УДК 621.317.7

Б. Ф. Мельников, С. В. Пивнева, М. А. Трифонов

ОЦЕНКА АЛГОРИТМОВ РАСЧЕТА РАССТОЯНИЯ СТРОК ДНК

Аннотация.

Актуальность и цели. Часто требуется измерить различие или расстояние между двумя строками (например, в эволюционных, структуральных или функциональных исследованиях биологических строк). Так как строковые последовательности митохондриальных ДНК приблизительно составляют 17 000 символов {a, g, c, t}, то для решения поставленной задачи были выбраны алгоритмы нечеткого сравнения, рассчитывающие расстояние за полиноминальное время. В рамках исследования при расчете метрик различными ранее известными алгоритмами неточного сравнения строк были получены различные результаты. Цель исследования: разработка методов качественной оценки полученных результатов. Разработка качественных оценок позволит сделать выбор более приемлемого алгоритма, что улучшит исследования в различных предметных областях.

Материалы и методы. В качестве метода исследования применятся теория треугольной нормы в метрическом пространстве.

Результаты. Исходные данные были получены из банка данных NCBI и случайным образом выбраны 30 строковых последовательностей митохондриальных ДНК. В результате работы алгоритмов сравнения 30 строковых последовательностей приведены качественные оценки.

Выводы. По полученным качественным оценкам метрик был определен наилучший алгоритм сравнения строковых последовательностей.

Ключевые слова: метрическая оценка, алгоритмы, мультиэвристический подход.

B. F. Mel'nikov, S. V. Pivneva, M. A. Trifonov

ESTIMATION OF ALGORITHMS FOR CALCULATION OF DISTANCE BETWEEN DNA LINES

Abstract.

Background. Often it is required to measure distinction or distance between two lines (for example, in evolutionary, structural or functional researches of biological lines). As line sequences of mitochondrial DNA approximately make 17 000 symbols {a, g, c, t}, in order to solve the set problem the authors chose objective algorithms of indistinct comparison that calculate the distance in polynomial time. In the research, when calculating the metrics of the known algorithms of inexact comparison of lines, there were received various results. The work purpose is to develop the methods of qualitative assessment of the received results. Development of qualitative assessment will allow to choose the most acceptable algorithm that will improve researches in various subject areas.

Materials and methods. The theory of triangular norm in metric space was used as a method of research.

Results. The initial data were obtained from the NCBI databank, and 30 line sequences of mitochondrial DNA were randomly chosen. As a result of perfomance of algorithms of comparison of 30 line sequences the authors adduced qualitative estimates.

Conclusions. Using the obtained qualitative estimates of metrics the best algorithm of comparison of line sequences has been determined.

Key words: metric evaluation, algorithms, multiheuristic approach.

Введение

Часто требуется измерить различие или расстояние между двумя строками (например, в эволюционных, структуральных или функциональных исследованиях биологических строк) [1]. Строки генетических последовательностей являются длинными строками, состоящими из символов {a, c, g, t} и могут достигать десятки миллионов символов. В таких случаях время получения точного сравнения строк может оказаться слишком большим, поэтому для сравнения используют алгоритмы неточного поиска ([2, 3] и др.).

Все известные алгоритмы неточного сравнения строк дают различные результаты, в этой связи необходимо давать оценку полученных результатов.

Для тестирования алгоритмов были использованы митохондриальные ДНК различных организмов. Эти молекулы ДНК, содержащиеся в митохондриях клетки, не подвержены рекомбинации и наследуются по материнской линии у большинства многоклеточных организмов, поэтому их изменение может происходить только за счет мутации. Была использована генетическая информация следующих организмов [4]: 1) Bison bison (бизон); 2) Bos taurus (дикий бык); 3) Canis lupus(волк); 4) Drosophila simulans (дрозофила); 5) Felis catus (кошка); 6) Gadus morhua (атлантическая треска); 7) Gallus gallus (курица); 8) His1 virus complete genome (вирус H1); 9) Homo sapiens (человек разумный); 10) Mus musculus (домовая мышь); 11) Orcaella brevirostris (иравадийский дельфин); 12) Orcinus orca (косатка); 13) Pan troglodytes (обыкновенный шимпанзе); 14) Peponocephala electra (широкомордый дельфин); 15) Rattus norvegicus (серая крыса); 16) Sus scrofa taiwanensis (кабан); 17) Rhincodon (китовая акула); 18) Equus caballus (домашняя лошадь); 19) Gorilla gorilla (западная горилла); 20) Danio rerio (данио-рерио); 21) Anopheles gambiae (комар); 22) Chrysemys picta bellii (расписная черепаха); 23) Apalone spinifera (чешуеспинная лебеда); 24) Tursiops truncatus (афалина); 25) Pantholops hodgsonii (Оронго); 26) Alligator sinensis (китайский аллигатор); 27) Octodon degus (дегу); 28) Trichechus manatus (американский ламантин); 29) Esox lucius (щука); 30) Echinops telfairi (малая тенрека).

На рис. 1–5 представлена метрическая оценка в процентах тестируемых алгоритмов в виде матрицы. Нами рассматривались следующие алгоритмы: мультиэвристический алгоритм, алгоритм Джаро – Винклера, расстояние Дамерау – Левенштейна и расстояние Дамерау – Левенштейна с использованием оценочной матрицы. Далее приведены краткие описания этих алгоритмов.

1. Применение мультиэвристического подхода

Мультиэвристический подход ([5–7] и др.) – новый подход к решению задач дискретной оптимизации, который заключается в сочетании незавершенного метода ветвей и границ с комбинацией различных эвристик, на результатах работы которых основывается выбор очередного шага. Оценки, полученные эвристиками, усредняются с помощью динамических функций рис-

ка. Для подбора коэффициентов усреднения применяются генетические алгоритмы, упрощенное самообучение которыми применяется также и для старта незавершенного метода ветвей и границ. Была использована эвристика выборов траектории, для которой выражение $(i'–i) + (j'–j)$ принимает минимальное либо близкое к минимальному значение. Например, сначала рассматриваем все траектории со сдвигом только одной из строк на один символ; затем – со сдвигом одной из строк на два символа или обеих на один символ и т.д.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	100	89	71	40	74	59	58	37	69	68	73	72	67	73	68	72	62	72	67	59	41	61	62	73	81	56	67	67	59	66
2	89	100	74	40	71	61	56	36	66	70	76	75	70	76	70	68	64	75	69	56	41	64	65	76	85	59	70	70	62	68
3	71	74	100	39	71	60	54	36	64	68	71	71	68	71	68	64	63	75	67	53	40	63	64	71	73	58	70	70	60	68
4	40	40	39	100	44	42	41	39	41	45	44	44	42	44	44	44	45	43	42	43	74	45	45	44	44	41	45	43	41	45
5	74	71	71	44	100	56	59	36	67	64	67	67	64	68	65	71	59	69	63	58	39	58	59	68	69	53	64	64	56	62
6	59	61	60	42	56	100	53	35	56	60	60	60	59	61	60	53	66	61	59	59	38	62	62	61	61	57	60	60	68	60
7	58	56	54	41	59	53	100	34	58	54	54	54	54	54	54	59	55	54	54	58	38	57	57	54	54	53	53	53	54	52
8	37	36	36	39	36	35	34	100	41	43	42	42	41	42	42	43	43	42	41	42	42	42	42	42	42	40	42	42	40	42
9	69	66	64	41	67	56	58	41	100	63	65	65	84	65	64	66	58	66	82	57	38	58	59	65	66	55	63	63	57	61
10	68	70	68	45	64	60	54	43	63	100	69	69	68	70	80	63	64	70	67	56	42	64	65	70	70	58	69	68	61	67
11	73	76	71	44	67	60	54	42	65	69	100	93	69	94	69	67	64	74	69	55	41	64	64	93	76	58	69	69	62	67
12	72	75	71	44	67	60	54	42	65	69	93	100	69	93	69	66	64	74	69	55	41	63	64	93	76	58	68	69	61	67
13	67	70	68	42	64	59	54	41	84	68	69	69	100	69	68	61	62	70	87	53	39	62	63	68	69	58	68	68	60	65
14	73	76	71	44	68	61	54	42	65	70	94	93	69	100	69	67	64	74	69	55	41	64	64	93	76	59	69	69	61	67
15	68	70	68	44	65	60	54	42	64	80	69	69	68	69	100	63	64	71	68	55	41	64	65	69	71	59	70	68	61	67
16	72	68	64	44	71	53	59	43	66	63	67	66	61	67	63	100	56	66	61	62	41	56	57	66	67	51	61	61	53	59
17	62	64	63	45	59	66	55	43	58	64	64	64	62	64	64	56	100	63	60	58	41	64	65	62	63	58	63	62	65	61
18	72	75	75	43	69	61	54	42	66	70	74	74	70	74	71	66	63	100	69	54	39	64	64	73	75	58	70	72	61	68
19	67	69	67	42	63	59	54	41	82	67	69	69	87	69	68	61	60	69	100	54	39	62	63	69	70	58	68	68	60	65
20	59	56	53	43	58	59	58	42	57	56	55	55	53	55	55	62	58	54	54	100	40	56	56	55	55	51	53	53	59	54
21	41	41	40	74	39	38	38	42	38	42	41	41	39	41	41	41	41	39	39	40	100	44	44	43	43	40	44	42	40	44
22	61	64	63	45	58	62	57	42	58	64	64	63	62	64	64	56	64	64	62	56	44	100	73	61	62	60	62	61	61	61
23	62	65	64	45	59	62	57	42	59	65	64	64	63	64	65	57	65	64	63	56	44	73	100	62	63	60	63	62	61	61
24	73	76	71	44	68	61	54	42	65	70	93	93	68	93	69	66	62	73	69	55	43	61	62	100	76	58	69	69	61	67
25	81	85	73	44	69	61	54	42	66	70	76	76	69	76	71	67	63	75	70	55	43	62	63	76	100	58	70	70	61	68
26	56	59	58	41	53	57	53	40	55	58	58	58	58	59	59	51	58	58	58	51	40	60	60	58	58	100	57	58	58	56
27	67	70	70	45	64	60	53	42	63	69	69	68	68	69	70	61	63	70	68	53	44	62	63	69	70	57	100	69	60	67
28	67	70	70	43	64	60	53	42	63	68	69	69	68	69	68	61	62	72	68	53	42	61	62	69	70	58	69	100	59	68
29	59	62	60	41	56	68	54	40	57	61	62	61	60	61	61	53	65	61	60	59	40	61	61	61	61	58	60	59	100	60
30	66	68	68	45	62	60	52	42	61	67	67	67	65	67	67	59	61	68	65	54	44	61	61	67	68	56	67	68	60	100

Рис. 1. Метрическая оценка мультиэвристического алгоритма

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	100	88	86	81	87	84	85	80	84	86	87	87	85	87	87	87	86	86	85	88	80	87	86	87	88	83	86	85	84	85
2	88	100	91	78	88	86	83	84	86	91	91	92	90	91	91	87	87	90	89	86	78	91	90	91	91	83	90	90	86	90
3	86	91	100	79	88	87	83	84	85	91	92	90	89	92	92	85	88	91	89	86	78	91	90	92	92	84	92	91	86	92
4	81	78	79	100	78	76	82	77	75	80	79	78	76	79	79	79	81	77	75	80	93	79	79	79	79	74	80	76	75	80
5	87	88	88	78	100	86	83	82	86	87	88	88	86	88	88	86	87	87	86	86	78	88	87	88	88	84	87	87	87	86
6	84	86	87	76	86	100	82	84	84	85	86	86	84	86	85	83	91	85	84	85	76	85	84	86	85	83	86	86	89	86
7	85	83	83	82	83	82	100	78	87	82	83	84	86	83	83	83	82	85	87	84	82	83	83	83	83	86	82	85	83	82
8	80	84	84	77	82	84	78	100	81	84	84	84	83	84	84	80	81	84	83	82	76	84	83	84	84	80	83	84	83	84
9	84	86	85	75	86	84	87	81	100	84	86	86	89	86	86	84	85	87	89	83	75	86	86	86	86	87	85	87	86	84
10	86	91	91	80	87	85	82	84	84	100	92	90	88	92	92	86	88	91	88	86	80	92	91	92	92	82	92	90	85	91
11	87	91	92	79	88	86	83	84	86	92	100	91	89	93	92	86	88	91	89	86	79	92	91	93	92	83	92	91	86	91
12	87	92	90	78	88	86	84	84	86	90	91	100	90	92	91	86	87	91	90	85	78	91	90	91	91	84	90	90	86	89
13	85	90	89	76	86	84	86	83	89	88	89	90	100	89	89	85	85	90	94	83	75	89	89	89	90	86	89	91	86	88
14	87	91	92	79	88	86	83	84	86	92	93	92	89	100	92	87	87	92	89	85	78	92	91	93	92	83	92	91	86	91
15	87	91	92	79	88	85	83	84	86	92	92	91	89	92	100	87	88	91	89	85	79	92	92	92	93	83	92	91	85	91
16	87	87	85	79	86	83	83	80	84	86	86	86	85	87	87	100	86	86	84	84	78	87	87	86	87	88	86	84	83	84
17	86	87	88	81	87	91	82	81	85	88	88	87	85	87	88	86	100	86	85	85	80	87	86	88	88	82	89	86	86	87
18	86	90	91	77	87	85	85	84	87	91	91	91	90	92	91	86	86	100	90	85	76	92	91	92	92	85	91	92	86	90
19	85	89	89	75	86	84	87	83	89	88	89	90	94	89	89	84	85	90	100	83	75	89	89	89	89	87	88	90	86	88
20	88	86	86	80	86	85	84	82	83	86	86	85	83	85	85	84	85	85	83	100	80	85	83	86	85	83	85	85	85	85
21	80	78	78	93	78	76	82	76	75	80	79	78	75	78	79	78	80	76	75	80	100	79	79	78	78	74	80	76	75	80
22	87	91	91	79	88	85	83	84	86	92	92	91	89	92	92	87	87	92	89	85	79	100	92	92	92	83	92	91	85	90
23	86	90	90	79	87	84	83	83	86	91	91	90	89	91	92	87	86	91	89	83	79	92	100	91	92	83	91	91	85	90
24	87	91	92	79	88	86	83	84	86	92	93	91	89	93	92	86	88	92	89	86	78	92	91	100	92	83	92	91	86	91
25	88	91	92	79	88	85	83	84	86	92	92	91	90	92	93	87	88	92	89	85	78	92	92	92	100	83	92	91	86	91
26	83	83	84	74	84	83	86	80	87	82	83	84	86	83	83	88	82	85	87	83	74	83	83	83	83	100	82	86	85	82
27	86	90	92	80	87	86	82	83	85	92	92	90	89	92	92	86	89	91	88	85	80	92	91	92	92	82	100	91	85	92
28	85	90	91	76	87	86	85	84	87	90	91	90	91	91	91	84	86	92	90	85	76	91	91	91	91	86	91	100	88	90
29	84	86	86	75	87	89	83	83	86	85	86	86	86	85	83	86	86	86	86	85	75	85	85	86	86	85	85	88	100	85
30	85	90	92	80	86	86	82	84	84	91	91	89	88	91	91	84	87	90	88	85	80	90	90	91	91	82	92	90	85	100

Рис. 2. Метрическая оценка алгоритма Джаро – Винклера

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	100	85	60	32	63	47	47	32	57	57	62	61	55	62	57	62	50	61	56	48	31	49	51	63	71	44	56	56	47	55
2	85	100	62	32	60	49	45	32	55	59	64	64	58	64	59	59	52	64	57	45	31	51	52	65	75	46	58	58	49	56
3	60	62	100	33	61	48	43	33	52	58	60	59	56	60	58	55	51	64	55	44	32	50	51	60	62	45	58	58	48	57
4	32	32	33	100	32	31	28	29	29	34	32	32	29	32	32	32	34	31	29	32	66	33	33	33	32	28	34	31	30	34
5	63	60	61	32	100	45	48	34	56	55	59	58	54	59	56	61	48	60	54	48	32	48	49	59	60	43	54	54	46	53
6	47	49	48	31	45	100	41	32	43	48	49	48	45	49	47	42	53	47	46	49	30	48	48	49	48	43	47	47	56	48
7	47	45	43	28	48	41	100	31	48	45	45	45	45	44	45	48	44	44	45	45	28	47	47	45	45	43	42	43	43	41
8	32	32	33	29	34	32	31	100	31	32	32	32	31	32	32	33	34	32	30	33	30	33	32	32	32	31	33	32	31	33
9	57	55	52	29	56	43	48	31	100	52	54	54	77	54	54	54	46	55	75	45	28	47	48	54	54	44	52	52	45	49
10	57	59	58	34	55	48	45	32	52	100	58	57	55	59	70	53	53	58	55	46	33	52	52	58	59	45	58	57	48	56
11	62	64	60	32	59	49	45	32	54	58	100	86	56	88	58	57	52	63	56	45	32	51	52	88	65	46	58	58	49	56
12	61	64	59	32	58	48	45	32	54	57	86	100	56	87	58	56	52	62	56	44	31	51	52	87	65	46	56	57	48	55
13	55	58	56	29	54	45	45	31	77	55	56	56	100	57	57	51	49	58	80	42	28	49	50	57	58	46	55	56	47	52
14	62	64	60	32	59	49	44	32	54	59	88	87	57	100	59	57	52	63	57	45	31	51	52	87	65	46	58	58	49	55
15	57	59	58	32	56	47	45	32	54	70	58	58	57	59	100	54	52	59	56	45	32	52	53	59	60	46	58	57	48	56
16	62	59	55	32	61	42	48	33	54	53	57	56	51	57	54	100	46	58	51	50	31	46	47	57	58	41	52	52	43	50
17	50	52	51	34	48	53	44	34	46	53	52	52	49	52	52	46	100	51	49	48	33	52	53	52	52	45	52	50	53	51
18	61	64	64	31	60	47	44	32	55	58	63	62	58	63	59	58	51	100	58	43	30	52	52	62	64	46	58	61	49	56
19	56	57	55	29	54	46	45	30	75	55	56	56	80	57	56	51	49	58	100	43	28	50	50	57	57	46	55	55	47	52
20	48	45	44	32	48	49	45	33	45	46	45	44	42	45	45	50	48	43	43	100	31	45	45	45	44	40	43	42	48	43
21	31	31	32	66	32	30	28	30	28	33	32	31	28	31	32	31	33	30	28	31	100	32	32	32	31	27	33	30	29	33
22	49	51	50	33	48	48	47	33	47	52	51	51	49	51	52	46	52	52	50	45	32	100	62	51	52	48	50	49	50	50
23	51	52	51	33	49	48	47	32	48	52	52	52	50	52	53	47	53	52	50	45	32	62	100	52	53	47	51	50	49	49
24	63	65	60	33	59	49	45	32	54	58	88	87	57	87	59	57	52	62	57	45	32	51	52	100	66	46	58	58	49	56
25	71	75	62	32	60	48	45	32	54	59	65	65	58	65	60	58	52	64	57	44	31	52	53	66	100	46	59	58	49	56
26	44	46	45	28	43	43	43	31	44	45	46	46	46	46	46	41	45	46	46	40	27	48	47	46	46	100	44	45	45	44
27	56	58	58	34	54	47	42	33	52	58	58	56	55	58	58	52	52	58	55	43	33	50	51	58	59	44	100	57	47	56
28	56	58	58	31	54	47	43	32	52	57	58	57	56	58	57	52	50	61	55	42	30	49	50	58	58	45	57	100	47	57
29	47	49	48	30	46	56	43	31	45	48	49	48	47	49	48	43	53	49	47	48	29	50	49	49	49	45	47	47	100	47
30	55	56	57	34	53	48	41	33	49	56	56	55	52	55	56	50	51	56	52	43	33	50	49	56	56	44	56	57	47	100

Рис. 3. Метрическая оценка алгоритма «Расстояние Дамерау – Левенштейна»

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	100	90	76	53	78	67	67	52	74	73	76	76	72	77	73	77	70	76	72	68	54	69	70	77	82	66	73	73	67	72
2	90	100	77	53	76	68	66	52	72	74	78	77	74	78	74	74	71	78	73	66	54	70	71	78	84	67	74	75	68	73
3	76	77	100	53	76	67	64	51	70	72	74	74	72	74	72	71	70	77	71	64	53	69	69	74	76	65	74	74	67	73
4	52	52	53	100	59	58	56	54	56	59	59	58	56	58	58	58	60	58	56	58	80	59	59	59	58	56	60	58	57	60
5	85	83	85	59	100	65	67	51	71	70	72	72	70	72	71	74	67	74	69	66	52	67	67	72	73	64	71	71	65	69
6	64	66	66	51	65	100	63	51	64	66	67	67	65	67	66	63	71	67	65	68	52	68	67	67	67	65	67	67	72	67
7	66	65	65	50	68	63	100	50	67	64	64	64	65	64	64	67	65	65	65	65	50	67	67	64	65	64	64	65	64	63
8	51	50	51	48	52	51	50	100	58	59	59	58	58	58	58	59	60	59	57	59	57	59	59	59	59	58	59	59	58	59
9	83	81	80	58	84	74	77	58	100	69	71	71	86	71	70	71	67	72	84	66	51	67	68	71	71	65	70	71	66	68
10	72	73	73	53	72	67	65	51	69	100	74	74	72	74	81	71	71	74	72	66	55	71	71	74	75	66	74	74	68	73
11	76	78	76	54	75	69	66	52	72	74	100	91	73	92	74	73	71	77	73	66	54	70	70	92	78	67	74	74	68	72
12	76	77	75	53	75	68	66	52	71	73	91	100	73	92	73	73	70	77	73	65	54	70	70	92	78	67	73	74	68	72
13	72	74	73	52	72	67	66	51	87	72	73	73	100	73	72	69	69	74	87	64	52	69	69	73	73	66	72	73	67	70
14	75	77	75	53	74	68	65	51	71	73	91	91	73	100	74	74	71	77	73	66	54	70	71	92	78	67	74	74	68	72
15	73	74	74	53	73	67	66	51	71	81	74	73	73	74	100	71	71	75	72	66	54	71	71	74	75	67	75	74	68	73
16	77	74	73	53	78	65	69	52	72	71	74	73	70	74	71	100	67	74	69	69	54	66	67	73	74	63	70	70	64	69
17	69	70	71	55	69	72	66	53	67	70	70	70	69	70	70	67	100	69	67	67	54	70	70	69	69	66	70	69	70	68
18	74	76	76	51	75	66	64	50	71	72	75	74	73	75	73	72	69	100	73	64	52	70	70	76	77	66	74	76	68	72
19	71	72	71	50	71	65	65	50	83	70	71	71	86	72	71	68	67	73	100	64	52	69	69	73	74	66	72	73	67	70
20	67	66	66	53	69	69	67	53	66	66	66	66	64	66	66	69	69	65	64	100	53	66	65	65	65	63	64	64	68	64
21	53	53	54	72	54	52	51	49	51	54	53	53	52	53	53	53	55	53	51	53	100	58	58	58	58	55	59	57	56	59
22	73	75	75	58	74	74	73	56	72	75	75	75	74	75	75	71	77	76	74	71	58	100	76	68	69	67	69	68	68	68
23	67	68	69	52	68	67	66	50	66	68	68	68	68	68	69	65	70	69	67	64	53	76	100	69	70	67	69	69	68	68
24	75	76	74	52	74	67	64	51	70	72	90	90	72	90	72	72	69	75	71	64	53	68	69	100	79	67	74	74	68	72
25	82	84	77	53	76	68	66	52	72	74	78	78	74	78	75	74	71	78	73	66	54	71	71	79	100	66	75	75	68	73
26	65	66	66	50	66	65	65	51	65	65	66	66	67	66	66	63	67	67	66	63	51	68	68	66	66	100	65	66	66	64
27	71	73	74	54	72	67	64	51	69	73	72	72	72	73	73	69	70	74	71	64	54	69	69	73	73	65	100	73	67	72
28	71	73	74	51	72	67	64	51	70	71	72	72	72	72	72	69	69	75	71	63	52	68	69	72	73	66	73	100	67	72
29	65	66	67	50	66	71	64	50	65	65	66	66	66	66	66	63	70	67	65	67	51	68	67	66	67	65	66	67	100	67
30	70	72	73	53	70	67	63	51	68	71	71	71	69	71	71	68	69	72	69	64	54	68	68	71	72	64	72	73	67	100

Рис. 4. Метрическая оценка алгоритма «Расстояние Дамерау – Левенштейна с использованием оценочной матрицы»

1. Сдвигаем ту строку, текущий символ которой реже встречается в другой строке. Для этой эвристики желательно знать вероятности появления символов в каждой из строк. Если они заранее не известны, можно считать их равными. В процессе работы можно корректировать вероятности или использовать алгоритм старения, чтобы вероятность символа определялась по некоторому фрагменту перед текущим символом, а не по целой строке. Если

вероятности для текущих символов окажутся равными, сдвигается строка, в которой осталось больше символов.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	**100**	**90**	**72**	**34**	**75**	**61**	**63**	**36**	**72**	**68**	**73**	**73**	**70**	**74**	**69**	**74**	**63**	**74**	**70**	**61**	**35**	**63**	**63**	**74**	**81**	**60**	**68**	**70**	**63**	**67**
2	**90**	**100**	**74**	**34**	**73**	**63**	**61**	**36**	**70**	**70**	**75**	**75**	**72**	**76**	**71**	**71**	**64**	**77**	**71**	**58**	**34**	**65**	**65**	**76**	**84**	**62**	**70**	**72**	**64**	**68**
3	**72**	**74**	**100**	**33**	**73**	**61**	**59**	**35**	**67**	**68**	**71**	**71**	**70**	**71**	**69**	**67**	**63**	**76**	**69**	**56**	**34**	**63**	**64**	**71**	**73**	**61**	**69**	**73**	**63**	**68**
4	**33**	**33**	**33**	**100**	**43**	**43**	**42**	**40**	**42**	**44**	**44**	**44**	**43**	**44**	**43**	**43**	**44**	**43**	**43**	**43**	**76**	**44**	**44**	**44**	**44**	**42**	**44**	**43**	**43**	**44**
5	**97**	**94**	**96**	**43**	**100**	**58**	**63**	**35**	**69**	**64**	**68**	**68**	**66**	**68**	**66**	**71**	**59**	**71**	**66**	**59**	**33**	**60**	**60**	**68**	**70**	**57**	**65**	**68**	**60**	**63**
6	**58**	**60**	**60**	**32**	**58**	**100**	**59**	**35**	**59**	**60**	**61**	**61**	**61**	**61**	**60**	**56**	**66**	**62**	**61**	**62**	**33**	**62**	**62**	**61**	**61**	**60**	**60**	**62**	**70**	**60**
7	**61**	**59**	**58**	**32**	**64**	**59**	**100**	**33**	**61**	**55**	**55**	**56**	**58**	**56**	**56**	**59**	**56**	**57**	**58**	**57**	**31**	**59**	**59**	**56**	**56**	**57**	**54**	**56**	**57**	**53**
8	**33**	**33**	**33**	**28**	**33**	**33**	**33**	**100**	**43**	**42**	**43**	**43**	**43**	**43**	**43**	**43**	**43**	**43**	**43**	**43**	**37**	**43**	**43**	**43**	**43**	**43**	**43**	**43**	**43**	**42**
9	**86**	**83**	**82**	**39**	**87**	**73**	**80**	**43**	**100**	**62**	**65**	**65**	**85**	**65**	**64**	**66**	**58**	**67**	**83**	**57**	**32**	**59**	**60**	**65**	**66**	**59**	**63**	**65**	**60**	**61**
10	**63**	**65**	**64**	**32**	**63**	**57**	**56**	**33**	**62**	**100**	**71**	**71**	**71**	**71**	**81**	**66**	**65**	**72**	**70**	**59**	**36**	**65**	**66**	**71**	**72**	**62**	**70**	**71**	**64**	**68**
11	**75**	**77**	**74**	**35**	**73**	**64**	**62**	**37**	**71**	**71**	**100**	**92**	**71**	**93**	**70**	**69**	**64**	**76**	**71**	**58**	**35**	**65**	**65**	**92**	**76**	**62**	**69**	**72**	**64**	**67**
12	**73**	**75**	**72**	**34**	**71**	**63**	**61**	**36**	**70**	**69**	**92**	**100**	**71**	**91**	**69**	**69**	**63**	**75**	**71**	**57**	**34**	**64**	**64**	**91**	**75**	**62**	**68**	**71**	**64**	**67**
13	**69**	**71**	**71**	**33**	**69**	**63**	**63**	**36**	**90**	**68**	**70**	**71**	**100**	**68**	**67**	**62**	**61**	**70**	**87**	**54**	**32**	**62**	**63**	**67**	**69**	**61**	**66**	**69**	**62**	**63**
14	**69**	**71**	**68**	**32**	**67**	**59**	**57**	**34**	**66**	**65**	**87**	**87**	**68**	**100**	**70**	**69**	**64**	**76**	**71**	**58**	**34**	**64**	**65**	**92**	**76**	**62**	**69**	**72**	**64**	**67**
15	**69**	**70**	**70**	**33**	**68**	**62**	**61**	**35**	**68**	**78**	**69**	**70**	**71**	**70**	**100**	**66**	**64**	**73**	**70**	**58**	**35**	**65**	**66**	**70**	**72**	**62**	**70**	**71**	**64**	**67**
16	**74**	**71**	**69**	**34**	**75**	**58**	**65**	**36**	**70**	**65**	**69**	**70**	**66**	**70**	**66**	**100**	**58**	**70**	**65**	**63**	**34**	**59**	**60**	**69**	**70**	**57**	**64**	**66**	**58**	**62**
17	**62**	**64**	**64**	**34**	**62**	**67**	**61**	**36**	**62**	**63**	**63**	**64**	**64**	**64**	**64**	**58**	**100**	**65**	**63**	**61**	**35**	**65**	**66**	**63**	**64**	**62**	**64**	**65**	**68**	**62**
18	**73**	**76**	**77**	**33**	**74**	**63**	**62**	**36**	**71**	**70**	**75**	**75**	**74**	**75**	**72**	**70**	**65**	**100**	**71**	**55**	**33**	**64**	**64**	**72**	**74**	**61**	**69**	**73**	**63**	**67**
19	**67**	**68**	**67**	**31**	**66**	**60**	**61**	**34**	**85**	**66**	**68**	**68**	**88**	**68**	**67**	**63**	**61**	**71**	**100**	**55**	**32**	**62**	**62**	**68**	**69**	**61**	**66**	**68**	**62**	**64**
20	**58**	**56**	**54**	**32**	**59**	**61**	**60**	**34**	**58**	**55**	**55**	**55**	**55**	**55**	**55**	**60**	**59**	**55**	**55**	**100**	**34**	**59**	**59**	**58**	**58**	**56**	**56**	**57**	**65**	**56**
21	**35**	**34**	**35**	**58**	**34**	**34**	**34**	**31**	**34**	**35**	**34**	**34**	**34**	**35**	**34**	**34**	**35**	**34**	**34**	**34**	**100**	**44**	**44**	**43**	**43**	**42**	**44**	**43**	**43**	**44**
22	**79**	**81**	**82**	**43**	**79**	**81**	**81**	**45**	**80**	**80**	**81**	**82**	**83**	**82**	**82**	**75**	**83**	**84**	**82**	**75**	**44**	**100**	**73**	**63**	**63**	**63**	**62**	**64**	**65**	**61**
23	**61**	**63**	**63**	**33**	**61**	**62**	**63**	**35**	**62**	**62**	**63**	**63**	**64**	**63**	**64**	**58**	**64**	**65**	**64**	**57**	**34**	**73**	**100**	**63**	**64**	**64**	**63**	**64**	**64**	**61**
24	**71**	**74**	**70**	**33**	**69**	**61**	**59**	**35**	**67**	**67**	**89**	**89**	**69**	**90**	**68**	**68**	**62**	**73**	**69**	**56**	**33**	**63**	**63**	**100**	**77**	**62**	**69**	**72**	**64**	**67**
25	**81**	**84**	**75**	**34**	**73**	**63**	**62**	**36**	**70**	**70**	**76**	**76**	**73**	**76**	**71**	**71**	**65**	**77**	**72**	**58**	**34**	**65**	**66**	**77**	**100**	**62**	**70**	**72**	**64**	**68**
26	**59**	**61**	**61**	**32**	**59**	**61**	**62**	**36**	**62**	**59**	**61**	**62**	**64**	**62**	**61**	**56**	**62**	**64**	**63**	**56**	**33**	**64**	**65**	**61**	**62**	**100**	**57**	**59**	**60**	**55**
27	**62**	**64**	**65**	**31**	**62**	**57**	**55**	**33**	**62**	**63**	**64**	**63**	**65**	**64**	**64**	**59**	**59**	**67**	**64**	**51**	**32**	**59**	**60**	**64**	**65**	**57**	**100**	**71**	**63**	**67**
28	**69**	**71**	**73**	**33**	**70**	**63**	**61**	**36**	**69**	**69**	**71**	**71**	**72**	**71**	**70**	**66**	**65**	**76**	**71**	**57**	**34**	**65**	**65**	**71**	**72**	**63**	**71**	**100**	**62**	**67**
29	**59**	**61**	**60**	**31**	**59**	**68**	**59**	**34**	**60**	**59**	**61**	**61**	**62**	**61**	**60**	**56**	**65**	**62**	**61**	**61**	**32**	**63**	**62**	**61**	**61**	**61**	**60**	**62**	**100**	**59**
30	**63**	**65**	**66**	**32**	**63**	**59**	**55**	**34**	**61**	**63**	**64**	**64**	**64**	**64**	**64**	**59**	**59**	**67**	**64**	**53**	**33**	**60**	**59**	**64**	**65**	**57**	**65**	**68**	**59**	**100**

Рис. 5. Метрическая оценка алгоритма Смита – Ватермана с использованием оценочной матрицы BLOSUM50

2. Комбинация двух предыдущих: результирующая оценка позиции складывается из ее оценок первой и второй эвристиками. Для определения оценки второй эвристики суммируются вероятности появления в другой строке для всех символов, которые придется пропустить при сдвиге.

3. Используем алгоритм для поиска наибольшей общей подпоследовательности строк $x[i..i+k]$ и $y[j..j+k]$, где $k \sim 15$. Для сдвига выбираем такие индексы i', j', в которых заканчивается наибольшая общая подпоследовательность. Если не будет найдено ни одной пары одинаковых символов, область поиска увеличивается. При использовании этой эвристики результат будет близок к значению наибольшей общей подпоследовательности.

5. Комбинация третьей и четвертой эвристик: оценка позиции складывается из ее оценок обеими эвристиками. Оценка позиции (i', j') четвертой эвристикой является отношением длины наибольшей общей подпоследовательности строк $x[i..i']$ и $y[j..j']$ к средней длине сдвига строк из позиции (i, j) в позицию (i', j').

6. Используем алгоритм Нидлмана – Вунша [8] для строк $x[i..i+k]$ и $y[j..j+k]$, где $k \sim 15$. Сдвигаем строки в позицию (i', j'), для которой соответствующее значение в таблице алгоритма Нидлмана – Вунша является наибольшим.

7. Комбинация третьей и шестой эвристик: оценка позиции складывается из ее оценок обеими эвристиками. Оценка позиции (i', j') шестой эвристикой является отношением значения в таблице алгоритма Нидлмана – Вунша, соответствующего этой позиции, к средней длине сдвига строк из позиции (i, j) в позицию (i', j').

2. Сравнение работы алгоритмов

Алгоритм Джаро – Винклера (Jaro – Winkler similarity) [9]. Для данных строк № 1 и 2 их сходство задается формулой

$$S = (m/3)\cdot a + (m/3)\cdot b + (m-t)/3\cdot m,$$

здесь m – число соответствующих символов; a – длина строки № 1; b – длина строки № 2; t – число перестановок.

Два символа считаются соответствующими, только если они находятся не дальше чем ($\max(a,b)/2 - 1$). Первый соответствующий символ в строке № 1 сравнивается с первым соответствующим символом в строке № 2; второй соответствующий символ в строке № 1 сравнивается со вторым соответствующим символом в строке № 2 и т.д. Число соответствующих символов, деленное на 2, дает число перестановок.

Расстояние Дамерау – Левенштейна [1]. Некоторым обобщением функции Левенштейна является возможность использования произвольной весовой *стоимости*, приписываемой каждой строковой операции, в том числе и совпадению.

Цены операций могут зависеть от вида операции (вставка, удаление, замена) и/или от участвующих в ней символов, отражая разную вероятность мутаций в биологии, разную вероятность разных ошибок при вводе текста и т.д. В общем случае:

$w(a, b)$ – цена замены символа a на символ b;

$w(\varepsilon, b)$ – цена вставки символа b;

$w(a, \varepsilon)$ – цена удаления символа a.

При произвольных весах операций задача об операционно-взвешенном расстоянии называется задачей о поиске редакционного предписания, которое переводит строку S1 в строку S2 с *минимальным полным весом* операций. На рис. 3 представлена метрическая оценка Дамерау – Левенштейна с произвольными весовыми операциями. На рис. 4 представлена метрическая оценка с оценочной матрицей *phred score* [10]. На рис. 5 представлена метрическая оценка с оценочной матрицей BLOSUM50 [11].

Проверялось выполнение условия неравенства треугольника относительно элементов матрицы. Приведенные исследования показывают, что критерии оценки неравенства треугольника сводят труднорешаемую задачу к более простой «метрической» задаче, что существенно упрощает оценку результатов. В связи с этим для количественной оценки авторами было предложено усиленное условие неравенства треугольников. Условие равнобедренности треугольника ультраметрического пространства удовлетворяет усиленному неравенству треугольника.

3. Ультраметрическое пространство в оценке алгоритмов подсчета метрик ДНК

Ультраметрическое пространство – это пара (M, d), где M – множество, а d: $M \times M \to R$ – вещественнозначная функция на нем, также называемая метрикой, удовлетворяющая следующим условиям:

1) $d(x,y) \geq 0,\ d(x,y) = 0;\ x \leftrightarrow y$ (*положительная определенность*);

2) $d(x,y) = d(y,x)$ (*симметричность*);

3) $d(x,z) \leq \max(d(x,y), d(y,z))$ (*усиленное неравенство треугольника*).

Ультраметрическое пространство отличается от метрического тем, что неравенство треугольника заменено на усиленное неравенство треугольника.

Для перехода от метрического пространства в ультраметрическое пространство в оценке алгоритмов подсчета метрик ДНК необходимо сделать важное допущение: если сопоставить человека, скажем, с кошкой или волком, к которым мы гораздо ближе, чем к дрозофилам, то окажется, что и по ДНК человек более схож с ними. Если отправиться по ветви млекопитающих дальше, к приматам, то по мере приближения к человеку родственные черты с человекообразными (орангутангом, гориллой и шимпанзе) становятся очевидными. Больше всего человек походит на шимпанзе. Если сопоставить ДНК, окажется, что они очень близки. Поэтому наше допущение основывается на предположении о равенстве значений метрик схожих организмов, например, Homo sapiens (Человек разумный) и Pan troglodytes (Обыкновенный шимпанзе) с другими организмами. Из этого следует, что для каждой пары схожести больше или равное схожести ДНК Homo sapiens (Человек разумный) и Pan troglodytes (Обыкновенный шимпанзе) должно выполняться условие равнобедренности треугольника в ультраметрическом пространстве.

Для подсчета количества треугольников в ультраметрическом пространстве с нарушением равнобедренности использовался следующий алгоритм:

1. Выбор значения метрики Homo sapiens (Человек разумный) и Pan troglodytes (Обыкновенный шимпанзе) в качестве наиболее схожих в каждой из таблиц.

2. Выборка пар организмов со значением метрики большее или равное метрике Homo sapiens (Человек разумный) и Pan troglodytes (Обыкновенный шимпанзе). Запись значения выбранной метрики в переменную P1, эта переменная будет вершиной треугольника.

3. Запись индексов строки и столбца выбранной метрики в переменные $i1, j1$.

4. Выборка пар организмов по матрице с индексами $[i,j1]$ и $[j1,i]$, где $i \in (1..30)$, $j \in (1..30)$, $i > j$.

5. Запись в переменные $P2$ и $P3$ значений метрик с индексами $[i,j1]$ и $[i1,j]$. Вершины $P1$, $P2$, $P3$ образуют треугольник.

6. Для каждой из вершин $P1$ проверка выполнения равенства $P2 - P = 0$.

7. Подсчет общего количества треугольников для выбранных организмов.

8. Вычисление оценки точности по формуле, где Pt – количество треугольников с нарушением равнобедренности $P2 - P3 > 0{,}1$; Pn – общее количество треугольников выборки.

В табл. 1–5 приведены результаты оценки работы алгоритмов.

Заключение

Приведенные исследования показывают, что наиболее точным алгоритмом является алгоритм Нидлмана – Вунша (табл. 1), так как риск подсчета неточного результата меньше, чем у остальных исследуемых алгоритмов.

Чем выше показатель «оценка точности», тем выше вероятность подсчета неверного результата алгоритмом. Приемлемым результатом считается оценка точности < 50 %.

Таблица 1

Оценка мультиэвристического алгоритма

Номер столбца	2	12	13	14	14	19	24	24	24	25
Номер строки	1	11	9	11	12	13	11	12	14	2
Метрика	0,893	0,9268	0,8409	0,94	0,93	0,874	0,933	0,927	0,932	0,85
Количество треугольников с нарушением равнобедренности > 0,1	50	6	52	0	16	14	12	26	10	32
Общее количество треугольников	56	56	56	56	56	56	56	56	56	56
Оценка точности, %	89,29	10,714	92,857	0	28,6	25	21,43	46,43	17,86	57,1

Таблица 2

Оценка алгоритма Джаро – Винклера

Номер столбца	3	10	10	11	11	11	12	12	12	12	13	13	13	13
Номер строки	2	2	3	2	3	10	2	3	10	11	2	3	9	10	11
Метрика	0,9	1	1	1	1	1	1	1	1	1	1	0,9	1	1	1
Количество треугольников с нарушением равнобедренности > 0,1	32	52	32	36	26	36	4	34	48	40	50	56	38	50	54
Общее количество треугольников	56	56	56	56	56	56	56	56	56	56	56	56	56	56	56
Оценка точности, %	57	93	57	64	46	64	7	61	86	71	89	100	68	89	96

Таблица 3

Оценка алгоритма «Расстояние Дамерау – Левенштейна»

Номер столбца	2	12	13	14	14	19	24	24	24
Номер строки	1	11	9	11	12	13	11	12	14
Метрика	0,846	0,86	0,77	0,88	0,867	0,796	0,877	0,87	0,875
Количество треугольников с нарушением равнобедренности > 0,1	50	26	52	2	36	18	6	30	8
Общее количество треугольников	56	56	56	56	56	56	56	56	56
Оценка точности, %	89,29	46,4	92,9	3,57	64,29	32,14	10,71	53,6	14,29

В результате можно сделать заключение о порядке точности в порядке убывания:

1) мультиэвристический алгоритм;
2) алгоритм Дамерау – Левенштейна;
3) алгоритм с использованием оценочной матрицы BLOSUM50;
4) алгоритм Джаро – Винклера;
5) алгоритм Дамерау – Левенштейна с использованием оценочных матриц.

Таблица 4

Оценка алгоритма «Расстояние Дамерау – Левенштейна» с использованием оценочных матриц

Номер столбца	2	5	5	5	9	9	9	9	9	11
Номер строки	1	1	2	3	1	2	3	5	7	2
Метрика	0,9	0,849	0,833	0,849	0,829	0,812	0,804	0,84	0,77	0,77
Количество треугольников с нарушением равнобедренности > 0,1	50	48	51	51	53	55	54	51	49	29
Общее количество треугольников	56	56	56	56	56	56	56	56	56	56
Оценка точности, %	89,3	85,71	91,07	91,07	94,64	98,21	96,43	91,07	87,5	51,79

Таблица 5

Оценка алгоритма с использованием оценочной матрицы BLOSUM50

Номер столбца	2	12	14	14	24	24	24
Номер строки	1	11	11	12	11	12	14
Метрика	0,9041	0,915	0,91	0,907	0,903	0,8959	0,902
Количество треугольников с нарушением равнобедренности > 0,1	50	17	11	15	31	42	29
Общее количество треугольников	56	56	56	56	56	56	56
Оценка точности, %	89,286	30,36	19,6	26,79	55,36	75	51,79

Таким образом, такого рода оценка позволяет определять качество алгоритмов подсчета метрик ДНК.

Список литературы

1. **Гасфилд, Д.** Строки, деревья и последовательности в алгоритмах. Информатика и вычислительная биология / Д. Гасфилд. – СПб. : Невский диалект, БХВ-Петербург, 2003. – 654 с.
2. **Бойцов, Л.** Использование хеширования по сигнатуре для поиска по сходству / Л. Бойцов // Прикладная математика и информатика. – 2000. – № 7.
3. **Мельников, Б. Ф.** Параллельная реализация мультиэвристического подхода в задаче сравнения генетических последовательностей / Б. Ф. Мельников, А. Г. Па-

нин // Вектор науки Тольяттинского государственного университета. – 2012. – № 4 (22). – С. 83–86.
4. NCBI: nucleotide database, 2015. – URL: http://www.ncbi.nlm.nih.gov/nuccore.
5. **Пивнева, С. В.** Моделирование задач дискретной оптимизации / С. В. Пивнева, М. А. Трифонов // Вектор науки Тольяттинского государственного университета. – 2010. – № 3. – С. 28–30.
6. **Мельников, Б. Ф.** Кластеризация ситуаций и принятие решений в задачах дискретной оптимизации / Б. Ф. Мельников, Е. А. Мельникова // Известия высших учебных заведений. Поволжский регион. Сер. Естественные науки. – 2007. – № 2. – С. 25–28.
7. **Сайфуллина, Е. Ф.** Об алгоритмах восстановления графа по вектору степеней второго порядка / Е. Ф. Сайфуллина, Р. И. Семенов // Эвристические алгоритмы и распределенные вычисления. – 2014. – Т. 1, № 2. – С. 43–57.
8. **Needleman, S.** A general method applicable to the search for simi-larities in the amino acid sequence of two proteins / S. Needleman, C. Wunsch // Journal of Molecular Bi-ology. – 1970. – № 48 (3). – P. 443–453.
9. **Winkler, W.** String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage / W. Winkler // Proceedings of the Section on Survey Research Methods. – American Statistical Association, 1990. – P. 354–359.
10. **Ewing, B.** Base-calling of automated sequencer traces using phred. I. Accuracy assessment / B. Ewing, L. Hillier, M. Wendl, P. Green // Genome Res. – 1998. – № 8 (3). – P. 175–185.
11. **Altschul, S. F.** Amino acid substitution matrices from an information theoretic perspective / S. F. Altschul // Journal of Molecular Biology. – 1991. – № 219 (3). – P. 555–565.

References

1. Gasfild D. *Stroki, derev'ya i posledovatel'nosti v algoritmakh. Informatika i vychislitel'naya biologiya* [Lines, trees and sequences in algorithms. Informatics and calculus biology]. Saint-Petersburg: Nevskiy dialekt, BKhV-Peterburg, 2003, 654 p.
2. Boytsov L. *Prikladnaya matematika i informatika* [Applied mathematics and informatics]. 2000, no. 7.
3. Mel'nikov B. F., Panin A. G. *Vektor nauki Tol'yattinskogo gosudarstvennogo universiteta* [Scientific vector of Togliatti State University]. 2012, no. 4 (22), pp. 83–86.
4. *NCBI: nucleotide database, 2015*. Available at: http://www.ncbi.nlm.nih.gov/nuccore.
5. Pivneva S. V., Trifonov M. A. *Vektor nauki Tol'yattinskogo gosudarstvennogo universiteta* [Scientific vector of Togliatti State University]. 2010, no. 3, pp. 28–30.
6. Mel'nikov B. F., Mel'nikova E. A. *Izvestiya vysshikh uchebnykh zavedeniy. Povolzhskiy region. Ser. Estestvennye nauki* [University proceedings. Volga region. Physical and mathematical sciences]. 2007, no. 2, pp. 25–28.
7. Sayfullina E. F., Semenov R. I. *Evristicheskie algoritmy i raspredelennye vychisleniya* [Heuristic algorithm and distributed calculations]. 2014, vol. 1, no. 2, pp. 43–57.
8. Needleman S., Wunsch C. *Journal of Molecular Biology*. 1970, no. 48 (3), pp. 443–453.
9. Winkler W. *Proceedings of the Section on Survey Research Methods*. American Statistical Association, 1990, pp. 354–359.
10. Ewing B., Hillier L., Wendl M., Green R. *Genome Res*. 1998, no. 8 (3), pp. 175–185.
11. Altschul S. F. *Journal of Molecular Biology*. 1991, no. 219 (3), pp. 555–565.

Мельников Борис Феликсович
доктор физико-математических наук, профессор, кафедра прикладной математики и информатики, Тольяттинский государственный университет (Россия, г. Тольятти, ул. Белорусская, 14)

E-mail: barmaley62@yandex.ru

Mel'nikov Boris Feliksovich
Doctor of physical and mathematical sciences, professor, sub-department of applied mathematics and informatics, Togliatti State University (14 Belorusskaya street, Togliatti, Russia)

Пивнева Светлана Валентиновна
кандидат педагогических наук, доцент, кафедра высшей математики и математического моделирования, Тольяттинский государственный университет (Россия, г. Тольятти, ул. Белорусская, 14)

E-mail: tlt.swetlana@rambler.ru

Pivneva Svetlana Valentinovna
Candidate of pedagogical sciences, associate professor, sub-department of mathematical modeling, Togliatti State University (14 Belorusskaya street, Togliatti, Russia)

Трифонов Максим Андреевич
аспирант, Тольяттинский государственный университет (Россия, г. Тольятти, ул. Белорусская, 14)

E-mail: trifonov_max@mail.ru

Trifonov Maksim Andreevich
Postgraduate student, Togliatti State University (14 Belorusskaya street, Togliatti, Russia)

УДК 621.317.7

Мельников, Б. Ф.

Оценка алгоритмов расчета расстояния строк ДНК / Б. Ф. Мельников, С. В. Пивнева, М. А. Трифонов // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. – 2015. – № 2 (34). – С. 57–67.