

Стабилизация взаимодействия или автоволны в параллельных вычислениях?

Д. В. Манаков, П. А. Васёв

Институт математики и механики им. Н.Н. Красовского УрО РАН, Екатеринбург, Россия

Аннотация. Цель данной работы состоит в том, чтобы отобразить (нарисовать) стабилизацию взаимодействия в виде волны, максимально абстрагируясь от значений вычисляемой функции (меняя только параметры распараллеливания) с целью выявления общих закономерностей. Для ускорения стабилизации рассматриваются сбалансированные вычисления (более общее требование, чем однотипность агентов). Таким образом, начальное распределение данных (равномерное с учетом гетерогенности вычислительной системы и графа задач) выбирается наиболее близким к «логистической кривой». Балансировка нагрузки была ранее рассмотрена как задача автоматического управления с очевидной стратегией – следующая задача из графа задач назначается тому процессору, у которого математическое ожидание нагрузки процессора меньше. Вообще говоря, это стандартная задача интерполяции (сглаживания). Стоит отметить, что рассматривалась не только задача Дирихле, но и частично задача Пуассона, поскольку на практике была замечена высокая чувствительность к коэффициенту, связанному с обработкой очереди сообщений.

Наибольший интерес в теоретическом и практическом плане представляет задача определения условий, при выполнении которых волна стабилизации взаимодействия будет близка к автоволне. Такая постановка задачи обоснована. Известно, что автоволны (а именно спиральные волны) возникают в химической реакции Белоусова-Жиботинского, а также что подобный процесс в некоторой степени описывается моделью Лотки-Вольтера. Получается, что, распараллеливая решение, например, системы двух линейных уравнений (модель хищник-жертва без учета нелинейной составляющей (взаимодействия)), автоматически получится численное решение стохастической модели Лотки-Вольтера, поскольку в параллельных вычислениях всегда имеются случайная составляющая и взаимодействие, а протяженные объемы соответствуют, например, линейке процессоров. На практике можно рассмотреть не только линейку процессоров, но и решетку процессоров (динамическая 3D-визуализация) и разные схемы взаимодействия, среди которых особый интерес представляют «недетерминированные» (без синхронизации, на бесконечном горизонте) стационарные процессы (предполагается, что схема распараллеливания явно не зависит от времени). Кроме того, в параллельных вычислениях можно ввести псевдослучайные помехи, например, запаздывание некоторого процессора по времени или по координате, в том числе через управление параметрами в режиме онлайн-визуализации.

В теоретическом плане важно, что оптимизационная задача максимизации энтропии (производительности параллельных вычислений) сводится к параметризованной модели белошумного случайного процесса. И вид отображения (волна) строится с учетом этой модели. Также хотелось обсудить задачу параллельной фильтрации данных (частный случай онлайн-визуализации, актуальный при работе с большими данными) в контексте задачи Канторовича (транспорт мер) и принципа динамического программирования.

Ключевые слова: параллельные вычисления, стабилизация взаимодействия, автоволны, стохастическая модель Лотки-Вольтера.

Stabilization of interaction or autowaves in parallel computing?

D. V. Manakov, P. A. Vasev

Krasovsky Institute of Mathematics and Mechanics, Ural Branch of the Russian Academy of Sciences, Yekaterinburg, Russia

Abstract. The purpose of this work is to display the stabilization of interaction in the form of a wave abstracting as much as possible from the values of the calculated function, changing only the parallelization parameters in order to identify general patterns. In order to accelerate stabilization balanced calculations are considered. It is a more general requirement than the uniformity of agents so the initial data distribution uniform taking into account the heterogeneity of the computing system and the task graph is chosen as the closest to the "logistic curve". Previously load balancing was considered as an automatic control problem with an obvious strategy the next task from the task graph is assigned to the processor with the lower mathematical expectation of the processor load. This is a standard interpolation (smoothing) problem. It is worth noting that not only the Dirichlet problem was considered but also partially, the Poisson problem since in practice high sensitivity to the coefficient associated with the processing of the message queue was noted. Of greatest theoretical and practical interest is the problem of determining the conditions under which the interaction stabilization wave will be close to an autowave. This formulation of the problem is justified. It is known that autowaves namely spiral waves arise in the Belousov-Zhibotinsky chemical reaction, and that a similar process is to some extent described by the Lotka-Volterra model. It turns out that by parallelizing the solution, for example of a system of two linear equations, it is the predator-prey model without taking into account the nonlinear component, interaction we will

automatically obtain a numerical solution of the stochastic Lotka-Volterra model since parallel calculations always have a random component and interaction, and extended volumes correspond, for example to a line of processors. In practice, it is possible to consider not only a processor line, but also a processor grid and various interaction schemes, among which “non-deterministic” it is without synchronization, on an infinite horizon stationary processes are of particular interest it is assumed that the parallelization scheme is clearly independent of time. In addition, pseudo-random noise can be introduced into parallel computations, for example a delay of some processor in time or coordinate, including through parameter control in the online visualization mode.

In theoretical terms, it is important that the optimization problem of maximization of entropy, that is parallel computing performance is reduced to a parameterized model of a white-noise random process, and that the view (wave) is constructed taking this model into account.

I would also like to discuss the problem of parallel data filtering that is a special case of online visualization relevant when working with big data in the context of the Kantorovich problem or transport of measures and the principle of dynamic programming.

Keywords: parallel computing, interaction stabilization, autowaves, Lotka-Volterra stochastic model.

Введение

Существует такая точка зрения, что численные методы в контексте параллельных вычислений морально устарели. Возникает вопрос: как правильно распараллеливать программы? Необходимо отталкиваться не только от практики, но и развивать математическую теорию. Известно, что хорошо работают интерполяционные алгоритмы, например, аналоги метода Ньютона, в которых определяется касательная в точке (точках) траектории (касательное расслоение). Таким образом, желательно для распараллеливания сразу выбирать асимптотически устойчивый алгоритм. С другой стороны, любая параллельная программа, если она масштабируется, является энтропийноустойчивой. Можно привести и другой пример. Так, значения решения последовательной программы методом Рунге-Кутты на тех же данных будут отличаться по точности порядка три вторых [1] от параллельного вращения, которое всегда имеет случайную составляющую. Получается так, что формализация параллельных вычислений хорошо ложится на стохастические дифференциальные уравнения (СДУ) [2]. Особый интерес для исследований представляют установившиеся возмущенные процессы (стабилизирующиеся). Так, в данной работе представлены практические и теоретические исследования по «автоволнам» в параллельных вычислениях.

В принципе, идея рассмотрения параллельных вычислений с точки зрения динамических систем не такая уж новая: аналогичный подход используется, например, для клеточных автоматов, кроме того актуальны исследования для мультиагентных систем, в которых требуется однотипность агентов, например, агентов программирования. Стоит подчеркнуть, что в данной работе декларируется более общее требование – балансировка нагрузки для графа задач.

С точки зрения визуализации задача заключается в том, чтобы нарисовать эти волны в зависимости, в первую очередь, от схемы распараллеливания, а также числовых параметров: количества процессов - p и количества данных - N для некоторых тестовых примеров задач. Формально такой перебор можно рассматривать как решение задачи методом Монте-Карло с выборкой по значимости, для которой выборка осуществляется на основе экспертного подхода. Причем геометрическое распараллеливание приводит к рассмотрению смещенного распределения (блочный подход, в котором с блоком ассоциируется процессор). Для начала надо разработать адекватный вид отображения, далее вид отображения, сопряженный с математической моделью (образ меры).

Постановка задачи

Цель данной работы состоит в том, чтобы отобразить (нарисовать) стабилизацию взаимодействия в виде волны, максимально абстрагируясь от значений вычисляемой функции (меняя только параметры распараллеливания) с целью выявления общих закономерностей. Наибольший интерес в теоретическом и практическом плане представляет задача определения условий, при выполнении которых волна стабилизации взаимодействия будет близка к автоволне. Вероятно, что триггерные автоволны могут возникнуть только в случае недетерминированных процессов на решетке процессоров.

Теория

1. Автоволны

Общее определение: автоволны представляют собой самоподдерживающиеся сигналы, которые индуцируют процессы локального высвобождения запасенной в среде энергии, затрачивающейся на выпуск аналогичных процессов в соседних областях. Параллельные (распределенные) вычисления можно интерпретировать как сигнальные графы (частный случай СДУ) [2], тогда передача информации (сообщений), рассматриваемой как кинетическая энергия [3], обеспечивает запуск аналогичных процессов на процессорах, которым предназначено это сообщение. Таким образом, стабилизацию взаимодействия можно рассматривать как триггерную автоволну, что и является основной целью данной работы. При изучении фазовых автоволн необходимо учитывать значения вычисляемой функции, хотя эта задача оставлена на будущее, но в разрабатываемый вид отображения заложена возможность изучения фазовых автоволн.

Еще одно важное определение. Ревербератор – это источник волн: он может существовать в ждущей среде, в которой нет автоколебательных элементов. Кроме того, поскольку он не связан с геометрией среды, то может мигрировать по ней. Начальное значение процессора $-p_0$, с которого начинается вычисление, так же как и номер процессора, на котором заданы псевдослучайные помехи, будем рассматривать как ревербератор.

2. Вид отображения

С целью масштабируемости изображения используются безразмерные переменные. Областью определения является распределение данных по процессорам (в случае линейки процессоров одна координата, в случае решетки процессоров – две). Областью значений является номер итерации при фиксированном астрономическом времени $I(t)$ соответственно по процессорам, также рассмотрен другой вариант с целью получения кусочно-непрерывной выпуклой функции: обратное значение к номеру итерации $-1/I(t)$, которое можно рассматривать как $1/T$. Значение вычисляемой функции отображается цветом, например, градиентная раскраска. Изменение астрономического времени дает волну. Естественно, значение вычисляемой функции на фронте волны имеет максимальную яркость. Стабилизации взаимодействия соответствует логистическая кривая: прямая линия (в случае линейки процессоров) или плоскость. Графики волны строятся в двух вариантах: линейная аппроксимация (рис. 1) и интеграл Стратоновича (столбчатая диаграмма). Примеры представлены в практической части в виде видео.

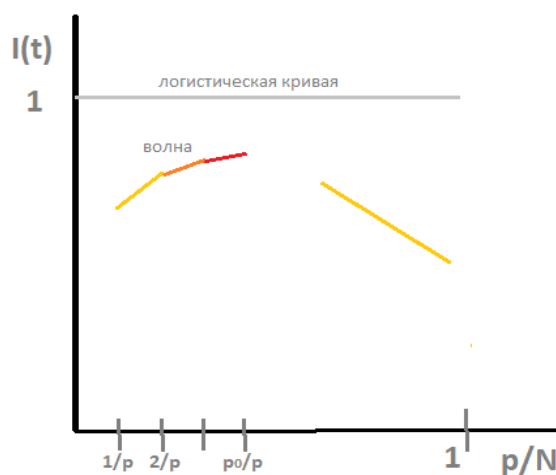


Рис. 1. Волна стабилизации взаимодействия для линейки процессоров

Основополагающей гипотезой является следующая: установившиеся триггерные автоволны могут возникнуть только в случае асинхронных итераций; в отличие от параллельных вычислений последовательные вычисления всегда синхронны, возможны только незначительные периодические колебания, взаимодействие всегда стабилизируется. В целях отладки вида отображения и более

простой асинхронной реализации рассмотрены, в первую очередь, итерации по пространственным координатам. В частности, блочное умножение матриц при количестве процессоров (блоков), стремящихся к бесконечности до этапа синхронизации. Реализация недетерминированных стационарных процессов не является стандартной схемой распараллеливания, но в принципе возможна. Приведем такой пример. Рассмотрим распараллеливание на решетке процессоров разностной схемы типа крест с использованием обещаний, рассматриваемых как условие слабой опережаемости. Предполагается, что значение вычисляемой функции явно не зависит от времени, тогда область определения функции на каждом процессоре сужается по итерациям по времени в виде пирамиды, а при получении ожидаемого сообщения область определения доопределяется. Таким образом, на каждом процессоре возможны более длительные вычисления, чем в стандартной схеме распараллеливания с синхронизацией по временному шагу, и, следовательно, средние значения времени на процессорах могут сильно отличаться. Реализации подобных схем распараллеливания будем называть сильной асинхронностью. В принципе такую сужающуюся область определения можно рассматривать как грубое множество, но основная цель – математическое моделирование; выбрав СДУ, будем придерживаться стохастической семантики [2].

3. Стохастическая модель Лотки-Вольтера

Известно, что автоволны, а именно спиральные волны, возникают в химической реакции Белоусова-Жиботинского, а также что подобный процесс в некоторой степени описывается моделью Лотки-Вольтера. Выходит, что, распараллеливая решение, например, системы двух линейных уравнений (модель хищник-жертва без учета нелинейной составляющей (взаимодействия)), автоматически получим численное решение стохастической модели Лотки-Вольтера [1], поскольку в параллельных вычислениях всегда имеются случайная составляющая и взаимодействие, а протяженные объемы соответствуют, например, линейке процессоров. На практике можно рассмотреть не только линейку процессоров, но и решетку процессоров (динамическая 3D-визуализация), и разные схемы взаимодействия, среди которых особый интерес представляют «недетерминированные» (без синхронизации, на бесконечном горизонте) стационарные процессы.

Стохастическая модель Лотки-Вольтера [1] имеет следующий вид:

$$\frac{d}{dt}x^{(i)}(t) = x^{(i)}(t) \left(\gamma_i + \sum_{j=1}^p \nu_{ij} x^{(j)}(t) \right) + \sigma_i x^{(i)} \xi^{(i)}(t),$$

где $x^{(i)}$ – численность i -й популяции или в i -м объеме (в параллельных вычислениях на i -м процессоре); взаимодействие $x^{(i)}x^{(j)}$ – передача сообщений между соответствующими процессорами; коэффициенты ν_{ij} определяют матрицу обмена сообщениями (матрицу смежности произвольного ранга, поскольку количество процессоров p является переменным параметром). Оправданность стохастических моделей заключается в том, что, например, для химических реакций, протекающих в протяженных объемах, значительную роль играет диффузия вещества. А в последовательном решении системы двух линейных уравнений (или уравнений первого порядка (модель хищник-жертва)) возможны только периодические колебания, в параллельных вычислениях существенную роль играет явление диффузии.

4. Параметризованная модель белошумного случайного процесса

Система дифференциальных уравнений для параметризованной модели белошумного случайного процесса имеет вид [1]

$$\dot{x}_t = a(x_t, u(t), t) + \sum (x_t, t) \frac{1}{\mu} \xi_{\frac{1}{\mu^2}}, \quad x_0 = x(0).$$

Малый параметр μ стремится к нулю. Стоит отметить, что случайная составляющая имеет вид $\xi_{\frac{1}{\mu^2}}$,

таким образом, дисперсия установившегося возмущенного (стабилизирующегося) процесса стремится к нулю, в то время как дисперсия «наблюдаемого» процесса (без квадрата) не ограничена. Наблюдаемый процесс, а именно задача визуализации цифровой модели поверхности (ЦМП), которая

является стационарной линейной, был рассмотрен и визуализирован в работе [2]. В настоящей статье рассматривается стабилизация взаимодействия в параллельных вычислениях.

Вышеприведённая система дифференциальных уравнений была получена в контексте СДУ Р. Л. Стратоновичем. В работе [1] установившиеся возмущенные процессы сначала рассматриваются для фундаментальных последовательностей (сходимость по вероятностной мере) и дробно-рациональных функций (в качестве случайной составляющей можно рассматривать остаток любого сходящегося ряда, далее приведены типовые примеры). Влиянием малого параметра активно занимался Л. С. Понтрягин, этой тематике посвящена его первая работа с соавторами [4], в которой рассмотрены стационарные процессы с возмущениями (толчками), аналогично в среднеквадратическом смысле.

С точки зрения вариационного исчисления не имеет значения, по какому именно малому параметру дифференцировать, довольно часто в качестве малого параметра рассматривается $1/T$, например, в теореме о магистралях. В параллельных вычислениях это функция энтропии (малый параметр) $p/N(p)$, которая является гармонической (свойство среднего) и аналитической, что важно в теоретическом плане.

В случае геометрического распараллеливания вероятностное пространство определено следующим образом: N – это мощность, множество Ω , p – это количество непересекающихся под- σ -алгебр \mathcal{F}_p (обозначение p является аббревиатурой от слова процессор (процесс), обычно в качестве непересекающихся под- σ -алгебр рассматривается декартово произведение подпространств, то есть это обозначение выбрано по аналогии с L_p), определена (редуцированная) геометрическая вероятность через отношение объемов (разной размерности).

Числовой параметр p аналогичен времени в смысле вариационного исчисления: $\dot{x}_p = dl/dp$, где l – длина сообщения. Важно подчеркнуть, что длина сообщения считается не в количестве байт, а в количестве буферов, так высокая латентность коротких сообщений обусловлена слишком большими помехами.

В качестве случайной составляющей можно рассматривать остаток любого сходящегося ряда. (Налицо явное отставание теории от практики, в которой преобладает применение всплесков для двумерных сигналов (изображений) и методов Монте-Карло, а не, например, унифицированного разложения Тейлора или преобразования Фурье). Если процесс по отношению к системе не является белым шумом, то рекомендуется сузить ширину пропускного канала (доверительный интервал), что можно сделать двумя способами: увеличить степень аппроксимации или увеличить количество данных. С учетом параметра N (разбиение по времени) в СДУ были ведены смешанные интегралы Стратоновича. Важно, что при доказательстве сильной сходимости численного решения СДУ интеграл, зависящий от N , должен быть мартингалом (дифференцируемость в среднеквадратическом смысле). (В том числе, можно рассматривать не стандартные кратные (по пространственным переменным) смешанные интегралы Стратоновича, а следовательно, не стандартные матрицы произвольного ранга – $N = 2N$ (для уравнений в частных производных по каждой размерности), а не $N = N + 1$, что связано с практической ориентацией на k -деревья и всплески.) В работе [2] предложена формула Ито для фильтра сечения плоскостью:

$$f'(B_N) = (\vec{n}, \nabla \prod x_i)|_{x_i=N} + N.$$

Соответственно для интеграла Стратоновича $ds = 2N$.

Теорема. Пусть $f(N, \vec{x})$ – возмущенный случайный процесс (Ито) по количеству данных. Если процесс Ито является мартингалом, то для любого фильтра $\varphi \in C^2$, дважды непрерывно дифференцируемого, $\varphi f(N, \vec{x})$ является мартингалом.

Доказательство этой теоремы стандартное, если фильтр сечения плоскостью рассматривать как производящий оператор [5]. На рисунке 2 представлена геометрическая интерпретация формулы Ито для фильтра сечения плоскостью (вид спереди). Малый параметр Θ отвечает за угол наклона секущей плоскости (фильтра) относительно регулярной сетки $0 < \Theta < \Delta z$. Если посчитать количество данных на секущей плоскости (проекции ближайших точек), то оно будет соответствовать вышеприведённой формуле. Если рассмотреть другую окрестность, через середину которой проходит секущая плоскость, то соответственно получится интеграл Стратоновича.



Рис. 2. Геометрическая интерпретация формулы Ито для фильтра сечения плоскостью

Важно, что суперпозиция функций $\varphi f(N, \vec{x})$, реализованная как пересечение и проекция, при дифференцировании ведет к интегрированию по частям (свертке). Возможно, в качестве φ (функции многих переменных) можно рассматривать абсолютно непрерывную функцию в сочетании с условием Лифшица, тогда в контексте СДУ необходимо применение формулы Гирсанова.

Задача параллельной фильтрации данных возникает в том случае, когда данные, рассчитанные на вычислителе при фиксированном значении времени, невозможно передать в полном объеме на рабочую станцию (например, они не входят в память видеокарты) и сжатие данных не решает поставленную проблему или ведет к значительной потере точности. Таким образом, невозможно построить отображение вида $R^3 \times \{t > 0\} \rightarrow R^3$. Наибольший интерес представляет однопараметрическая фильтрация данных, например, сечения плоскостью (сферой, цилиндрами) или изоповерхности, сокращающая объем данных в размерность (принцип динамического программирования). В итоге надо предложить ε -проксимальную стратегию (расставить контрольные точки в зависимости от скорости анализа информации на клиенте). Задача параллельной фильтрации данных напоминает задачу транспорта мер (перемещение данных из одной области в другую с минимальной работой). На вычислителе фильтрация данных реализована как реактивное вычисление (выборка), что вселяет надежду на то, что затраты (помехи) на ее реализацию будут незначительны по отношению к основным вычислениям. Для наглядности будем рассматривать трехмерную регулярную сетку со скалярным значением функции в узлах. С целью сокращения длины сообщения передаются только значения функции, а координаты заново пересчитываются. Визуализация осуществляется сопоставимо с режимом реального времени, что ведет к ограничению на длину передаваемого сообщения. Если длина сообщения все равно большая, можно дополнительно использовать сжатие, в том числе разложение в ряд. Изменение параметра ведет к отображению другой плоскости. Вообще, не хотелось бы использовать полный перебор по параметру, в этом случае напрашивается применение методов Монте-Карло, в зависимости от задачи возможно решение некоторой оптимизационной задачи, например, отображение поверхностей экстремальных кривых.

Итак, с учетом итераций по времени надо построить отображение, абсолютно-непрерывную функцию (фильтр) $\phi: R^3 \times \{t > 0\} \rightarrow R^2$. Если бы эта функция имела вид $\phi(x, t): R^3 \times \{t > 0\} \rightarrow R^1$, то это была бы стохастическая функция Ляпунова [1], дважды непрерывно дифференцируемая по x и один раз по t , применяемая для решения задач стохастического оптимального управления. Возможно, решение поставленной задачи связано с рассмотрением задачи транспорта мер, не обязательно в контексте СДУ, другой вариант – функциональная сходимости для методов Монте-Карло.

В заключение теоретической части стоит привести простую теорему [4] в плане доказательства (дерево выбора ограничено сверху k -деревом), но важную в применении, в теории информации получившую название – оптимальное кодирование дешифруемых кодов Крафта.

Теорема 2.3. Можно указать такой способ кодирования (передачи) равномерных независимых сообщений, что

$$l_{cp} < \frac{H_{\xi}}{\ln D} + 1,$$

где H_{ξ} – средняя энтропия (количество информации).

В нашей интерпретации l_{cp} – средняя длина пути между узлами вычислителя, D – количество каналов процессора (в оригинале количество букв в алфавите). Приведем классификацию каналов по значению в знаменателе. В случае обмена сообщениями через общую шину (гауссовские каналы) можно считать, что $D = \infty$, следовательно, средняя длина пути между узлами вычислителя равна единице, что соответствует экспертному подходу. Причем гауссовские каналы с условием равновероятности сообщений дадут недетерминированные процессы [2]. $D = 2$ – двоичные каналы. $D = 1$ – узкое место – вырожденный случай, необходимо рассматривать задачу стохастического оптимального управления.

Рассмотрение задачи Канторовича (транспорт мер) и принципа динамического программирования актуально не только для задачи фильтрации данных, но и для изучения автоволн на решетке процессоров. В принципе нейронные сети можно рассматривать как конвейер фильтров, где размерности подпространств соответствует количество персептронов в слое. Для обучения нейронных сетей проблематично предложить равномерное распределение по данным, поэтому актуально рассмотрение устойчивости для методов Монте-Карло. Аналогично методы эластичных сетей (карт [6]) можно рассматривать как задачу Канторовича в сочетании с принципом динамического программирования, поскольку они имеют квадратичную часть (регуляризация Тихонова), линейную часть (метод лассо) и аддитивные помехи, связанные с понижением размерности.

Результаты экспериментов

Практическая часть находится в стадии реализации. Будем надеяться, что удастся получить визуализацию волн для блочного умножения матриц при количестве процессоров (блоков), стремящихся к бесконечности до этапа синхронизации. Поскольку рассматриваются не типовые схемы распараллеливания (в частности, применение обещаний), вероятны трудности в плане программирования.

Обсуждение результатов

Известно, что автоволны, а именно спиральные волны, возникают в химической реакции Белоусова-Жиботинского. Подобный процесс в некоторой степени описывается моделью Лотки-Вольтера. Эта модель при рассмотрении параллельных вычислений (в частности, триггерных автоволн) является не полной, поскольку представляет интерес отображение данных на решетку процессоров, следовательно, надо рассматривать уравнение в частных производных (возможно, транспорт мер), поскольку задача имеет физический смысл, должен выполняться принцип наименьшего действия (полный дифференциал по количеству процессоров должен быть равен нулю). Возможно рассмотрение и других подходов, например уравнения неразрывности, адекватность которых неочевидна. Поэтому авторы намерены отталкиваться от распараллеливания конкретных задач, что ведет к рассмотрению функциональных уравнений. Хотелось бы начать обсуждение с практических результатов.

Заключение

Эта работа является исследовательской, и делать какие-то выводы преждевременно. Анализ результатов показывает, что при распараллеливании намеренно допускается логическая ошибка: недетерминированные процессы сводятся к детерминированным. Именно моделирование и изучение недетерминированных процессов представляет интерес. Хотелось бы получить ответы на ряд вопросов:

1. Бывают ли стабилизирующие недетерминированные процессы?
2. Как недетерминизм повлияет на производительность?
3. Верна ли гипотеза, что триггерные автоволны могут возникнуть только в случае недетерминированных процессов?

Актуальным направлением является создание виртуального испытательного стенда (цифровых двойников). Вряд ли возможно свести, например, рой роботов к детерминированным процессам, поскольку чем больше количество роботов, тем больше время синхронизации, в результате во время синхронизации все роботы должны стоять на месте или смещаться с малым приращением.

Список литературы

1. Кузнецов Д.Ф. Численное моделирование стохастических дифференциальных уравнений и стохастических интегралов. Санкт-Петербург: Наука, 1999. 458 с.
2. Manakov D.V., Vasev P.A. Stochastic semantics of big data (parallel computing and visualization) // Scientific visualization. 2024. Quarter 4. Vol. 16, no. 5. P. 120-150. DOI: 10.26583/sv.16.5.09
3. Стратонович Р.Л. Теория информации. М.: Сов. радио, 1975. 424 с.
4. Андронов А.А., Витт А.А., Понтрягин Л.С. О статистическом рассмотрении динамических систем // ЖЭТФ. 1933. 1.3, 3.
5. Oksendal B. Stochastic differential equations: an introduction with applications. Berlin: Springer, 2003. ISBN 3-540-04758-1.
6. Бондарев А.Е., Галактионов В.А., Шапиро Л.З. Визуальный анализ и обработка многомерных данных // GraphiCon 2017: 27-я Международная конференция по компьютерной графике и машинному зрению: труды конференции. Пермь: ПГНИУ, 2017. С. 103-107.