

# Моделирование распределенной импульсной системы управления на гибридном суперкомпьютере

Ю. В. Ильюшин

Санкт-Петербургский горный университет  
ilyushin\_y@spmi.ru

И. М. Новожилов

Санкт-Петербургский государственный  
электротехнический университет  
«ЛЭТИ» им. В.И. Ульянова (Ленина)  
novozhilovim@list.ru

**Аннотация.** В статье проведен синтез закона распределения температурного поля в изотропном стержне на основе импульсных переходных функций – функции Грина. Проведено математическое моделирование полученного закона на гибридном суперкомпьютере по технологии NVidia CUDA. На основании математического моделирования сделан вывод о возможности оптимизации температурных режимов в рассматриваемой системе управления. Сделаны выводы о количестве нагревательных элементов и числе членов ряда Фурье.

**Ключевые слова:** синтез; системный анализ; математическая модель

Синтез программных управлений осуществляется на основе моделей процессов путём формализации требований в виде односточечных или многоточечных целевых условий (равенств, неравенств) представленных требованиями минимизации функционалов качества. Для взаимоотношения между управляемыми координатами (факторами) используются аналитические решения операторов теплопроводности. Или же разностные схемы для соответствующих им задач. Что позволяет приобрести большое многообразие алгоритмов программного управления. Такие алгоритмы при соответствующем обобщении могут предназначаться для создания систем локально-оптимального или локально-целевого управления. Подход к синтезу локально-оптимального управления для температурных процессов может базироваться на задачах теплопроводности с использованием разностных схем. Эти схемы дают возможность «алгебраизовать» оптимизационные задачи синтеза управлений. Что в свою очередь даст возможность свести их к решению счётного числа конечномерных задач математического программирования. В модальном изображении аналитическое решение представляет собой бесконечные или конечные линейные комбинации «мод». Как совокупности экспоненциально-периодических базисных функций времени и координат.

Подходы к синтезу стабилизирующих управлений могут рассматриваться на основе вариантных целевых условий и разрешением задач математического программирования численными методами. Данная задача

была решена А.Н. Тихоновым и А.А. Самарским при рассмотрении первой краевой задачи, математическая модель которой задана следующими уравнениями [1]:

$$\begin{cases} u_t = a^2 u_{xx} + f(x, t), 0 < x < l, 0 < t \leq T, \\ u(0, t) = \mu_1(t), 0 \leq t \leq T, \\ u(l, t) = \mu_2(t), 0 \leq t \leq T, \\ u(x, 0) = \varphi(x), 0 \leq x \leq l. \end{cases} \quad (1)$$

где:  $t$  – время;  $x$  – точка (координата по оси  $X$ ) расположения датчика температуры;  $\xi$  – точка (координата по оси  $X$ ) расположения нагревательного элемента,  $f(x, t)$  – входное воздействие в точке  $x$  во время  $t$ ,  $a^2$  – заданный коэффициент теплопроводности материала объекта управления.

Была получена функция Грина (2), отражающая поведение температурного поля в любой момент времени в любой точке объекта управления. Приводя ее А.Н. Тихонов и А.А. Самарский исследовали вопрос сходимости рядов Фурье, на которые раскладывается функция Грина [2, 3].

$$G(x, t) = \sum_{n=1}^{\infty} \frac{2}{l} \sin\left(\frac{\pi n}{l} s\right) \sin\left(\frac{\pi n}{l} \xi\right) \exp\left\{-a^2 \left(\frac{\pi n}{l}\right)^2 t\right\} \quad (2)$$

где  $n$  – номер члена ряда Фурье;  $l$  – длина стержня;  $t$  – время;  $x$  – точка (координата по оси  $X$ ) расположения датчика температуры;  $\xi$  – точка (координата по оси  $X$ ) расположения нагревательного элемента;  $\tau$  – момент включения точечного источника,  $a^2$  – заданный коэффициент теплопроводности материала объекта управления.

Таким образом, формула данная позволяет производить расчет поведения температурного поля в произвольной точке изотропного стержня в произвольный момент времени. Однако для исследования температурного поля с течением времени необходимо учитывать функцию начального нагрева, так как значение температуры есть сумма значений функции Грина в текущий момент времени и функции начального нагрева [4, 5].

Таким образом, для анализа температурного поля необходимо применять формулу (3), учитывающую функцию начального нагрева [4]:

$$T(\mathbf{x}_j, t) = \sum_{i=1}^d \sum_{n=1}^k \frac{2}{l} \exp \left[ - \left( \frac{\pi n a}{l} \right)^2 t \right] \sin \frac{\pi n}{l} \mathbf{x}_j \sin \frac{\pi n}{l} \xi_i + \\ + \sum_p \sum_{n=1}^k \frac{2}{l} \exp \left[ - \left( \frac{\pi n a}{l} \right)^2 (t - \tau_p) \right] \sin \frac{\pi n}{l} \mathbf{x}_j \sin \frac{\pi n}{l} \xi_{z(p)} \quad (3)$$

Рассмотрим процесс моделирования на примере объекта управления, состоящего из цилиндрического стержня, пренебрежимо малой толщины, длиной  $l_1$  и расположенном на нем нагревательных элементов  $k$  и датчиков температуры  $d$ .

Промоделируем поведение температурного поля в изотропном стержне на гибридном суперкомпьютере по технологии CUDA [8, 9]. В качестве исходных данных возьмем следующие:  $l_1=10$  метров,  $k=100$  штук,  $d=100$  штук,  $T_{зад} = 1...5000 \text{ } ^\circ\text{C}$ ,  $a^2=0,00024 \text{ м}^2/\text{с}$ ,  $x_1 = y_1 = z_1 = v_1 = p_1 = Q_1 = 1$ ,

$y, x, v, p_i \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ ,  $\tau=1 \text{ сек.}$  Время математического моделирования зададим равным 170800 секунд (48 часов).

Требуется определить время, через которое металл потеряет свойства твердого тела. Математическое моделирование данного поля не возможно на обычном компьютере в связи с чрезмерно высокой сложностью вычислений. Для этих целей возьмем 96 ядерный процессор GF108 на базе графического вычислителя GT 630. Следует отметить тот факт, что структура чипа CUDA отличается от структуры процессоров для PC.

Процессор Intel Nehalem содержит 4 независимых процессорных ядра, каждое из которых обладает полной функциональностью центрального процессора. Такое ядро способно обрабатывать системные прерывания, работать с устройствами ввода – вывода, то есть абсолютно полноценно поддерживать операционную систему. Каждое ядро содержит КЭШ первого уровня для данных и инструкций, содержит логику выборки инструкций и КЭШ данных второго уровня. Все ядра абсолютно симметрично присоединены к КЭШу третьего уровня и к QPI (QuickPath Interconnect) – системе присоединения процессоров к чипсету. Так же они присоединены к IMC (Integrated Memory Controller) – системе связи с памятью, пришедшая взамен северного моста.

В некоторых версиях современных процессоров Intel так же присутствует встроенный графический контроллер [8, 9].

Тогда же, как в свою очередь чип CUDA представляет собой процессор GF108 имеющий 32 потоковых процессоров (рис. 1).

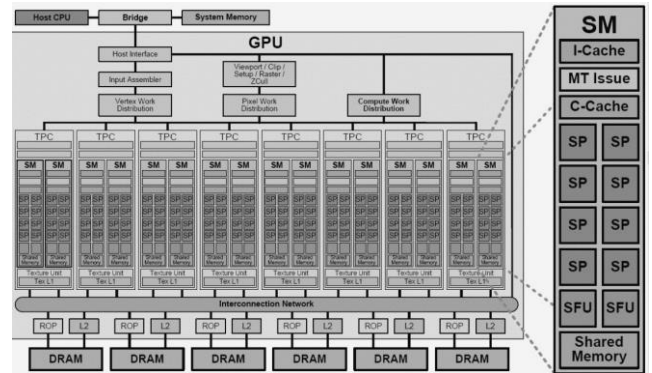


Fig. 1.

Так же имеет два КЭШа объемами 32 и 256 КБ соответственно с разрядностью шины 128 бит. Данный чип позволяет использовать два гигабайта оперативной памяти на процессор. Также в нем имеется возможность поддержки кластерного моделирования потоков и SIMD-инструкции.

Следующий блок программы производит вычисление температурного поля

```
for (i = 1; i <= *d; i++) {
    for (n = 1; n <= *k; n++) {
        vspom = (2/(*l))*__expf(((0-
PI*n*(a)/(*l))*(PI*n*(a)/(*l))*t1)*
__sinf(PI*n*x[j]/
(*l))*__sinf(PI*n*ksi[i]/(*l))/((*l)+ksi1;
        g[j] = g[j] + vspom;
    }
}
```

Следует отметить, что для сохранения объемов памяти была выделена зарезервированная область памяти:

```
HANDLE_ERROR( cudaMalloc( (void**)&dev_t_mas,
I*J* sizeof( float ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_z,
sizeof( int ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_d,
sizeof( int ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_k,
sizeof( int ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_s,
sizeof( int ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_k1,
sizeof( float ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_k2,
sizeof( float ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_l,
sizeof( float ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_a,
sizeof( float ) ) );
HANDLE_ERROR( cudaMalloc( (void**)&dev_t_zad, sizeof( float ) ) );
```

После ее копирования данных на вычислительный процессор с помощью программного оператора `cudaMemcpyDeviceToHost` произошло выделение слеш памяти для организации вычислений по векторным типам

данных. Копирование данных осуществлял следующий программный код:

```

cudaMemcpy( dev_z, &z,
sizeof(int),cudaMemcpyHostToDevice );
cudaMemcpy( dev_d, &d,
sizeof(int),cudaMemcpyHostToDevice );
cudaMemcpy( dev_k, &k, sizeof( int ),
cudaMemcpyHostToDevice );
cudaMemcpy( dev_s, &s, sizeof( int ),
cudaMemcpyHostToDevice );
cudaMemcpy( dev_k1, &k1, sizeof( float ),
cudaMemcpyHostToDevice );
cudaMemcpy( dev_k2, &k2, sizeof( float ),
cudaMemcpyHostToDevice );
cudaMemcpy( dev_l, &l, sizeof( float ),
cudaMemcpyHostToDevice );
cudaMemcpy( dev_a, &a, sizeof( float ),
cudaMemcpyHostToDevice );
cudaMemcpy( dev_t_zad, &t_zad, sizeof( float ),
cudaMemcpyHostToDevice);

```

Вызов модуля расчета производился по схеме с жесткой привязкой к ядру исполнителю:

```

add<<<1,1>>>( dev_t_mas , dev_z, dev_k1, dev_k2,
dev_l, dev_a, dev_t_zad, dev_k, dev_d, dev_s);

```

Что позволило обеспечить функционирование данного модуля исключительно на первом процессоре вычислителя. Изменяя первый параметр функции можно производить вычисления на других вычислителях, если таковые имеются в системе. Также комбинация данных параметров позволит создавать программные управления для высокопроизводительных систем на базе вычислителей NVidia tesla. Однако в связи с тем, что вычислители tesla обладают встроенными 4 процессорами по 2000 ядер каждое, необходимо производить дополнительный расчет потоковых операции. Использование слеш памяти позволило помимо расчета температурного поля экспериментальным путем установить необходимое число членов ряда Фурье необходимого для обеспечения заданной точности вычислений. В системе происходил расчет выборки, для 1000, 100 и 10 членов ряда Фурье. Результаты математического моделирования на суперкомпьютере и на обычном персональном компьютере приведены в таблице. Как видно из таблицы значения идентичны, но время, затраченное на расчет значительно меньше. Следовательно, математическое моделирование систем автоматического управления существенно увеличит производительность систем. В большей степени это необходимо для систем реального времени.

TABLE I.

Шаг	Значение на суперкомпьютере	Значение на персональном компьютере
1.	120,38692	120,38692
2.	129,38892	129,38892
3.	137,39692	137,39692
*****		
100	870,19692	870,19692
101	890,31292	890,31292
102	970,29692	970,29692
*****		
7405	2010,25698	2010,25698
7406	2013,36551	2013,36551
7407	2040,15454	2040,15454

На основе результатов математического моделирования температурного поля изотропного стержня можно сделать следующие выводы.

1. При минимальном воздействии (0,1 Вт) на объект управления изотропный стержень потеряет свойства физически твердого тела на 8652 секунде эксперимента.

2. При уменьшении числа нагревательных воздействий время эксперимента увеличивается, на основании чего можно сделать вывод о возможности разработки системы автоматического управления температурным полем изотропного стержня. При этом можно управлять температурным полем за счет числа и времени включения нагревательных элементов.

3. При моделировании процесса нагрева изотропного стержня импульсными источниками нагрева необходимо и достаточно использовать 5 членов ряда Фурье. Данный результат подтвердил данные из других источников [1, 2, 3, 6, 7].

4. Моделирование данного процесса на гибридном суперкомпьютере привело к реальному времени расчета в 10 минут. В свою очередь линейный алгоритм на обычном компьютере моделирует данный процесс в течение 6 часов. Данная скорость реализуется главным образом за счет 96 ядерного процессора работающего с более высокой суммарной частотой, чем обычный процессор персонального компьютера.

Проведенное анализ результатов математического моделирования показывает, что моделирование систем автоматического управления и систем прогрессирования, в том числе робастного управления, можно значительно расширить за счет применения гибридных суперкомпьютеров. Их применение позволит не только расширить число управляющих воздействий, но и время моделирования переходных характеристик.

#### СПИСОК ЛИТЕРАТУРЫ

- [1] Ilyushin Y., Pervukhin D., Afanasieva O., Klavdiev A., & Kolesnichenko, S. (2014). Designing of Distributed Control System with Pulse Control. Middle-East Journal of Scientific Research, 21(3), 436-439. <http://dx.doi.org/10.5829/idosi.mejsr.2014.21.03.21433>
- [2] Ilyushin Y., Pervukhin D., Afanasieva O., Klavdiev A., & Kolesnichenko, S. (2014). The Methods of the Synthesis of the Nonlinear Regulators for the Distributed One-Dimension Control

- Objects. *Modern Applied Science* 9 (2), 42-61. <http://dx.doi.org/10.5539/mas.v9n2p42>
- [3] Chernishev A. (2009). Adaptation of absolute stability frequency criterion to systems with distributed parameters. *Mechatronics, automatization, control*, 7, 13-18.
- [4] Chernishev A. (2009). Modified absolute stability criterion for nonlinear distributed systems. *IHL News – North Caucasian region. Technical sciences*, 3(151), 38-41.
- [5] Chernishev A. (2010). Interpretation of absolute stability criterion for nonlinear distributed systems. *Automatization and modern technologies*, 2, 28-32.
- [6] Chernishev A., Antonov V., & Shurakov D. (2010). System of temperature field stabilization criterion in the process of heat utilization in contact welding. *Scientific-technical news of S.Pt.SPI*, 6(113), 151-155.
- [7] Ilyushin Y. (2011). Designing of temperature field control system of tunnel kilns of conveyor type. *Scientific-technical news of S.Pt.SPI*, 3(126), 67-72.
- [8] Kolesnikov A. (2009). Nonlinear Oscillations Control. Energy Invariants. *Journal of Computer and Systems Sciences International*, 48(2), 185-198. <http://dx.doi.org/10.1134/S1064230709020038>
- [9] Kolesnikov A., Zarembo Ya., & Zarembo V. (2007). Discharge of a Copper-Magnesium Galvanic Cell in the Presence of a Weak Electromagnetic Field. *Russian Journal of Physical Chemistry A*, 81(7), 1178-1180. <http://dx.doi.org/10.1134/s003602440707031x>
- [10] Kolesnikov A., Zarembo Ya., Puchkov L., & Zarembo V. (2007). Zinc Electrochemical Reduction on a Steel Cathode in a Weak Electromagnetic Field. *Russian Journal of Physical Chemistry A*, 81(10), 1715-1717. <http://dx.doi.org/10.1134/s0036024407100330>