (in English).

Probabilistic Method and Model for the Intermodular Interaction Parameters Estimating in Vibration Protection Control Systems of Turbine Units

D.A. Plotnikov

*Platov South-Russian State Polytechnic University (NPI), Prosveshcheniya St., 101,
Novocherkassk, Rostov region, Russia 346428
ORCID iD: 0000-0001-7551-1364
WoS Researcher ID: B-1248-2018
e-mail: dpl68@mail.ru*

Abstract – The paper briefly describes the structure of a turbine unit's modular vibration protection control system (VPCS), formulates the problems of estimating the intermodular interaction parameters that arise at the early stages of VPCS design. Based on the algorithm analysis for the operation of various CANopen protocol services, a probabilistic method is proposed for determining the main communication processes parameters in the VPCS using the queuing theory elements. The mathematical model of the VPCS using the proposed method is considered, its input parameters, the functioning algorithm and output data are described. A technique for studying a real VPCS using a model is described, the results of modeling are presented, their analysis is carried out, based on which methods for optimizing some parameters of intermodular interaction are proposed.

Keywords: vibration protection, vibration monitoring, modular system, intermodular interaction, queuing system, CANopen modeling, CAN bus.