

# Обзор литературы по ассемблерам

Дмитрий Яковлев

EPAM Systems

18 октября 2016 г.

- 1 Введение
- 2 Модель ошибки на данных BioNano
- 3 Ассемблеры
  - TWIN
  - OPTIMA
  - MAligner
  - OMBlast
- 4 Ссылки

# Введение



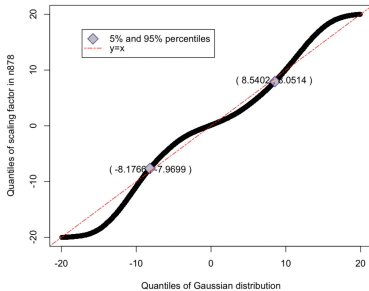
# Модель ошибки на данных BioNano

- Было рассмотрено 3 датасета карт от BioNano
- С помощью RefAligner был построен референс
- Далее был проведён анализ ошибок

# Модель ошибок: ошибка в длине фрагмента

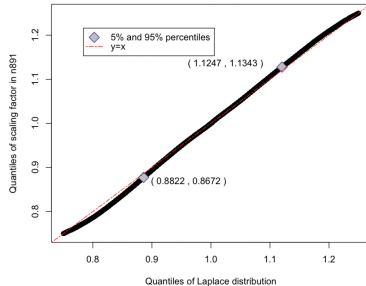
Валуев:

$$e_k = \frac{o_k - r_k}{\sqrt{r_k}} \sim N(0, \sigma)$$
$$o_k \sim N(r_k, \sigma^2 r_k)$$



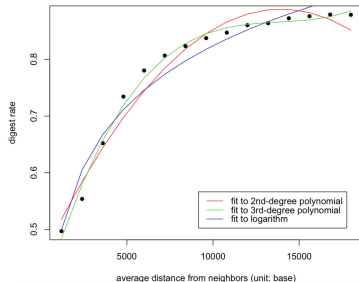
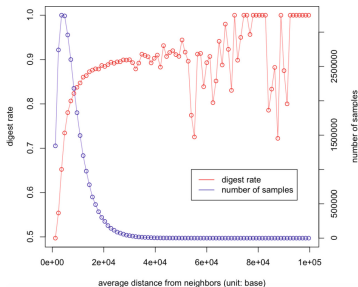
Новый подход:

$$s_k = \frac{o_k}{r_k}$$
$$s_k \sim Laplace(\mu, \beta)$$



# Модель ошибок: пропущенные разрезы

Было замечено, что вероятность пропущенного разреза зависит от длины до соседних разрезов.

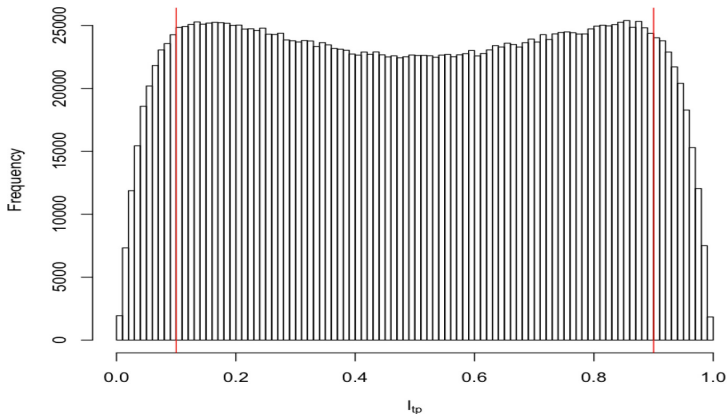


$$p_c(d_{avg}) = \alpha_3 d_{avg}^3 + \alpha_2 d_{avg}^2 + \alpha_1 d_{avg} + \alpha_0$$

$$d_{avg} = \frac{\text{среднее расстояние до соседей}}{1200}$$



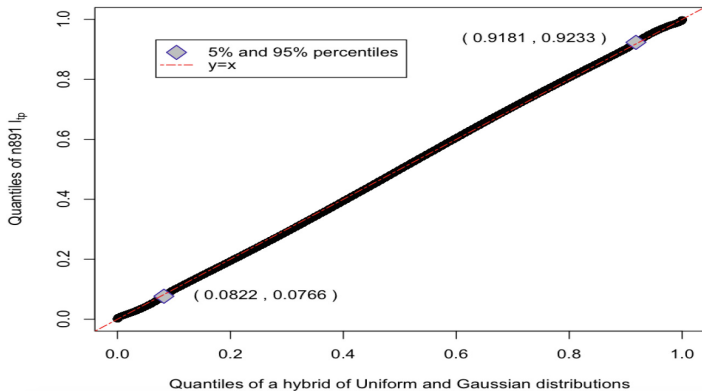
# Модель ошибок: лишние разрезы (1)



$$l_{fp} = \frac{\text{расстояние от лишнего разреза до конца карты}}{\text{длина оптической карты}}$$

$$n_{fp} \sim 0.18 \text{ Poisson}(0) + 0.6 \text{ Poisson}(1) + 0.22 \text{ Poisson}(3)$$

# Модель ошибок: лишние разрезы (2)



$$l_{fp} \sim \begin{cases} U[0.1, 0.9], & 0.1 \leq l_{fp} \leq 0.9, \text{ w.p. } 0.8852 \\ N(0.1, 0.044186), & l_{fp} < 0.1, \text{ w.p. } 0.0574 \\ N(0.9, 0.044186), & l_{fp} > 0.9, \text{ w.p. } 0.0574 \end{cases}$$

# Ассемблеры

TWIN



# OPTIMA

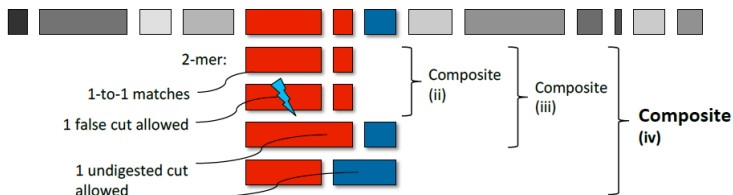


Этапы выравнивания:

- Поиск стартовых мест (сидов) для начала выравнивания
- Парное выравнивание карты с референсом
- Определение значимых выравниваний
- Объединение пересекающихся выравниваний



## Композитные сиды:



Множество фрагментов  $o_k, o_{k+1}, \dots, o_s$  возможно совпадает с множеством фрагментов  $r_l, r_{l+1}, \dots, r_t$ :

$$\frac{\left| \sum_{i=k}^s o_i - \sum_{j=l}^t r_j \right|}{\sqrt{\sum_{j=l}^t \sigma_j^2}} \leq C_\sigma \quad (1)$$

Алгоритм поиска сидов для выравнивания:

- По референсу строятся композитные сиды и сортируются по первому элементу
- У карты берётся произвольный сид, по которому будем искать множество подходящих локаций(1) в референсе
- Бинарным поиском (по первому элементу) ищем множество подходящих сидов в референсе
- Далее линейно проверяем и оставляем только те, которые удовлетворяют (1)
- Таким образом получаем множество сидов на референсе, где карта может быть выравнена
- Сложность алгоритма  $O(m(\log n + c \#seeds_{c=1}))$

После обнаружения схожих сидов на референсе происходит парное выравнивание алгоритмом динамического программирования:

$$Score_{s,t} = \min_{k \leq s, l \leq t} C_{ce}(s - k + t - l) + \chi_{k\dots s, l\dots t}^2 + Score_{k-1, l-1}$$

$$\chi_{k\dots s, l\dots t}^2 = \frac{\left( \sum_{i=k}^s o_i - \sum_{j=l}^t r_j \right)^2}{\sum_{j=l}^t \sigma_j^2}$$

$C_{se}$  - штраф за пропущенные фрагменты

Пусть  $a$  - выравнивание из множества выравниваний  $\mathcal{A}$

$$Z - score(a \in \mathcal{A}, f) = \frac{f_a - \text{Mean}(f_{\mathcal{A}})}{SD(f_{\mathcal{A}})}$$

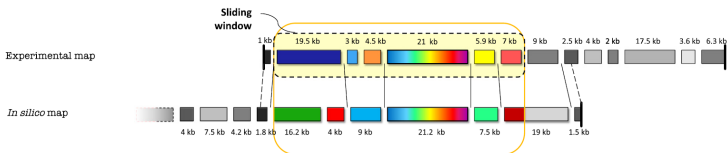
где  $f$  - характеристика выравнивания.

Тогда статистическая значимость выравнивания:

$$\begin{aligned} \vartheta(a \in \mathcal{A}) = & Z - score(-Z - score(a, \#matches) \\ & + Z - score(a, \#cuterrors) \\ & + Z - score(a, WHT(\chi^2, \#matches))) \end{aligned}$$

$$\text{где } WHT(\chi^2, \#matches) = \frac{\sqrt[3]{\frac{\chi^2}{\#matches}} - \left(1 - \frac{1}{9} \frac{2}{\#matches}\right)}{\sqrt{\frac{1}{9} \frac{2}{\#matches}}}$$

Разбиение на блоки:



## Результаты для 2100 карт:

Algorithm	<i>Drosophila</i> (A)		<i>Drosophila</i> (B)		Human (A)		Human (B)	
	S	P	S	P	S	P	S	P
OPTIMA	<b>90</b>	<b>100</b>	<b>49</b>	<b>99</b>	<b>83</b>	<b>100</b>	<b>43</b>	<b>98</b>
Gentig v.2 (d)	59	<b>100</b>	24	<b>99</b>	53	96	20	80
Gentig v.2 (tp)	59	<b>100</b>	24	98	54	95	20	88
SOMA v.2 (v)	72	73	31	39	50	50	17	20
Likelihood (d+a)	49	49	29	30	24	24	14	14
Likelihood (d+a+t)	64	65	38	39	33	34	18	19
Likelihood (p+a+t)	75	75	39	39	62	62	19	20

S - чувствительность

P - точность

tp - найстрока параметров в соответствии с генерацией данных

p - параметры, указанные в статьях авторов

d - стандартные настройки

t - обрезание концов карт

# MAligner

Два подхода:

- На основе алгоритма Смита-Ватермана
  - ❶ Построение множества выравниваний на референсе
  - ❷ Отклонение выравниваний с помощью M-Score
- На основе индексации



Пусть имеются два выравненных участка с  $n$  и  $m$  пропущенными фрагментами длины  $r$  и  $m$  на референсе и карте соответственно. Тогда выравнивание имеет следующее значение:

$$\text{Score}(q, r, m, n) = S(q, r) + C_q m + C_r n$$

$$S(q, r) = \left( \frac{q - r}{\sigma(r)} \right)^2$$

$$\sigma(r) = \max(\alpha r, \sigma_{min})$$

$C_q$  - штраф за пропущенные фрагменты на карте

$C_r$  - штраф за пропущенные фрагменты на референсе

$\sigma_{min}$  - для фрагментов малой длины, ошибка больше

$\alpha$  - доля референса, которая будет использоваться как стандартное отклонение

Предложена оценка M-Score для определения значимости выравнивания:

$$m_{\mathcal{A}} = \underset{A \in \mathcal{A}}{\text{median}}\{\text{Score}(A)\}$$

$$MAD_{\mathcal{A}} = \underset{A \in \mathcal{A}}{\text{median}}\{|\text{Score}(A) - m_{\mathcal{A}}|\}$$

$$M - \text{Score}_{\mathcal{A}}(A) = \frac{\text{Score}(A) - m_{\mathcal{A}}}{MAD_{\mathcal{A}}}$$

$\text{Score}(A)$  - значение выравнивания  $A$

$\mathcal{A}$  - 100 лучших выравниваний по  $\text{Score}(A)$

# MAligner: Алгоритм на основе индексов



# OMBlast



Этапы выравнивания:

- Поиск стартовых мест (сидов) для начала выравнивания
- Расширение сидов
- Объединение пересекающих выравниваний
- Построение итогового выравнивания

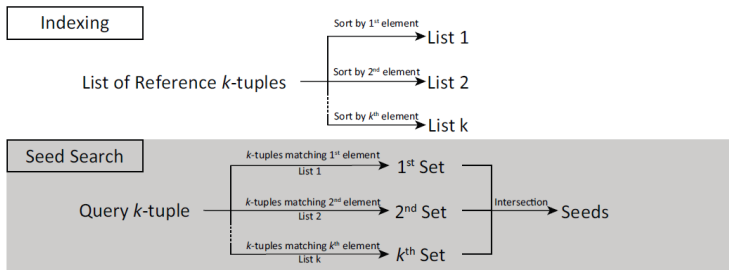
# OMBlast: Поиск стартовых сидов - индексация

Фрагмент  $q$  на карте совпадает с фрагментом  $r$  на референсе:

$$r(1 - T_s) - T_m \leq q \leq r(1 + T_s) + T_m$$

$T_s$  - ошибка масштабирования

$T_m$  - ошибка измерений





# OMBlast: Поиск стартовых сидов - бины

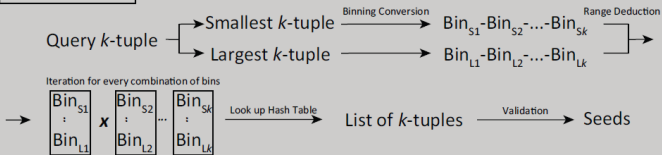
## Binning

Bin	Interval
A	1-5000
B	5001-10000
C	10001-15000
...	...

## Indexing

Reference  $k$ -tuple  $\xrightarrow{\text{Binning Conversion}}$   $\text{Bin}_1\text{-Bin}_2\text{-...-Bin}_k \xrightarrow{\text{Hashing}}$  Hash Table

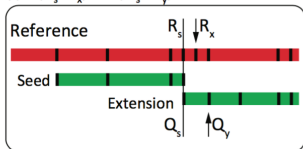
## Seed Search



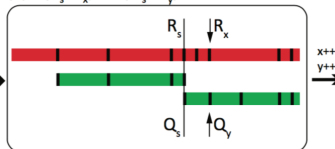
# OMBlast: Расширение сидов



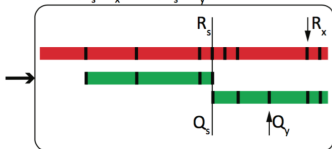
(i)  $D(R_s, R_x) < D(Q_s, Q_y)$



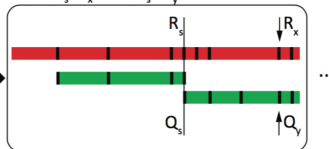
(ii)  $D(R_s, R_x) \sim D(Q_s, Q_y)$



(iii)  $D(R_s, R_x) > D(Q_s, Q_y)$



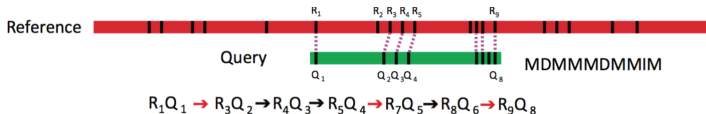
(iv)  $D(R_s, R_x) \sim D(Q_s, Q_y)$



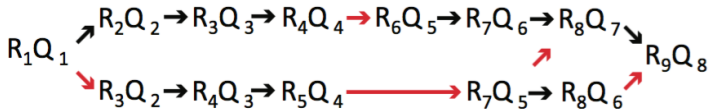
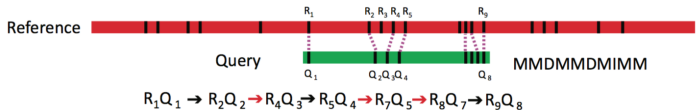
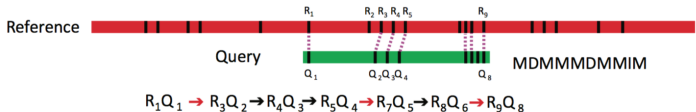
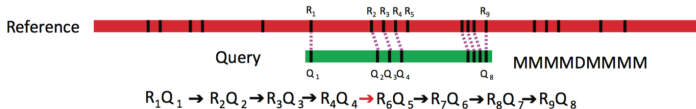
# OMBlast: Объединение выравниваний (1)

Строится взвешенный ациклический граф:

- Вершины - выравненные разрезы
- Рёбра - между двумя парами последовательно (на одной карте) выравненных разрезов
- Веса -  $t_m u_m - t_{es} u_{es} - t_{ms} u_{ms}$   
 $u_m$  - количество совпадений  
 $u_{es}$  - количество лишних разрезов  
 $u_{ms}$  - количество пропущенных разрезов

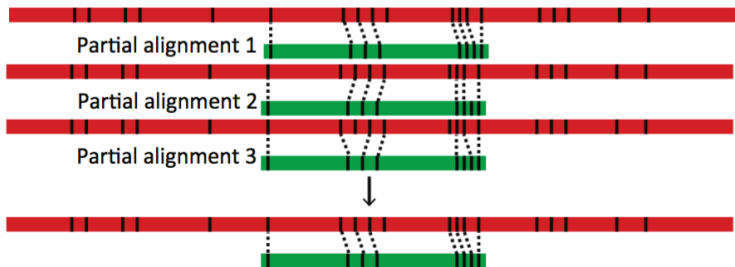


# OMBlast: Объединение выравниваний (2)



# OMBlast: Объединение выравниваний (3)

С помощью динамического программирования определяется путь в графе с наибольшим весом



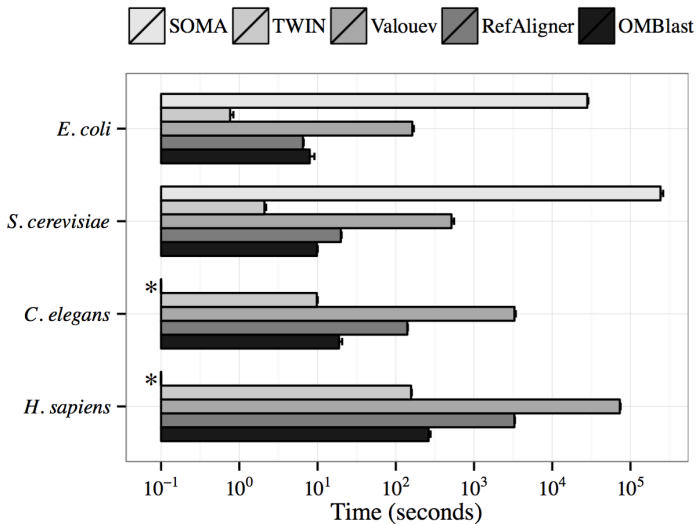
# OMBlast: Построение итогового выравнивания

# OMBlast: Результаты - входные данные

Organism	Genome Size (Mbp)	Total Signals	Average Bases Between Signals (kbp)
<i>E. coli</i>	4.6	683	$6.8 \pm 7.3$
<i>S. cerevisiae</i>	12.1	1953	$6.2 \pm 6.7$
<i>C. elegans</i>	100.3	14837	$6.8 \pm 8.0$
<i>H. sapiens</i>	3088.3	377143	$8.2 \pm 83.2$

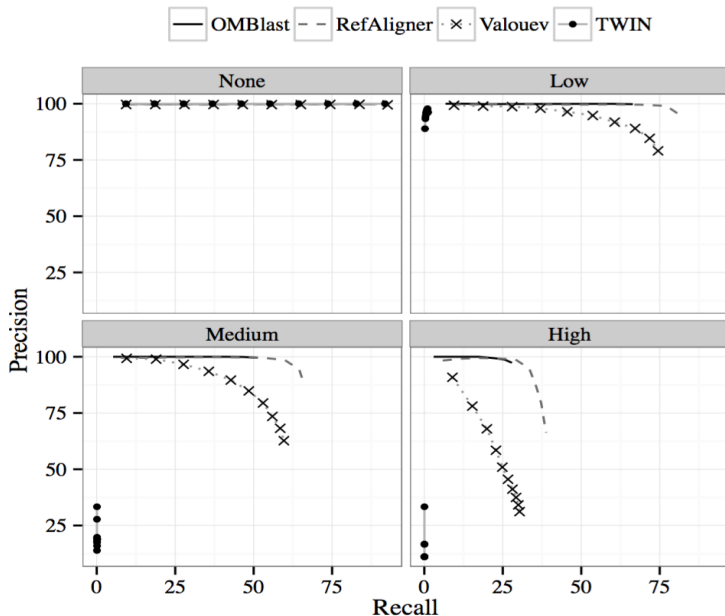
Error Rate	None	Low	Medium	High
Extra Signal Rate	0	0.000005	0.00001	0.00002
Missing Signal Rate	0	0.05	0.1	0.2
Scaling	0	0.02	0.04	0.08
Measurement (bp)	0	500	500	500
Resolution (bp)	0	1200	1200	1200

# OMBlast: Результаты - время работы

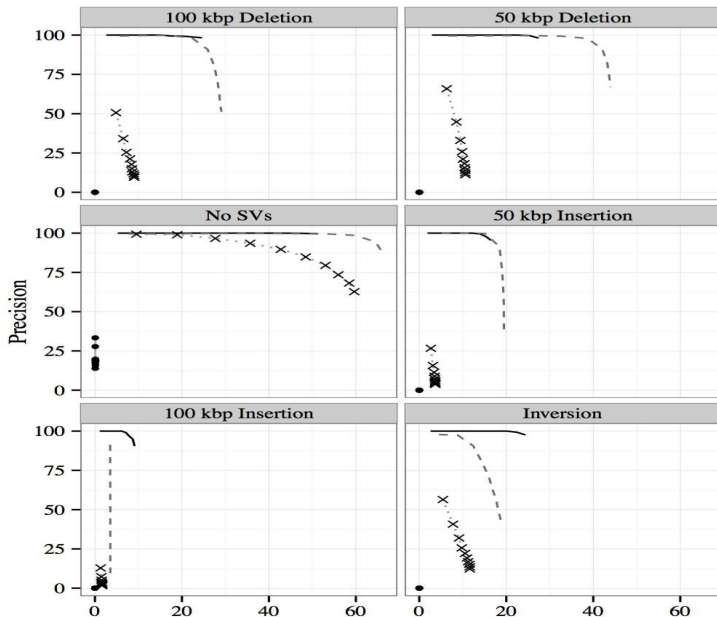




# OMBlast: Результаты - точность и полнота



# OMBlast: Результаты - наличие SV



## Ссылки

В открытом доступе:

- TWIN
- OPTIMA
- MAligner
- OMBlast

Спасибо за внимание!