Разработка и анализ высокопроизводительных алгоритмов решения кооперативных игр

Толмачёв Дмитрий, гр. 6057/2

Теория игр

- Изучает процессы, при которых каждая из сторон ведет борьбу за достижение своих интересов, используя определенную стратегию из набора
- Стратегия – это полный план действий стороны при всевозможных ситуациях
- Набор стратегий – множество единственных стратегий для каждой из сторон

Немного истории

- XIX в: Задачи производства и ценообразования. А.Курно, Ж.Бертран
- XX в: Теория игр и экономическое поведение. Д.Нейман, О.Моргенштерн
- 50-е: Исследования антагонистических игр. Д.Нэш
- Исследование “Стратегия конфликта” Т.Шеллинга(Нобелевская премия, 2005)
- “Люди, которые играют в игры”, Эрик Берн

Кооперативные игры

- Стороны могут объединяться в коалиции, принимать совместные решения и перераспределять выигрыш между собой
- Результат решения кооперативной игры: оптимальная коалиция, оптимальные смешанные стратегии участников и выигрыш коалиции

Формальная постановка задачи

- Кооперативная игра: $\Gamma = (N, v)$
- Конечное множество игроков: $N = \{1, \ldots, n\}$
- Характеристическая функция:
  наиболее уверенно получаемый выигрыш для коалиции

$$\forall K \in 2^N : K \to R$$

- Выигрыш коалиции зависит от стратегии каждого из участников
- Найти оптимальную характеристическую функцию на основе стратегий участников

Биматричные игры

- Биматричная игра: $\Gamma_b = (N, B)$
- Конечное множество игроков: $N = \{1, \dots, n\}$
- Набор биматричных игр:

$$B = \{B_{xy} | x \in N, y \in N, x \neq y\}$$

$$B_{xy} = \begin{pmatrix} b_{11} & \cdots & b_{1k} \\ \vdots & \ddots & \vdots \\ b_{k1} & \cdots & b_{kk} \end{pmatrix}$$

b_{ij} - выигрыш x со стратегией i в игре с y со стратегией j

Биматричные игры

- Все, кто не с нами против нас:

$$P = N \setminus S = \{p \in N, p \notin S\}$$

- Выигрыш коалиции $S: v(S) = |A_{SP}|$ - решение матричной игры A_{SP} в смешанных стратегиях

$$A_{SP} = \begin{pmatrix} a_{11} & \cdots & a_{1k^{n-m}} \\ \vdots & \ddots & \vdots \\ b_{k^m 1} & \cdots & b_{k^m k^{n-m}} \end{pmatrix}$$

Алгоритм решения

- Вычислить каждый элемент a_{ij} по правилам
- $(i-1)_{10} = (c_m c_{m-1} \dots c_1)_k$,
  тогда $c_\nu + 1, \nu = \overline{1,m}$ - стратегия $\nu \in S$
- $(j-1)_{10} = (c_{n-m} c_{n-m-1} \dots c_1)_k$,
  тогда $c_w + 1, \nu = \overline{1,n-m}$ - стратегия $w \in P$

$$a_{ij} = \sum_{\substack{b_{c_x c_y} \in B_{xy}, \\ x,y \in S, x \neq y}} b_{c_x c_y} + \sum_{\substack{b_{c_x c_y} \in B_{xy}, \\ x \in S, y \in P}} b_{c_x c_y}$$

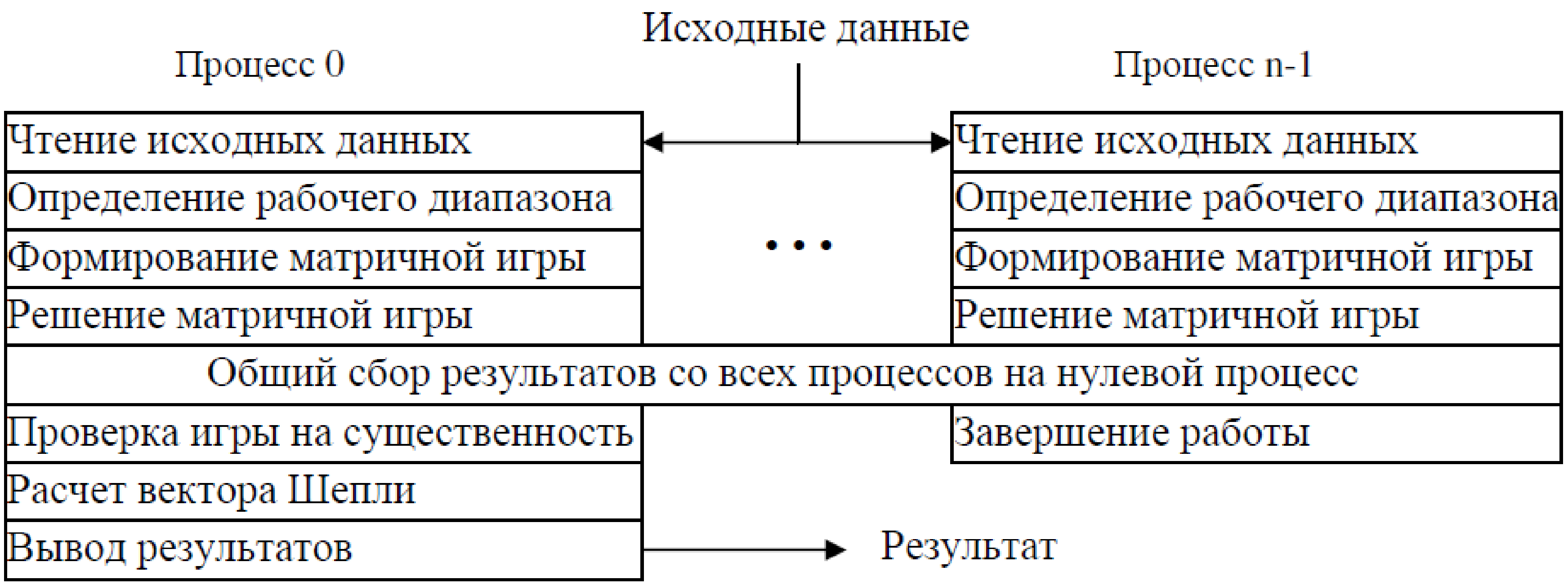
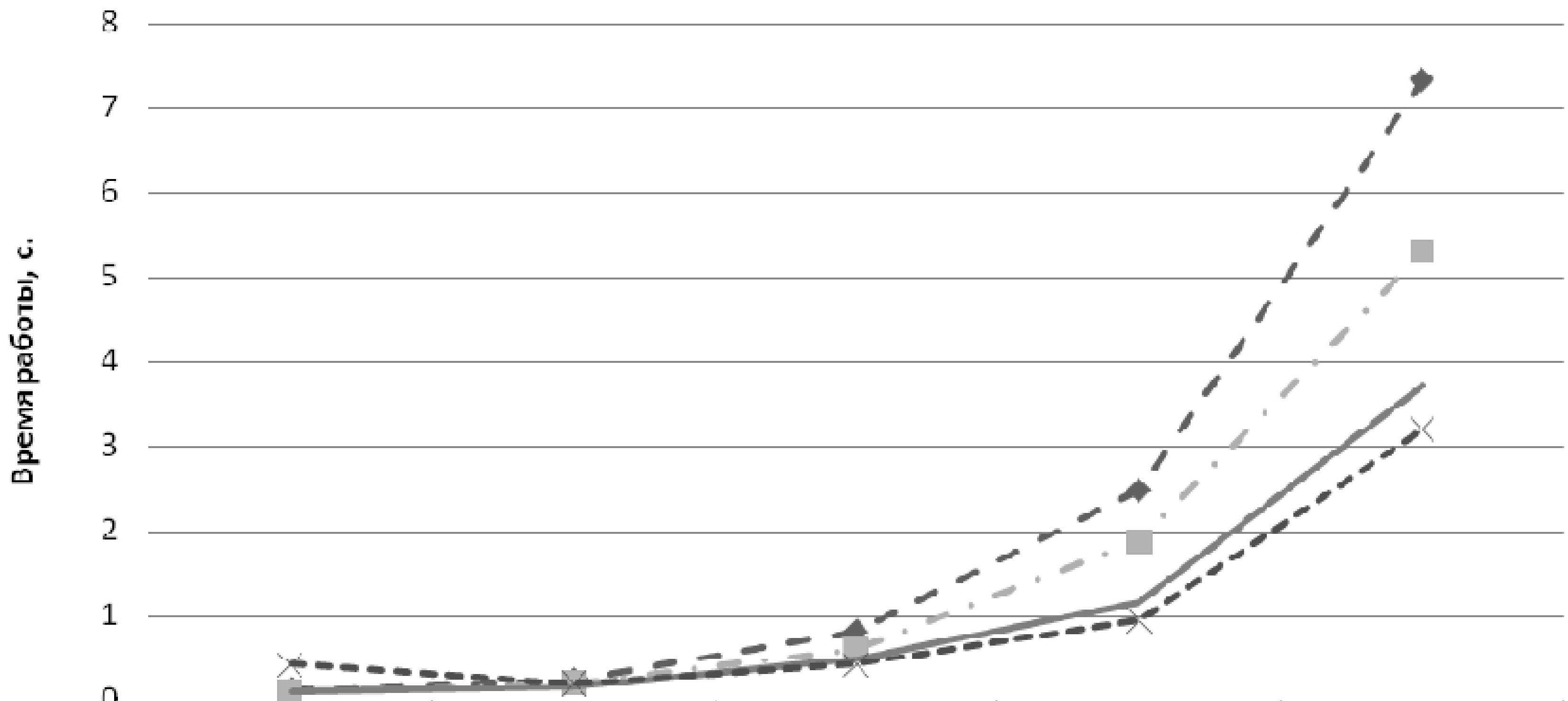


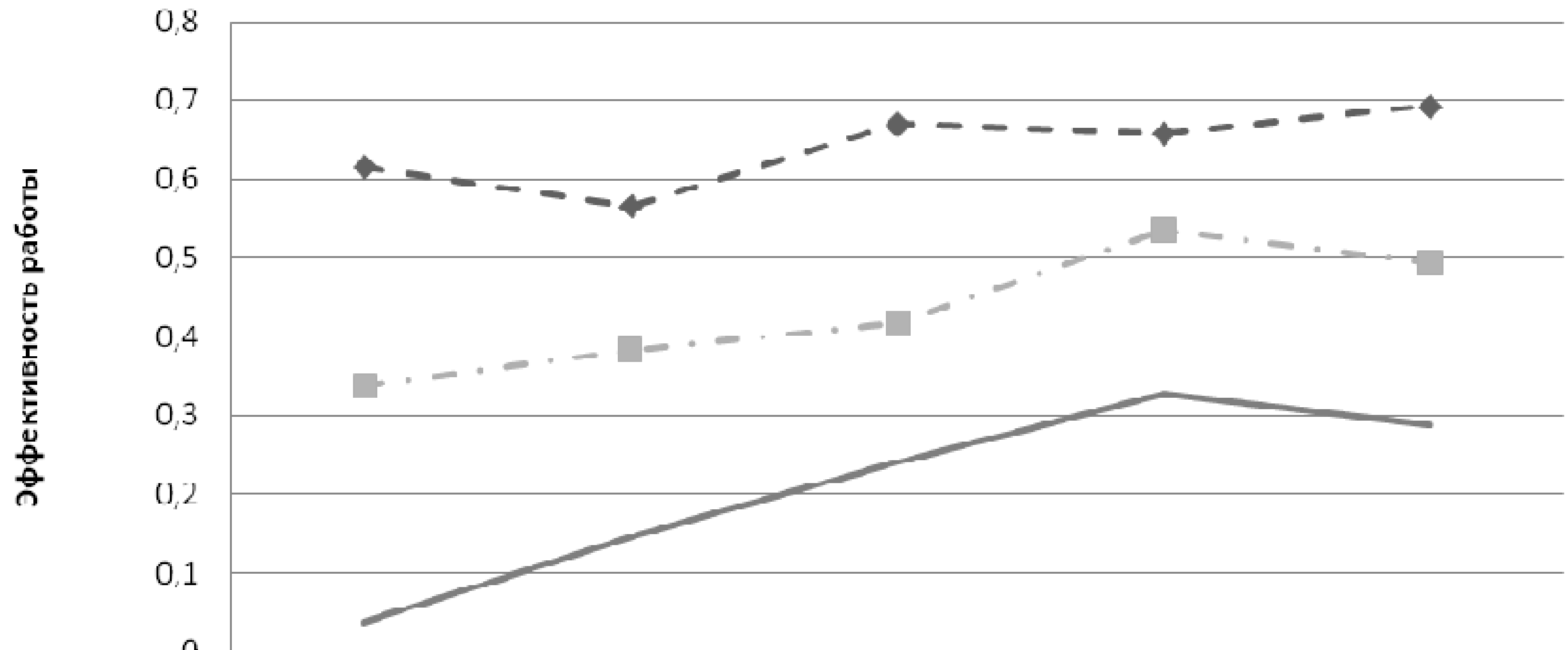
Рис. 1. Схема работы параллельного алгоритма

Результаты



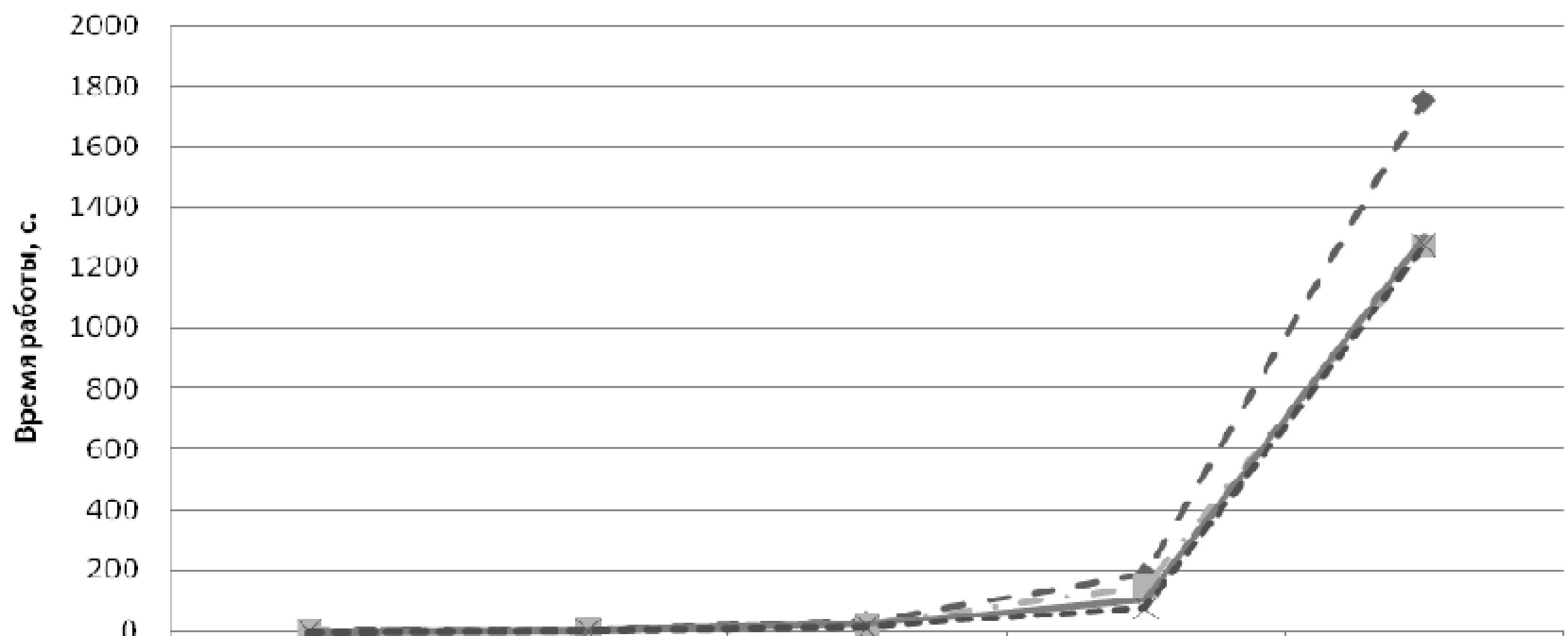
	4 стратегии	5 стратегий	6 стратегий	7 стратегий	8 стратегий
1	0,1228	0,2254	0,8023	2,4732	7,3355
2	0,0997	0,1993	0,599	1,8774	5,2963
4	0,0911	0,1474	0,4805	1,1566	3,7116
8	0,4156	0,1932	0,4165	0,9472	3,19

Результаты



	4 стратегии	5 стратегий	6 стратегий	7 стратегий	8 стратегий
2 ядра	0,615847543	0,565479177	0,669699499	0,658676894	0,692511753
4 ядра	0,336992316	0,38229308	0,417429761	0,534584126	0,494092844
8 ядер	0,036934552	0,145833333	0,240786315	0,326383021	0,287441223

Результаты



	4 стратегии	5 стратегий	6 стратегий	7 стратегий	8 стратегий
1	0,5717	2,8744	30,7895	187,4967	1755,5704
2	0,4173	2,2582	25,0761	145,6267	1266,913
4	0,3315	1,5705	21,04	101,4809	1306,8114
8	0,2452	1,1176	13,0794	75,4896	1275,1027

Выводы

- Эффективность – 40%
- Низкий показатель эффективности алгоритма, по причине недостаточной балансировки