

<sup>1</sup>Канд. техн. наук, старший преподаватель кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета, г. Пермь, Россия

<sup>2</sup>Старший преподаватель кафедры «Микропроцессорные средства автоматизации» Пермского национального исследовательского политехнического университета, г. Пермь, Россия

## МОДЕЛЬ ЦИКЛИЧЕСКОГО АЛГОРИТМА ПЛАНИРОВАНИЯ ЗАДАЧ В РАСПРЕДЕЛЕННОЙ СИСТЕМЕ РЕАЛЬНОГО ВРЕМЕНИ

**Актуальность.** Решена задача количественной оценки вероятностных и временных характеристик подсистемы планирования задач узлов для проектирования распределенных систем реального времени. Объектом исследования являлся циклический алгоритм планирования задач на основе приоритетов (round robin priority driven preemptive scheduler), используемый в распределенных fieldbus-сетях.

**Цель работы** – построение модели и разработка способа количественной оценки вероятностных и временных характеристик подсистемы планирования задач узла: задержки и интервалов времени решения задач, интенсивности информационной нагрузки на каналы промышленной fieldbus-сети.

**Метод.** Для решения задачи использован аппарат теории вероятностей. Предложен граф состояний и переходов модели описывающий функционирование циклического алгоритма планирования задач узла. Получены аналитические соотношения для оценки основных вероятностных и временных характеристик модели подсистемы планирования задач: среднего времени задержки обработки задачи, среднего интервала времени между решением задач, интенсивности информационной нагрузки на каналы fieldbus-сети. Предложенные модель и способ оценки отличаются от известных детализацией ранее не анализируемыми в совокупности значимыми параметрами функционирования алгоритма: разнородными уровнями приоритетов, количеством задач по приоритетам, конфигурационными свойствами задач, задержками операций и частотой наступления событий планировщика, процедурами прерывания и дообслуживания задач, и другими параметрами.

**Результаты.** Разработанная модель и способ количественной оценки вероятностных и временных характеристик подсистемы планирования задач с циклическим алгоритмом обработки на основе приоритетов.

**Выводы.** Проведенные эксперименты подтвердили работоспособность и адекватность предложенного математического обеспечения и позволяют рекомендовать его для использования на практике при решении задач проектирования распределенных систем реального времени с заданными временными характеристиками. В ходе проектирования инструментарий будет использован для определения: допустимого числа программных модулей планировщика и распределения среди них приоритетных и неприоритетных задач обрабатываемых узлом в зависимости от частоты технологических событий и заданных требований по задержке обработки.

**Ключевые слова:** планирование задач, реальное время, циклический алгоритм, приоритет-ориентированный алгоритм, модель, вероятностные и временные характеристики, распределенная система, промышленная сеть, fieldbus.

### НОМЕНКЛАТУРА

ТЗКС – точка завершения критической секции (точка входа);

ЦНПЗ – цикл неприоритетных задач;

ЦПЗ – цикл приоритетных задач;

$a$  – приоритет модуля планировщика задач;

$b_i$  – индикатор формирования модулем  $i$  информации для передачи по сети;

$c_i$  – среднее количество подряд следующих циклов обработки задачи  $i$  планировщика;

$d_i$  – интервал времени между проверкой текущего и предшествующего модулей;

$i$  – текущий номер модуля планировщика;

$j$  – номер анализируемого модуля планировщика;

$N$  – количество модулей планировщика задач;

$n$  – количество приоритетных модулей планировщика задач;

$m$  – количество неприоритетных модулей планировщика задач;

$p_i$  – вероятность наступления технологического события для модуля  $i$  планировщика;

$T_n$  – время проверки технологического условия;

$T_3$  – время обработки задачи планировщика;

$T_b$  – время обработки информации в точке входа;

$T_{ci}$  – средний интервал времени между событиями модуля  $i$  планировщика задач;

$T_i$  – среднее полное время обработки задачи модуля  $i$  планировщика;

$t_i$  – средний интервал времени между проверками модуля с номером  $i$ ;

$t_i^y$  – средний интервал времени между успешными проверками модуля  $i$ ;

$t_n^H$  – средняя продолжительность пустого (неуспешного) цикла приоритетных задач;

$t_{zi}$  – среднее время завершения цикла после проверки модуля  $i$ ;

$t_{ni}$  – среднее время с начала цикла (точки входа) и до начала обработки модуля  $i$ ;

$u_i$  – порядок закона распределения Эрланга интервала времени проверки модуля  $i$ ;

$y$  – начальный порядок закона распределения Эрланга времени между проверками модулей;

$\lambda_\Sigma$  – суммарная интенсивность нагрузки узла на канал передачи сети.

### ВВЕДЕНИЕ

Циклический алгоритм (Round-robin) – один из наиболее распространенных алгоритмов, используемых

программными планировщиками задач в вычислениях реального времени. Широкое распространение данный алгоритм получил благодаря детерминированности интервалов времени решения задач в системах мягкого и жесткого реального времени. Различные модификации алгоритма применяются при пакетном планировании задач передачи данных в компьютерных и сенсорных сетях, для сбора, обработки и передачи технологической (измерительной и управляющей) информации в распределенных системах управления (Distributed control system), в том числе построенных на основе сетевых fieldbus-технологий. Соответственно применяются алгоритмы: Round Robin, Weighted Round Robin, Deficit Round Robin and Modified Deficit Round-Robin [1, 2]; Priority-based, Rate Monotonic Analysis, Earliest Deadline First [3–6]. Область применения алгоритмов определена эффективностью обработки планируемых задач в зависимости от их количества, приоритетности, периодичности/спорадичности [5, 6], вычислительной сложности, продолжительности и других параметров. Количественная оценка эффективности алгоритма производится по характеристикам детерминированности временных интервалов и времени реакции на их выполнение, эффективности использования времени. Для анализируемого в работе алгоритма планирования на основе приоритетов (priority-based), применяемого в распределенных системах с событийно-ориентированным (event-driven) принципом обработки задач и передачи информации узлами (контроллерами) fieldbus-сети (LonWorks, BacNet, CAN, EIB/KNX и др.), важнейшей характеристикой является время реакции и решения планируемой задачи. Данное время представляет собой интервал между спорадическим моментом необходимости решения задачи (называемым в работе событием) и ее фактическим решением. Возможное запаздывание в решении задачи негативно влияет на показатели качества управления (время переходного процесса, перерегулирование) техническими объектами и технологическими процессами, отражается на качестве реализуемой продукции. Таким образом, необходимым является проектирование планировщика задач узла с временными характеристиками не хуже требуемых. Это актуализирует решение задачи построения адекватной модели циклического алгоритма планирования задач и оценки ее вероятностных и временных характеристик.

### 1 ПОСТАНОВКА ЗАДАЧИ

В работе производится анализ и построение адекватной аналитической модели алгоритма планирования задач [7, 8], управляемого приоритетом (priority-driven preemptive scheduler), применяемого на узлах распределенных (на основе fieldbus) систем управления, и разработка способа количественной оценки времени решения задач. Решение требует принятия во внимание множества значимых параметров и особенностей функционирования планировщика узла систем, ранее не анализируемых в совокупности в известной литературе: разнородные уровни приоритетов, количество задач по приоритетам, конфигурационные свойства задач, задержки на реализацию элементарных операций пла-

нировщика, включая проверки и задачи, частота (закон, случайные интервалы времени) наступления событий планировщика, прерывание и дообслуживание задач, и другие параметры [7, 8].

### 2 ОБЗОР ЛИТЕРАТУРЫ

Проектированию и оценке информационных характеристик распределенных систем управления на основе fieldbus-сетей посвящено большое число публикаций, в том числе распространенной промышленной сети LonWorks [7–15]. В значительной части публикаций выполняется оценка транспортной задержки, как основной характеристики системы [9–11, 13, 15]. Это обусловлено применением в анализируемых fieldbus-сетях случайного метода множественного доступа с контролем несущей (CSMA). Примерами таких fieldbus сетей являются сети LonWorks, BacNet с протоколами доступа predictive p-persistent CSMA, EtherNet/IP, Modbus/TCP с протоколами CSMA/CD, Can, Knx/Eib и DeviceNet с протоколами CSMA/CA (NDA) [7, 8]. При этом известна и высокая значимость задержки планировщика задач узла сети, вносимой обработкой измерительной и управляющей информации, которая часто превышает транспортное запаздывание [7, 8, 12, 14]. Не смотря на отмеченное, анализу данной тематики посвящено значимо меньшее число публикаций. Как будет показано, это обусловлено многопараметричностью создаваемой модели и сложностью ее количественного расчета. Данная трудноразрешимая задача решается учеными по разному.

В работах [7, 12] выполнен анализ и систематизированы факторы работы планировщика, потенциально влияющие на задержку обработки информации. В работе [12] проведен натурный эксперимент и получены типовые значения задержек и законы их распределения для различных элементарных операций. Полученные результаты подтверждают и расширяют результаты полученные в ходе испытаний технологии, опубликованные в [7, 8]. В [7] также предложены аналитические соотношения для приближенной оценки задержки вносимой планировщиком задач в обработку информации. Недостаток предложенных формул заключается в отсутствии учета значимых параметров (см. постановку задачи) функционирования планировщика узла системы, что нарушает адекватность оценок. В [14] предложена имитационная модель планировщика задач в системе моделирования AnyLogic. В модели учтены основные значимые факторы функционирования и выполнена адекватная оценка задержки. Недостаток модели заключается в высоких временных затратах получения статистически корректных оценок. Таким образом, малая проработанность темы и проблема адекватности, высокие временные затраты на оценку, требуют разработки новой аналитической модели функционирования алгоритма планирования задач и способа оценки его вероятностных и временных характеристик.

### 3 МАТЕРИАЛЫ И МЕТОДЫ

Циклический алгоритм планирования задач определяет последовательность обработки собранной узлом измерительной и поступившей из сети управляющей информации. На каждом такте цикла выполняется рабо-

та с программним модулем планировщика, состоящим из проверки «when» условия события (изменения технологической информации) и решения задачи «task» обработки данной информации (рис. 1).

Управление планированием осуществляется на основе приоритета и номера задачи. Выделяют два уровня приоритета: приоритетный и неперитетный. Цикл решения задач состоит из цикла приоритетных задач (ЦПЗ), в рамках которого происходит последовательная обработка приоритетных модулей, и цикла неперитетных задач (ЦНПЗ), который наступает только после завершения ЦПЗ и характеризуется обработкой только одного модуля планировщика (в порядке следования) (рис. 1). При успешной проверке «when» события модуля выполняется решение соответствующей задачи и переход в точку завершения критической секции (ТЗКС, точку входа), после чего начинается новый ЦПЗ.

Модель алгоритма планирования может быть представлена в виде графа (рис. 2), основными элементами которого являются: вершины – состояния входа «в», проверки «п», задачи «з» (сокращения используются в индексах параметров), характеризуемые соответствующими задержками  $T$  обработки информации; ребра – вероятности  $p$  переходов между состояниями модели.

Предложенный граф модели позволяет учитывать следующие параметры планировщика. Основные параметры планировщика: количество модулей  $N$ , включая  $n$  – приоритетных,  $m$  – неперитетных; последовательность и приоритеты модулей  $a \in \{0,1\}$ . Индивидуальные параметры модулей: тип модуля (входящий, исходящий, комбинированный), конфигурационные свойства, функция изменения контролируемой величины [7, 8].

Количественная оценка характеристик алгоритма планирования требует выполнения стохастического анализа модели с использованием математического аппарата теории вероятностей. Для упрощения решаемой задачи в работе вначале будет выполнен анализ модели цикла приоритетных задач алгоритма ( $n=N, m=0$ ) (рис. 3).

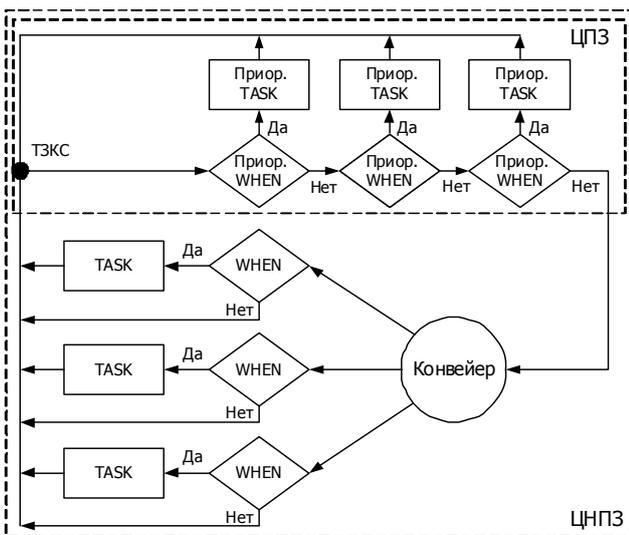


Рисунок 1 – Схема циклического алгоритма планирования задач

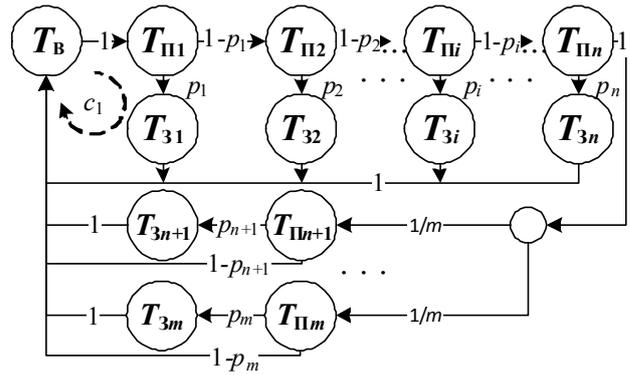


Рисунок 2 – Модель алгоритма

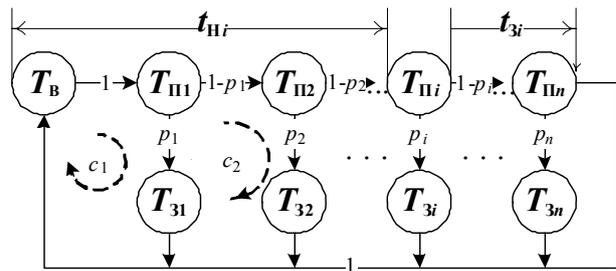


Рисунок 3 – Модель цикла приоритетных задач алгоритма

Время реакции планировщика на произошедшее (потенциальное) в системе событие изменяется в широких диапазонах. Это связано с спорадическими моментами наступления событий в системе. Поэтому время до начала обработки модуля может изменяться от 0, когда событие наступает в момент начала обработки проверки модуля, и до значения, когда событие наступает в момент завершения обработки модуля планировщика. Предполагая статистическую независимость временных интервалов работы планировщика задач от случайных моментов событий, среднее время  $T_i$  обработки задачи  $i \in \{1..N\}$  (рис. 3):

$$T_i = [t_{zi} + t_{ni} + (T_{ni} + T_{zi} + T_в)]/2 = [t_i + (T_{ni} + T_{zi} + T_в)]/2, \quad (1)$$

где  $t_{zi}$  – время завершения текущего цикла приоритетных задач,  $t_{ni}$  – время от начала цикла до начала  $i$  проверки,  $t_i$  – средний интервал времени между проверками модуля  $i$ .

Типовое время задержки  $T_{п}$ ,  $T_z$ ,  $T_в$  состояний модели [7, 12] примем распределенным по экспоненциальному закону. Это позволит получить оценку наивысшей средней задержки решения задач. Для упрощения аналитических выкладок и уменьшения количества исходных данных в модели примем равными для всех модулей задержки обработки информации в точке входа, проверки и задачи, а их сумму, определяющую полное время обработки одного модуля, обозначим  $T_{пзв}$ . Выполним оценку неизвестных в формуле (1) величин.

Среднее время завершения цикла  $t_{zi}$  после проверки модуля  $i$ , безотносительно к результату проверки, можно определить суммой времени завершения цикла на текущем и на каждом последующем модуле  $j = i + 1..n$  с уче-

том вероятности наступления события  $p_j$  и при условии, что все предшествующие модулю  $j$  проверки неуспешны:

$$t_{zi} = \begin{cases} T_3 \cdot p_i & \text{при } i = n, \\ T_3 \cdot p_i + \sum_{j=i+1}^n \left[ \prod_{k=i}^{j-1} (1-p_k) [T_{\text{п}} + p_j \cdot T_3] \right] & \text{при } i < n. \end{cases} \quad (2)$$

Вероятность  $p_i$  успешного результата проверки (наступления события) модуля  $i$  определяется временем  $t_i$ , прошедшим с момента последней проверки модуля, и средним временем  $T_i$  между событиями этого модуля. Учитывая экспоненциальный характер интервалов времени между событиями, вероятность:

$$p_i = 1 - e^{-t_i/T_i}. \quad (3)$$

Среднее время  $t_{\text{н}i}$  с начала цикла (точки входа) и до начала обработки модуля  $i$  представляет собой сумму задержек: точки входа, проверки  $(i-1)$  предшествующих модулей, решения промежуточных задач с вероятностью  $p_j$ , что создает фрагментацию цикла задач с возможным повторным выполнением данного фрагмента цикла  $c_j$  раз подряд. Таким образом для  $i = 1$  время с начала цикла равняется времени обработки точки входа  $T_{\text{в}}$ . Проверке модуля с номером  $i = 2$  предшествует время обработки точки входа  $T_{\text{в}}$  и  $(i-1)$  предшествующих модулю  $i$  неуспешных проверок, а также количество  $c_{i-1}$  фрагментов цикла, каждый из которых продолжительностью  $T_{\text{пзв}}$ . При  $i > 2$  задержка до начала обработки модуля  $t_{\text{н}i}$  возрастает за счет фрагментации и выражается в увеличении количества проверок и решаемых задач, возникающих в результате повторного обслуживания модулей предшествующих обработке целевого модуля  $i$ :

$$t_{\text{н}i} = \begin{cases} T_{\text{в}} & \text{при } i = 1, \\ T_{\text{в}} + T_{\text{п}} \cdot (i-1) + T_{\text{пзв}} \cdot c_{i-1} & \text{при } i = 2, \\ T_{\text{в}} + T_{\text{п}} \cdot \left[ i-1 + \sum_{j=2}^{i-1} ((j-1) \cdot c_j) \right] + \\ + T_{\text{пзв}} \cdot \left[ \sum_{j=1}^{i-1} \prod_{k=j}^{i-1} \begin{cases} c_k, k = j, \\ 1 + c_k, k > j \end{cases} \right] & \text{при } i > 2, \end{cases} \quad (4)$$

где  $c_i$  – среднее число подряд следующих циклов обработки модуля  $i$ ,

$$c_i = p_i + p_i^2 + p_i^3 + p_i^4 + \dots = \frac{p_i}{1-p_i}. \quad (5)$$

Средний интервал времени  $t_i$  между проверками модуля  $i > 1$  может быть оценен суммой времени между проверками предыдущего модуля и разницы  $d_i$  между временем текущей и предыдущей проверок:

$$t_i = t_1 + \sum_{k=1}^i d_k = t_{i-1} + [d_i] = t_{i-1} + [(t_{zi} + t_{\text{н}i}) - (t_{zi-1} + t_{\text{н}i-1})] = t_{i-1} + [d_{\text{н}i} + d_{zi}]. \quad (6)$$

Разница  $d_{\text{н}i}$  во времени  $t_{\text{н}}$  от начала цикла для модулей  $i$  и  $(i-1)$  по формуле (4):

$$d_{\text{н}i} = \begin{cases} 0 & \text{при } i = 1, \\ T_{\text{п}} + T_{\text{пзв}} \cdot c_{i-1} & \text{при } i = 2, \\ T_{\text{п}} \cdot [1 + (i-2) \cdot c_{i-1}] + T_{\text{пзв}} \times \\ \times \left[ \sum_{j=1}^{i-1} \prod_{k=j}^{i-1} \begin{cases} c_k, k = j \vee k = i-1 \\ 1 + c_k, \text{ иначе} \end{cases} \right] & \text{при } i > 2. \end{cases} \quad (7)$$

Разница  $d_{zi}$  времени завершения цикла для соседних модулей по формуле (2):

$$d_{zi} = \begin{cases} 0 & \text{при } i = 1, \\ (p_i - p_{i-1}) \cdot T_3 - (1 - p_{i-1}) \cdot (T_{\text{п}} + p_i \cdot T_3) + \\ + p_{i-1} \cdot \sum_{j=i}^{n-1} \left[ \prod_{k=i}^j (1-p_k) \cdot (T_{\text{п}} + p_j \cdot T_3) \right] & \text{при } i > 1. \end{cases} \quad (8)$$

Упрощенно, с учетом малости величин  $p_j$ ,  $p_{i-1}$ , формула (8):

$$d_{zi} = \begin{cases} 0 & \text{при } i = 1, \\ -(1 - p_{i-1}) \cdot T_{\text{п}} & \text{при } i > 1. \end{cases} \quad (9)$$

Тогда разница  $d_i$  между средним временем проверки текущего и предыдущего модулей определяется суммой выражений (7) и (9):

$$d_i = \begin{cases} 0 & \text{при } i = 1, \\ T_{\text{п}} + T_{\text{пзв}} \cdot c_{i-1} - (1 - p_{i-1}) \cdot T_{\text{п}} & \text{при } i = 2, \\ T_{\text{п}} \cdot [1 + (i-2) \cdot c_{i-1}] + T_{\text{пзв}} \times \\ \times \left[ \sum_{j=1}^{i-1} \prod_{k=j}^{i-1} \begin{cases} c_k, k = j \vee k = i-1 \\ 1 + c_k, \text{ иначе} \end{cases} \right] - (1 - p_{i-1}) \cdot T_{\text{п}} & \text{при } i > 2. \end{cases} \quad (10)$$

Расчет  $t_i$  по формуле (6) требует оценки значения интервала времени  $t_1$  между моментами проверки первого модуля планировщика. Величина изменяется в диапазоне  $[T_{\text{пзв}} \dots T_{\text{в}} + n \cdot T_{\text{п}} + T_3]$ . Нижняя граница  $T_{\text{пзв}}$  характерна для высокой вероятности ( $=1$ ) наступления события первого модуля (режим перегрузки). Верхняя граница диапазона обусловлена вариантом ожидания завершения полного цикла приоритетных задач (режим низкой загрузки) с решением последней. Близким к верхней границе является значение времени  $(T_{\text{в}} + nT_{\text{п}})$ , характеризующее практическую малую загруженность планировщика задач. С увеличением загруженности планировщика (без перегрузки) величина времени изменяется за счет возможного, с вероятностью  $p$ , выполнения задачи и составляет величину  $(T_{\text{в}} + jT_{\text{п}} + pT_3)$ , где  $j$  – номер проверки, на которой закончится цикл. Указанное означает, что интервал времени между проверками первого модуля определяется нагрузкой, создаваемой всеми модулями  $1 \dots N$  планировщика. Таким образом, для типовых низкой и средней загрузки интервал времени между

проверками первого модуля может быть выбран равным времени завершения полного «пустого» (без успешных проверок) цикла обработки  $(T_b + nT_n)$ . Значение данной величины сильно влияет на характеристики модели и поэтому при большой нагрузке планировщика оценки временных задержек могут быть занижены. Таким образом, определены все неизвестные величины для определения времени задержки (1) для алгоритма с приоритетными задачами.

Выполним анализ модели с учетом  $m > 0$  неприоритетных модулей (рис. 2), образующих множество  $M$  дополнительных модулей планировщика. С учетом дисциплины обслуживания только одного модуля планировщика за время неприоритетного цикла, интервал времени между проверками первого приоритетного модуля составит величину  $t_1$  равную  $(T_b + (n + 1) \cdot T_n)$ . При этом выражение (6) для оценки времени  $t_i$  приоритетных модулей остается без изменений. Средний интервал времени  $t_i$  между проверками каждого неприоритетного модуля  $i \in M = \{n + 1..n + m\}$  представляет сумму продолжительности обработки  $m$  циклов приоритетных задач и времени  $t$  обработки циклов неприоритетных задач:

$$t_i = m \cdot t_n^H + t, \quad (11)$$

где  $t_n^H$  – полное время цикла приоритетных задач:

$$t_n^H = t_n(1 + c_n), \quad (12)$$

$$t = T_n + \sum_{\forall j \in (M \setminus \{i\})} (T_n + p_j \cdot T_3). \quad (13)$$

Расчет времени задержки  $t$  по формуле (13) может быть выполнен с учетом пренебрежимо малой величины вероятности  $p_i$ . Тогда  $t = (m-1) \cdot T_n$ , а вероятности срабатывания каждого модуля  $p$  могут быть последовательно определены по формуле (3). Вероятности могут быть уточнены путем итеративного пересчета с учетом времени  $t_i$  оцениваемого по формулам (11–13). Таким образом, интервал времени между проверками неприоритетных модулей  $i \in M$

$$t_i = t_n(1 + c_n) \cdot m + T_n + \sum_{\forall j \in (M \setminus \{i\})} (T_n + p_j \cdot T_3). \quad (14)$$

Расчет времени задержки решения планируемых приоритетных и неприоритетных задач выполняется по формуле (1).

Частным вариантом планировщика является вариант с отсутствием приоритетных задач. В данном случае обработка цикла неприоритетных задач предшествует точка входа, и поэтому в (12), (14)  $t_n = t_0 = T_b$ , и  $c_n = 0$ .

При проектировании планировщика задач узла определенной системы реального времени важно выполнять оценку не только создаваемой им временной задержки, но и интенсивности  $\lambda_\Sigma$  формируемых им информационных сообщений, которые будут передаваться по каналам промышленной сети. Анализ данной величины необходим для оценки транспортных задержек [15]. Расчет интенсивности можно произвести суперпозицией интенсивности каждого модуля планировщика задач. Индивидуальную интенсивность модуля можно оценить

через величину обратную интервалу времени  $t_i^y$  между его последовательными успешными проверками (задачами). Суммарная интенсивность нагрузки узла на канал передачи сети:

$$\lambda_\Sigma = \sum_{i=1}^N \lambda_i \cdot b_i = \sum_{i=1}^N \frac{1}{t_i^y} \cdot b_i, \quad (15)$$

где  $b_i$  – индикатор (0/1) формирования модулем  $i$  информации для передачи.

Интервал времени  $t_i^y$  ( $t_i^y > t_i$ ) потока успешных проверок модуля  $i$  может быть найден путем прореживания потока всех проверок модуля средним значением времени  $T_i$  между информационными событиями модуля. Прореживание должно производиться экспоненциальным потоком событий модуля с известной интенсивностью  $1/T_i$ . Сложность корректного прореживания заключается в неопределенности типа распределения времени проверок исходного потока проверок модулей. Известно только среднее время между проверками  $t_i$ , определяемое формулой (11). Основываясь на результатах имитационных исследований [14], примем, что распределение интервалов времени потока проверок модулей подчиняется закону Эрланга, а его параметры зависят от номера модуля и создаваемой им интенсивности нагрузки. Эмпирически порядок  $u_i$  закона распределения для модуля  $i$  выбран:

$$u_i = \left[ y + 1 - \frac{i}{y} \right], \quad (16)$$

где, квадратные скобки в выражении определяют целую часть значения;  $y$  – начальный порядок закона распределения Эрланга, зависит от вероятности срабатывания модулей и предшествующего им числа проверок  $i$ :

$$y = n - \sum_{i=1}^n (i \cdot p_i). \quad (17)$$

Таким образом, поток Эрланга успешных проверок  $t_i^y$  модуля  $i$  представляет собой сумму  $u_i$  экспоненциальных потоков с параметром интенсивности  $u_i/t_i$ . Прореживание потока осуществляется простой выборкой из указанного распределения случайных значений, больших или равных значениям экспоненциального потока событий модуля с интенсивностью  $1/T_i$ , что определяет условие превышения среднего интервала времени между событиями модуля  $i$ . Математическое ожидание среднего интервала времени между успешными проверками модуля  $i$  определяется расчетом среднего арифметического значения полученного прореженного распределения. Интенсивность информационной нагрузки узла на канал промышленной сети определяется по формуле (15).

#### 4 ЭКСПЕРИМЕНТЫ

Выполним анализ среднего времени задержки обработки задач планировщика узла с использованием предложенной модели и способа оценки характеристик. Исследуем три различных сценария загрузки планировщи-

ка задач (табл. 1): С1) однородная информационная нагрузка модулей с низкой частотой технологических событий (большим интервалом времени между событиями); С2) однородная нагрузка с высокой частотой событий; С3) разнородная (контрастная) нагрузка модулей. Анализ предложенных сценариев и сравнение полученных результатов с результатами имитационного моделирования [14] позволит установить адекватность разработанной аналитической модели и способа оценки. Примем типовые исходные данные для моделирования (табл. 1) [7]:  $N (n=N), T_b, T_n, T_z, T_{ct}$ .

## 5 РЕЗУЛЬТАТЫ

Результаты оценки времени задержки обработки задач планировщика узла промышленной сети систематизированы в табл. 2 по анализируемым сценариям функционирования. Результаты аналитического (А) моделирования дополнены оценками полученными имитационным (И) моделированием [14], а также расчетом относительного отклонения полученных результатов.

## 6 ОБСУЖДЕНИЕ

Сравнительный анализ результатов моделирования (табл. 2) позволил сделать следующие основные выводы. Анализ характера изменения среднего времени задержки обработки задач подтверждает известные из литературы [7] и создает новые положения. Получены следующие основные результаты:

- с ростом номера решаемой задачи возрастает время ее обработки, что объясняется сокращением частоты проверок модулей с большими номерами за счет фрагментации;
- с изменением номера модуля время обработки задачи изменяется практически линейно (при однородной нагрузке модулей планировщика задач);
- рост времени задержки обработки задачи, следующей за модулем с высокой частотой событий, носит скачкообразный характер.

Отклонение результатов разработанной аналитической модели от имитационной [14] не превышает значения 15%. Исследование других сценариев функционирования, не представленных в работе, при различном количестве модулей программы  $N$ , подтверждает указанный результат. Во многом полученные отклонения

результатов аналитической модели объясняются принятыми допущениями (16), (17) в расчетах порядка распределения Эрланга интервалов времени  $t_i$  между последовательными проверками модулей планировщика задач. Полученные результаты подтверждают адекватность разработанной аналитической модели планировщика задач и способа оценки его задержек.

Предложенная в работе модель и способ анализа задержек будут использоваться при проектировании подсистем планирования задач на узлах распределенных систем реального времени с целью обеспечения заданных временных характеристикам обработки технологической информации. Указанное предполагает использование предложенного математического аппарата для определения: допустимого числа программных модулей планировщика и распределения среди них приоритетных и не приоритетных задач обрабатываемых узлом в зависимости от частоты технологических событий и заданных требований по задержке обработки.

## ВЫВОДЫ

В работе построена аналитическая модель и предложен способ количественной оценки вероятностных и временных характеристик циклического алгоритма планирования задач узла, управляемого приоритетами, в распределенной fieldbus-системе реального времени. Впервые при оценке характеристик учтена ранее не анализируемая в известных аналитических моделях наиболее полная совокупность значимых параметров и особенностей функционирования алгоритма: разнородные уровни приоритетов, количество задач по приоритетам, конфигурационные свойства задач, задержки операций и частота наступления событий планировщика, процедуры прерывания и дообслуживания задач, и другие параметры. Как показано, это обеспечивает адекватность, приемлемую точность и низкую вычислительную сложность аналитических расчетов характеристик подсистемы планирования задач узла по сравнению с известной имитационной моделью. Практическая значимость полученных результатов заключается в возможности эффективного проектирования программной подсистемы планирования задач узла распределенной системы мягкого реального времени с требуемыми временными характеристиками.

Таблица 1 – Исходные данные (временные параметры указаны в мс)

| Сцен. | $N$ | $T_b$ | $T_n$ | $T_z$ | $T_{c1}$ | $T_{c2}$ | $T_{c3}$ | $T_{c4}$ | $T_{c5}$ | $T_{c6}$ | $T_{c7}$ | $T_{c8}$ | $T_{c9}$ | $T_{c10}$ |
|-------|-----|-------|-------|-------|----------|----------|----------|----------|----------|----------|----------|----------|----------|-----------|
| C1    | 10  | 1     | 0,3   | 1     | 1000     |          |          |          |          |          |          |          |          |           |
| C2    |     |       |       |       | 50       |          |          |          |          |          |          |          |          |           |
| C3    |     |       |       |       | 200      | 10       | 200      | 10       | 200      |          |          |          |          |           |

Таблица 2 – Результаты сравнения аналитических и имитационных оценок времени задержки

| № задачи | Сценарии функционирования |       |       |        |        |        |        |        |        |
|----------|---------------------------|-------|-------|--------|--------|--------|--------|--------|--------|
|          | С1                        |       |       | С2     |        |        | С3     |        |        |
|          | А                         | И     | О (%) | А      | И      | О (%)  | А      | И      | О (%)  |
| 1        | 4,189                     | 4,266 | 1,805 | 4,322  | 4,268  | 1,265  | 4,263  | 4,211  | 1,235  |
| 2        | 4,214                     | 4,256 | 0,987 | 4,664  | 4,460  | 4,574  | 4,313  | 4,234  | 1,866  |
| 3        | 4,198                     | 4,240 | 0,991 | 4,866  | 4,715  | 3,203  | 4,266  | 4,336  | 1,614  |
| 4        | 4,234                     | 4,269 | 0,820 | 5,109  | 5,047  | 1,228  | 5,432  | 5,508  | 1,380  |
| 5        | 4,225                     | 4,264 | 0,915 | 5,384  | 5,468  | 1,536  | 5,739  | 5,629  | 1,954  |
| 6        | 4,237                     | 4,305 | 1,580 | 6,141  | 6,041  | 1,655  | 5,850  | 5,821  | 0,498  |
| 7        | 4,224                     | 4,296 | 1,676 | 6,569  | 6,826  | 3,765  | 5,669  | 5,945  | 4,643  |
| 8        | 4,252                     | 4,337 | 1,960 | 7,079  | 7,919  | 10,607 | 8,799  | 10,346 | 14,953 |
| 9        | 4,285                     | 4,340 | 1,267 | 9,381  | 9,498  | 1,232  | 9,935  | 11,047 | 10,066 |
| 10       | 4,280                     | 4,373 | 2,127 | 10,354 | 11,882 | 12,860 | 10,280 | 12,048 | 14,675 |

## СПИСОК ЛІТЕРАТУРИ

- David Neels Pon Kumar D. An Enhanced Scheduling Scheme for QoS in Mobile WiMAX Networks / D. David Neels Pon Kumar, Jithin Raj, K. Arun Kumar, K. Murugesan // International Journal of Engineering and Innovative Technology (IJEIT). – 2012. – Vol. 2, № 5. – P. 163–169.
- Park W. Scheduler Design for Multiple Traffic Classes in OFDMA Networks / W. Park, S. Cho, S. Bahk // in Proc. IEEE Int. Conf. Communications, Istanbul, Turkey. – 2006. – Vol. 2. – P. 790–795.
- Кавалеров М. В. Повышение эффективности планирования с фиксированными приоритетами задач жесткого реального времени на основе применения г-алгоритма для формирования запросов этих задач / М. В. Кавалеров, Н. Н. Матушкин // Нейрокомпьютеры: разработка, применение. – 2014. – №12. – С. 14–19.
- Кавалеров М. В. Новый алгоритм назначения параметров задач реального времени с линейными интервальными ограничениями в условиях планирования с фиксированными приоритетами, основанный на сокращенном переборе приоритетов / М. В. Кавалеров, Н. Н. Матушкин // Нейрокомпьютеры: разработка, применение. – 2013. – №11. – С. 012–017.
- Embedded System Software. Module 6. Version 2. EE IIT, Kharagpur 1 Lesson 30 Real-Time T. 31437008-Realtime-Task-Scheduling-2. – 128 p. – Access mode: <http://www.nptel.ac.in/courses/108105057/Pdf/Lesson-30.pdf>
- Marcus Völp. Real-Time Systems: Event-Driven Scheduling / Marcus Völp, Hermann Härtig [Electronic resource]. – 2003. – Access mode: <https://os.inf.tu-dresden.de/Studium/RTS/WS2013/05-EventDriven.pdf>
- Дитрих Д. LON-технология, построение распределенных приложений / Д. Дитрих, Д. Лой, Г. Ю. Швайнциер // Пер. с нем. – под ред. О. Б. Низамутдинова. – Пермь : Звезда, 1999 – 242 с.
- LonTalk protocol specification: ANSI/CEA-709.1-B. – United States, 2006.
- Moshe Kam. Collision Resolution Simulation for Distributed Control Architectures using LonWorks / Moshe Kam // IEEE International Conference on Automation Science and Engineering. – Edmonton, Canada: IEEE, 2005. – P. 319–326.
- Mi kowicz M. Analysis of Mean Access Delay in Variable-Window CSMA / M. Mińkiewicz // Sensors. Schweiz: Molecular Diversity Preservation International, 2007. – Vol. 7. – P. 3535–3559.
- Miśkiewicz M. Access delay in LonTalk MAC protocol / M. Miśkiewicz // Computer Standards & Interfaces. Nederland: Elsevier Science Publishing Company, 2009. – P. 548–556.
- Miśkiewicz M. Latency characteristics of event-driven task scheduler embedded in neuron chip / M. Miśkiewicz // IJCSNS International Journal of Computer Science and Network Security. – 2007. – Vol. 7, № 12. – P. 132–149.
- Buchholz P. Analytical analysis of access-schemes of the CSMA type / P. Buchholz, J. Plonnigs // Proc. of IEEE International Workshop on Factory Communication Systems WFCS'2004, Wien, 2004. – P. 127–136.
- Даденков С. А. Разработка имитационной модели уровня приложенной узла LonWorks сети / С. А. Даденков, В. В. Чмыков / / Материалы конференции «Автоматизированные системы управления и информационные технологии», 2014. – Режим доступа: [http://etf.pstu.ru/conf/files/stud\\_conf\\_2014.pdf](http://etf.pstu.ru/conf/files/stud_conf_2014.pdf).
- Даденков С. А. Подход к построению аналитической модели информационно-управляющей сети LonWorks на основе нейронных сетей / С. А. Даденков, Е. Л. Кон // Нейрокомпьютеры: разработка, применение. – 2013. – №11. – С. 64–69.

Статья поступила в редакцию 10.05.2017  
После доработки 21.08.2017.

Даденков С. О.<sup>1</sup>, Даденков Д. О.<sup>2</sup>

<sup>1</sup>Канд. техн. наук, старший преподаватель кафедры «Автоматика і телемеханіка» Пермського національного дослідницького політехнічного університету, м. Перм, Росія

<sup>2</sup>Старший преподаватель кафедры «Мікропроцесорні засоби автоматизації» Пермського національного дослідницького політехнічного університету, м. Перм, Росія

## МОДЕЛЬ ЦИКЛИЧНОГО АЛГОРИТМУ ПЛАНУВАННЯ ЗАДАЧ У РОЗПОДІЛЕНІЙ СИСТЕМІ РЕАЛЬНОГО ЧАСУ

**Актуальність.** Вирішено задачу кількісного оцінювання ймовірнісних і часових характеристик підсистеми планування задач вузлів для проектування розподілених систем реального часу. Об'єктом дослідження є циклічний алгоритм планування задач на основі пріоритетів (round robin priority driven preemptive scheduler), використовуваний у розподілених fieldbus-мережах.

**Мета роботи** – побудова моделі і розробка способу кількісного оцінювання ймовірнісних і часових характеристик підсистеми планування задач вузла: затримки й інтервалів часу рішення задач, інтенсивності інформаційного навантаження на канали промислової fieldbus-мережі.

**Метод.** Для вирішення задачі використано апарат теорії ймовірностей. Запропоновано графа станів і переходів моделі, що описує функціонування циклічного алгоритму планування задач вузла. Отримано аналітичні співвідношення для оцінки основних ймовірнісних і часових характеристик моделі підсистеми планування задач: середнього часу затримки обробки задачі, середнього інтервалу часу між рішенням задач, інтенсивності інформаційного навантаження на канали fieldbus-мережі. Запропоновані модель і спосіб оцінки відрізняються від відомих деталізацією раніше не аналізованими в сукупності значимими параметрами функціонування алгоритму: різнорідними рівнями пріоритетів, кількістю задач по пріоритетах, конфігураційними властивостями задач, затримками операцій і частотою настання подій планувальника, процедурами переривання і дообслуговування задач, та іншими параметрами.

**Результати.** Розроблені модель і спосіб кількісної оцінки ймовірнісних та часових характеристик підсистеми планування задач з циклічним алгоритмом обробки на основі пріоритетів.

**Висновки.** Проведені експерименти підтвердили працездатність і адекватність запропонованого математичного забезпечення і дозволяють рекомендувати його для використання на практиці при вирішенні задач проектування розподілених систем реального часу з заданими тимчасовими характеристиками. У ході проектування інструментарій буде використаний для визначення: припустимого числа програмних модулів планувальника і розподілу серед них пріоритетних і непріоритетних задач оброблюваних вузлом у залежності від частоти технологічних подій і заданих вимог по затримці обробки.

**Ключові слова:** планування задач, реальний час, циклічний алгоритм, пріоритет-орієнтований алгоритм, модель, ймовірнісні та часові характеристики, розподілена система, промислова мережа, fieldbus.

Dadenkov S. A.<sup>1</sup>, Dadenkov D. A.<sup>2</sup>

<sup>1</sup>PhD, Assistant professor of department «Automatics and telemechanics», Perm National Research Polytechnic University, Perm, Russia

<sup>2</sup>Assistant professor of department «Microprocessor automation equipment», Perm National Research Polytechnic University, Perm, Russia

#### ROUND-ROBIN PREEMPTIVE SCHEDULER MODEL IN DISTRIBUTED REAL-TIME SYSTEM

**Context.** The problem of quantitative estimation of probabilistic and temporal characteristics of the planning node's tasks subsystem for the design of distributed real-time systems has been solved. Round robin priority driven preemptive scheduler is the subject of research, used in distributed fieldbus-networks.

**The goal** is to create a model and a method of quantifying the probabilistic and temporal characteristics of the node's task scheduling subsystem: delays and time intervals for solving problem, intensity of the information load on the industrial fieldbus-network channels.

**Method.** The method of probability theory has been used there to solve problems. State and transition graph of model has been offered, describing the functioning of cycle algorithm for scheduling node tasks. Analytical relations for estimating the main probabilistic and temporal characteristics of the task scheduling subsystem model have been got. Specifically, an average delay time for task processing, an average time interval between solving problems, an intensity of the information load on the fieldbus-network channels. The proposed model and method are different from well-known ones: with the detailing of the important parameters of the algorithm functioning, that weren't previously analyzed: dissimilar levels of priorities, number of priority tasks, configuration characteristics of tasks, operation delays and frequency of events of scheduler, etc.

**Results.** The developed model and method of quantifying the probabilistic and temporal characteristics of tasks scheduling subsystem with round robin priority driven preemptive scheduler.

**Conclusions.** The conducted experiments have confirmed the proposed software operability and adequacy, allow recommending it for use in practice to solve problems of designing distributed real-time systems with specified time characteristics. During the design process tools will be used for determining: the allowable number of scheduler software modules and the distribution among them of the priority and non-priority tasks handled by the node, depending on the frequency of the technological events and the specified processing delay requirements.

**Keywords:** task scheduling, real-time, round-robin, priority-driven preemptive scheduler, model, probable and time response characteristics, distributed system, industrial network, fieldbus.

#### REFERENCES

1. David Neels Pon Kumar D., Jithin Raj, K. Arun Kumar, K. Murugesan An Enhanced Scheduling Scheme for QoS in Mobile WiMAX Networks, *International Journal of Engineering and Innovative Technology (IJEIT)*, 2012, Vol. 2, No. 5, pp. 163–169.
2. Park W., Cho S., Bahk S. Scheduler Design for Multiple Traffic Classes in OFDMA Networks, in *Proc. IEEE Int. Conf. Communications, Istanbul, Turkey*, 2006, Vol. 2, pp. 790–795.
3. Kavalerov M. V., Matushkin N. N. Povyshenie effektivnosti planirovaniya s fiksirovannymi prioritetami zadach zhestkogo real'nogo vremeni na osnove primeneniya r-algoritma dlya formirovaniya zaprosov e'tix zadach, *Nejrokomp'yutery: razrabotka, primeneniye*, 2014, No. 12, pp. 14–19.
4. Kavalerov M. V., Matushkin N. N. Novyj algoritm naznacheniya parametrov zadach real'nogo vremeni s linejnymi interval'nymi ogranicheniyami v usloviyax planirovaniya s fiksirovannymi prioritetami, osnovannyj na sokrashhennom perebore prioritetov, *Nejrokomp'yutery: razrabotka, primeneniye*, 2013, No. 11, pp. 012–017.
5. Embedded System Software. Module 6. Version 2. EE IIT, Kharagpur 1 Lesson 30 Real-Time T. 31437008-Realtime-Task-Scheduling-2, 128 p. Access mode: <http://www.nptel.ac.in/courses/108105057/Pdf/Lesson-30.pdf>
6. Marcus Völp., Hermann Härtig [Electronic resource], Real-Time Systems: Event-Driven Scheduling, 2003, Access mode: <https://os.inf.tu-dresden.de/Studium/RTS/WS2013/05-EventDriven.pdf>
7. Ditrix D., Loj D., Shvajncer G. Yu. LON-tehnologiya, postroenie raspredelennyx prilozhenij, Per. s nem. pod red. O. B. Nizamutdinova. Perm', Zvezda, 1999, 242 p.
8. LonTalk protocol specification: ANSI/CEA-709.1-B. United States, 2006.
9. Moshe Kam. Collision Resolution Simulation for Distributed Control Architectures using LonWorks, *IEEE International Conference on Automation Science and Engineering*. Edmonton, Canada, IEEE, 2005, pp. 319–326.
10. Miśkiewicz M. Analysis of Mean Access Delay in Variable-Window CSMA, Sensors. Schweiz: Molecular Diversity Preservation International, 2007, Vol. 7, pp. 3535–3559.
11. Miśkiewicz M. Access delay in LonTalk MAC protocol, *Computer Standards & Interfaces. Nederland: Elsevier Science Publishing Company*, 2009, pp. 548–556.
12. Miśkiewicz M. Latency characteristics of event-driven task scheduler embedded in neuron chip, *IJCSNS International Journal of Computer Science and Network Security*, 2007, Vol. 7, No. 12, pp. 132–149.
13. Buchholz P., Plonnigs J. Analytical analysis of access-schemes of the CSMA type, *Proc. of IEEE International Workshop on Factory Communication Systems WFCS'2004, Wien*, 2004, pp. 127–136.
14. Dadenkov S. A., Chmykov V. V. Razrabotka imitacionnoj modeli urovnya prilozhenij uzla LonWorks seti, *Materialy konferencii «Avtomatizirovannye sistemy upravleniya i informacionnye tehnologii»*, 2014. Rezhim dostupa: [http://etf.pstu.ru/conf/files/stud\\_conf\\_2014.pdf](http://etf.pstu.ru/conf/files/stud_conf_2014.pdf).
15. Dadenkov S. A., Kon E. L. Podxod k postroeniyu analiticheskoj modeli informacionno-upravlyayushhej seti LonWorks na osnove nejrochipov, *Nejrokomp'yutery: razrabotka, primeneniye*. Moscow, Radiotekhnika, 2013, No. 11, pp. 64–69.