

УДК 51.007.004

А.А. Копыльцов  
Санкт-Петербург, Россия

## ОБОБЩЕННЫЙ АЛГОРИТМ ОБРАБОТКИ СЛАБО ФОРМАЛИЗОВАННОЙ ИНФОРМАЦИИ И ЕГО ПРИЛОЖЕНИЯ

**Аннотация.** Предлагается применение обобщенного алгоритма обработки слабо формализованной информации, поступающей от датчиков, для обеспечения безопасного движения автотранспорта. Под слабо формализованной понимается информация, которая не может быть загружена на компьютер и/или не может обрабатываться на компьютере без предварительной обработки, которая и называется формализацией информации. Поступающие с датчиков данные часто являются слабо формализованной информацией. Требования, предъявляемые к системам обработки подобной информации, следующие: системы обработки слабо формализованной информации должны из слабо формализованной информации, путем ее переработки, получать формализованную информацию, которую можно обрабатывать на компьютере. Для обработки слабо формализованной информации предложен обобщенный алгоритм, который может быть использован для обработки информации в различных технических системах. Одним из таких приложений может быть безопасное движение автотранспорта. Предполагается, что на автомобилях установлены датчики в различных диапазонах (например, видеодатчики, звуковые и ультразвуковые датчики), с которых поступает информация о препятствиях, находящихся на пути движения автомобиля (постройках, деревьях, других транспортных средствах, людях, животных и т.д.). Недостаток информации может проявляться в том, что одно препятствие может заслонять другое, например, человек может находиться за машиной вне зоны видимости. Задача состоит в том, чтобы не произошло столкновения машины с каким-либо препятствием (другая машина, здание, человек, животное и др.). Для этого существуют два параметра: скорость и направление движения автомобиля. В случае уменьшения расстояния между автомобилем и другим автомобилем или каким-либо препятствием, автомобиль может изменить либо направление движения, либо скорость. Изменяя скорость и направление движения, можно добиться безопасного движения автомобиля. В качестве примера рассмотрено движение машин через Т-образный перекресток. Показано, что применение обобщенного алгоритма позволяет повысить безопасность движения автотранспорта.

**Ключевые слова:** слабо формализованная информация; алгоритм; обработка информации; безопасность движения автотранспорта.

**Сведения об авторе:** Антон Александрович Копыльцов, старший преподаватель кафедры автоматизированных систем обработки информации и управления СПб ГЭТУ «ЛЭТИ».

**Место работы:** Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина) (СПб ГЭТУ «ЛЭТИ»).

**Контактная информация:** 197198, г. Санкт-Петербург, ул. Лизы Чайкиной, д. 18, кв. 8; тел: 8-921-4019427. E-mail: kopyl2001@mail.ru.

### Введение

В последнее время много работ посвящено обработке слабо формализованной информации [3–22]. Под слабо формализованной понимается информация, которая не может быть загружена на компьютер и/или не может обрабатываться на компьютере без предварительной обработки, которая и называется формализацией информации [2]. Поступающие с датчиков данные часто являются слабо формализованной информацией. Требования, предъявляемые к системам обработки подобной информации, следующие: системы обработки слабо формализованной информации должны из сла-

бо формализованной информации, путем ее переработки, получать формализованную информацию, которую можно обрабатывать на компьютере. Для обработки слабо формализованной информации предложен обобщенный алгоритм [4; 6; 7; 10], который может быть использован для обработки информации в различных технических системах. Одним из таких приложений может быть безопасное движение автотранспорта.

### Безопасное движение автотранспорта

Предполагается, что на автомобилях установлены датчики в различных диапазо-

нах (например, видеодатчики, звуковые и ультразвуковые датчики). Если погода хорошая, то видеодатчики позволяют хорошо наблюдать обстановку в окружении автомобиля. Если же погода плохая (туман, снег, дождь и т.д.), то видеодатчики с этой задачей могут не справиться, а совместная работа видео- и аудиодатчиков может привести к желаемому результату. С этих датчиков поступает информация о препятствиях, которые находятся на пути движения автомобиля (постройках, деревьях, других транспортных средствах, людях, животных и т.д.). Недостаток информации может проявляться в том, что одно препятствие может заслонять другое, например, человек может находиться за машиной вне зоны видимости.

Задача состоит в том, чтобы не произошло столкновения машины с каким-нибудь препятствием (другая машина, здание, человек, животное и др.). Для этого существуют два параметра: скорость и направление движения автомобиля. В случае уменьшения расстояния между одним автомобилем и другим автомобилем или каким-либо препятствием автомобиль может изменить либо направление движения, либо скорость. Изменяя скорость и направление движения, можно достичь безопасного движения автомобиля.

### Приложения обобщенного алгоритма

Применим обобщенный алгоритм обработки слабо формализованной информации [4; 6; 7; 10].

Этап 1. Информация, которая собирается с различных датчиков, поступает в модуль «Сбор информации». В нашем случае это могут быть видео-, аудио-, ультразвуковые и другие датчики, которые установлены на машинах.

Этап 2. Информация из модуля «Сбор информации» поступает в модуль «Распознавание информации». В нашем случае машины распознаются по видео-, аудио- и ультразвуковым датчикам.

Этап 3. Из модуля «Распознавание информации» информация поступает в модуль «Классификация информации», где осуществляется ее классификация по классам ( $n$  – количество классов). В нашем случае имеем видеoinформацию (первый класс), аудиoinформацию (второй класс), ультразвуковую информацию (третий класс), а также их сочетания: видео-аудиоинформацию (четвертый класс), видео-ультразвуковую информацию (пятый класс), аудио-ультразвуковую информацию (шестой класс) и видео-аудио-ультразвуковую информацию (седьмой класс), т.е.  $n = 7$ .

Этап 4. В каждом из  $n = 7$  модулей «Свертка информации» информация проходит свою, характерную для данного класса, обработку по определенному алгоритму [1; 23; 24]. В нашем случае (например, когда рассматриваем две машины) считаем, что свертка – среднее расстояние между двумя машинами за некоторый промежуток времени. В итоге получаем для каждой двух машин среднее расстояние между ними за некоторый промежуток времени.

Этап 5. Оценивание достоверности информации в модулях «Оценивание достоверности информации». Предполагается, что ранее была проведена серия экспериментов (например, при пересечении машинами перекрестка), в результате которых было выявлено, что среднее расстояние  $H_{cpi}$  между каждыми двумя машинами

$$H_{min} \leq H_{cpi} \leq H_{max},$$

где  $i$  – номер пары машин,  $N$  – всего пар машин,  $H_{min}$ ,  $H_{max}$  – минимальное и максимальное значение (определяется на основании предыдущих экспериментов). Если  $H_{min} \leq H_{новаяi} \leq H_{max}$  для всех  $i$ , то информация достоверная. Если это верно для  $Nl$  пар машин, то достоверность равна  $Nl/N$ .

Этап 6. Оценивание безопасности информации в каждом из  $n$  классов происходит в модулях «Оценивание безопасности информации». Проверяем  $N-Nl$  пар машин,

где  $H_{mini} > H_{новая_i}$  и/или  $H_{новая_i} > H_{maxi}$ . Возможны варианты: это либо неисправность системы, или так и должно быть. Прогоняем старый тестовый вариант. Если он дает правильные результаты, то  $H_{mini}$  и/или  $H_{maxi}$  корректируются. Если же нет, то эта конкретная пара машин работает неправильно (например, водитель пьяный). Выявляем число неправильно работающих пар машин  $N_2$ , и тогда безопасность равна  $(N-N_2)/N$ .

Этап 7. Установление связей между недавно полученной информацией в каждом из  $n$  классов и прежде полученной информацией происходит в модулях «Установление связей». Новая информация – набор усредненных расстояний между парами машин в какой-то момент (промежуток) времени. Сравниваем этот набор с наборами, хранящимися в памяти, и ищем наиболее близкий (минимизируем среднее квадратичное отклонение). Если задать некоторый интервал  $\Delta$ , то в него может попасть несколько старых вариантов. Число таких вариантов (связей) пусть будет равно  $M$ .

Этап 8. В модулях «Оценивание вероятности» происходит оценивание вероятности, с которой можно доверять поступившей информации в каждом из  $n$  классов. В нашем случае: если  $M \geq M^*$ , где  $M^*$  задано, то  $P = 1$ . В противном случае переходим на этап «Распознавание информации». Пусть, например,  $M^* = 1$ , и  $M = 2$ . Тогда  $M \geq M^*$  и, следовательно,  $P = 1$ .

Этап 9. В модулях «Поддержка принятия решений» происходит поддержка принятия решений в каждом из  $n$  классов. Это происходит следующим образом. Если выполнены следующие условия:

- достоверность поступающей слабо формализованной информации  $I_{дпи} > I_{дпи}^*$  для некоторого  $I_{дпи}^*$ ;
- безопасность поступающей слабо формализованной информации  $I_{бпи} > I_{бпи}^*$  для некоторого  $I_{бпи}^*$ ;
- число связей  $M > M^*$  для некоторого  $M^*$ ;

▪ вероятность, с которой можно доверять вновь полученной слабо формализованной информации  $P > P^*$  для некоторого  $P^*$ , то эта новая поступающая информация включается в хранилище для данного класса.

Нормируем число связей  $M$  следующим образом

$$M^N = \frac{M}{M_{max}}$$

где  $M^N$  – нормированное значение  $M$ ,  $M_{max}$  – максимальное значение  $M$ .

Таким образом, достоверность поступающей информации  $I_{дпи}$ , безопасность поступающей информации  $I_{бпи}$ , нормированное число связей  $M^N$ , вероятность, с которой можно доверять вновь полученной информации  $P$ , находятся в промежутке от 0 до 1, т.е.  $0 \leq I_{дпи} \leq 1$ ,  $0 \leq I_{бпи} \leq 1$ ,  $0 \leq M^N \leq 1$ ,  $0 \leq P \leq 1$ .

Для того, чтобы новую поступающую информацию включить в модуль «Хранилище» для данного класса, нужно, чтобы выполнялись неравенства

- $I_{дпи} > I_{дпи}^*$  для некоторого  $I_{дпи}^*$ ;
- $I_{бпи} > I_{бпи}^*$  для некоторого  $I_{бпи}^*$ ;
- $M^N > M^{N*}$  для некоторого нормированного  $M^{N*}$ ;
- $P > P^*$  для некоторого  $P^*$ .

В нашем случае информацию (набор средних расстояний между парами машин в некоторый момент (промежуток) времени) включаем в модуль «Хранилище», поскольку все 4 условия выполнены.

Этап 10. В модуле «Обобщенная поддержка принятия решений» проводится сбор сгенерированных решений из  $n$  классов и после этого на их основе осуществляется генерация новой совокупности решений. В нашем случае имеем видеодатчики, аудиодатчики и ультразвуковые датчики (т.е. датчики трех видов). То есть нашу новую информацию (набор средних расстояний между парами машин в некоторый момент (промежуток) времени) включаем в модуль «Хранилище».

Этап 11. Сравнение недавно принятого решения с ранее принятыми решениями

происходит в модуле «Определение числа связей». Определение числа связей, подтверждающих правильность принятого решения, осуществляется следующим образом. При поступлении новой информации в модуль «Хранилище» осуществляется ее свертка  $C_N$  и сравнение со свертками ( $C_1, C_2, \dots, C_k$ ) ранее полученной информации, хранящимися в модуле «Хранилище», следующим образом. Если  $\delta_i = |C_N - C_i| < \varepsilon$  для некоторого  $i \in \{1, 2, \dots, k\}$  и некоторого  $\varepsilon > 0$ , то считаем, что новая информация с номером  $N$  и старая информация с номером  $i$  между собой связаны. После установления связей считаем число этих связей  $MM$ . Если  $MM > MM^*$  для некоторого  $MM^*$ , то считаем, что вероятность правильности принятого решения  $P = 1$ . В противном случае  $P < 1$ . В нашем случае (на основе предыдущего)  $P = 1$ .

Этап 12. В модуле «Выработка устойчивой реакции» происходит выработка устойчивой реакции на неоднократно поступающую информацию и ее запоминание. Если для некоторого  $\varepsilon > 0$  среднее квадратичное отклонение между новым вариантом и несколькими старыми вариантами меньше  $\varepsilon$ , то считается, что эти фрагменты информации (набор средних расстояний между парами машин в некоторый момент (интервал) времени) образуют устойчивую информацию, и их записываем в хранилище с индексом «устойчивая»  $I_{ул}$ .

Этап 13. В модуле «Генерация решений» генерируются решения. Собираем в модуле «Хранилище» все фрагменты информации с индексом «устойчивая». Рассмотрим сеть связей в модуле «Хранилище» для набора устойчивой информации  $I_{ул_i}, i=1, \dots, n$ . Если между фрагментами информации с номерами  $i$  и  $j$  имеется  $C_{ij}$  связей и  $C_{ij} > C^*$ , где  $C^*$  – некоторое число, то считаем, что фрагменты информации с номерами  $i$  и  $j$  связаны непосредственно (уровень связности  $C_{ij}$ ).

Этап 14. В модуле «Хранилище» на основе информации, которая записана по-

следней, и ранее записанной информации генерируется новая информация. Сведения о том, что фрагменты информации с номерами  $I_i$  и  $I_j$  связаны, могут рассматриваться, как новая информация с номером  $I_k$ , и ее можно направить на этап «Сбор информации» с целью распознавания, классификации информации и т.д. Это означает, что если фрагмент информации с номером  $I_i$  определяется показателями  $i_1, i_2, \dots, i_{l1}$ , а фрагмент информации с номером  $I_j$  – показателями  $j_1, j_2, \dots, j_{j1}$ , то фрагмент информации с номером  $I_k = I_{i \cup j}$  определяется показателями  $i_1, i_2, \dots, i_{l1}, j_1, j_2, \dots, j_{j1}$ , представляющими собой объединение показателей, которыми определяются фрагменты информации с номерами  $I_i$  и  $I_j$ . Это означает, что свертка в дальнейшем делается по показателям  $i_1, i_2, \dots, i_{l1}, j_1, j_2, \dots, j_{j1}$ . В нашем случае объединяем две совокупности двух экспериментов и вычисляем в каждом случае средние значения, т.е. это означает, что такой вариант средних значений тоже возможен. Это новая информация, которую можно переслать на вход в систему (в модуль «Сбор информации») и прогнать аналогично. В итоге можно вычислить среднее значение по всем парам машин, дисперсию и среднее квадратичное отклонение. Таким образом, если среднее значение достаточно мало, а дисперсия велика, то нужно изменять режим движения (например, изменять скорость при пересечении перекрестка (установить соответствующие знаки)). Через некоторый интервал времени (когда часть машин пересечет перекресток) снова вычисляем среднее значение (математическое ожидание). Снова изменяем режим движения (например, через перекресток), и т.д.

#### **Движение через Т-образный перекресток.**

Рассмотрим, например, Т-образный перекресток (рис. 1).

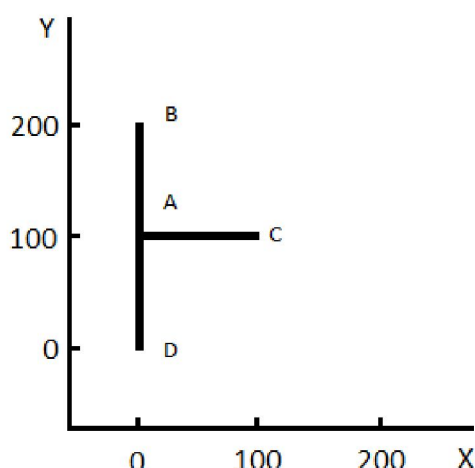


Рис. 1. Перекресток трех дорог АВ, АС, АД в системе координат (X, Y)

Если рассмотреть две машины, которые движутся из пунктов В и D через перекресток А, то возможны варианты: первая машина – маршруты ВАС и ВAD, вторая машина – маршруты DAB и DAC. Причем, если интервал времени между прохождением перекрестка больше некоторого  $\Delta$ , то они пройдут его, не изменяя скорости. Если же нет, то возможны варианты: маршруты ВAD и DAC (машины разойдутся), маршруты ВAD и DAB (машины разойдутся), маршруты ВАС и DAC (машины не разойдутся, если не изменят скорость), маршруты ВАС и DAB (машины не разойдутся, если не изменят скорость). Проводим несколько экспериментов с различными вариантами движения и запоминаем условия и результаты экспериментов (скорости и направления движения, успешно ли пересекли перекресток и т.д.). В итоге, на основе запомненной информации формируются интервалы скоростей, при которых возможны аварийные ситуации на дороге и при которых невозможны.

Если в следующий раз возникает похожая ситуация (например, к перекрестку приближаются две машины со скоростями  $V_1$ ,  $V_2$ ), то рассматриваются интервалы, выбирается наиболее близкий вариант и

генерируются рекомендации об изменении (или не изменении) скорости. В случае необходимости интервалы корректируются. Однако, хотя и генерируются рекомендации об изменении скорости, машины могут: изменить скорость на ту величину, которая рекомендуется; изменить скорость на другую величину; не изменять скорость. В любом случае в память записывается полная информация о результатах пересечения перекрестка машинами.

В качестве примера рассмотрим вариант, когда скорость первой машины  $V_1 = 10$  м/с (36 км/ч) и она движется по маршруту ВАС, а вторая машина со скоростью  $V_2$  движется по маршруту DAB. Предполагается, что  $AB = AC = AD = 100$  м. В этом случае первая машина пересечет перекресток через 10 с. Тогда, проводя эксперименты, т.е. запуская вторую машину с различными скоростями, получим следующие экспериментальные данные (табл. 1). Например, пусть первая машина движется по маршруту ВАС со скоростью 10 м/с, а вторая – по маршруту DAB со скоростью 20 м/с. Тогда координаты машин и расстояние между ними будут следующие (табл. 2, рис. 2).

Таблица 1

Скорости и время прохождения перекрестка

№	Скорость второй машины $V_2$ (м/с)	Время прохождения перекрестка А (с)	Возможность столкновения
1	2	50,00	
2	4	25,00	
3	6	16,67	
4	8	12,50	
5	10	10,00	Возможно столкновение
6	12	8,33	
7	14	7,14	
8	16	6,25	
9	18	5,55	
10	20	5,00	

Поскольку машины представляют собой не материальные точки, а имеют размеры, и в целях безопасности можно взять, например, интервал для времени  $(t_0 - t^*, t_0 + t^*)$ , в течение которого может сложиться опасная ситуация на дороге, где  $t_0$  – время пересечения курсов машин, например,  $t^* = 3$  с. Тогда интервалы опасных скоростей (при которых возможна аварийная ситуация на

дороге) будут для перекрестка А следующие:

$$t \in (10 - 3 \text{ с}; 10 + 3 \text{ с}) = (7 \text{ с}; 13 \text{ с}),$$

$$V_1 = 10 \text{ м/с}, V_2 \in (8 \text{ м/с}; 14 \text{ м/с}).$$

Это означает, что если скорость  $V_2$  принадлежит этому интервалу, то при приближении к перекрестку рекомендуется ее изменить в целях обеспечения безопасности движения.

Таблица 2

Координаты машин и расстояние между ними

Время (с)	Координаты первой машины (м)	Координаты второй машины (м)	Расстояние между машинами (м)	Комментарии
0,00	(0,0; 200,0)	(0,0; 0,0)	200,00	
5,00	(0,0; 150,0)	(0,0; 100,0)	50,00	
6,67	(0,0; 133,34)	(0,0; 133,34)	0,0	Машины встречаются
10,00	(0,0; 100,0)	(0,0; 200,0)	100,00	
15,00	(50,0; 100,0)	(0,0; 300,0)	206,16	
20,00	(100,0; 100,0)	(0,0; 400,0)	316,23	
25,00	(150,0; 100,0)	(0,0; 500,0)	427,20	

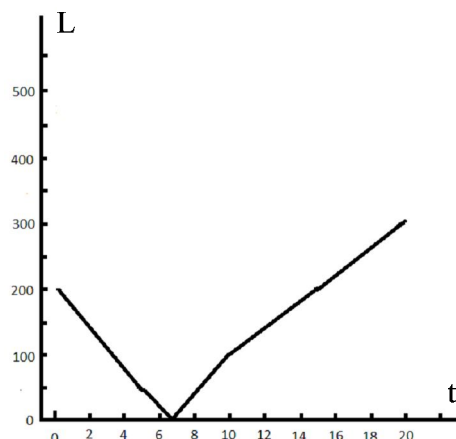


Рис. 2. Изменение расстояния (L, м) между двумя машинами с течением времени (t, с)

Рассматриваем различные варианты (проводим эксперименты) и запоминаем их. Запомненные варианты обрабатываем, т.е. получаем некоторые интервалы скоростей, при которых возможна аварийная ситуация на дороге. Если возможна аварийная ситуация, то одна из машин должна изменить скорость на некоторую величину, чтобы избежать аварии. Эту информацию можно рассматривать, как новую информацию. В случае проведения нового эксперимента, например,  $V_1 = 12$  м/с,  $V_2 = 11$  м/с, определяем, принадлежит ли скорость  $V_2$  опасному интервалу (8 м/с; 14 м/с). Поскольку  $V_2 = 11$  м/с принадлежит опасному интервалу, то не рекомендуется с такой скоростью ехать через перекресток А. То есть одна из машин должна изменить скорость при подъезде к перекрестку, чтобы избежать аварийной ситуации. Аналогично, проводя следующие эксперименты, о каждом эксперименте информацию сохраняем в памяти. В случае необходимости границы опасных интервалов корректируются.

### Заключение

Предложено применение обобщенного алгоритма обработки слабо формализованной информации к задаче безопасного движения автотранспорта, в частности, через Т-образные перекрестки. Для более сложных ситуаций, например, когда к перекрестку приближается несколько машин с различными скоростями, все делается аналогично:

– Вся информация о пересечении перекрестка сохраняется и накапливается (например, можно определять скорость машин за 100, 200, 300 и т.д. метров от перекрестка);

– На основе накопленной информации генерируются безопасные и опасные интервалы скоростей для каждой машины;

– В случае рассмотрения нового варианта пересечения перекрестка определяются интервалы скоростей, в которые попадают скорости машин, приближающихся к перекрестку;

– Если интервалы не опасные, то изменение скоростей не происходит;

– Если интервалы опасные, то в памяти осуществляется поиск ближайшего неопасного варианта, т.е. такого варианта, чтобы изменение скоростей было минимальным;

– Новый сценарий пересечения перекрестка записывается в память;

– Если необходимо, то происходит коррекция интервалов скоростей;

– Далее происходит ожидание следующего пересечения машинами перекрестка и т.д.

Таким образом, применяя обобщенный алгоритм, можно получить на выходе некоторые рекомендации по поддержке принятия решений, в частности, определить, кому, по какому маршруту и с какой скоростью перемещаться. Аналогичная ситуация наблюдается в железнодорожном, речном и морском транспорте, где также можно применять обобщенный алгоритм.

### ЛИТЕРАТУРА

1. Воробьев В.И., Копыльцов А.В., Пальчун Б.П., Юсупов Р.М. Методы и модели оценивания качества программного обеспечения. – СПб., 1992.
2. Воройский Ф.С. Информатика. Энциклопедический словарь-справочник: введение в современные информационные и телекоммуникационные технологии в терминах и фактах. – М., 2011.
3. Копыльцов А.А. Алгоритм коррекции связей между фрагментами слабо формализованной информации и генерация новой информации // Вестник Нижневартковского гос. ун-та. – 2014. – № 3. – С. 28–34.
4. Копыльцов А.А., Копыльцов А.В. Алгоритм обработки слабо формализованной информации, поступающей от технических систем // Известия СПбГЭТУ «ЛЭТИ» (известия государственного электротехнического университета). Серия «Информатика, управление и компьютерные технологии». – 2012. – № 8. – С. 30–36.

5. Копыльцов А.А., Копыльцов А.В. Качественная и количественная слабо формализованная информация и ее обработка // *Материалы XIV Санкт-Петербургской международной конференции «Региональная информатика – 2014»* (29–31 октября 2014 г.). – СПб., 2014. – С. 557.
6. Копыльцов А.А., Копыльцов А.В. Обобщенный алгоритм обработки слабо формализованной информации и его применение // *Вестник Нижневартковского гос. ун-та.* – 2014. – № 35. – С. 35–44.
7. Копыльцов А.А., Копыльцов А.В. Обработка слабо формализованной информации, поступающей от технических систем // *Вестник Нижневартковского гос. гуманитар. ун-та.* – 2013. – № 1. – С. 32–36.
8. Копыльцов А.А., Копыльцов А.В. Технические системы и слабо формализованная информация // *Материалы XIII Санкт-Петербургской международной конференции «Региональная информатика – 2012»* (24–26 октября 2012 г.). – СПб., 2012. – С. 375.
9. Копыльцов А.А., Копыльцов А.В. Цифровой образовательный ресурс «Краткая история моделирования» и его применение в учебном процессе // *Материалы сетевой международной научно-практической конференции «Электронное обучение в вузе и школе»* (16–19 апреля 2014 г.). – СПб., 2014. – С. 153–155.
10. Копыльцов А.А. Модель классификации информации и алгоритм ее предварительной обработки для статических и динамических объектов // *Известия СПбГЭТУ «ЛЭТИ»* (известия государственного электротехнического университета). Серия «Информатика, управление и компьютерные технологии». – 2013. – № 6. – С. 134–139.
11. Копыльцов А.А. Обработка информации в живых и технических системах // *Материалы XIII Санкт-Петербургской международной конференции «Региональная информатика – 2012»* (24–26 октября 2012 г.). – СПб., 2012. – С. 373–374.
12. Копыльцов А.А. Обработка слабо формализованной информации в живых и технических системах // *Материалы Всероссийской научно-практической конференции студентов, магистров, аспирантов «Современное программирование»* (16–17 апреля 2014 г.). – Нижневартковск, 2014. – С. 117–121.
13. Копыльцов А.А. Обработка слабо формализованной информации при недостатке информации // *Материалы XIII Санкт-Петербургской международной конференции «Региональная информатика – 2012»* (24–26 октября 2012 г.). – СПб., 2012. – С. 374.
14. Копыльцов А.А. Особенности обработки слабо формализованной информации // *Материалы XIV Санкт-Петербургской международной конференции «Региональная информатика – 2014»* (29–31 октября 2014 г.). – СПб., 2014. – С. 555–556.
15. Копыльцов А.А. Понятие слабо формализованной информации и особенности ее обработки // *Материалы сетевой международной научно-практической конференции «Электронное обучение в вузе и школе»* (20–24 апреля 2015 г.). – СПб., 2015. – С. 140–144.
16. Копыльцов А.А. Применение обобщенного алгоритма обработки слабо формализованной информации для управления неравновесной химической реакцией // *Инженерный вестник Дона.* – 2015. – № 1. – Ч. 2. – URL: [ivdon.ru/ru/magazine/archive/n1p2y2015/2812](http://ivdon.ru/ru/magazine/archive/n1p2y2015/2812).
17. Копыльцов А.А. Программа для обработки слабо формализованной информации, поступающей от технических систем // Свидетельство о государственной регистрации программы для ЭВМ № 2013617310. Правообладатель: ФГБОУ ВПО «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им В.И. Ульянова (Ленина)» (СПб ГЭТУ «ЛЭТИ»). Заявка № 2013615041. Дата поступления 19 июня 2013 г. Дата государственной регистрации в Реестре программ для ЭВМ 08 августа 2013 г.
18. Копыльцов А.А. Слабо формализованная информация и алгоритмы ее обработки // *Материалы XIV Санкт-Петербургской международной конференции «Региональная информатика – 2014»* (29–31 октября 2014 г.). – СПб., 2014. – С. 556–557.
19. Копыльцов А.А. Сохранение конфиденциальности данных при поддержке принятия решений, на основе извлекаемой специальным образом информации // *Материалы VIII Санкт-Петербургской межрегиональной конференции «Информационная безопасность регионов России»* (23–25 октября 2013 г.). – СПб., 2013. – С. 104.
20. Копыльцов А.А. Цифровой образовательный ресурс «Обработка слабо формализованной информации в живых и технических системах» // *Материалы сетевой международной научно-практической конференции «Электронное обучение в вузе и школе»* (16–19 апреля 2014 г.). – СПб., 2014. – С. 151–153.
21. Копыльцов А.А., Нечитайленко Р.А. Кластерное атрибутивное описание объектов информационной обработки по понятийным частным и интегральным признакам // *Материалы XII Санкт-Петербургской международной конференции «Региональная информатика – 2010»* (20–22 октября 2010 г.). – СПб., 2010. – С. 49–50.
22. Копыльцов А.А., Нечитайленко Р.А. Обеспечение пороговой безопасности обработки слабо формализованной информации при распределенном управлении с целью принятия решений // *Труды VII Санкт-Петербургской межрегиональной конференции «Информационная безопасность регионов России»* (26–28 октября 2011 г.). – СПб., 2011. – С. 154–155.



23. Копыльцов А.В. Об оценке качества программных продуктов // Проблемы информатизации (теоретический и научно-практический журнал). – 1994. – Вып. 3–4. – С. 46–48.
24. Хованов Н.В. Статистические модели теории квалиметрических шкал. – Л., 1986.

#### REFERENCES

1. Vorobjev, V.I., Kopyltsov, A.V., Palchun, B.P., Yusupov, R.M. Metody i modeli otsenivaniya kachestva programmnoy obespecheniya [Methods and models of assessing software quality]. Saint Petersburg, 1992. (In Russian).
2. Vorobjev, F.S. Informatika. Entsiklopedicheskiy slovar-spravochnik: vvedenie v sovremennyye informatsionnyye i telekommunikatsionnyye tekhnologii v terminakh i faktakh [Computer science. Encyclopedic dictionary: introduction to modern information and communication technologies in terms of facts]. Moscow, 2011. (In Russian).
3. Kopyltsov, A.A. Algoritm korrektsii svyazey mezhdru fragmentami slabo formalizovannoy informatsii i generatsiya novoy informatsii [Algorithm for correcting the links between fragments of weakly formalized data and generation of new data]// Vestnik Nizhnevartovskogo gosudarstvennogo universiteta. 2014. № 3. P. 28–34. (In Russian).
4. Kopyltsov, A.A., Kopyltsov, A.V. Algoritm obrabotki slabo formalizovannoy informatsii, postupayushchey ot tekhnicheskikh sistem [The algorithm for processing weakly formalized information from technical systems]// Izvestiya SPbGETU «LETI» (Izvestiya gosudarstvennogo elektrotekhnicheskogo universiteta). Informatika, upravlenie i kompyuternyye tekhnologii. 2012. № 8. P. 30–36. (In Russian).
5. Kopyltsov, A.A., Kopyltsov, A.V. Kachestvennaya i kolichestvennaya slabo formalizovannaya informatsiya i ee obrabotka [Qualitative and quantitative weakly formalized information and its processing]// Materialy XIV Sankt-Peterburgskoy mezhdunarodnoy konferentsii “Regionalnaya informatika – 2014” (29–31 oktyabrya 2014 g.). Saint Petersburg, 2014. P. 557. (In Russian).
6. Kopyltsov, A.A., Kopyltsov, A.V. Obobshchennyy algoritm obrabotki slabo formalizovannoy informatsii i ego primeneniye [Generalized algorithm of processing weakly formalized information and application of such algorithm]// Vestnik Nizhnevartovskogo gosudarstvennogo universiteta. 2014. № 35. P. 35–44. (In Russian).
7. Kopyltsov, A.A., Kopyltsov, A.V. Obrabotka slabo formalizovannoy informatsii, postupayushchey ot tekhnicheskikh sistem [Processing of weakly formalized information from the technical systems]// Vestnik Nizhnevartovskogo gosudarstvennogo gumanitarnogo universiteta. 2013. № 1. P. 32–36. (In Russian).
8. Kopyltsov, A.A., Kopyltsov, A.V. Tekhnicheskiye sistemy i slabo formalizovannaya informatsiya [Technical systems and weakly formalized information]// Materialy XIII Sankt-Peterburgskoy mezhdunarodnoy konferentsii «Regionalnaya informatika – 2012» (24–26 oktyabrya 2012 g.) Saint Petersburg, 2012. P. 375. (In Russian).
9. Kopyltsov, A.A., Kopyltsov, A.V. Tsifrovoy obrazovatelnyy resurs «Kratkaya istoriya modelirovaniya» i ego primeneniye v uchebnoy protsesse [Digital educational resource Brief History of Modeling and use of this resource in the educational process]// Materialy setevoy mezhdunarodnoy nauchno-prakticheskoy konferentsii «Elektronnoye obucheniye v VUze i shkole» (16–19 aprelya 2014 g.) Saint Petersburg, 2014. P. 153–155. (In Russian).
10. Kopyltsov, A.A. Model klassifikatsii informatsii i algoritm ee predvaritelnoy obrabotki dlya staticheskikh i dinamicheskikh objektov [Information classification models and its algorithm of preprocessing information for static and dynamic objects]// Izvestiya SPbGETU «LETI» (izvestiya gosudarstvennogo elektrotekhnicheskogo universiteta). Seriya «Informatika, upravlenie i kompyuternyye tekhnologii». 2013. № 6. P. 134–139. (In Russian).
11. Kopyltsov A.A. Obrabotka informatsii v zhivykh i tekhnicheskikh sistemakh [Processing information in living and technical systems]// Materialy XII Sankt-Peterburgskoy mezhdunarodnoy konferentsii «Regionalnaya informatika – 2012» (24–26 oktyabrya 2012 g.). Saint Petersburg, 2012. P. 373–374. (In Russian).
12. Kopyltsov, A.A. Obrabotka slabo formalizovannoy informatsii v zhivykh i tekhnicheskikh sistemakh [Processing weakly formalized information in living and technical systems]// Materialy Vserossiyskoy nauchno-prakticheskoy konferentsii studentov, magistrrov, aspirantov «Sovremennoye programmirovaniye» (16–17 aprelya 2014 g.) Nizhnevartovsk, 2014. P. 117–121. (In Russian).
13. Kopyltsov A.A. Obrabotka slabo formalizovannoy informatsii pri nedostatke informatsii [Processing weakly formalized information with lack of information] // Materialy XII Sankt-Peterburgskoy mezhdunarodnoy konferentsii «Regionalnaya informatika – 2012» (24 – 26 oktyabrja 2012 g.) Saint Petersburg, 2012. P. 374. (In Russian).
14. Kopyltsov, A.A. Osobennosti obrabotki slabo formalizovannoy informatsii [Peculiarities of processing weakly formalized information]// Materialy XIV Sankt-Peterburgskoy mezhdunarodnoy konferentsii «Regionalnaya informatika – 2014» (29 – 31 oktyabrya 2014 g.) Saint Petersburg, 2014. P. 555–556. (In Russian).
15. Kopyltsov, A.A. Poyatiye slabo formalizovannoy informatsii i osobennosti ee obrabotki [Weakly formalized information: concept and processing peculiarities]// Materialy setevoy mezhdunarodnoy nauchno-

prakticheskoy konferentsii «Elektronnoe obuchenie v VUZe i shkole» (20–24 aprelya 2015 g.) Saint Petersburg, 2015. P. 140–144. (In Russian).

16. Kopyltsov, A.A. Primenenie obobshchennogo algoritma obrabotki slabo formalizovannoy informatsii dlya upravleniya neravnovesnoy khimicheskoy reaktsiy [Generalized algorithm of processing weakly formalized information applied to control a nonequilibrium chemical reaction]// Inzhenerny vestnik Dona. 2015. № 1. P. 2. URL: ivdon.ru/ru/magazine/archive/n1p2y2015/2812. (In Russian).

17. Kopyltsov, A.A. Programma dlya obrabotki slabo formalizovannoy informatsii, postupajushchey ot tekhnicheskikh sistem [A program for processing weakly formalized information from technical systems]// Svidetelstvo o gosudarstvennoy registratsii programmy dlya EVM № 2013617310. Pravoobladatel: FGBOU VPO «Sankt-Peterburgskiy gosudarstvenny elektrotehnicheskij universitet «LETI» im V.I. Ulyanova (Lenina)» (SPb GETU «LETI»). Zayavka № 2013615041. Data postupleniya 19 iyunya 2013 g. Data gosudarstvennoy registratsii v Reestre programm dlya EVM 08 avgusta 2013 g. [Software Application State Registration Certificate № 2013617310] (In Russian).

18. Kopyltsov, A.A. Slabo formalizovannaya informatsiya i algoritmy ee obrabotki [Weakly formalized information and processing weakly formalized information]// Materialy XIV Sankt-Peterburgskoy mezhdunarodnoy konferentsii «Regionalnaja informatika – 2014» (29–31 oktyabrya 2014 g.) Saint Petersburg, 2014. P. 556–557. (In Russian).

19. Kopyltsov, A.A. Sokhranenie konfidentsialnosti dannykh pri podderzhke prinyatiya resheniy, na osnove izvlekaemoy spetsialnym obrazom informatsii [Maintaining data privacy with the support of decision-making, based on the specially extracted information]// Materialy VIII Sankt-Peterburgskoy mezhhregionalnoy konferentsii «Informatsionnaya bezopasnost regionov Rossii» (23–25 oktuyabrya 2013 g.) Saint Petersburg, 2013. P. 104. (In Russian).

20. Kopyltsov, A.A. Tsifrovoy obrazovatelny resurs «Obrabotka slabo formalizovannoy informatsii v zhivyykh i tekhnicheskikh sistemakh» [Digital educational resource Processing Weakly Formalized Information in Living and Technical Systems]// Materialy setevoy mezhdunarodnoy nauchno-prakticheskoy konferentsii «Elektronnoe obuchenie v VUZe i shkole» (16–19 aprelya 2014 g.) Saint Petersburg, 2014. P. 151–153. (In Russian).

21. Kopyltsov, A.A., Nechitajlenko, R.A. Klasternoe atributirovanie objektov informatsionnoy obrabotki po ponyatijnyim chastnym i integralnym priznakam [Cluster attribution of information processing objects in terms of private and integral conceptual dimensions]// Materialy XII Sankt-Peterburgskoy mezhdunarodnoy konferentsii «Regionalnaja informatika – 2010» (20–22 oktyabrya 2010 g.) Saint Petersburg, 2010. P. 49–50. (In Russian).

22. Kopyltsov, A.A., Nechitajlenko, R.A. Obespechenie porogovoy bezopasnosti obrabotki slabo formalizovannoy informatsii pri raspredelennom upravlenii s tseliyu prinyatiya reshenij [Ensuring threshold safety of processing weakly formalized information in the distributed control to provide for decision-making]// Trudy VII Sankt-Peterburgskoy mezhhregionalnoy konferentsii «Informatsionnaya bezopasnost regionov Rossii» (26–28 oktyabrya 2011 g.) Saint Petersburg, 2011. P. 154–155. (In Russian).

23. Kopyltsov, A.V. Ob otsenke kachestva programmnykh produktov [Assessing the quality of software products]// Problemy informatizatsii (teoreticheskij i nauchno-prakticheskij zhurnal).1994. Issue. 3–4. P. 46–48. (In Russian).

24. Khovanov, N.V. Statisticheskie modeli teorii kvalimetricheskikh shkal [Statistical models of the qualitative scales theory]. Leningrad, 1986. (In Russian).

*A.A. Kopyltsov*  
*Saint Petersburg, Russia*

## GENERALIZED ALGORITHM FOR PROCESSING WEAKLY FORMALIZED INFORMATION AND APPLICATIONS OF SUCH ALGORITHM

**Abstract.** This paper offers applications of the generalized algorithm for processing weakly formalized information from sensors to ensure safe road traffic. Weakly formalized information is information that cannot be loaded on the computer and/or cannot be processed on the computer without preliminary processing, namely, information formalization. The data arriving from sensors are often weakly formalized information, which means that in order to upload the data they have to be processed. There are special requirements imposed on the systems of processing such data, particularly, the systems have to process weakly formalized information and obtain formalized information which can be further processed on the computer. We have proposed a generalized algorithm for processing weakly formalized data that can be used for information processing in various technical systems, particularly, to ensure safe road traffic. Vehicles are thought to be equipped with various sensors (video, sound, ultrasonic sensors, etc.) providing data on obstacles on the road (buildings, constructions, trees, other vehicles, people, animals, etc.). The data may be insufficient due to such cases when one obstacle covers another, for example, a person

can be behind car in the blind spot. The point is in avoiding collision of the car with any obstacle (other car, building, person, animal, etc.). There are two parameters to be considered: driving speed and direction. If the distance between one car and other car or any obstacle gets smaller, a vehicle can change either driving speed or direction, thus ensuring safe road traffic. This paper considers vehicle movement through a T-junction as an example and shows that applying the considered generalized algorithm allows ensuing safe road traffic.

**Key words:** weakly formalized information; algorithm; information processing; road traffic safety.

**About the author:** Anton Alexandrovich Kopyltsov, Senior Lecturer at the Department of Automated Information Processing Systems, Saint-Petersburg State Electrotechnical University «LETI».

**Place of employment:** Saint-Petersburg State Electrotechnical University «LETI».

---

---

УДК 51:004.942

*А.С. Трунов, Л.И. Воронова, В.И. Воронов*  
*Москва, Россия*

### РАЗРАБОТКА ПАРАЛЛЕЛЬНОГО АЛГОРИТМА ДЛЯ ИНФОРМАЦИОННО-ИССЛЕДОВАТЕЛЬСКОЙ СИСТЕМЫ «MD-SLAG-MELT» НА ОСНОВЕ ТЕХНОЛОГИИ CUDA

**Аннотация.** Одним из приоритетных направлений современной науки является создание новых материалов с заранее заданными свойствами. В этой области широко применяется компьютерное моделирование (КМ), в том числе метод молекулярной динамики, позволяющий определять целый комплекс свойств (структурные, термодинамические, транспортные) и исследовать взаимосвязи наноструктуры и физико-химических свойств.

Для проведения КМ создаются автоматизированные информационные системы (АИС), главной целью которых является расширение границы исследований, оптимизация научной работы и ускорение проведения исследований. Одной из таких систем является ИИС «Шлаковые расплавы» (ИИС «MD-SLAG-MELT») [1].

Особенности предметной области ИИС таковы, что в настоящее время без применения методов распределения вычислений удастся просчитывать за приемлемое время групповое поведение систем, содержащих в лучшем случае десятки тысяч частиц. Однако существует ряд задач, в частности, связанных с определением пространственных наноразмерных неоднородностей, для которых необходимо увеличение размерности модельной системы до миллионов частиц. Промоделировать подобную систему на локальном компьютере в последовательном режиме практически не удастся, т.к. это связано с большими временными затратами, эксперимент может занять несколько месяцев. Решение подобных задач требует использования распределенных вычислений.

В статье описан разработанный авторами параллельный алгоритм, адаптирующий существующий (legacy application) линейный алгоритм расчета сил межмолекулярного взаимодействия под распределенные вычисления на центральном и графическом процессорах вычислительного устройства и имплементированный в программную среду ИИС «MD-SLAG-MELT». В основе разработанного распределенного алгоритма лежит технология параллельного программирования CUDA.

**Ключевые слова:** Информационно-исследовательская система (ИИС) «MD-SLAG-MELT»; молекулярная динамика; распределенные вычисления; параллельный алгоритм; CUDA

**Сведения об авторах:** Трунов Артем Сергеевич<sup>1</sup>, старший преподаватель кафедры информационных систем и моделирования РГГУ; Воронова Лилия Ивановна<sup>2</sup>, заведующая кафедрой информационных систем и моделирования РГГУ, профессор кафедры математической кибернетики и информационных технологий МТУСИ; Воронов Вячеслав Игоревич<sup>3</sup>, доцент кафедры информационных систем и моделирования РГГУ.

**Место работы:** Российский государственный гуманитарный университет (ФГОБУ ВПО «РГГУ») <sup>1,2,3</sup>; Московский технический университет связи и информатики (ФГОБУ ВПО «МТУСИ») <sup>2</sup>.

**Контактная информация:** 111672, г. Москва, ул. Новокосинская, д. 40, кв. 204; тел: 8-916-068-68-38. E-mail voronova2001@mail.ru.

---