

УДК 553.98:519.612

## ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ РАЗЛИЧНЫХ ПРЕДОБУСЛАВЛИВАТЕЛЕЙ ПРИ ПАРАЛЛЕЛЬНОМ РЕШЕНИИ РАЗРЕЖЕННЫХ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ НА ГРАФИЧЕСКИХ ПРОЦЕССОРАХ

Р. К. Газизов<sup>1</sup>, Р. Р. Губайдуллин<sup>2</sup>, Н. В. Репин<sup>3</sup>, А. В. Юлдашев<sup>4</sup>

<sup>1</sup>gazizovrk@gmail.com, <sup>2</sup>hfnvbhu@gmail.com, <sup>3</sup>meujin@gmail.com, <sup>4</sup>art@ugatu.su

ФГБОУ ВО «Уфимский государственный авиационный технический университет» (УГАТУ)

Поступила в редакцию 22.05.2018

**Аннотация.** Проведена оценка производительности центральных процессоров Intel Xeon и графических процессоров NVIDIA Tesla различных поколений при решении разреженных систем линейных алгебраических уравнений (СЛАУ) итерационным методом BiCGStab с предобуславливателем BILU(0); показано, что использование графических процессоров в качестве ускорителей позволяет снизить время решения СЛАУ в 4,3–7,7 раза. Рассмотрен метод AIPS, идея которого заключается в аппроксимации обратной матрицы на основе степенного разложения в ряд Неймана, и его применимость в качестве первой ступени двухступенчатого предобуславливателя CPR. Проведена оценка эффективности процедуры решения СЛАУ с различными предобуславливателями (BILU(0), CPR-AMG, CPR-AIPS) на вычислительной системе с двумя графическими процессорами NVIDIA Tesla P100; показано, что за счет параллельной реализации метода AIPS предобуславливатель CPR-AIPS достаточно хорошо масштабируется и позволяет снизить время решения большинства тестовых СЛАУ.

**Ключевые слова:** графические процессоры; итерационные методы; параллельные вычисления; предобуславливатели; разреженные матрицы.

### ВВЕДЕНИЕ

Одним из актуальных трендов в суперкомпьютерной индустрии является применение наряду с центральными процессорами (CPU) массивно-параллельных процессорных устройств, используемых для ускорения вычислений. К примеру, в 28 редакции списка Топ-50 мощнейших компьютерных систем России и стран СНГ [1] от 03.04.2018 г. 26 суперкомпьютеров (52%) оснащены ускорителями, причем 18 из них – графическими процессорами (GPU) NVIDIA Tesla.

Возможность использования графических процессоров для существенного уско-

рения расчетов при решении ряда научных и производственных задач обусловила появление в составе вычислительного кластера УГАТУ [2] гибридных вычислительных узлов, оснащенных GPU NVIDIA Tesla, в результате чего пиковая производительность суперкомпьютерного комплекса за 2011–2017 гг. была увеличена в 2,5 раза с 20 до 50 Тфлопс. Ключевые характеристики центральных и графических процессоров, имеющих в составе университетского вычислительного кластера, приведены в табл. 1 (установка GPU NVIDIA Tesla V100 на базе новейшей архитектуры с кодовым названием Volta планируется в 2018 г.).

Таблица 1

## Характеристики центральных и графических процессоров вычислительного кластера УГАТУ

<i>Tun</i>	<i>CPU Intel Xeon</i>					<i>GPU NVIDIA Tesla</i>				
Наименование	E5345	X5670	E5-2670	E5-2650 v4	Gold 6126	M2050	K20	K20X	P100	V100
Кодовое название	Clovertown	Westmere	Sandy Bridge	Broadwell	Sky-lake	Fermi	Kepler		Pascal	Volta
Год выпуска	2007	2010	2012	2016	2017	2010	2013		2016	2017
Баз. тактовая частота (ГГц)	2,33	2,93	2,6	2,2	2,6	1,15	0,71	0,73	1,19	1,25
Количество ядер	4	6	8	12	12	448	2 496	2 688	3 584	5 120
Пик. производитель (Гфлопе)	37,33	70,32	166,4	480	883,2	515	1 170	1 310	4 700	7 000
Кол-во процессоров	532	2	16	4	4	2	10	2	2	–
Суммарная производитель (Тфлопе)	19,833	0,141	2,662	1,690	2,611	1,030	11,700	2,620	9,400	–
Тип, частота (ГГц) оперативной памяти	DDR2 0,67	DDR3 1,33	DDR3 1,6	DDR4 2,4	DDR4 2,67	GDDR5 1,55	GDDR5 2,6	GDDR5 2,6	HBM2 0,72	HBM2 0,88
Пик. пропускная способность, ГБ/с	10,66	32	51,2	76,8	128	148	208	250	732	900

В последнее время нами ведется разработка и реализация параллельных алгоритмов, ориентированных на архитектуру GPU, решения разреженных СЛАУ, возникающих при дискретизации дифференциальных уравнений в частных производных, к примеру, в задаче моделирования многофазных фильтрационных потоков углеводородов в пористых средах [3].

Сначала [4] был реализован решатель, позволяющий выполнять как на центральных, так и на графических процессорах решение разреженных СЛАУ итерационным методом бисопряженных градиентов со стабилизацией (BiCGStab) [5] с блочной модификацией неполного LU-разложения [6] (BILU(0)) в качестве предобуславливателя.

Далее [7] была разработана кластерная версия решателя, позволяющая проводить расчеты на множестве графических процессоров, распределенных по узлам вычислительного кластера. В ней, наряду с BILU(0), был реализован двухступенчатый предобуславливатель Constrained Pressure Residual (CPR) [8], а точнее его популярная версия, чаще всего обозначаемая в литературе аббревиатурой CPR-AMG, в которой на первой ступени (локальный предобуславлива-

тель) используется алгебраический многосеточный метод (AMG) [9], а на второй (глобальный предобуславливатель), как правило, неполное LU-разложение.

В теоретической части данной работы рассмотрен метод AIPS, идея которого заключается в аппроксимации обратной матрицы на основе степенного разложения в ряд Неймана, и его применимость в качестве первой ступени предобуславливателя CPR, являющегося одним из наиболее эффективных при решении СЛАУ в рамках задачи гидродинамического моделирования нефтегазовых месторождений.

В экспериментальной части сначала представлена оценка производительности CPU Intel Xeon и GPU NVIDIA Tesla при решении СЛАУ методом BiCGStab с предобуславливателем BILU(0), откуда следует актуальность использования графических процессоров в качестве ускорителей вычислений. Затем проведена сравнительная оценка эффективности предобуславливателей BILU(0), CPR-AMG и CPR-AIPS, демонстрирующая преимущества применения предобуславливателя CPR-AIPS на вычислительных системах, оснащенных несколькими графическими процессорами.

### МЕТОД АППРОКСИМАЦИИ ОБРАТНОЙ МАТРИЦЫ НА ОСНОВЕ СТЕПЕННОГО РАЗЛОЖЕНИЯ И ЕГО ПРИМЕНЕНИЕ В РАМКАХ ДВУХСТУПЕНЧАТОГО ПРЕДОБУСЛАВЛИВАТЕЛЯ CPR

Большой потенциал распараллеливания имеют предобуславливатели на основе аппроксимации обратной матрицы [10], в том числе путем разложения в ряд Неймана [11]:

$$(E - B)^{-1} = E + B + B^2 + B^3 + \dots,$$

где  $E$  – единичная матрица. Соответственно обратная матрица  $A^{-1} = (E - B)^{-1}$  может быть аппроксимирована некоторым количеством членов ряда Неймана.

Существуют и другие представления матрицы  $A$  (см., например, [11] и ссылки в работе, [12]), в том числе в виде разности матриц общего вида:  $A = P - R$ . Если же представить  $A$  в виде

$$A = P + R = P(E - (-1)(P^{-1}R)),$$

то с использованием разложения в ряд Неймана можно получить следующую формулу для построения предобуславливателя, аппроксимирующего обратную матрицу при заданном  $N$ :

$$A^{-1} \approx M_N^{-1} = \left[ E + \sum_{k=1}^N (-1)^k (P^{-1}R)^k \right] P^{-1}. \quad (1)$$

В данной работе будем называть метод построения предобуславливателя по формуле (1) методом аппроксимации обратной матрицы на основе степенного разложения (Approximation of Inverse by Power Series, AIPS). Хотя далее в экспериментальной части работы  $N$  положено равным 10, выработка подхода к выбору оптимального значения данного параметра является предметом дальнейшего исследования.

Отметим, что нами реализован неявный способ применения AIPS в рамках итерационного метода решения СЛАУ: на каждой итерации производится вычисление произведения  $M_N^{-1}$  на вектор путем выполнения последовательности матрично-векторных операций в соответствии с (1).

Одним из наиболее эффективных предобуславливателей при решении СЛАУ,

возникающих в рамках задачи гидродинамического моделирования нефтегазовых месторождений в результате дискретизации уравнений многофазной фильтрации по времени с использованием полностью неявной разностной схемы, считается специализированный двухступенчатый предобуславливатель CPR и его модификации.

В данной работе рассмотрена эффективность предобуславливателя CPR-AIPS, в котором в качестве альтернативы AMG на первой ступени используется описанный выше метод AIPS. Разложение по формуле (1) строится для подматрицы на давление ( $A_p$ ), которая формируется из исходной матрицы СЛАУ с помощью алгоритма метода CPR в предположении о малом изменении насыщенностей в расчетных ячейках.

При использовании метода AIPS в рамках CPR в качестве матрицы  $P$  предлагается использовать трехдиагональную часть  $A_p$ . Это позволяет в рамках неявного способа применения AIPS свести вычисление выражений вида  $z = P^{-1}b$  к решению систем  $Pz = b$  с трехдиагональной матрицей. Следовательно, для повышения быстродействия предобуславливателя CPR-AIPS необходим параллельный алгоритм, рассчитанный на многократное решение СЛАУ с неизменной трехдиагональной матрицей, сформированной из подматрицы на давление.

С учетом структуры трехдиагональной части матрицы линейной системы, получаемой в результате дискретизации уравнений в частных производных на трехмерной сетке, нами был разработан ориентированный на архитектуру GPU параллельный алгоритм решения линейных систем с трехдиагональной матрицей, состоящей из множества независимых блоков различного размера. Наличие такого рода блочности в структуре матрицы  $P$  позволило организовать эквивалентное распараллеливание решения соответствующих систем, а также построение степенного разложения в целом с минимальным объемом коммуникаций. Также благодаря тому, что в матрице  $P$  имеется множество независимых блоков, возможно эквивалентное распараллеливание метода AIPS на нескольких GPU.

### ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

В табл. 2 приведены характеристики разреженных матриц СЛАУ, задействованных в дальнейших экспериментах. Данные матрицы получены при решении уравнений трехфазной фильтрации, дискретизированных по времени с использованием полностью неявной схемы.

Таблица 2  
Характеристики тестовых матриц

Матрица	Размерность	Кол-во ненулевых элементов	Среднее кол-во элементов в строке
imsh	1 500 000	55 815 624	37,21
immn	2 304 102	42 859 314	18,6
krrv	4 320 921	85 471 137	19,78
mmnt	5 637 747	109 595 799	19,44
fdrv	6 610 263	118 221 633	17,89

Параметры тестов: критерий остановки итерационного процесса – достижение относительной невязкой величины  $\varepsilon = 10^{-6}$ ; начальное приближение – нулевой вектор; расчеты проводились с двойной точностью.

#### Оценка производительности различных вычислительных архитектур

Рассмотрим эксперимент, направленный на оценку производительности различных поколений серверных центральных процессоров Intel Xeon и графических процессоров NVIDIA Tesla на примере решения разреженных СЛАУ итерационным методом BiCGStab с предобуславливателем ВІLU(0). В основе программной реализации данного метода в нашем решателе лежит библиотека Intel Math Kernel Library из состава Intel Parallel Studio XE для CPU, а также библиотеки cuBLAS и cuSPARSE из состава NVIDIA CUDA Toolkit для GPU.

При расчетах на многоядерных CPU были задействованы все процессорные ядра,

количество блоков при использовании ВІLU(0) равнялось количеству ядер. Увеличение количества блоков в ВІLU(0) приводит к упрощению предобуславливателя, что может негативно сказываться на скорости сходимости итерационного процесса. Этим объясняется то, что в табл. 3 количество итераций колеблется при переходе с одного CPU на другой. Минимальное количество итераций было достигнуто при расчетах на 4-ядерном E5345, а максимальное на 12-ядерных E5-2650 v4 и Gold 6126. В то же время от используемого GPU скорость сходимости не зависит, т.к. количество блоков ВІLU(0) равно 1, а для распараллеливания операций неполного LU-разложения на GPU в cuSPARSE применяются эквивалентные параллельные алгоритмы, основанные на разделении по уровням (level scheduling).

В табл. 4 приведены времена решения СЛАУ на различных CPU и GPU. Информация о решении СЛАУ с матрицами mmnt и fdrv на GPU M2050 отсутствует из-за того, что для решения этих СЛАУ необходимо более чем 3 Гбайт оперативной памяти, что выходит за пределы объема памяти, имеющегося на данном GPU.

Сравнение времени решения СЛАУ на CPU и GPU приблизительно одного и того же года выпуска (X5670 и M2050, E5-2670 и K20X, E5-2650 v4 и P100, Gold 6126 и V100) показывает, что применение GPU позволяет существенно ускорить вычисления, например, GPU V100 на базе архитектуры Volta позволяет ускорить решение СЛАУ от 4,3 до 7,7 раза, в среднем в 5,8 раза относительно CPU Gold 6126 из семейства серверных процессоров Intel Xeon Scalable на базе архитектуры Skylake. В связи с этим считаем перспективным применение GPU NVIDIA в качестве ускорителей в рамках рассматриваемой задачи.

Таблица 3  
Количество итераций метода BiCGStab с предобуславливателем ВІLU(0) на различных CPU и GPU

Матрица	Архитектура								
	CPU Intel Xeon					GPU NVIDIA Tesla			
	E5345	X5670	E5-2670	E5-2650 v4	Gold 6126	M2050	K20X	P100	V100
imsh	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,5
immn	25,5	27,5	24,5	27,5	27,5	24,5	24,5	24,5	24,5
krrv	16	16	16	16	16	16	16	16	16
mmnt	22,5	29	22,5	30	30	–	22,5	22,5	22,5
fdrv	4	4	4	4	4	–	4	4	4

Таблица 4

Время решения СЛАУ методом BiCGStab с предобуславливателем BILU(0) на различных CPU и GPU, с

Матрица	Архитектура								
	CPU Intel Xeon					GPU NVIDIA Tesla			
	E5345	X5670	E5-2670	E5-2650 v4	Gold 6126	M2050	K20X	P100	V100
imsh	2,396	0,408	0,296	0,239	0,151	0,347	0,150	0,053	0,035
immn	29,299	5,029	3,037	2,489	1,612	5,116	2,049	0,524	0,325
krrv	37,354	9,213	3,967	2,921	1,894	6,429	2,574	0,631	0,322
mmnt	65,999	20,504	6,892	6,832	4,392	–	4,645	1,130	0,567
fdrv	16,392	4,314	1,873	1,446	0,897	–	1,121	0,281	0,152

Далее рассматривается эффективность решения на системе с двумя GPU разреженных СЛАУ методом BiCGStab с различными предобуславливателями, реализованными в нашем решателе. Результаты получены на одном из узлов вычислительного кластера УГАТУ (2 x CPU Intel Xeon Gold 6126, 2 x GPU NVIDIA Tesla P100).

#### Эффективность процедуры решения СЛАУ с предобуславливателем BILU(0)

При решении СЛАУ с предобуславливателем BILU(0) (табл. 5 и 6) наиболее низкое ускорение (1,45) при расчете на 2 GPU относительно 1 GPU получается на матрице imsh. На матрице immn коэффициент ускорения достигает 1,58, а на остальных матрицах становится более 1,8. Это связано с тем, что imsh и immn являются матрицами с наименьшими размерностями, что приводит к более низкой загруженности GPU, а также более высокой доле коммуникаций.

Таблица 5

Количество итераций BiCGStab при использовании предобуславливателя BILU(0)

Матрица	Количество GPU	
	1	2
imsh	1,5	1,5
immn	24,5	25
krrv	16	16
mmnt	22,5	22,5
fdrv	4	4

Таблица 6

Время решения СЛАУ при использовании предобуславливателя BILU(0), с

Матрица	Количество GPU	
	1	2
imsh	0,053	0,037
immn	0,524	0,332
krrv	0,631	0,345
mmnt	1,130	0,592
fdrv	0,281	0,152

В среднем использование 2 GPU позволяют ускорить решение СЛАУ при использовании предобуславливателя BILU(0) в 1,69 раза относительно 1 GPU.

#### Эффективность процедуры решения СЛАУ с предобуславливателем CPR-AMG

В данном подразделе рассматривается решение СЛАУ с предобуславливателем CPR-AMG, на первой ступени которого применяется классический алгоритм AMG, реализованный с помощью функционала открытой библиотеки AmgX. В качестве глобального предобуславливателя в CPR-AMG используется BILU(0), протестированный в предыдущем подразделе.

Необходимо отметить высокую скорость сходимости итерационного метода при использовании CPR-AMG (табл. 7), в частности, для решения СЛАУ с матрицами immn и mmnt требуется значительно меньше итераций, чем в случае с BILU(0).

Таблица 7

Количество итераций BiCGStab при использовании предобуславливателя CPR-AMG

Матрица	Количество GPU	
	1	2
imsh	1,5	1,5
immn	4	5
krrv	3,5	3,5
mmnt	4	4
fdrv	1,5	2

Таблица 8

Время решения СЛАУ при использовании предобуславливателя CPR-AMG, с

Матрица	Количество GPU	
	1	2
imsh	0,149	0,155
immn	0,222	0,231
krrv	0,360	0,291
mmnt	0,489	0,360
fdrv	0,351	0,299

Время решения СЛАУ (табл. 8) с матрицами *immn* и *mmnt* снижается благодаря использованию CPR-AMG относительно *BILU(0)* в 2,36 и 2,31 раза соответственно. Однако ускорение при решении СЛАУ с предобуславливателем CPR-AMG на 2 GPU составляет не более 1,36 раза относительно 1 GPU, а на матрицах *imsh* и *immn* коэффициент ускорения не превосходит 1. Столь слабые показатели преимущественно обусловлены низкой масштабируемостью процедуры построения иерархии многосеточного метода.

В среднем 2 GPU позволяют ускорить решение СЛАУ при использовании CPR-AMG лишь в 1,1 раза относительно 1 GPU.

Таким образом, несмотря на значительное снижение числа итераций и вызванное этим уменьшение времени решения некоторых СЛАУ на 1 GPU по сравнению с предобуславливателем *BILU(0)*, уже на двух GPU преимущество CPR-AMG перед более простым предобуславливателем *BILU(0)* становится менее очевидным.

#### Эффективность процедуры решения СЛАУ с предобуславливателем CPR-AIPS

В данном подразделе рассматривается решение СЛАУ с предобуславливателем CPR-AIPS, на первой ступени которого применяется реализованный нами для GPU метод AIPS. В качестве глобального предобуславливателя все так же используется *BILU(0)*.

Данные, приведенные в табл. 9 и 10, позволяют сделать вывод о достаточно высокой эффективности предобуславливателя CPR-AIPS. Время решения СЛАУ с CPR-AIPS на 2 GPU снижается от 1,6 (*imsh*) до 1,87 (*mmnt*) раза относительно 1 GPU, в среднем в 1,76 раза.

Можно заметить, что масштабируемость предобуславливателя CPR-AIPS ближе к масштабируемости *BILU(0)*, нежели CPR-AMG. Это связано с тем, что первая ступень в случае CPR-AIPS получилась менее затратной, чем в CPR-AMG, благодаря эффективной параллельной реализации на GPU алгоритмов метода AIPS, в том числе разработанного параллельного алгоритма решения линейных систем с трехдиагональной матрицей.

Таблица 9

**Количество итераций BiCGStab при использовании предобуславливателя CPR-AIPS**

Матрица	Количество GPU	
	1	2
<i>imsh</i>	1,5	1,5
<i>immn</i>	4,5	4,5
<i>krrv</i>	4,5	4,5
<i>mmnt</i>	5	5
<i>fdrv</i>	1	1

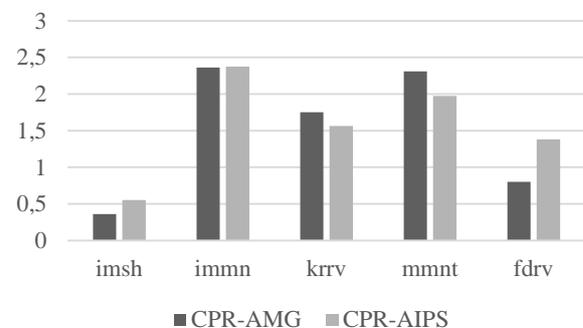
Таблица 10

**Время решения СЛАУ при использовании предобуславливателя CPR-AIPS, с**

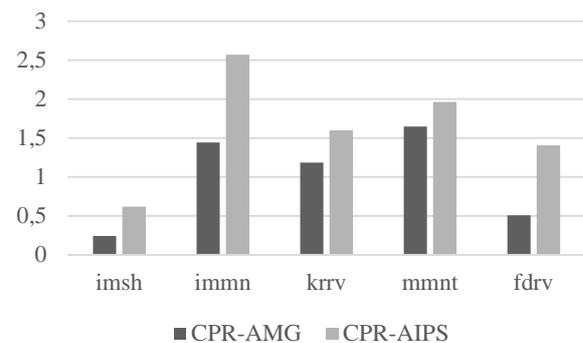
Матрица	Количество GPU	
	1	2
<i>imsh</i>	0,089	0,056
<i>immn</i>	0,193	0,117
<i>krrv</i>	0,350	0,192
<i>mmnt</i>	0,495	0,265
<i>fdrv</i>	0,185	0,099

#### Сравнение работы различных предобуславливателей на GPU

Рис. 1 и 2 иллюстрируют снижение времени решения СЛАУ при использовании рассмотренных вариантов предобуславливателя CPR относительно *BILU(0)* при расчетах на одном и двух GPU соответственно.



**Рис. 1.** Ускорение относительно *BILU(0)* при использовании CPR-AMG/AIPS (1 GPU)



**Рис. 2.** Ускорение относительно *BILU(0)* при использовании CPR-AMG/AIPS (2 GPU)

Проведенные эксперименты показали, что представленный в данной работе предобуславливатель CPR-AIPS:

- обеспечивает высокую скорость сходимости, сравнимую с CPR-AMG;
- за счет более легковесной первой ступени в большинстве случаев позволяет ускорить решение СЛАУ относительно CPR-AMG на 1 GPU;
- обладает хорошей масштабируемостью и обеспечивает минимальное время решения большинства тестовых СЛАУ на 2 GPU среди всех рассмотренных предобуславливателей.

### ЗАКЛЮЧЕНИЕ

Проведенная в работе оценка производительности центральных процессоров Intel Xeon и графических процессоров NVIDIA Tesla различных поколений позволила продемонстрировать возможность существенного снижения времени решения СЛАУ с разреженными матрицами итерационным методом BiCGStab с предобуславливателем VILU(0) за счет использования GPU в качестве ускорителей вычислений.

Также было показано, что для обеспечения высокой скорости сходимости итерационного процесса, а также достижения хорошей масштабируемости процедуры решения СЛАУ на вычислительных системах с несколькими GPU актуально использовать двухступенчатый предобуславливатель CPR-AIPS, в котором в качестве локального предобуславливателя реализован метод аппроксимации обратной матрицы на основе степенного разложения, а в качестве глобального – блочная модификация неполного LU-разложения без заполнения.

### СПИСОК ЛИТЕРАТУРЫ

1. **Топ50** URL: <http://top50.supercomputers.ru> (дата обращения: 15.04.2018). [ *Top50* [Online], (in Russian). Available: <http://top50.supercomputers.ru> ]
2. **Суперкомпьютер УГАТУ** URL: <http://www.ugatu.su/supercomputer/> (дата обращения: 15.04.2018). [ *USATU Supercomputer* [Online], (in Russian). Available: <http://www.ugatu.su/supercomputer/> ]
3. **Азиз Х., Сеттари Э.** Математическое моделирование пластовых систем. Москва – Ижевск: Институт компьютерных исследований, 2004. 411 с. [ К. Aziz, A. Settari, *Petroleum reservoir simulation*, (in Russian). Moscow – Izhevsk: Institute of Computer Science, 2004. ]
4. **Газизов И. И., Юлдашев А. В.** Разработка параллельного линейного решателя для задачи гидродинамического моделирования нефтегазовых месторождений // Эвристические алгоритмы и распределенные вычисления, 2014. Т. 1. № 1. С. 88–96. [ I. I. Gazizov, A. V. Yuldashev, “Development of parallel linear solver for hydro-dynamic modeling of oil and gas fields”, (in Russian), in *Evrst. algoritmy i raspredel. Vichisl.*, vol. 1, no. 1, pp. 88-96, 2014. ]
5. **AlgoWiki:** Открытая энциклопедия свойств алгоритмов. Стабилизированный метод бисопряженных градиентов (BiCGStab) URL: [http://algowiki-project.org/ru/Стабилизированный\\_метод\\_бисопряженных\\_градиентов\\_\(BiCGStab\)](http://algowiki-project.org/ru/Стабилизированный_метод_бисопряженных_градиентов_(BiCGStab)) (дата обращения: 31.05.2018). [ *AlgoWiki: Open Encyclopedia of Parallel Algorithmic Features. Biconjugate gradient stabilized method (BiCGStab)* [Online], (in Russian). Available: [http://algowiki-project.org/ru/Стабилизированный\\_метод\\_бисопряженных\\_градиентов\\_\(BiCGStab\)](http://algowiki-project.org/ru/Стабилизированный_метод_бисопряженных_градиентов_(BiCGStab)) ]
6. **Saad Y.** Iterative methods for sparse linear systems. 2nd ed. Philadelphia, PA, USA: SIAM, 2003. 528 p. [ Y. Saad, *Iterative methods for sparse linear systems*. 2nd ed. Philadelphia, PA, USA: SIAM, 2003. ]
7. **Специализированный** решатель разреженных систем линейных алгебраических уравнений на вычислительных кластерах, оснащенных графическими процессорами / А. В. Юлдашев [и др.] // Суперкомпьютерные дни в России: труды междунар. конф. М.: МГУ, 2016. С. 673–682. [ A. V. Yuldashev, et al., “Specialized solver of sparse linear systems of algebraic equations on GPU clusters”, (in Russian), *Superkompyuternye dni v Rossii*, pp. 673-682, 2016. ]
8. **Wallis J. R., Kendall R. P., Little T. E.** Constrained residual acceleration of conjugate residual methods // SPE 1353. 1985. P. 415–428. [ J.R. Wallis, R.P. Kendall, T. E. Little, “Constrained residual acceleration of conjugate residual methods”, in *SPE 13536*. 1985. ]
9. **Ruge J. W., Stuben K.** Algebraic multigrid (AMG) // Multigrid Methods. Philadelphia, PA, USA: SIAM, 1987. V. 3, p. 73–130. [ J. W. Ruge, K. Stuben, “Algebraic multigrid (AMG)”, in *Multigrid Methods*, Philadelphia, PA, USA: SIAM, 1987. vol. 3, p. 73-130. ]
10. **Недожогин Н. С., Копысов С. П., Новиков А. К.** Параллельное формирование предобуславливателя на основе обращения Шермана-Моррисона // Параллельные вычислительные технологии (ПаВТ’2015): труды междунар. конф. Челябинск: ЮУрГУ, 2015. С. 436–441. [ N. S. Nedozhogin, S. P. Kopysov, A. K. Novikov, “Parallel forming of preconditioners based on the approximation of the Sherman-Morrison inversion”, (in Russian), in *Parallelnye vychislitelnye tekhnologii*, pp. 436-441, 2015. ]
11. **Chen K.** Matrix Preconditioning Techniques and Applications (Cambridge. Monographs on Applied and Computational Mathematics), Cambridge University. Press, July 2005., 592 pp. [ K. Chen, *Matrix Preconditioning Techniques and Applications (Cambridge. Monographs on Applied and Computational Mathematics)*, Cambridge University. Press, 592 pp., 2005 ]
12. **Fung L. S. K., Dogru A. H.** Parallel Unstructured Solver Methods for Complex Giant Reservoir Simulation // SPE 106237, proceedings of the SPE Reservoir Simulation Symposium, Houston, Texas, 26-28 February 2007. [ L. S. K. Fung, A. H. Dogru, “Parallel Unstructured Solver Methods for Complex Giant Reservoir Simulation”, in *SPE 106237*, proceedings of the SPE Reservoir Simulation Symposium, 2007. ]

## ОБ АВТОРАХ

**ГАЗИЗОВ Рафаил Кавыевич**, проф., зав. каф. высокопроизводительных вычислительных технол. и систем. Дипл. математик (БГУ, 1983). Д-р физ. мат. наук по диф. уравнениям (ИММ Уральск. Отд. РАН, 1999). Иссл. в обл. группового анализа диф. уравнений и матем. моделирования.

**ГУБАЙДУЛЛИН Ратмир Расилевич**, магистрант по направлению «Прикладная математика и информатика». Дипл. бакалавр прикладной матем. и информ. (УГАТУ, 2016).

**РЕПИН Никита Вячеславович**, магистрант по направлению «Прикладная математика и информатика». Дипл. бакалавр прикладной матем. и информ. (УГАТУ, 2016).

**ЮЛДАШЕВ Артур Владимирович**, ст. преп. каф. высокопроизводительных вычислительных технол. и систем. Дипл. инж.-мат-к (УГАТУ, 2006). Иссл. в обл. высокопроизводительных выч.

**YULDASHEV, Artur Vladimirovich**, Senior lecturer, Dep. of high-performance computing technologies and systems. Dipl. eng.-mathematician (UGATU, 2006). Research in high-performance computing.

## METADATA

**Title:** Study of efficiency of several preconditioners for parallel solving sparse linear systems on graphics processors.

**Authors:** R. K. Gazizov<sup>1</sup>, R. R. Gubaidullin<sup>2</sup>, N. V. Repin<sup>3</sup>, A. V. Yuldashev<sup>4</sup>

**Affiliation:**

Ufa State Aviation Technical University (UGATU), Russia.

**Email:** <sup>1</sup>gazizovrk@gmail.com <sup>2</sup>hfnvbhu@gmail.com, <sup>3</sup>meujin@gmail.com, <sup>4</sup>art@ugatu.su

**Language:** Russian.

**Source:** Vestnik UGATU (scientific journal of Ufa State Aviation Technical University), vol. 22, no. 2 (80), pp. 105-112, 2018. ISSN 2225-2789 (Online), ISSN 1992-6502 (Print).

**Abstract:** Performance evaluation of Intel Xeon CPUs and NVIDIA Tesla GPUs in solving of sparse linear systems using BiCGStab iteration method with BILU(0) preconditioner is performed; it is shown that using of GPUs allows to decrease linear solve time in 4,3–7,7 times. AIPS method approximating inverse matrix using Neumann power series and its applicability as first stage of two-stage CPR preconditioner is considered. Effectiveness evaluation of linear solve procedure with different preconditioners (BILU(0), CPR-AMG, CPR-AIPS) on computing system with two NVIDIA Tesla P100 GPUs is performed; it is shown that due to parallel implementation of AIPS method CPR-AIPS preconditioner has good scalability that leads to decrease of solve time for majority of test linear systems.

**Key words:** graphics processors; iterative methods; parallel computing; preconditioners; sparse matrices.

**About authors:**

**GAZIZOV, Rafail Kavyevich**, Prof., head of Dep. of high-performance computing technologies and systems. Dipl. mathematician (BGU, 1983). Dr. Sci. (Phys.-Math). Research in group analysis of differential equations and mathematical modelling.

**GUBAIDULLIN, Ratmir Rasilevich**, grad. student, "Applied Mathematics and Informatics" direction. Bachelor of Applied Mathematics and Informatics (UGATU, 2016).

**REPIN, Nikita Vyacheslavovich**, grad. student, "Applied Mathematics and Informatics" direction. Bachelor of Applied Mathematics and Informatics (UGATU, 2016).