

**МОДИФИЦИРОВАННЫЙ ГИБРИДНЫЙ АЛГОРИТМ СИНХРОНИЗАЦИИ  
ВРЕМЕНИ В СРЕДЕ МОДЕЛИРОВАНИЯ ДИАНА****Д.Ю. Волканов, А.Б. Глонина, В.Н. Кибитова (Москва)****Введение**

В системах распределенного дискретно-событийного моделирования существенное влияние на скорость выполнения моделей оказывает алгоритм синхронизации времени. Выбрав наиболее подходящий для соответствующей модели алгоритм синхронизации, можно существенно сократить время проведения экспериментов с ней, и, следовательно, время на разработку моделируемой системы. В Лаборатории Вычислительных Комплексов факультета ВМК МГУ имени М.В. Ломоносова была разработана распределенная система дискретно-событийного моделирования (РС ДСИМ) DYANA[1], в которой изначально был реализован классический консервативный алгоритм синхронизации времени. Данный алгоритм характеризуется низким уровнем параллелизма выполнения моделей, что приводит к большой длительности имитационных экспериментов. Для того, чтобы ускорить процесс моделирования, была разработана и реализована модификация одного из известных алгоритмов гибридных алгоритмов синхронизации времени, которая способна на определенных моделях бороться с недостатками консервативных алгоритмов.

**Постановка задачи синхронизации времени в РС ДСИМ**

В РС ДСИМ модель представляет собой совокупность логических процессов (ЛП), взаимодействующих посредством приёма и отправки сообщений. У каждого ЛП есть свои локальные часы. Каждое сообщение имеет временную метку. Существенными для синхронизации являются следующие события ЛП: получение сообщения, отправка сообщения и продвижение модельного времени. При обработке сообщения меняется состояние ЛП, в соответствии с временной меткой, обновляется модельное время ЛП и, возможно, генерируются новые сообщения с метками в будущем.

Для корректного выполнения имитационной модели необходимо, чтобы все причинно-следственные связи между событиями были сохранены. То есть для любых событий  $a$  и  $b$  если  $a$  – причина, а  $b$  – следствие, то физическое время события  $a$  меньше физического времени события  $b$ . При этом из того, что  $a$  – причина, а  $b$  – следствие, следует, что модельное время события  $a$  меньше или равно модельного времени события  $b$ .

Из-за задержек на передачу сообщений и различной производительности узлов РС ДСИМ без применения специальных алгоритмов синхронизации причинно-следственные связи могут нарушаться.

Таким образом, для обеспечения корректности имитационного моделирования в РС ДСИМ необходимо использовать алгоритм синхронизации времени, гарантирующий сохранение причинно-следственных связей, то есть обработку пришедших сообщений в порядке неубывания их временных меток[2].

**Система имитационного моделирования DYANA.**

РС ДСИМ DYANA – это среда имитационного моделирования, которая была разработана в соответствии со стандартом HLA[3]. В качестве компонента RTI в ней используется среда CERTI[4].

RTI CERTI включает в себя: глобальный процесс RTIG, локальные процессы RTIA и библиотеку libRTI. Каждый федерат, то есть процесс, представляющий ЛП, обращается к библиотеке libRTI, которая предоставляет ему сервисы RTI. Для обработки запроса федерата libRTI подключается посредством Unix-сокетов к локальному процессу RTIA. Если запрос связан с работой других федератов, то происходит соединение RTIA с процессом RTIG, являющимся центральным элементом CERTI. Как только RTIG получает сообщение от некоторого RTIA, RTIG доставляет это сообщение тем RTIA, для которых оно было предназначено.

В каждом RTIA федерата существует класс TimeManagement, который ответственен за продвижение времени федерата. К сожалению, CERTI не предоставляет оптимистических сервисов управления временем, поэтому все федераты продвигают время согласно консервативному алгоритму.

### **Модифицированный гибридный алгоритм спекулятивного вычисления с локальными откатами.**

Исходя из особенностей РС ДСИМ DYANA, описанных в предыдущем разделе для реализации и доработки был выбран алгоритм, минимально использующий оптимистические сервисы синхронизации времени.

Основная идея базового гибридного алгоритма [5] состоит в том, что ЛП разрешено продвигать время сверх его LBTS (LBTS – наименьшее значение временных меток сообщений, которые может получить ЛП). Отправка сообщений другим ЛП невозможна до тех пор, пока существует возможность получения данным ЛП сообщения с временной меткой, меньшей текущего времени данного ЛП, то есть пока его текущее время превышает LBTS. Это гарантирует, что все откаты будут строго локальными и каскадных откатов не произойдет.

Для сокращения величины откатов авторами было предложено ввести ограничение на свободу выполнения ЛП. Данная величина обозначается LIMIT и вычисляется по формуле:

$$\text{LIMIT} = \text{LBTS} + (N - \text{LBTS}) - (N - \text{LBTS}) * (1 - (1 - \alpha)(1 - \beta)),$$

где  $N$  – время окончания работы ЛП,  $\alpha = \max_j \alpha_{js}$ ,  $\alpha_{js}$  – вероятность отправки сообщения  $j$ -ым ЛП на шаге  $s_j$   $i$ -му ЛП,  $\beta = \max_j \beta_j$ ,  $\beta_{js}$  – вероятность того, что на всех предыдущих шагах и на шаге  $s_i$ , приращение часов  $i$ -го ЛП больше приращения часов  $j$ -го ЛП.

Величина  $\alpha$  задается исходя из свойств модели и рассчитывается до ее запуска. Пусть  $X_i$  – приращение часов заданного ЛП, для которого рассчитывается величина LIMIT на  $i$ -м шаге. Если все ЛП имеют одинаковое распределение приращений часов, то вероятность того, что на  $i$ -м шаге приращение часов в заданном ЛП больше, чем в любом ЛП, который может отправлять ему сообщения, равна  $P(X < X_i) = F(X_i)$ , где  $F(X)$  – функция распределения приращений часов ЛП. Тогда величина  $\beta$  на  $i$ -м шаге определяется как  $\beta_i = F(X_i)\beta_{i-1}$ ,  $\beta_0 = 1$ .

На рис.1 представлено формальное описание разработанного алгоритма, где используются следующие обозначения:  $Q_i$  – действие выполняемое ЛП,  $T_m$  – временная метка полученного сообщения,  $L_i$  – текущее время ЛП,  $T_{tm}$  – временная метка сообщения, вызвавшего откат,  $T_s$  – временная метка контрольной точки,  $T_{Q_i}$  – временная метка действия  $Q_i$ .

```

DO
  F(Qi):
    CHECKPOINT()
    IF (Qi == SEND)
      WHILE(Li > LBTSi)
        WHILE HAS_MESSAGE(Li)
          F(RECEIVE)
        IF (RETRACT_HAPPEND) GO_TO_PREVIOUS_STATE()
      DELETE_CHECKPOINTS()
      SEND()
    ELSE IF(Qi == RECEIVE)
      IF (Tm <= Li)
        RETRACT()
      ELSE
        WHILE (Tqi > LIMIT)
          PROCESS_INCOMING_MESSAGES()
          IF (RETRACT_HAPPEND) GO_TO_PREVIOUS_STATE()
        EXECQi()
    ELSE IF (Qi == END)
      WHILE(Li > LBTSi)
        PROCESS_INCOMING_MESSAGES()
        IF (RETRACT_HAPPEND) GO_TO_PREVIOUS_STATE()
      STATE = FINISH
    ELSE
      WHILE (Tqi > LIMIT)
        PROCESS_INCOMING_MESSAGES()
        IF (RETRACT_HAPPEND) GO_TO_PREVIOUS_STATE()
      EXECQi()
  WHILE (STATE != FINISH)

GO_TO_PREVIOUS_STATE(Tmr):
  FOR STATE IN REVERSED(CHECKPOINTS):
    IF Ts > Trm
      DELETE_STATE(STATE)
    ELSE
      CONTINUE_IN_STATE(STATE)

```

Рис. 1. Описание модифицированного алгоритма с локальными откатами.

### Практическая реализация модифицированного гибридного алгоритма в среде имитационного моделирования DYANA.

Для реализации модифицированного гибридного алгоритма РС ДСИМ DYANA была доработана таким образом, чтобы сохранить возможность использования ранее разработанных моделей без их дополнительной адаптации.

Для этого были реализованы следующие функции:

- callTAR(t) продвигает время федерата до отметки t, не принимая во внимания текущее значение LBTS, доставляя федерату все сообщения с временной меткой, не превышающей t.
- callNMR(t) продвигает время федерата либо до времени t, либо до времени пришедшего ему сообщения с минимальной временной меткой, даже если эта метка превышает LBTS данного федерата.
- saveNewPoint() сохраняет текущее состояние федерата (в том числе его очередь сообщений). Данная функция вызывается неявно при запуске федерата и после каждого продвижения его времени.
- retractToPreviousPoint(t) возвращает состояние федерата с временной меткой, меньшей либо равной t, и осуществляет откат до этого состояния. Вызов этой функции происходит неявно при приеме сообщений от других федератов, в том случае, если временная метка пришедшего сообщения меньше, чем его текущее время.

callTAR и callNMR не продвигают время федерата сверх величины LIMIT.

На рис.2 приведена диаграмма классов, реализующих федерат.

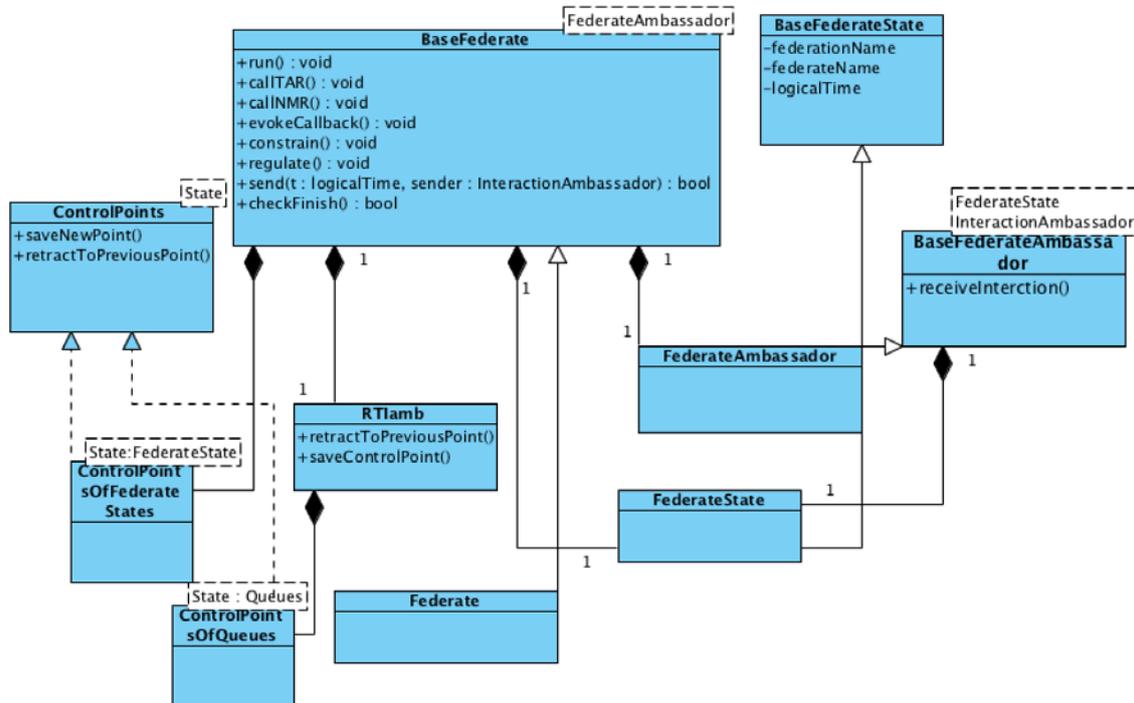


Рис. 2. Архитектура федерата в среде моделирования DYANA.

### Экспериментальное исследование модифицированного гибридного алгоритма в среде имитационного моделирования DYANA.

С целью проведения экспериментального исследования был выбран ряд тестовых моделей. Каждая тестовая модель характеризуется: 1) топологией, 2) вероятностью отправки сообщения одним процессом другому на каждом шаге ( $\alpha$ ).

Рассматривались 4 типа топологий: 1) полносвязная из 3-х процессов, 2) полносвязная из 13 процессов, 3) кольцо из 13 процессов, 4) звезда из 13 процессов.

Все исследования проводились в PC ДСИМ DYANA с использованием системы GRANIT, где под каждый процесс была выделена отдельная виртуальная машина.

Сравнивался консервативный алгоритм с базовым гибридным и модифицированным гибридным алгоритмами. Таблицы 1 и 2 представляют характеристики моделей, при которых целесообразно использовать разработанные алгоритмы вместо консервативного. Используются следующие обозначения:

$A1 \ll A2$  – ограничения на значения параметра  $\alpha$ , при которых алгоритм  $A1$  работает гораздо хуже, чем  $A2$  (различия в производительности – более 5 %)

$A1 \gg A2$  – ограничения на значения параметра  $\alpha$ , при котором алгоритм  $A1$  работает гораздо лучше, чем  $A2$  (различия в производительности – более 5 %)

$A1 \approx A2$  – ограничения на значения параметра  $\alpha$ , при которых эффективность алгоритмов примерно одинакова (различия в производительности не превосходят 5 %)

Тип топологии	$A1 \ll A2$	$A1 \approx A2$	$A1 \gg A2$
1	$\alpha \leq 0.416$	$0.416 \leq \alpha \leq 0.65$	$\alpha \geq 0.65$
Полносвязная	$\alpha \leq 0.225$	$0.225 \leq \alpha \leq 0.512$	$\alpha \geq 0.512$

Кольцо	$\alpha \leq 0.299$	$0.299 \leq \alpha \leq 0.56$	$\alpha \geq 0.56$
Звезда	$\alpha \leq 0.322$	$0.322 \leq \alpha \leq 0.612$	$\alpha \geq 0.612$

Таблица 1. Сравнение алгоритма гибридного алгоритма (A2) и консервативного алгоритма (A1).

Тип топологии	$A1 \ll A2$	$A1 \approx A2$	$A1 \gg A2$
1	$\alpha \leq 0.462$	$0.462 \leq \alpha \leq 0.68$	$\alpha \geq 0.68$
Полносвязная	$\alpha \leq 0.276$	$0.276 \leq \alpha \leq 0.53$	$\alpha \geq 0.53$
Кольцо	$\alpha \leq 0.315$	$0.415 \leq \alpha \leq 0.584$	$\alpha \geq 0.584$
Звезда	$\alpha \leq 0.34$	$0.34 \leq \alpha \leq 0.63$	$\alpha \geq 0.63$

Таблица 2. Сравнение модифицированного гибридного алгоритма (A2) и консервативного алгоритма (A1).

В ходе экспериментального исследования были выделены классы моделей, на которых гибридный и модифицированный гибридный алгоритмы эффективней, чем консервативный. При этом для модифицированного алгоритма данный класс шире.

### Выводы

В данной работе представлен модифицированный гибридный алгоритм синхронизации времени и его реализация в РС ДСИМ DYANA, разработанной в Лаборатории вычислительных комплексов факультета ВМК МГУ. Среда моделирования применяется в промышленности при разработке бортовых ВС РВ.

Использование предложенного в работе модифицированного гибридного алгоритма синхронизации позволяет ускорить выполнение экспериментов, а значит и ускорить процесс разработки бортовой ВСПВ.

В качестве направлений дальнейшего исследований можно выделить следующие:

- Рассмотреть другие алгоритмы, основанные не только на локальных откатах.
- Исследовать алгоритм на большем классе топологий, для выделения большего числа классов моделей, на которых гибридный алгоритм работает лучше, чем консервативный.
- Улучшить оценку величины LIMT, которая позволит получить еще больший прирост в производительности.

### Литература

1. Волканов Д.Ю., Антоненко В.А., Герасёв А.В., Глонина А.Б., Захаров В.А., Зорин Д.А., Коннов И.В., Пашков В.Н., Подымов В.В., Савенков К.О., Смелянский Р.Л., Чемерицкий Е.В.. Методика использования системы имитационного моделирования РВС РВ ДИАНА, основанной на стандарте HLA. // Сборник докладов шестой всероссийской научно-практической конференции "Имитационное моделирование. Теория и практика" (ИММОД-2013), Издательство "ФЭН" Академии наук РТ, Казань, 2013 - Т 1 - С. 322–326.
2. Волканов Д.Ю., Глонина А.Б., Кибитова В.Н. Исследование гибридного алгоритма синхронизации времени для среды имитационного моделирования ДИАНА // Программные системы и инструменты / Под ред. А. Н. Терехин. —

- Т. 15. — Издательский отдел факультета ВМиК МГУ Москва, МГУ, 2014. — С. 105–115
3. 1516-2010 IEEE Standard for Modeling and Simulation High Level Architecture (HLA), 2010.
  4. Noulard E., Rousselot J. Y., Siron P. CERTI, an Open Source RTI, why and how. // Spring Simulation Interoperability Workshop, 2009, 23-27.
  5. Dickens P. M., Reynolds Jr P. F. SRADS with local rollback. // In Proceedings of the SCS Multiconference on Distributed Simulation., 1990.