

# Обзор методов анализа параметров видеокодеков и систем обработки видео

---

Кира Рагулина

*Video Group*  
*CS MSU Graphics & Media Lab*



# Содержание

---

- **Введение**

- Введение
- Основные понятия
- Постановка задачи
- Общие проблемы
- Область применения
- Классификация алгоритмов
- Поиск оптимальных пресетов
- Анализ различных значений параметров
- Результаты
- Планы

# Введение

Видео

Битрейт  
T



Закодированное  
видео с заданным  
битрейтом

**Максимальное  
качество!**

# Основные понятия

- Параметры видеокодека настраивают некую часть работы видеокодека. Их значения влияют на качество и скорость кодирования
- Пресет – набор значений параметров видеокодека
- Качество пресета для нас характеризуется:
  - Временем кодирования
  - Качеством закодированного видео



# Постановка задачи

---

1. Проанализировать влияние различных параметров на время кодирования и качество закодированного видео
2. Найти оптимальные пресеты

# Общие проблемы

- Способ оценки качества пресета
- Большая размерность и мощность пространства возможных пресетов
- Отсутствие метрики в пространстве пресетов
- Сложные и неизвестные зависимости между параметрами
- Оценка качества и верификация рассматриваемых алгоритмов



# Область применения

---

- Подбор наилучших настроек кодека
- Проверка эффективности реализации параметров
- Выявление зависимостей между параметрами
- Нагрузочное тестирование кодека
- Выбор настроек с заданными характеристиками

# Проблемы алгоритмов

- Поиск оптимальных пресетов
  - Качество пресета
  - Способ неполного перебора
  - Сложность
  - Полнота
  - Точность
  - Стабильность
- Анализ различных значений параметров
  - Качество пресета
  - Неполные данные
  - Оценка значений параметров
  - Выявление зависимостей
  - Верификация
  - Стабильность



# Классификация алгоритмов



## Поиск оптимальных пресетов

## Анализ различных значений параметров

### Существующие методы

Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)

Алгоритм, основанный на симплекс-методе

Подсчет средних характеристик по параметрам\*

### Разработанные методы

Итеративный градиентный спуск\*

Разноцветные облака точек

Лямбда анализ

Анализ с помощью выкидывания огибающих

# Содержание

- Введение
- **Поиск оптимальных пресетов**
  - **Основные понятия**
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
    - Базовый алгоритм GBFOS
    - Итеративный алгоритм GBFOS
  - Алгоритм, основанный на симплекс-методе
    - Симплекс-метод
    - Алгоритм на основе симплекс-метода
  - Итеративный градиентный спуск
- Анализ различных значений параметров
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- Результаты
- Планы

# Оптимальные пресеты

- Оптимальные по Парето пресеты

$$\{p \mid \nexists p' : T(p') \leq T(p), Q(p') \leq Q(p), p' \neq p\}$$

- Метод свертки критериев

- Свертка критериев – любая числовая функция критериев

- Наиболее важные свертки критериев  $y = (y_1, \dots, y_n)$

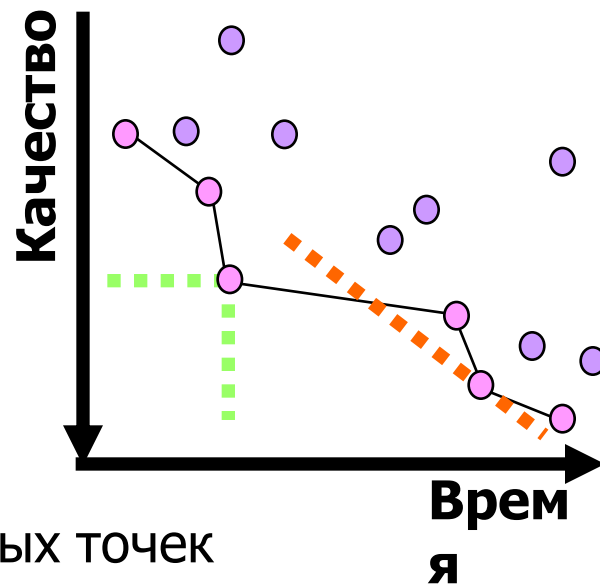
- Линейная свертка  $\langle c, y \rangle = \sum_{i=1}^n c_i y_i, c_i \geq 0, i = \overline{1, n}$

- Свертка Гермейера  $\min_{i=1, n} \frac{y_i}{a_i}, a_i > 0, i = \overline{1, n}$

- Свертка на основе отклика от идеальной точки  $\rho(y^*, y)$

# Свертки критериев

- Линейная свертка  $\langle \mathbf{c}, \mathbf{y} \rangle$ 
  - Порождает подмножество оптимальных точек (огibaющую) (случай невыпуклой границы)
  - Линии уровня – прямые
- Свертка Гермейера  $\min(\mathbf{y}_i / \mathbf{a}_i)$ 
  - Порождает все множество оптимальных точек
  - Линии уровня – конусы
- Свертка на основе отклика от идеальной точки  $-\rho(\mathbf{y}^*, \mathbf{y})$ 
  - Требуется явного введения метрики в пространстве критериев



# Определение лямбды

- Сформулируем нашу задачу следующим образом:

Найти оптимальные пресеты при заданных ограничениях на качество  $Q_c$  (битрейт)

$$p^* = \operatorname{argmin}_{p: Q(p) \leq Q_c} T(p)$$

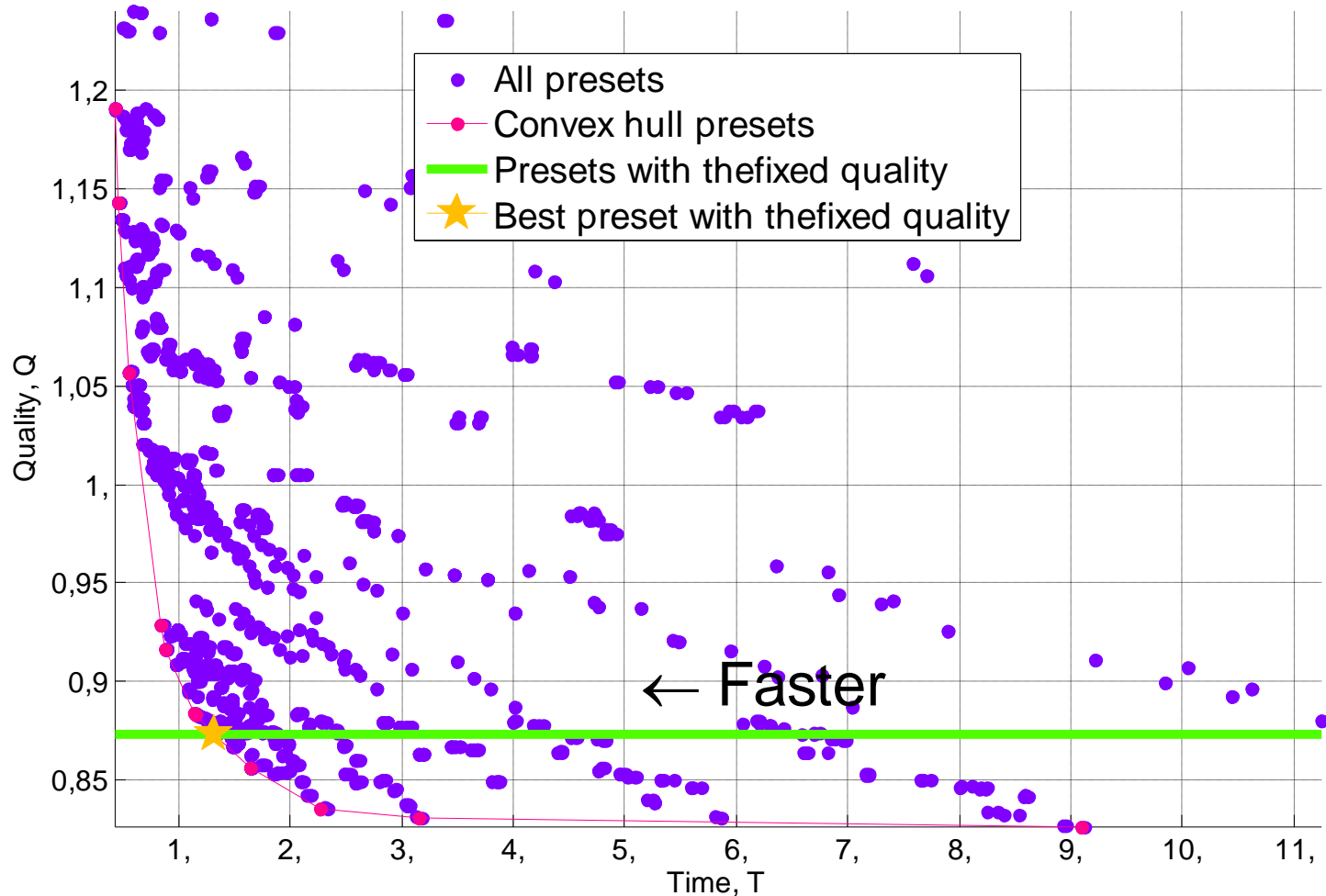
➡  $T(p^*) + \lambda(Q(p^*) - Q_c) \rightarrow \min$

➡  $p^* = \operatorname{argmin} M(p), M(p) = \lambda' T(p) + Q(p), \lambda' = \frac{1}{\lambda}$

- $M(p)$  - линейная свертка для 2-х критериев
- $\lambda$  – задает желаемое соотношение между качеством и временем кодирования

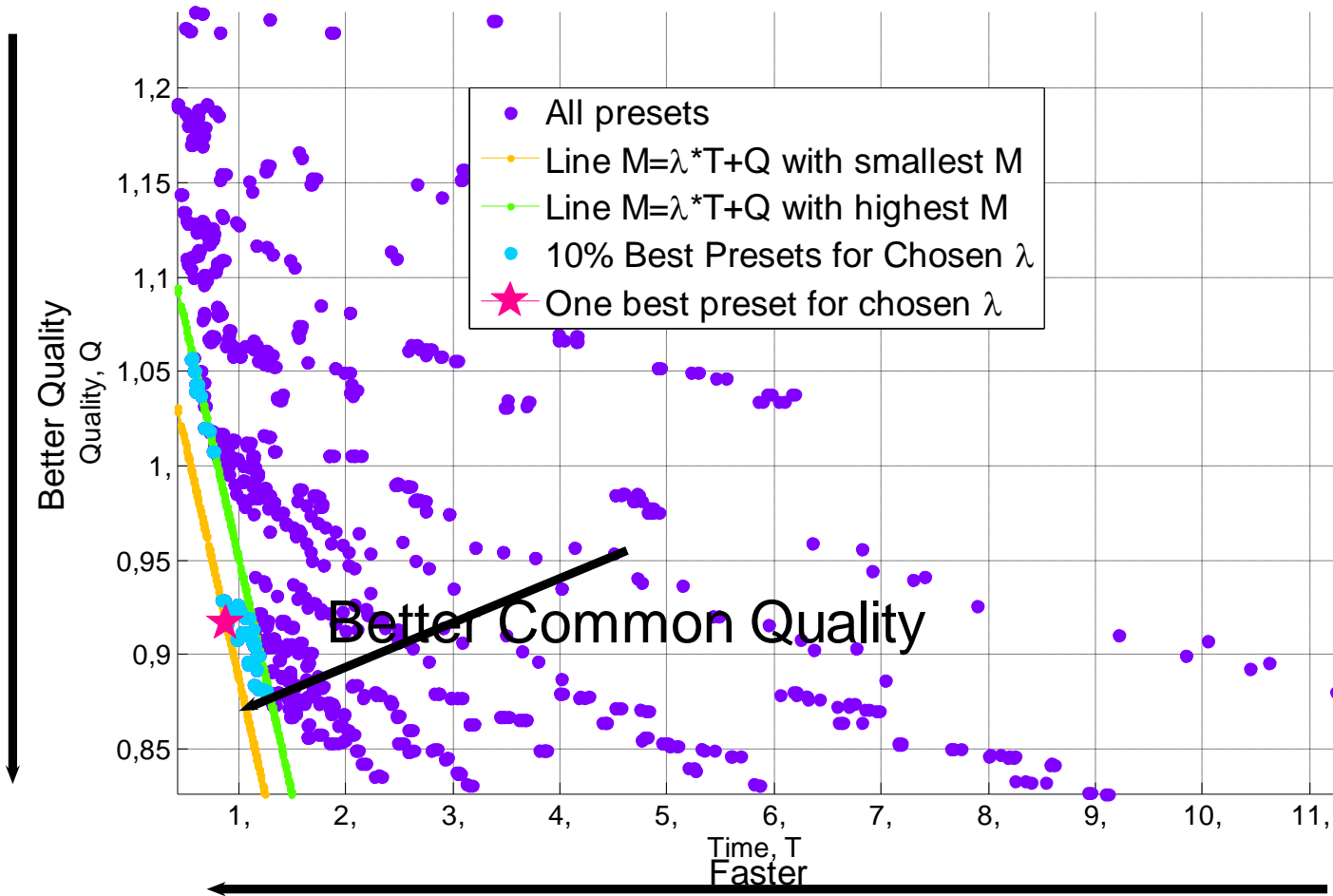
# Интерпретация новой постановки задачи

## Physical Interpretation of the Convex Hull Presets



# Интерпретация лямбды

10% Best Presets for Chosen  $\lambda = 0,25$



# Содержание

- Введение
- **Поиск оптимальных пресетов**
  - Основные понятия
  - **Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)**
    - **Базовый алгоритм GBFOS**
    - Итеративный алгоритм GBFOS
  - Алгоритм, основанный на симплекс-методе
    - Симплекс-метод
    - Алгоритм на основе симплекс-метода
  - Итеративный градиентный спуск
- Анализ различных значений параметров
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- Результаты
- Планы

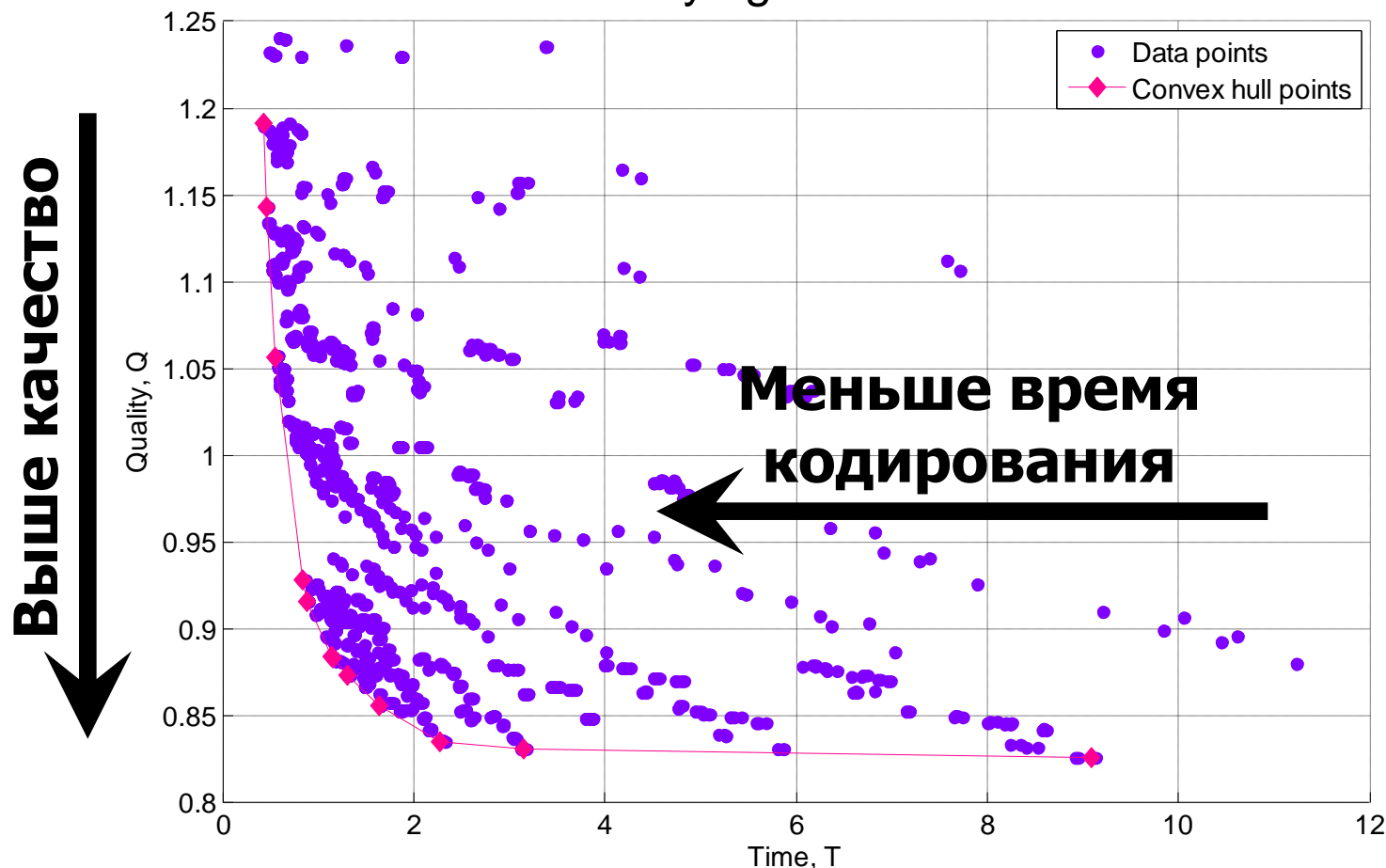


# Обозначения

- Пусть имеется **M** параметров
- Параметр **i** имеет значения **1, ..., n<sub>i</sub>**, которые упорядочены по улучшению качества (ухудшению времени)
- Значит, пресет – набор **a = (a<sub>1</sub>, ..., a<sub>M</sub>)**, где **a<sub>i</sub> ∈ {1, ..., n<sub>i</sub>}**
- Каждому пресету **a** соответствует **T(a), Q(a)** – его время кодирования и качество закодированного видео
- Построим график зависимости **Q** от **T**

# Огибающая линия

Presets Lying on Convex Hull



R. Vanam, E. Riskin, S. Hemami, R. Ladner. Distortion-Complexity Opti-mization of the H.264/MPEG-4 AVC Encoder using the GBFOS Algorithm. Proc. IEEE, 2007.

# Базовый алгоритм GBFOS

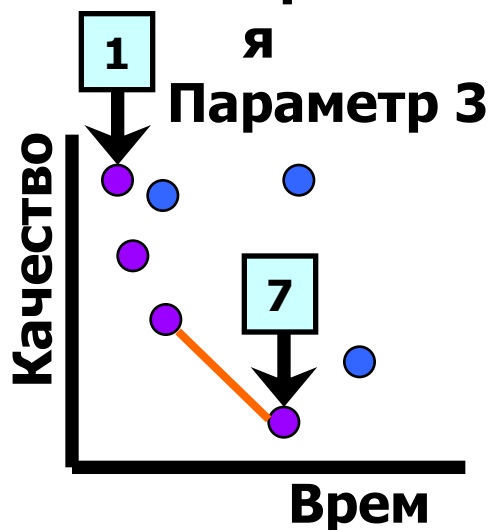
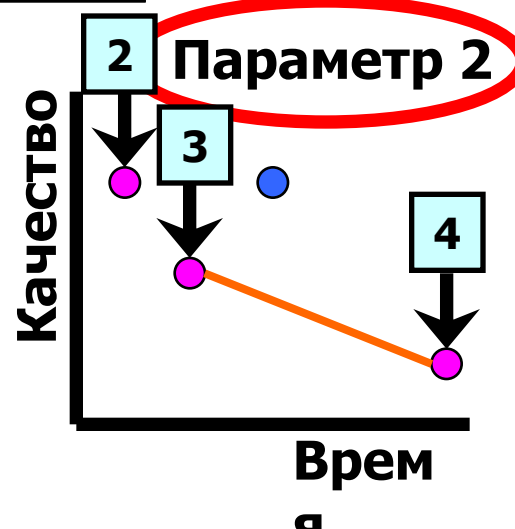
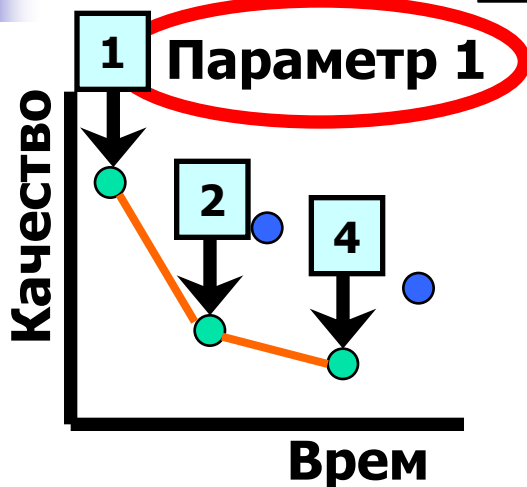


- Алгоритм предполагает, что параметры независимы и аддитивны
- Замеряются пресеты вида  $\mathbf{m}_{i,j} = (\mathbf{n}_1, \dots, \mathbf{n}_{i-1}, \mathbf{j}, \mathbf{n}_{i+1}, \dots, \mathbf{n}_M)$ , где  $\mathbf{j} = \mathbf{1}, \dots, \mathbf{n}_i$
- Алгоритм работает итеративно, в конце каждой итерации получается очередной оптимальный пресет

R. Vanam, E. Riskin, S. Hemami, R. Ladner.  
Distortion-Complexity Optimization of the  
H.264/MPEG-4 AVC Encoder using the GBFOS  
Algorithm. Proc. IEEE, 2007.

# Базовый алгоритм GBFOS

## Итерация N



**Повторяем, пока не получим последний оптимальный пресет**

# Итеративный алгоритм GBFOS

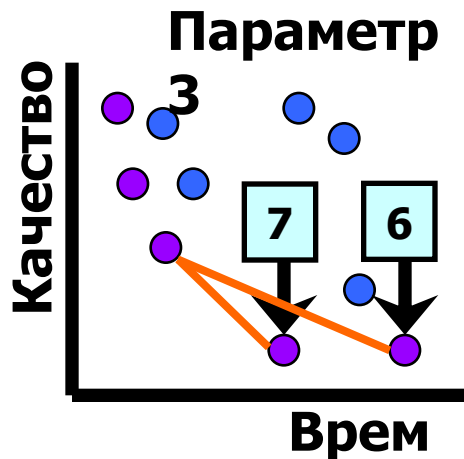
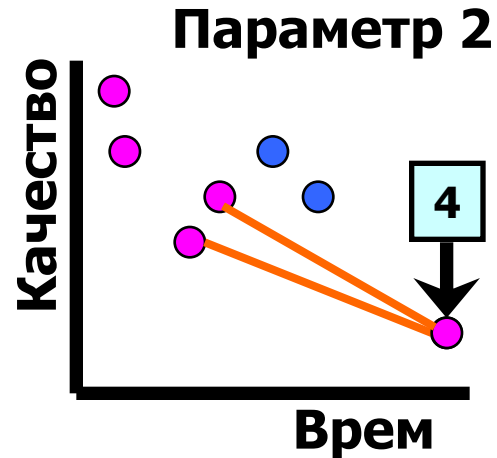


- Усовершенствованный базовый алгоритм GBFOS
- Отличается от базового алгоритма тем, что в конце каждой итерации обновляет точки, делая дополнительные замеры

R. Vanam, E. Riskin, S. Hemami, R. Ladner.  
Distortion-Complexity Optimization of the  
H.264/MPEG-4 AVC Encoder using the GBFOS  
Algorithm. Proc. IEEE, 2007.

# Итеративный алгоритм GBFOS

## Итерация 2



**Обновляем точки,  
используя значение  
2 параметра 1**

# Условия тестирования

- Для тестирования использовались:
  - кодек h264 (Март 26, 2006 версия)
  - значения параметров:
    - ref = 1,...,16
    - partitions = 10 значений
    - subme = 1,...,7
    - trellis = 0, 1, 2
  - 3 набора данных:
    - ASL-1 (10 QCIF последовательностей)
    - ASL-2 (10 320x240 последовательностей)
    - Standart (15 QCIF последовательностей)
  - 3 битрейта:
    - 30, 150 и 300 Кбит/с

R. Vanam, E. Riskin, S. Hemami, R. Ladner.  
Distortion-Complexity Optimization of the  
H.264/MPEG-4 AVC Encoder using the GBFOS  
Algorithm. Proc. IEEE, 2007.

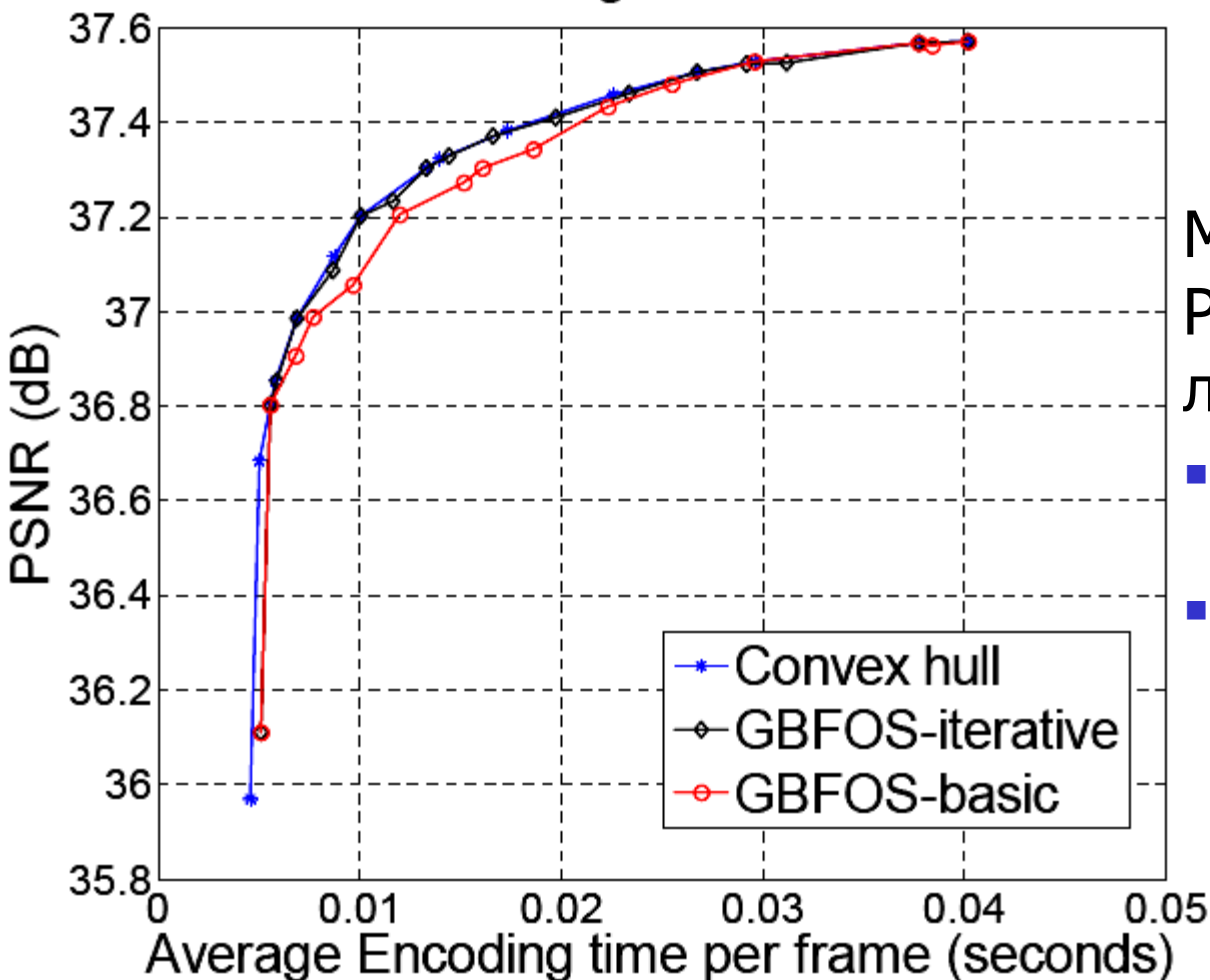
# Результаты: сложность

- Пусть имеется **M** параметров
- Параметр **i** имеет значения **1, ..., n<sub>i</sub>**
- Значит, пресет – набор **a = (a<sub>1</sub>, ..., a<sub>M</sub>)**, где **a<sub>i</sub> ∈ {1, ..., n<sub>i</sub>}**
  
- Количество замеров:
  - Полный перебор:  $\prod_{i=1}^M n_i = 3360$
  - Базовый алгоритм GBFOS:  $\sum_{i=1}^M n_i \quad M + 1 = 33 \approx 1\%$
  - Итеративный алгоритм GBFOS: **268 ≈ 8%**



# Результаты: точность

PSNR vs. encoding time for x264 encoder



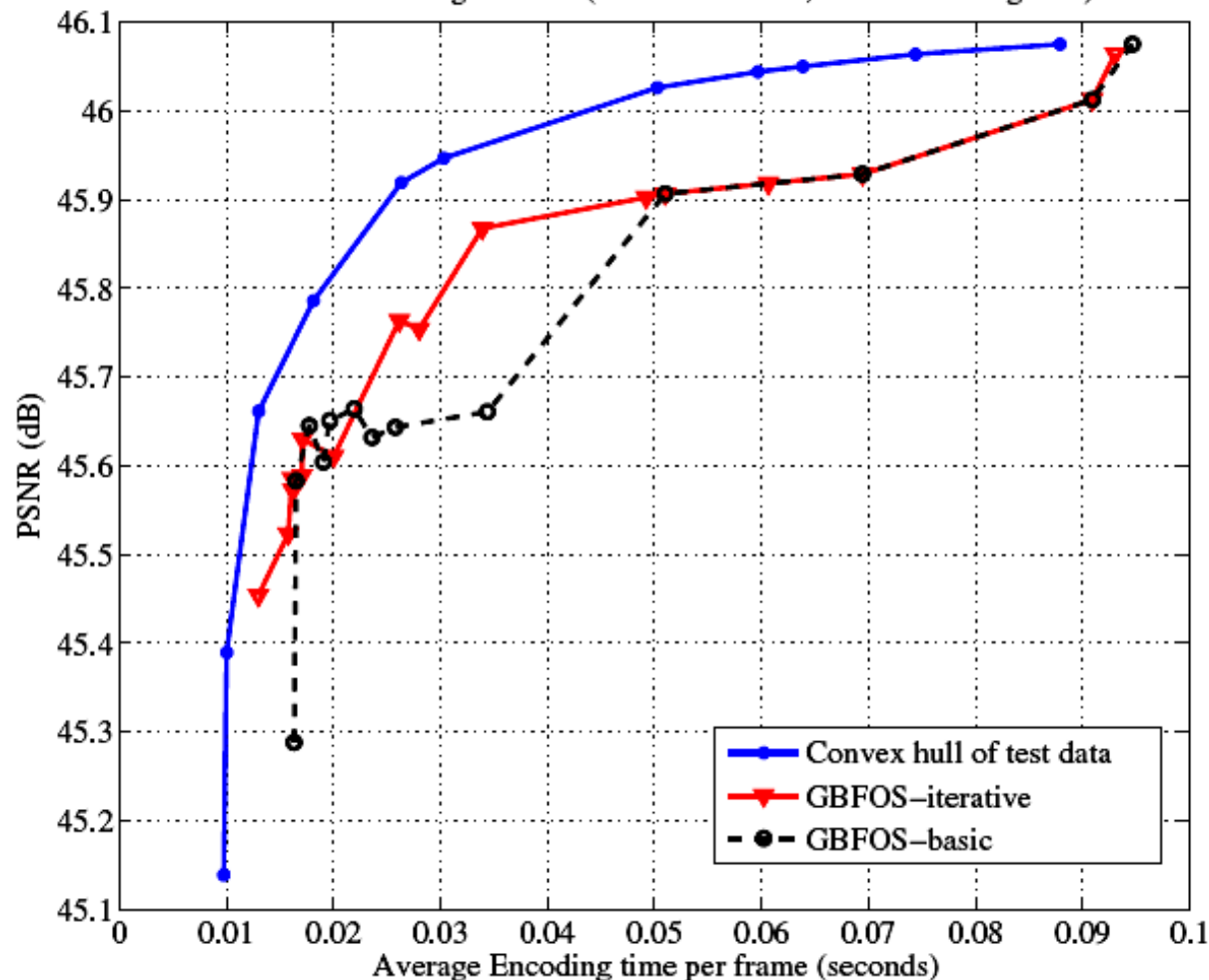
Максимальная разница PSNR между огибающей линией и

- Базовым алгоритмом GBFOS: **0.707 dB**
- Итеративным алгоритмом GBFOS: **0.575 dB**

R. Vanam, E. Riskin, S. Hemami, R. Ladner. Distortion-Complexity Optimization of the H.264/MPEG-4 AVC Encoder using the GBFOS Algorithm. Proc. IEEE, 2007.

# Результаты: стабильность

PSNR vs. encoding time for (ASL-1 test data, ASL-2 training data)



Максимальная разница PSNR между огибающей линией и обоими алгоритмами:  
**< 0.55 dB**

R. Vanam, E. Riskin, S. Hemami, R. Ladner. Distortion-Complexity Optimization of the H.264/MPEG-4 AVC Encoder using the GBFOS Algorithm. Proc. IEEE, 2007.

# Результаты: стабильность

Максимальная разница PSNR для обоих алгоритмов

- для одинакового тестового и тренировочного набора данных: **< 0.71 dB**
- для разных: **< 0.55 dB**

Bitrates	Maximum PSNR difference(dB)					
	(ASL-1, ASL-2)		(ASL-1, Standard)		(Standard, ASL-1)	
	Basic	Iterative	Basic	Iterative	Basic	Iterative
30 kb/s	0.17	0.265	0.422	0.422	0.243	0.18
150 kb/s	0.3	0.217	0.538	0.211	0.549	0.145
300 kb/s	0.223	0.051	0.314	0.288	0.213	0.167

# Содержание

- Введение
- **Поиск оптимальных пресетов**
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
    - Базовый алгоритм GBFOS
    - Итеративный алгоритм GBFOS
  - **Алгоритм, основанный на симплекс-методе**
    - **Симплекс-метод**
      - Алгоритм на основе симплекс метода
  - Итеративный градиентный спуск
- Анализ различных значений параметров
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- Результаты
- Планы

# Симплекс-метод

- Другие названия:
  - метод Нелдера — Мида
  - метод деформируемого многогранника

- Задача:

$$\mathbf{f}(\mathbf{x}_1, \dots, \mathbf{x}_n) \rightarrow \min$$

- Суть метода:
  - последовательное перемещение и деформирование симплекса вокруг точки экстремума

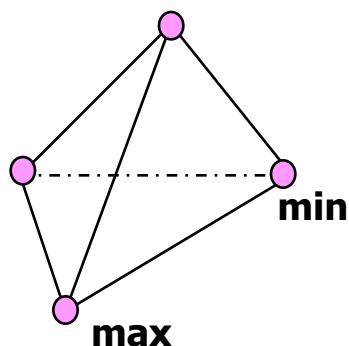
- Характеристики:

- + не использует градиентов функции
- + применим к негладким функциям
- находит локальный минимум
- $\mathbf{f}$  – скалярная функция

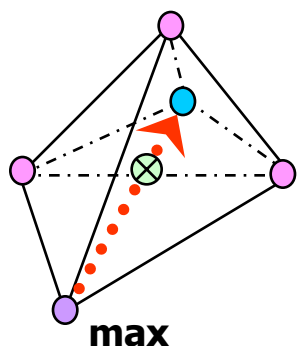
# Симплекс-метод

- Симплекс - многогранник, имеющий  $n+1$  вершину, где  $n$  - число факторов, влияющих на процесс (параметров)

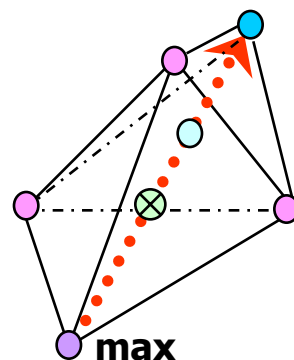
- Примитивы:



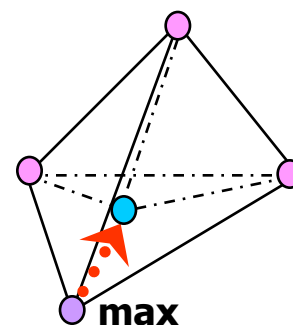
симплекс



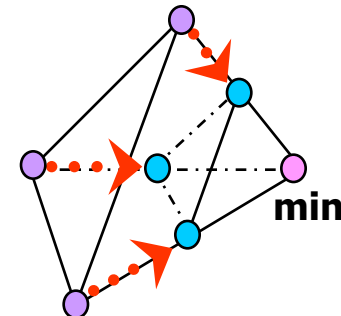
отражение ( $\alpha$ )



отражение и  
растяжение ( $\gamma$ )



сжатие ( $\beta$ )



глобальное  
сжатие (0.5)

- Параметры метода:

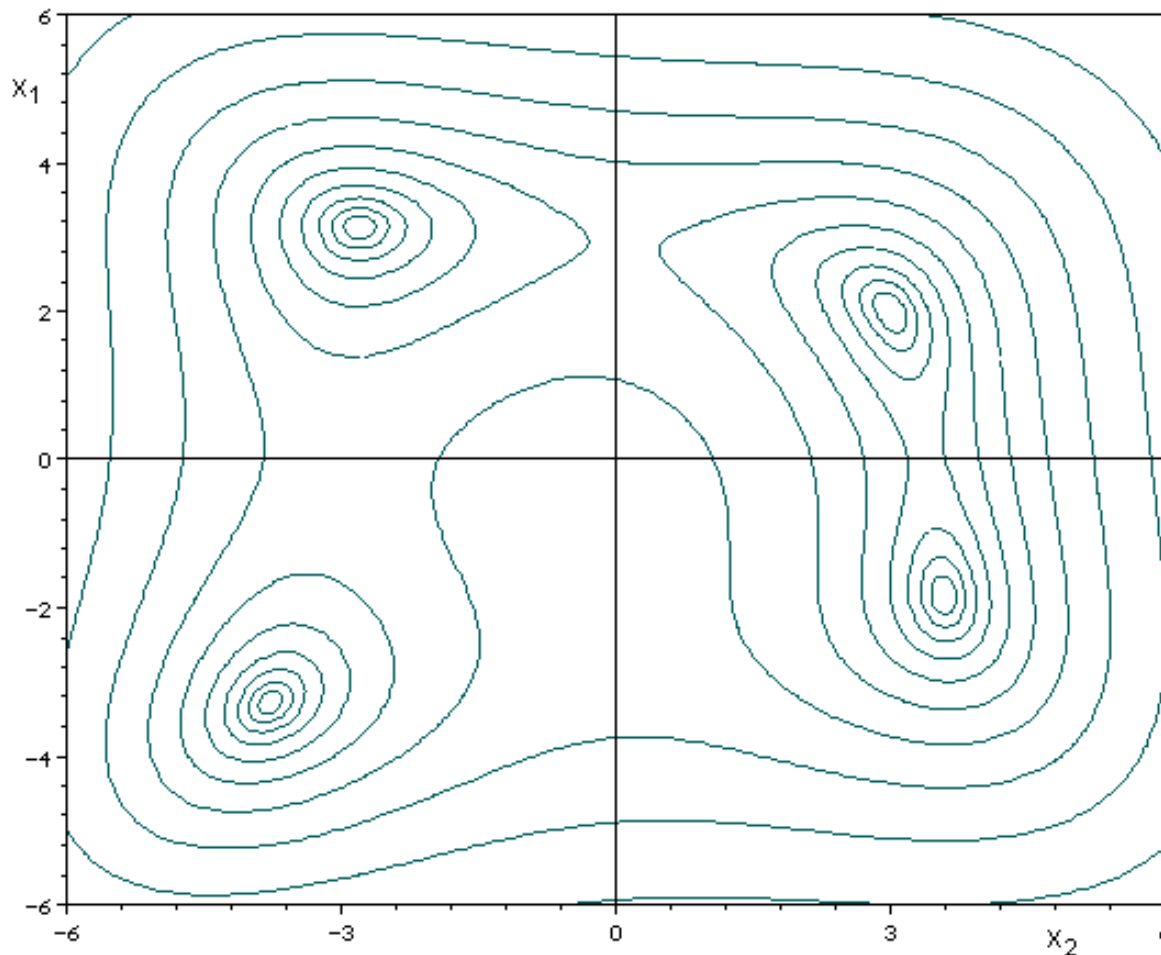
- коэффициент отражения  $\alpha > 0$  (обычно  $\alpha=1$ )
- коэффициент сжатия  $\beta > 0$  (обычно  $\beta=0,5$ )
- коэффициент растяжения  $\gamma > 0$  (обычно  $\gamma=2$ )

# Симплекс-метод

1. Инициализация:  $\mathbf{x}_1, \dots, \mathbf{x}_{n+1}$  - симплекс  $n$ -мерного пространства
2. Вычисляем  $\mathbf{f}(\mathbf{x}_i)$ ,  $i=1, \dots, n+1$
3. Выбираем  $m: \mathbf{f}(\mathbf{x}_m) = \max_{i=1, n+1} \mathbf{f}(\mathbf{x}_i)$
4. Вычисляем  $\mathbf{x}_c = \frac{1}{n} \sum_{i \neq m} \mathbf{x}_i$
5. Проводим оптимизацию шага в направлении от  $\mathbf{x}_m$  к  $\mathbf{x}_c$  (отражение и растяжение или сжатие)
6. Переходим на шаг 3, пока оптимизация изменяет положения вершин
7. Выбираем вершину с наименьшим значением  $\mathbf{f}$  и вокруг нее строим симплекс меньшего размера (сжатие)
8. Переходим на шаг 2, пока размер очередного симплекса больше требуемой точности

# Симплекс-метод

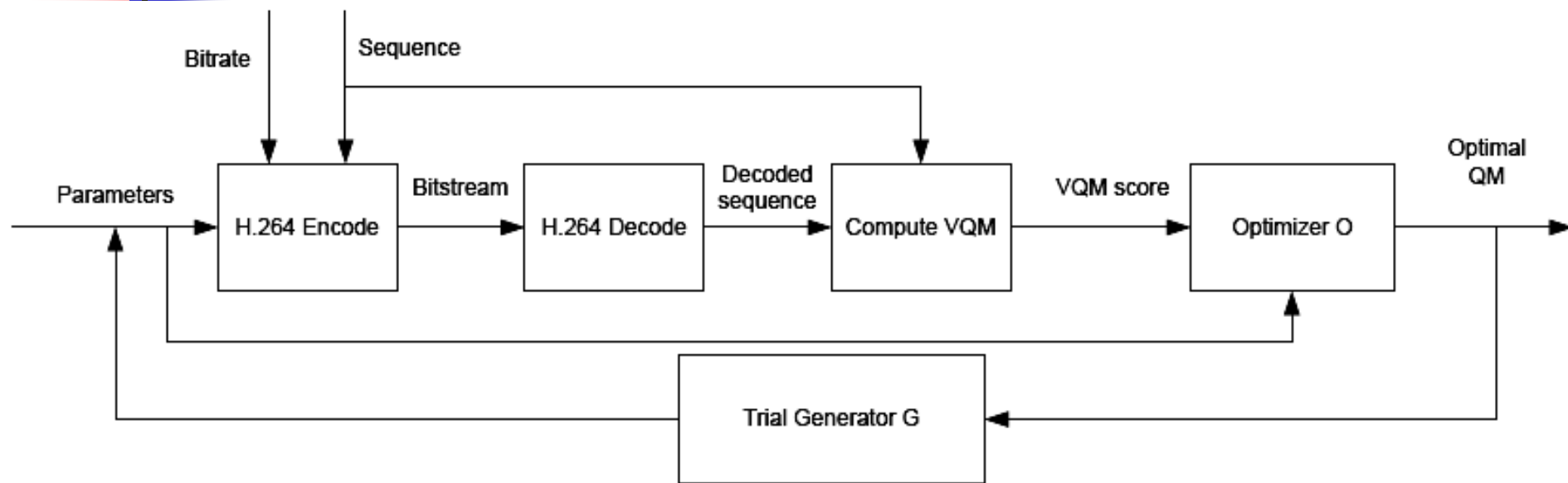
Nelder-Mead Simplex search over Himmelblau function



(c) P.A. Simionescu 2006



# Алгоритм на основе симплекс-метода



- Optimizer O контролирует выход из цикла
- Trial generator G генерирует очередной набор значений параметров с помощью симплекс-метода

Zhang, Huipin; Cote, Guy. Determining optimal configuration of video encoding parameters using numerical search algorithms. Proc. SPIE, 2008.

# Алгоритм на основе симплекс-метода

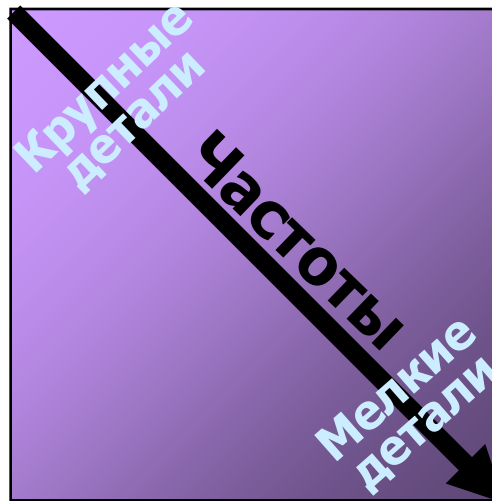


- Алгоритм на основе симплекс-метода был применен авторами статьи для решения следующих задач:
  - + Выбор адаптивного параметра квантования макроблоков
  - Выбор матрицы квантования

Zhang, Huipin; Cote, Guy. Determining optimal configuration of video encoding parameters using numerical search algorithms. Proc. SPIE, 2008.

# Процесс квантования при кодировании

- Макроблок после применения DCT преобразования:



- Для увеличения сжатия, отсекают наименее значимые детали с помощью квантования
- Квантование – деление величины на некоторое целое число, называемое квантом
- Операция квантования применяется дважды

# Процесс квантования при кодировании



1. С помощью матрицы квантования:

$$X_{\text{after\_QM}} = X_{\text{before\_QM}} \cdot / \text{QM}$$

- $X_{\text{after\_QM}}$  – макроблок до 1 квантования
- $X_{\text{before\_QM}}$  – макроблок после 1 квантования
- **QM** - матрица квантования (параметр видео)

2. С помощью порога квантования

$$X_{\text{after\_QP}} = \textit{round}(X_{\text{after\_QM}} / Q_{\text{step}})$$

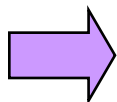
- $X_{\text{after\_QP}}$  – макроблок до 1 квантования
- $Q_{\text{step}}$  – шаг квантования, стандарт задает 52 возможных значения которые проиндексированы параметром квантования **QP** (параметр макроблока)
- Чем больше **QP**, тем выше степень сжатия и ниже качество

# Адаптивный параметр квантования макроблоков



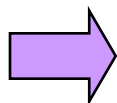
Система визуализации для человека (HVS) менее чувствительна к областям:

- с большим движением
- с высокими частотами
- где DC коэффициент (несущая частота) сильно отличается от DC коэффициента всего кадра



$$Q_{\text{step\_MB}} = Q_{\text{step\_PIC}} f_{\text{mot}} f_{\text{hf}} f_{\text{dc}}$$

- $Q_{\text{step\_MB}}$ ,  $Q_{\text{step\_PIC}}$  – шаги квантования макроблока и всего кадра
- $f_{\text{mot}}$ ,  $f_{\text{hf}}$ ,  $f_{\text{dc}}$  – модифицирующие коэффициенты движения, высоких частот и DC коэффициента



$$PQ_{\text{MB}} = PQ_{\text{PIC}} + \mathit{round}(6 \log_2(f_{\text{mot}} f_{\text{hf}} f_{\text{dc}}))$$

- $PQ_{\text{MB}}$ ,  $PQ_{\text{PIC}}$  – параметры квантования макроблока и всего кадра

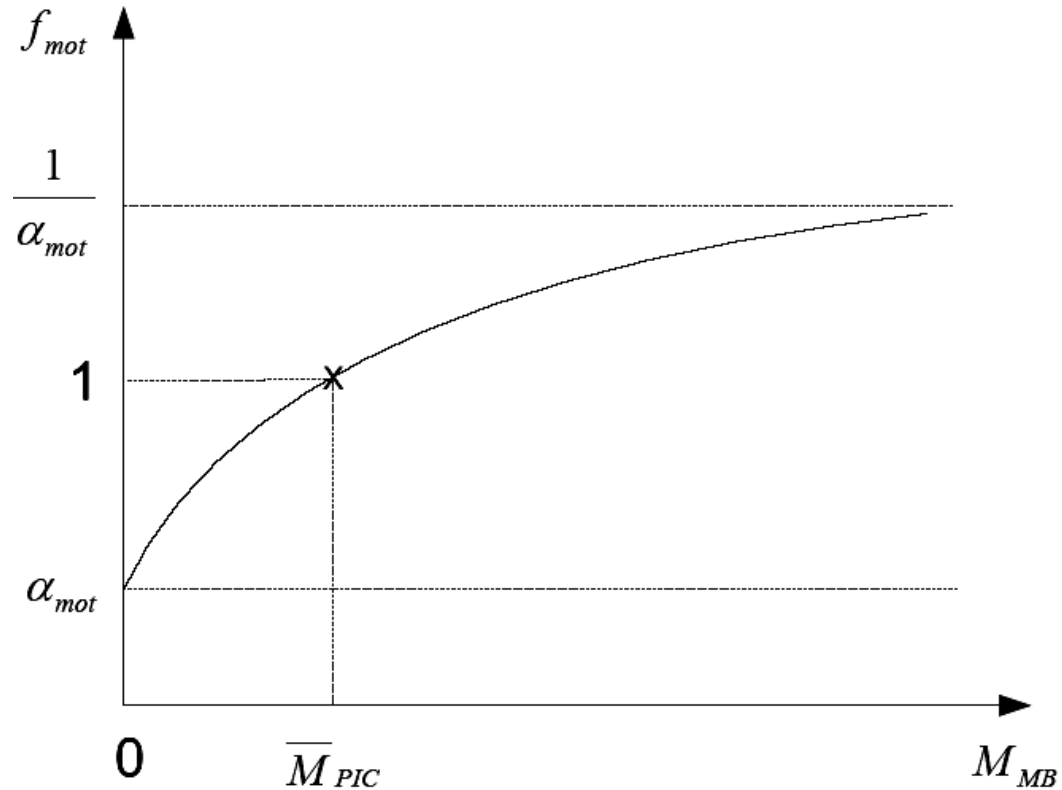
Zhang, Huipin; Cote, Guy. Determining optimal configuration of video encoding parameters using numerical search algorithms. Proc. SPIE. 2008.

# Модификационные коэффициенты

$$f_{mot} = \frac{M_{MB} + \alpha_{mot} \overline{M}_{PIC}}{\alpha_{mot} M_{MB} + \overline{M}_{PIC}}$$

$$f_{hf} = \frac{H_{MB} + \alpha_{hf} \overline{H}_{PIC}}{\alpha_{hf} H_{MB} + \overline{H}_{PIC}}$$

$$f_{dc} = \frac{D_{MB} + \alpha_{dc} \overline{D}_{PIC}}{\alpha_{dc} D_{MB} + \overline{D}_{PIC}}$$



# Алгоритм на основе симплекс-метода



- Симплекс-метод был применен для нахождения оптимальных коэффициентов  $\mathbf{a}_{\text{mot}}$ ,  $\mathbf{a}_{\text{hf}}$ ,  $\mathbf{a}_{\text{dc}}$  для 4-х последовательностей одновременно
- Начальные значения:  $(\mathbf{a}_{\text{mot}}, \mathbf{a}_{\text{hf}}, \mathbf{a}_{\text{dc}}) = (1, 1, 1)$
- Условие останова: VQM уменьшается менее чем на 0.001

Zhang, Huipin; Cote, Guy. Determining optimal configuration of video encoding parameters using numerical search algorithms. Proc. SPIE, 2008.

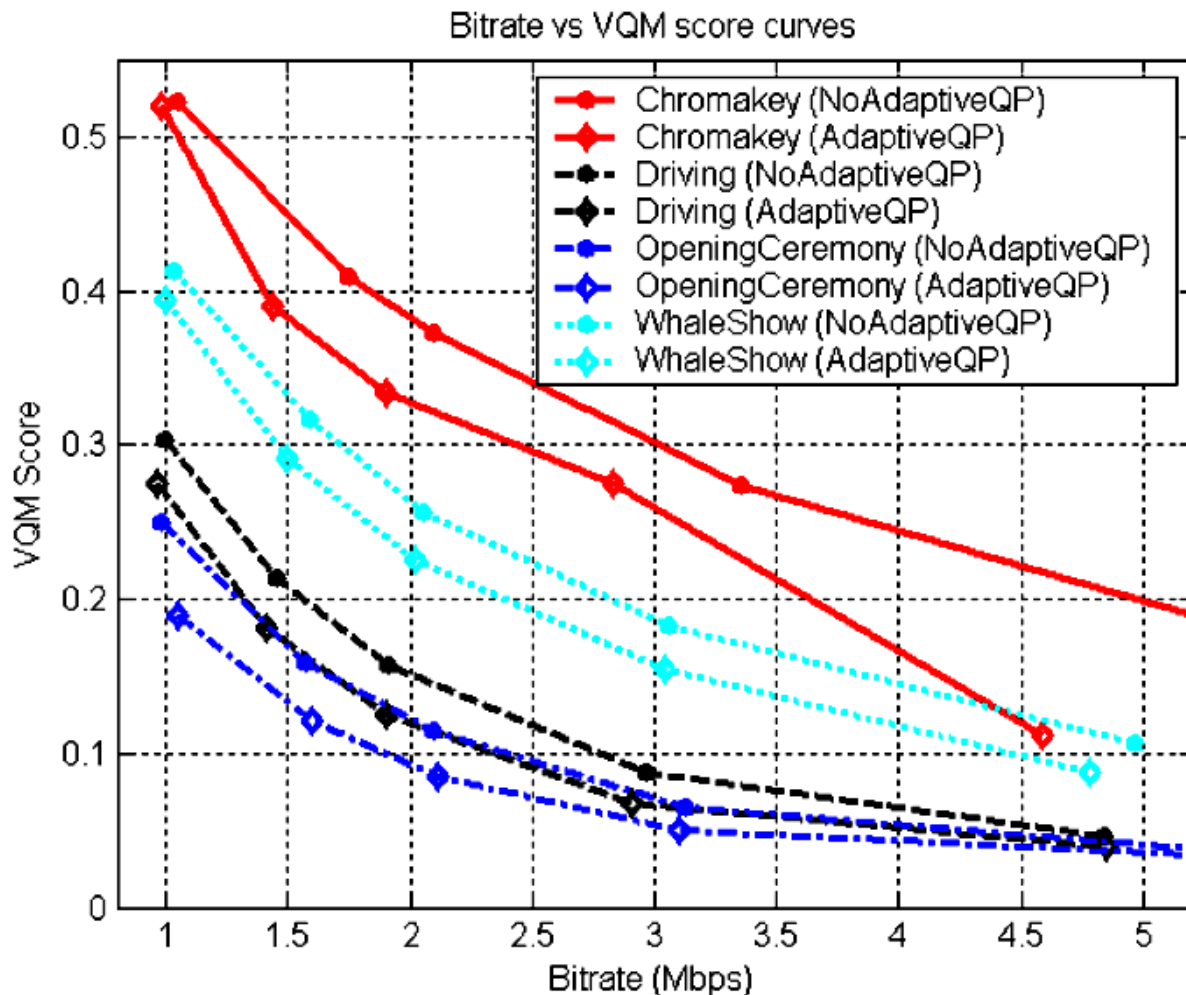
# Результаты

- high profile H.264 видеокодек
- 2-х уровневая B GOP структура длины 16
- Битрейт:
  - 1.5Mbps для определения оптимальных значений
  - 1.0, 1.5, 2.0, 3.0, 5.0 Mbps для сравнения
- Метрика качества – VQM
- 4 последовательности 480P (720x480 progressive), 30fps, 450 кадров
  - Chromakey
  - Driving
  - Opening Ceremony
  - Whale Show

Zhang, Huipin; Cote, Guy. Determining optimal configuration of video encoding parameters using numerical search algorithms. Proc. SPIE, 2008.



# Результаты



Zhang, Huipin; Cote, Guy. Determining optimal configuration of video encoding parameters using numerical search algorithms. Proc. SPIE, 2008.

# Содержание

- Введение
- **Поиск оптимальных пресетов**
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
    - Базовый алгоритм GBFOS
    - Итеративный алгоритм GBFOS
  - Алгоритм, основанный на симплекс-методе
    - Симплекс-метод
    - Алгоритм на основе симплекс метода
  - **Итеративный градиентный спуск**
- Анализ различных значений параметров
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- Результаты
- Планы

# Итеративный градиентный спуск

$p$  – текущий пресет

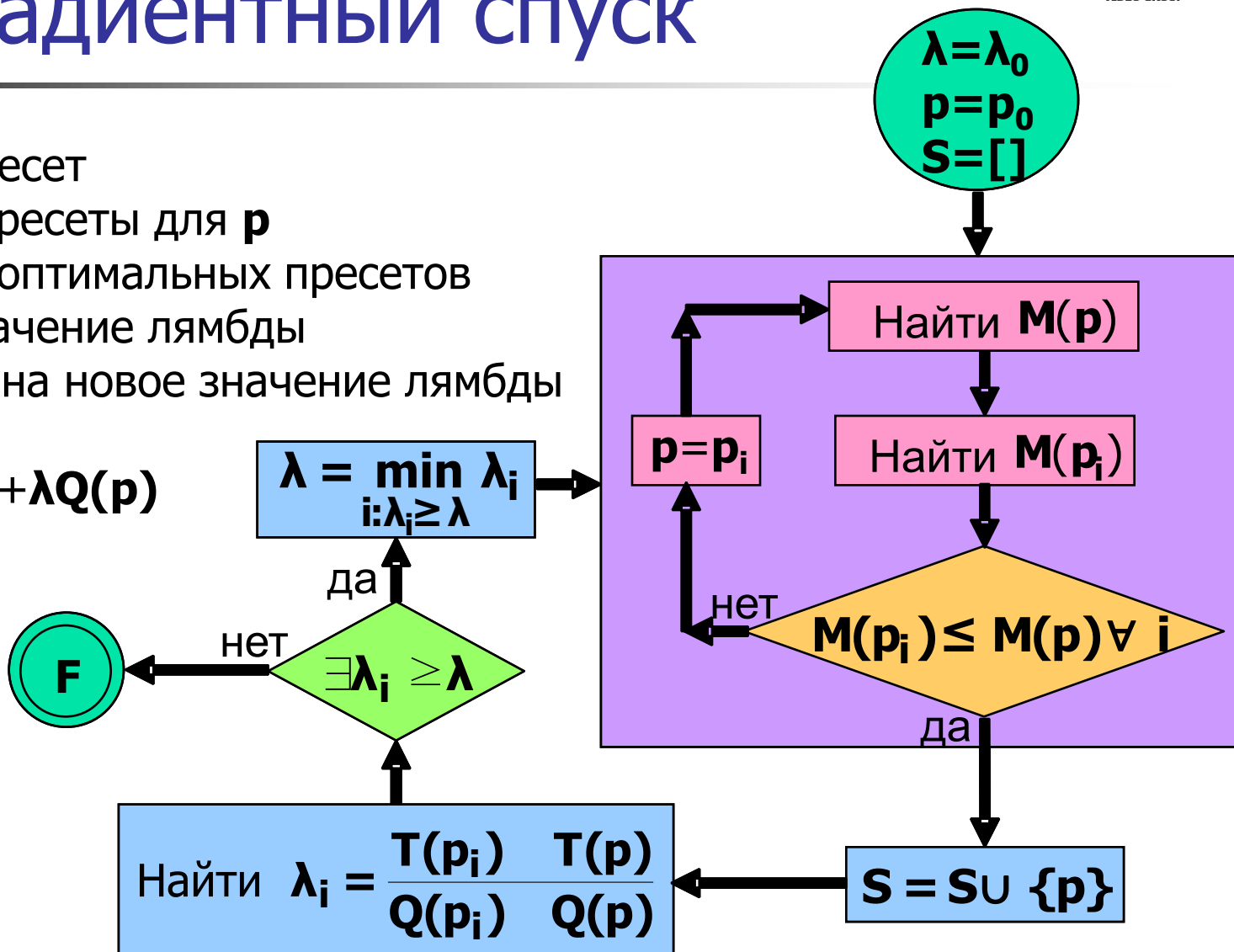
$p_i$  – соседние пресеты для  $p$

$S$  – множество оптимальных пресетов

$\lambda$  – текущее значение лямбды

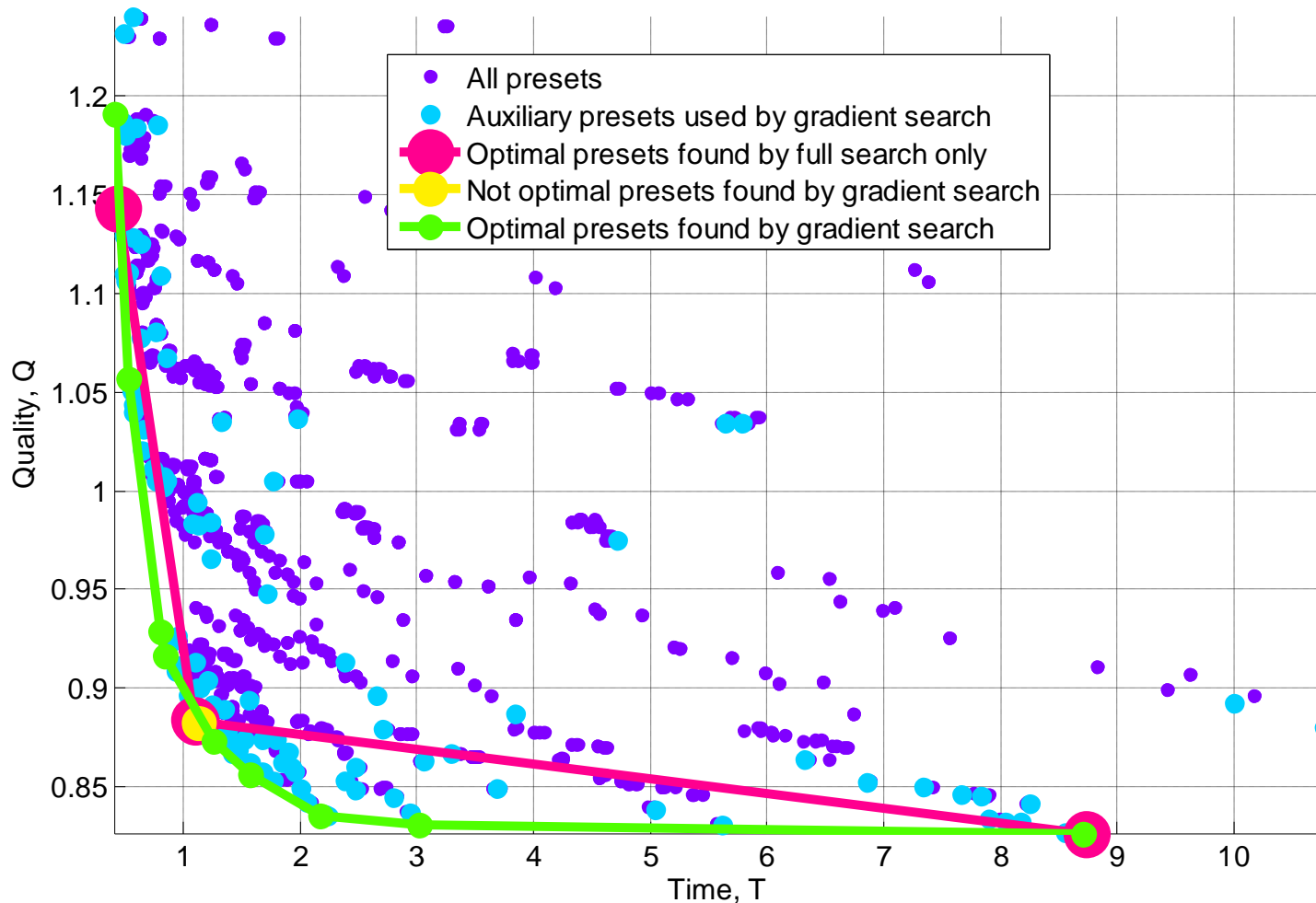
$\lambda_i$  – кандидаты на новое значение лямбды

$$M(p) = T(p) + \lambda Q(p)$$



# Результаты

## Gradient Search vs Full Search ("bus")



# Содержание

- Введение
- Поиск оптимальных пресетов
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
  - Алгоритм, основанный на симплекс-методе
  - Итеративный градиентный спуск
- **Анализ различных значений параметров**
  - **Подсчет средних характеристик по параметрам**
  - Разноцветные облака точек
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- Результаты
- Планы

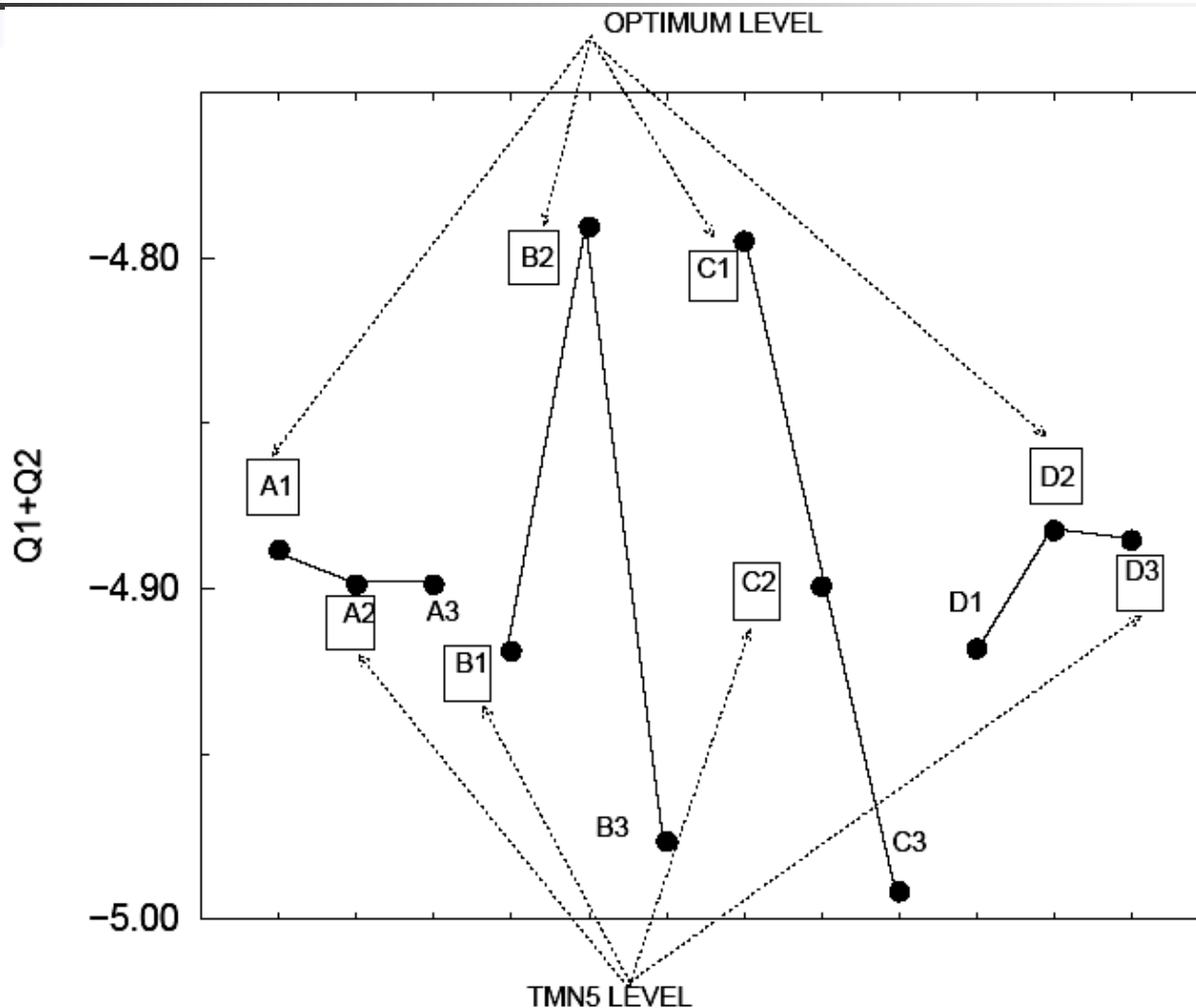
# Подсчет средних характеристик по параметрам

- В результате полного перебора пресетов, каждое значение параметра присутствует в замерах одинаковое количество раз
- Считается среднее значение и дисперсия
  - По PSNR
  - По битрейту
- Время кодирования не учитывается

$$Q1 = 10 \log \left( \frac{1}{n} \sum_{i=1}^n PSNR_i^2 \right) \quad Q2 = -10 \log \left( \frac{1}{n} \sum_{i=1}^n bps_i^2 \right)$$

- Для ускорения перебора используются **ортогональные МАССИВЫ**

# Результаты



# Результаты

- Обучение: "foreman"
- Тестирование: "carphone", "mother-daughter", "suzie", "salesman", "trevor"
- По сравнению с рекомендациями TMN5 (H.263) получено +0.2 dB в среднем (по PSNR)



# Содержание

- Введение
- Поиск оптимальных пресетов
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
  - Алгоритм, основанный на симплекс-методе
  - Итеративный градиентный спуск
- **Анализ различных значений параметров**
  - Подсчет средних характеристик по параметрам
  - **Разноцветные облака точек**
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- Результаты
- Планы

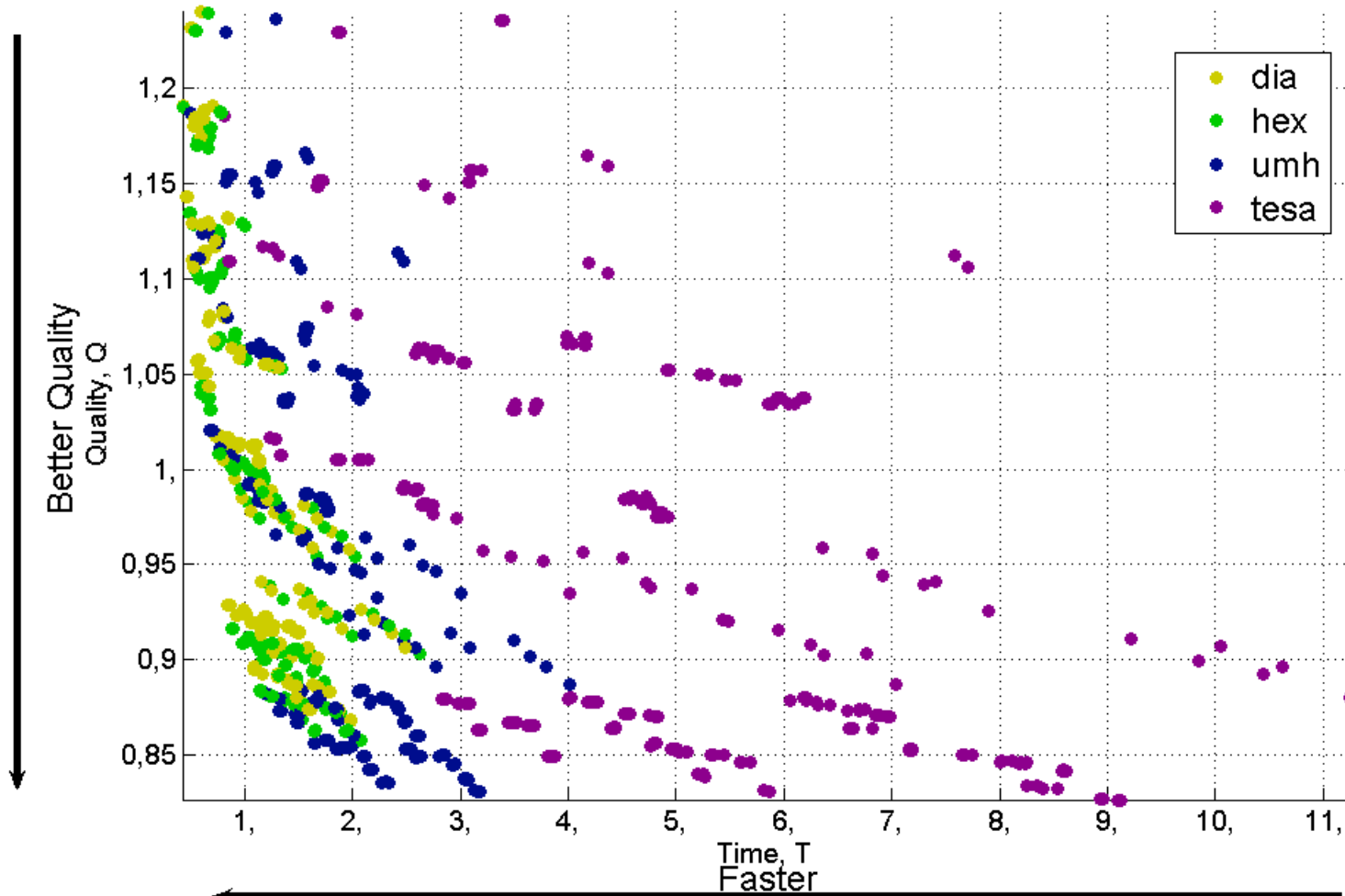
# Анализ раскрашенных облаков точек



- Графики строятся для каждого рассматриваемого параметра
- Точки, соответствующие одному значению выбранного параметра, раскрашены в один цвет
- Точки раскрашиваются в произвольном порядке
- Анализируют расположения облаков точек одного цвета:
  - Чем больше точек одного цвета лежит рядом с огибающей линией, тем лучше соответствующее этому цвету значение параметра

# Анализ раскрашенных облаков точек

Motion Estimation Method



# Анализ раскрашенных облаков точек



## Характеристики:

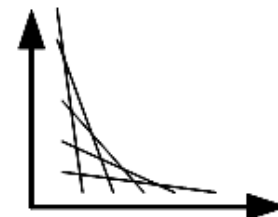
- + Простота и наглядность метода
- Выводы совершаются на основе субъективного визуального анализа расположения облаков точек одного цвета

# Содержание

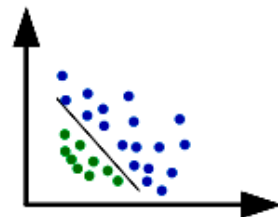
- Введение
- Поиск оптимальных пресетов
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
  - Алгоритм, основанный на симплекс-методе
  - Итеративный градиентный спуск
- **Анализ различных значений параметров**
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - **Лямбда анализ**
  - Анализ с помощью выкидывания огибающих
- Результаты
- Планы

# Лямбда анализ

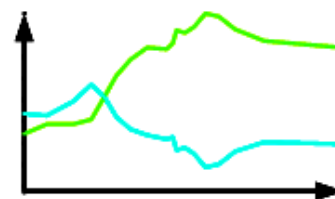
Selecting  $\lambda$



Best Presets for  $\lambda$



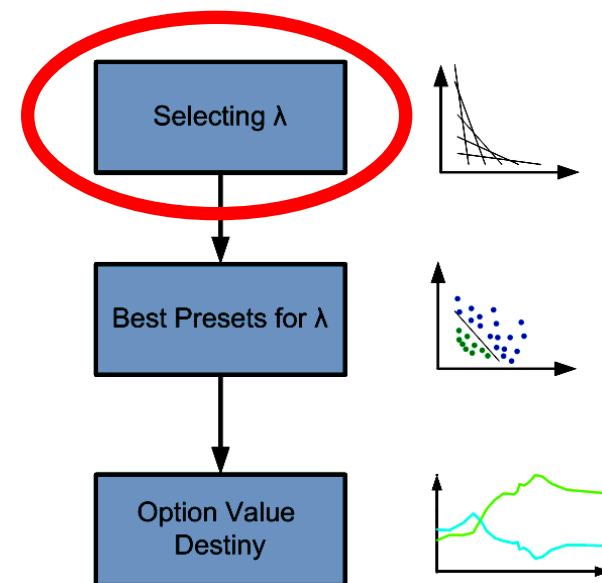
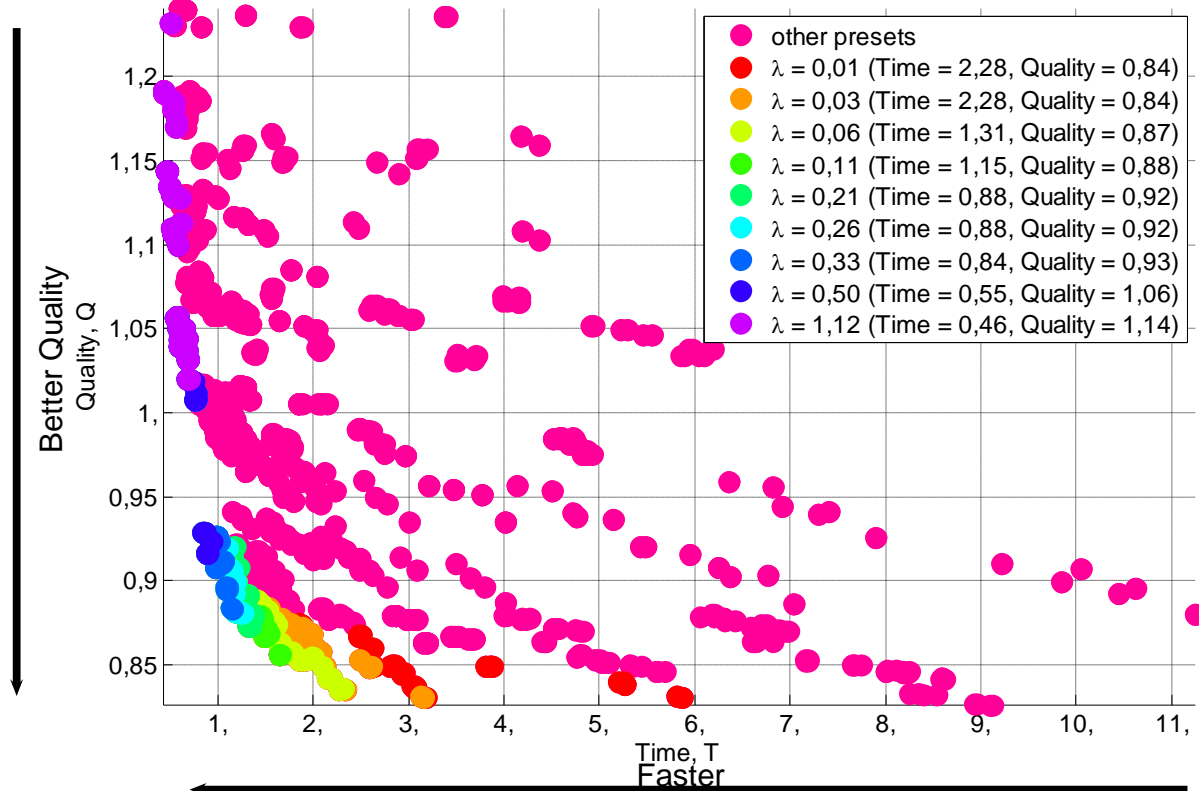
Option Value  
Destiny



# Выбор лямбд

Выбираем набор значений  $\lambda$  так, чтобы количество новых лучших пресетов для соседних  $\lambda$  было постоянным

10% Best Presets for Chosen  $\lambda$

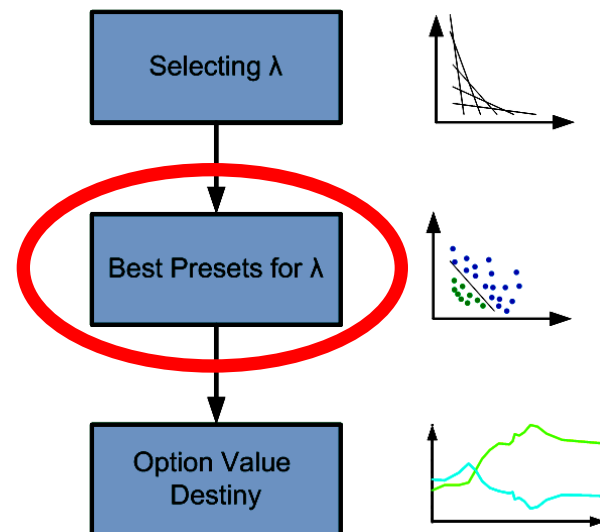


# Выбор лучших пресетов

- Для каждого значения  $\lambda$  рассматривается величина:

$$M(p) = T(p) + \lambda Q(p)$$

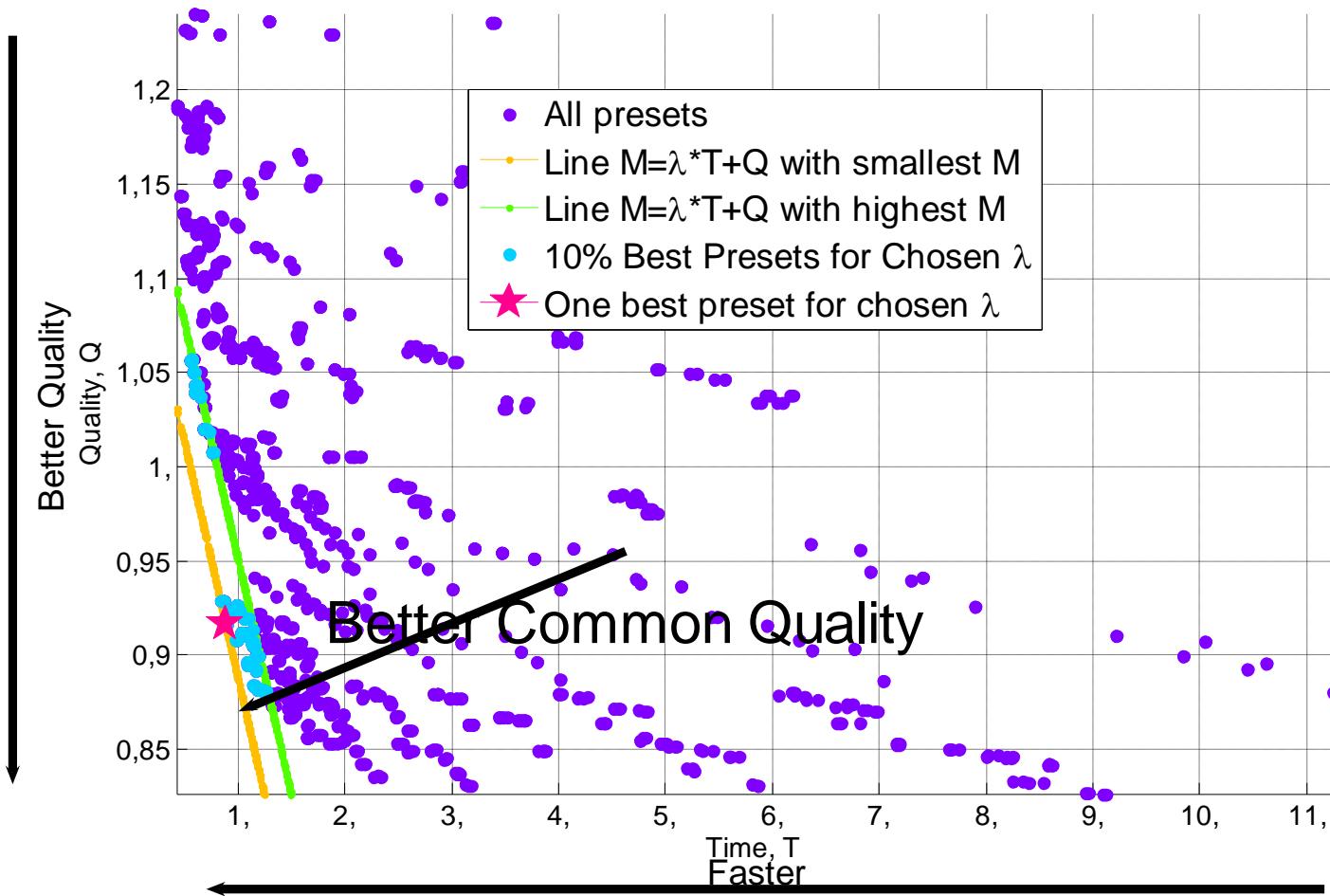
- Лучшими считаются 10% пресетов с минимальным значением  $M(p)$





# Интерпретация лямбды

10% Best Presets for Chosen  $\lambda = 0,25$



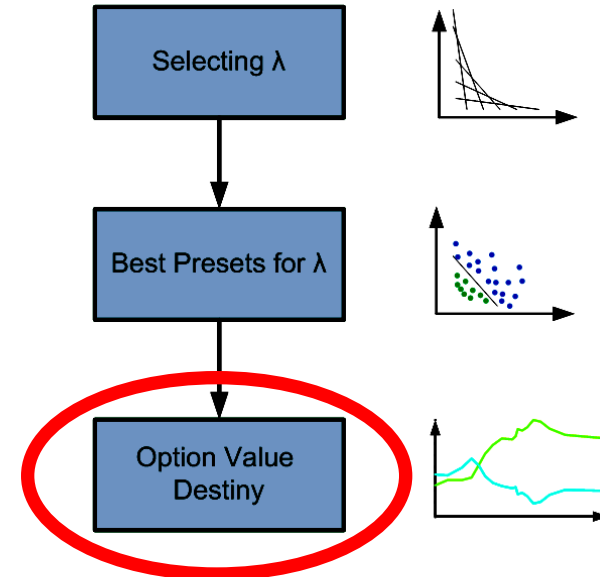
# Вычисление плотности пресетов

Для каждого параметра строится график зависимости плотности пресетов с каждым возможным значением  $k$  от  $\lambda$ :

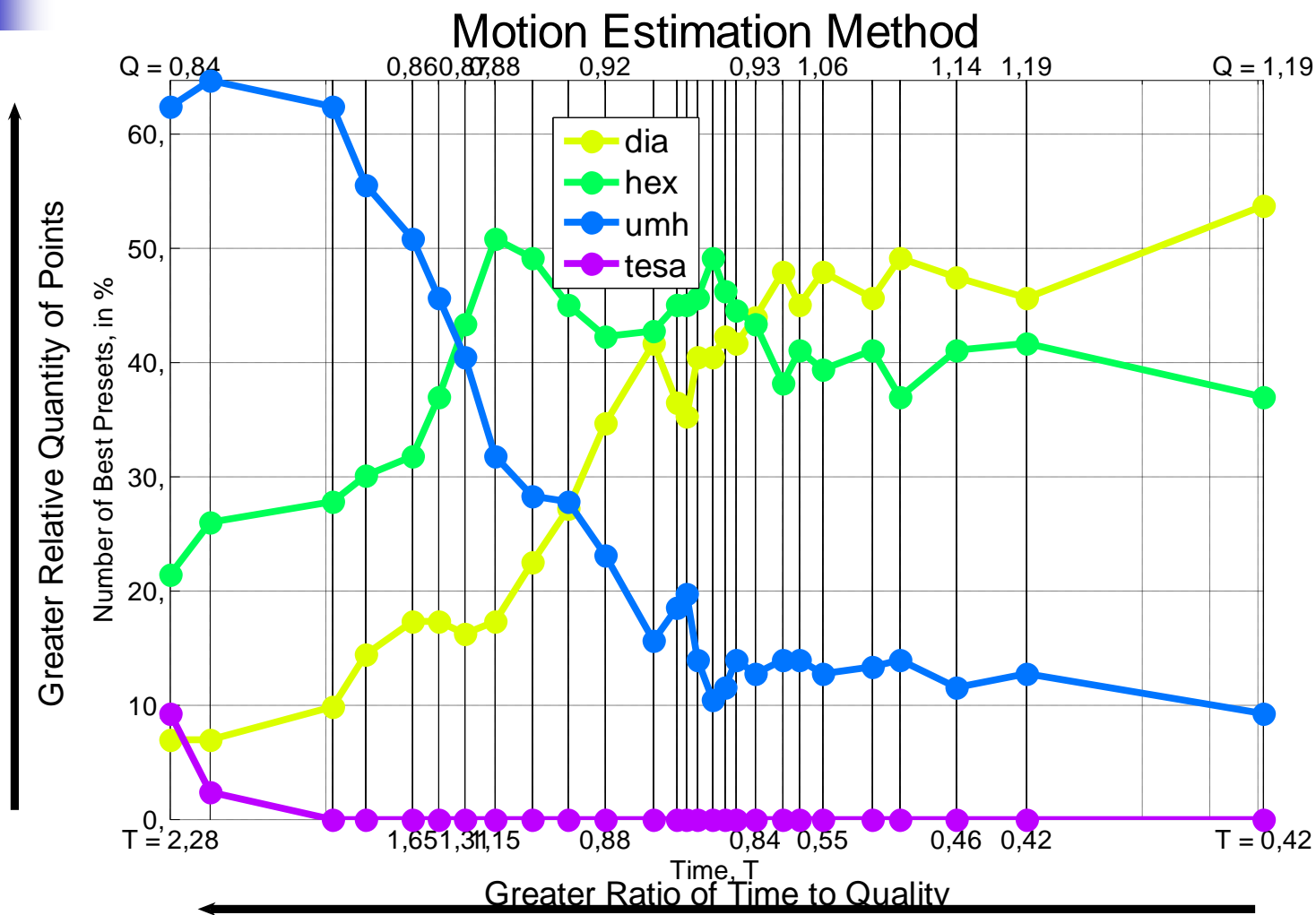
$$N_{\%}(\lambda, k) = 100 \frac{N(\lambda, k)}{\sum_m N(\lambda, m)}$$

$$N(\lambda, k) = \frac{N_k^\lambda}{N_k}$$

- $N_{\%}(\lambda, k)$  - плотность в % пресетов со значением параметра равным  $k$  для заданной  $\lambda$
- $N_k^\lambda$  - количество пресетов со значением параметра равным  $k$  среди 10% лучших пресетов для заданной  $\lambda$
- $N_k$  - количество пресетов со значением параметра равным  $k$  среди всех пресетов



# Лямбда анализ





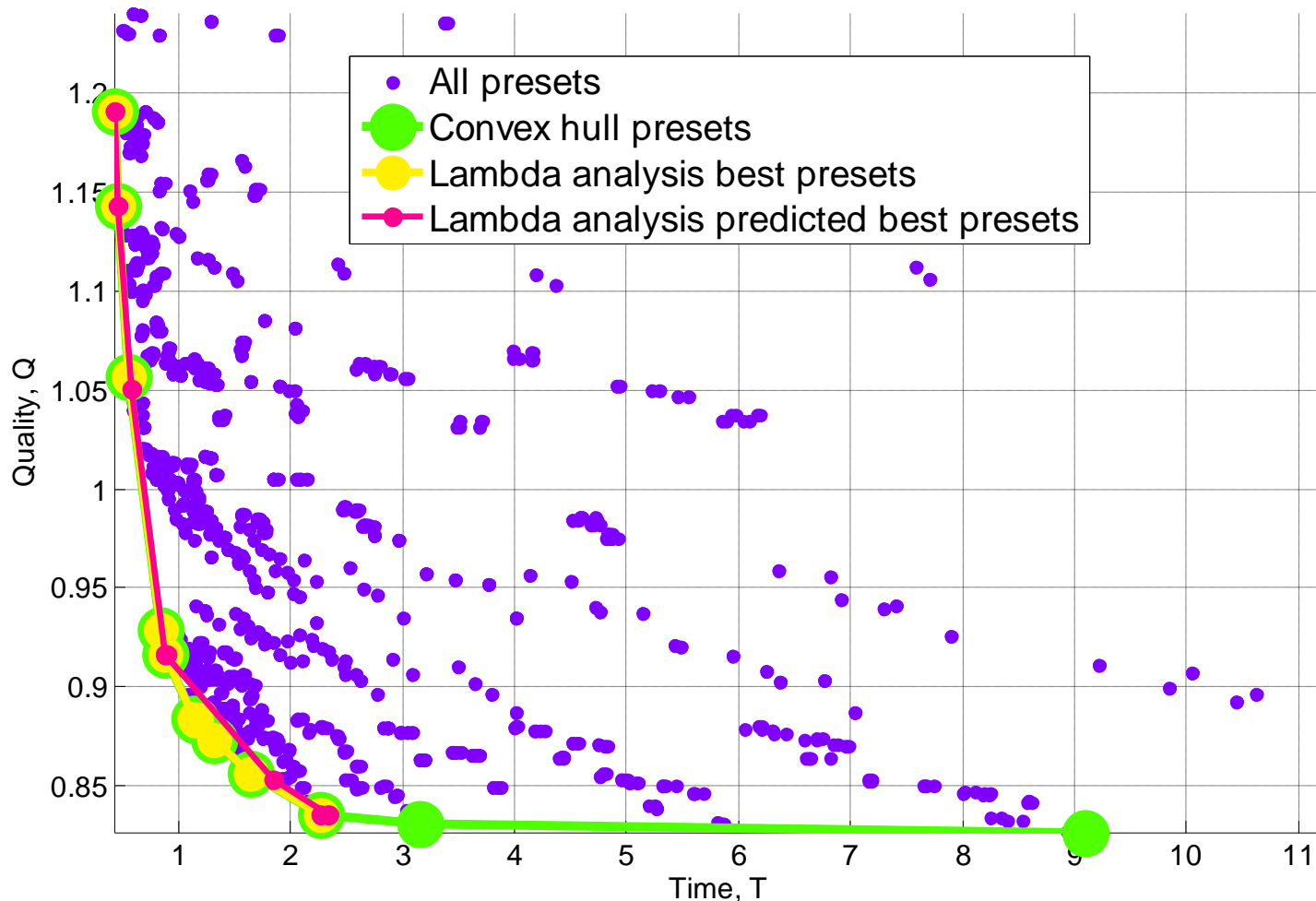
# Лямбда анализ

---

- Анализируют линии на графике соответствующие разным значениям заданного параметра:
  - Чем выше линия, тем лучше значение параметра
  - Если у линии есть максимум при каком-то  $\lambda$ , значит это значение  $\lambda$  оптимально, если мы хотим использовать это значение параметра

# Лямбда анализ

## Lambda Analysis vs Convex Hull Presets



# Лямбда анализ

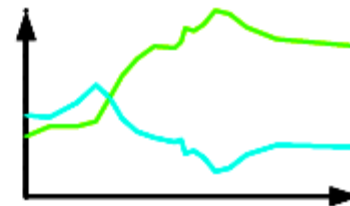
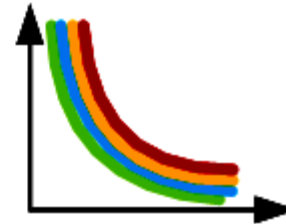
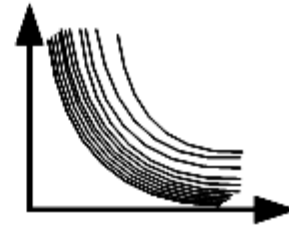
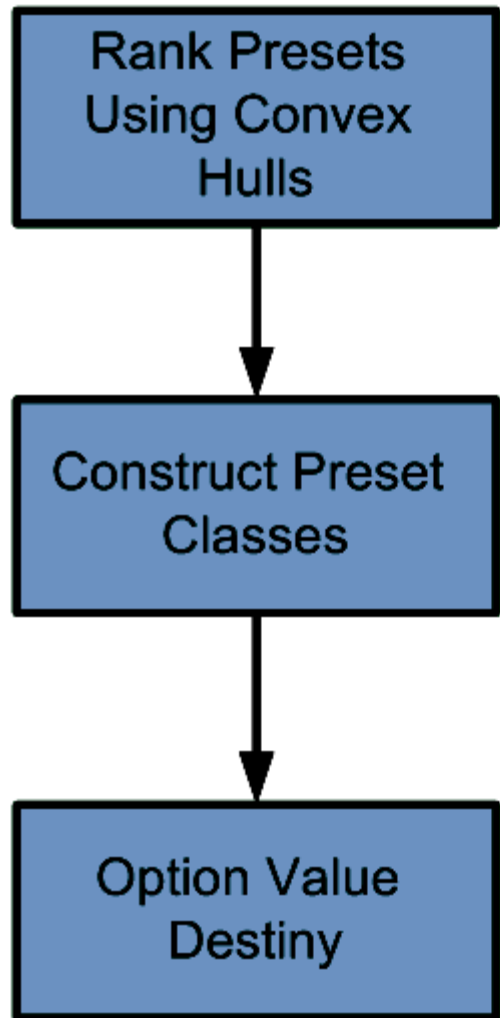
## Характеристики:

- + Позволяет оценивать эффективность кодирования при различном качестве кодирования
- + Позволяет находить оптимальное соотношение времени и качества для конкретного значения параметра (точка максимума на графике)
- Выбор набора значений  $\lambda$
- Информация о параметрах, лучших в экстремальных случаях может потеряться
- Выводы анализа зависят от выбора значений параметров\*

# Содержание

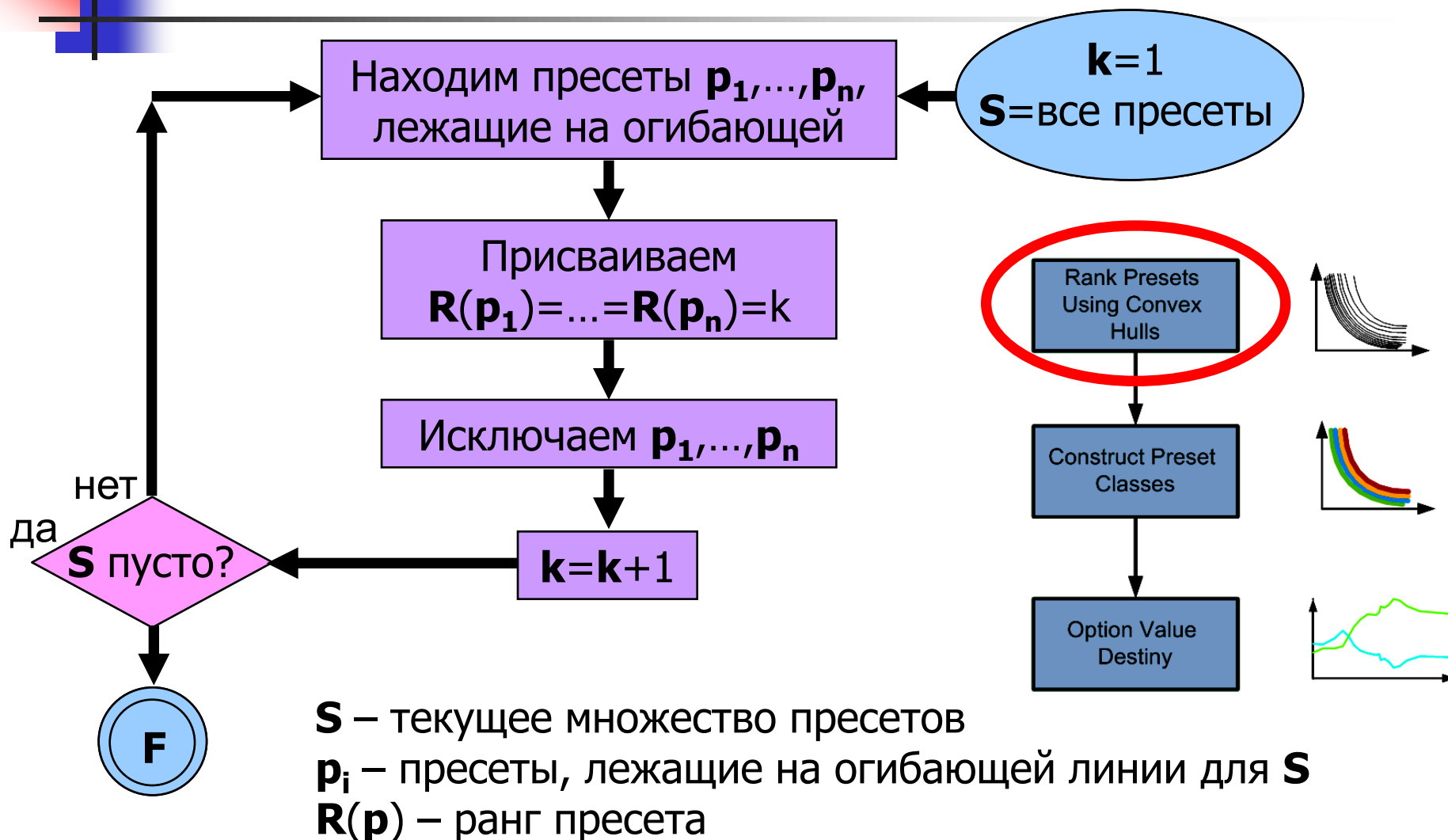
- Введение
- Поиск оптимальных пресетов
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
  - Алгоритм, основанный на симплекс-методе
  - Итеративный градиентный спуск
- **Анализ различных значений параметров**
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - Лямбда анализ
  - **Анализ с помощью выкидывания огибающих**
- Результаты
- Планы

# Анализ с помощью выкидывания огибающих





# Ранжирование пресетов

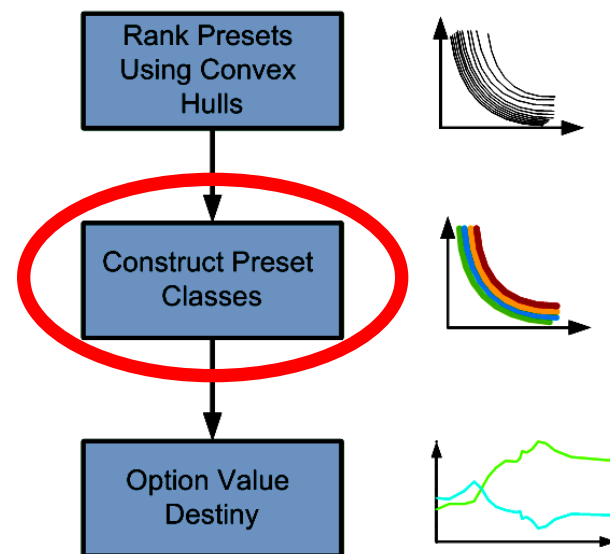
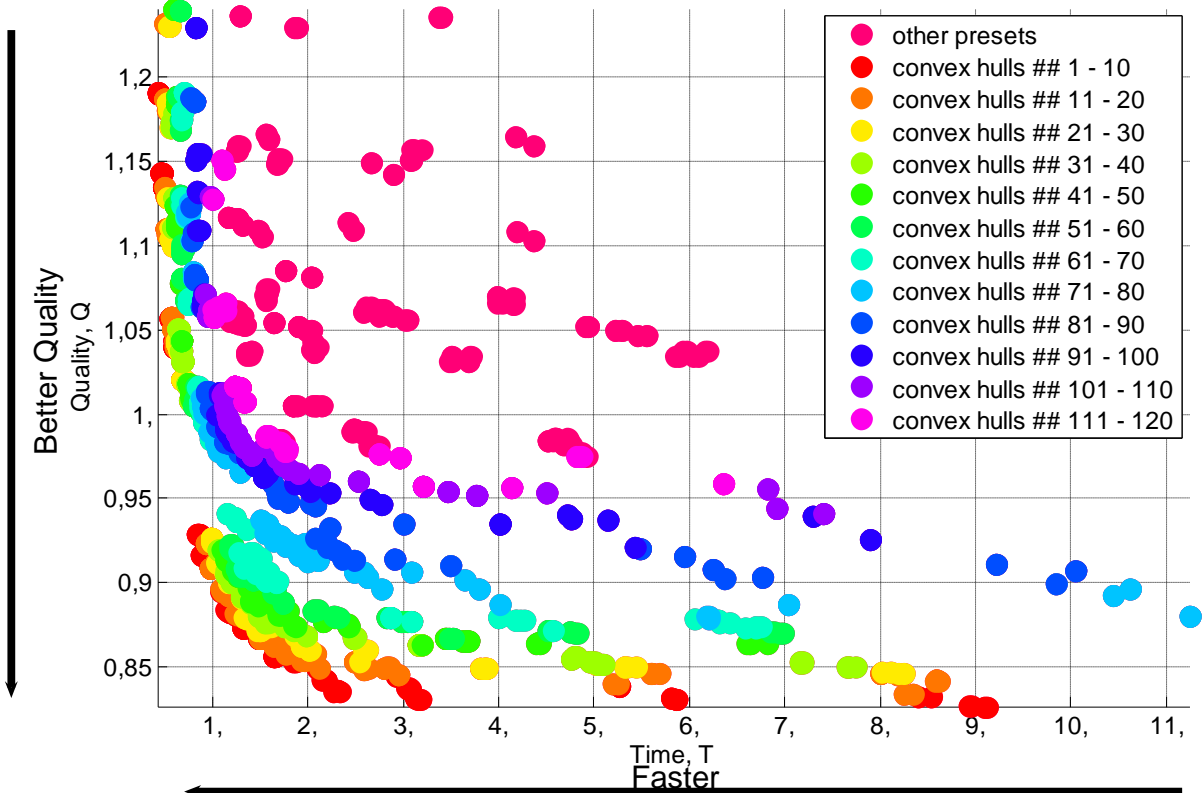


# Разбиение пресетов на классы

Разбиваем все пресеты на классы по рангу R:

$$\text{class}_n = \{p \mid R(p) \in [1 + 10n, 11n]\}, n \geq 0$$

Distance from Convex Hull



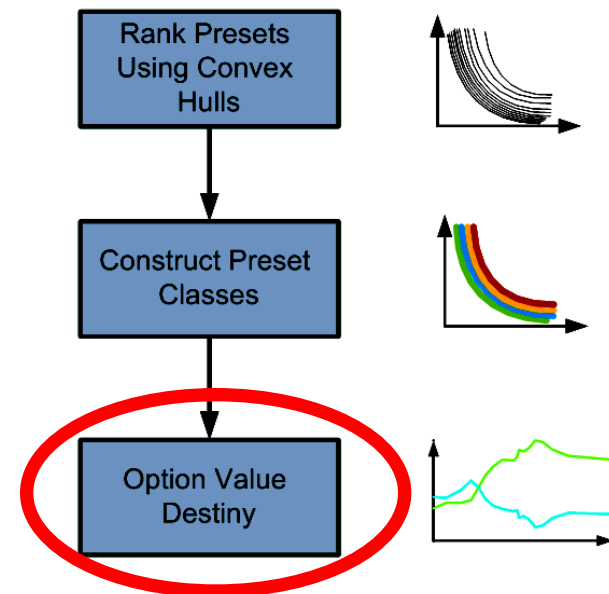
# Вычисление плотности пресетов

Для каждого параметра строится график зависимости плотности пресетов с каждым возможным значением  $k$  от номера класса  $n$ :

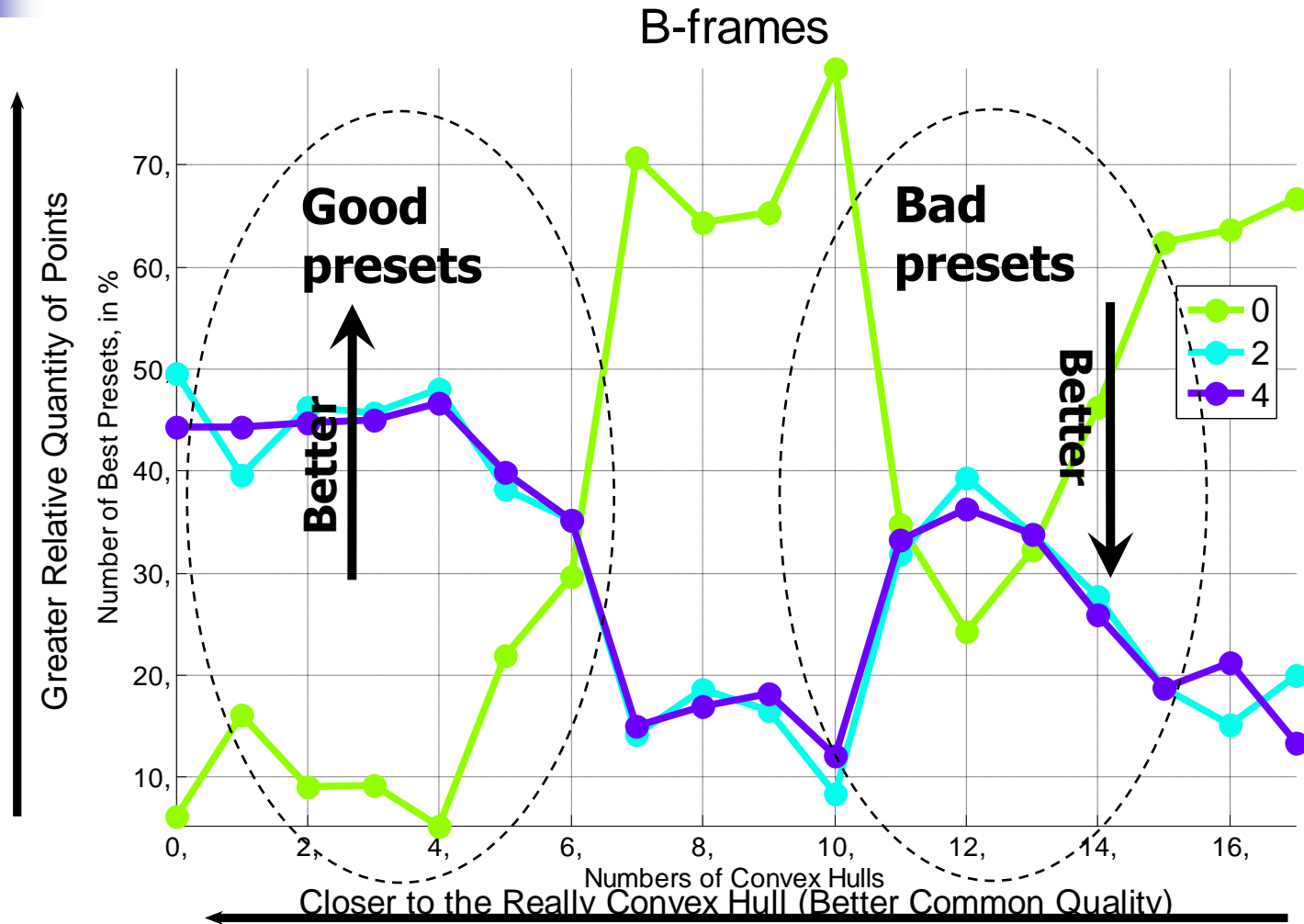
$$N_{\%}(n, k) = 100 \frac{N(n, k)}{\sum_m N(n, m)}$$

$$N(n, k) = \frac{N_k^n}{N_k}$$

- $N_{\%}(n, k)$  - плотность в % пресетов со значением параметра равным  $k$  в классе  $n$
- $N_k^n$  - количество пресетов со значением параметра равным  $k$  среди 10% лучших пресетов в классе  $n$
- $N_k$  - количество пресетов со значением параметра равным  $k$  среди всех пресетов



# Анализ с помощью выкидывания огибающих



# Анализ с помощью выкидывания огибающих



- Анализируют линии на графике соответствующие разным значениям заданного параметра:
  - Чем выше линия для маленьких номеров классов, тем лучше значение параметра
  - Чем ниже линия для больших номеров классов, тем лучше значение параметра
  - Если две линии пересекаются, то можно сказать, что одно значение параметра лучше другого

# Анализ с помощью выкидывания огибающих



## Характеристики:

- + Позволяет находить оптимальные значения параметров, когда нет априорных знаний о желаемом соотношении между скоростью кодирования и качеством закодированного видео
- Выводы анализа зависят от выбора значений параметров\*

# Содержание

- Введение
- Поиск оптимальных пресетов
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
  - Алгоритм, основанный на симплекс-методе
  - Итеративный градиентный спуск
- Анализ различных значений параметров
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- **Результаты**
- Планы

# Результаты (вместе с Володей)



- Разработана система для массовых замеров параметров видеокодека
- Разработан метод поиска оптимальных пресетов, требующий запуска около 15% пресетов
- Разработаны среда для анализа параметров 2-х критериальной задачи:
  - Анализ раскрашенных облаков
  - Лямбда анализ
  - (Анализ с помощью выкидывания огибающих)
- Проведен анализ параметров видеокодеков с помощью среды
- Проведен анализ стабильности методов анализа относительно:
  - Выбора последовательностей
  - Ошибок замеров



# Содержание

- Введение
- Поиск оптимальных пресетов
  - Основные понятия
  - Алгоритм, основанный на методе отсечения ветвей дерева (GBFOS)
  - Алгоритм, основанный на симплекс-методе
  - Итеративный градиентный спуск
- Анализ различных значений параметров
  - Подсчет средних характеристик по параметрам
  - Разноцветные облака точек
  - Лямбда анализ
  - Анализ с помощью выкидывания огибающих
- Результаты
- **Планы**



# Планы

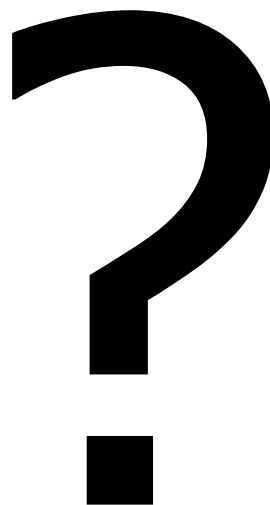
---

- Верификация методов анализа
- Анализ стабильности методов анализа относительно:
  - выбора последовательностей
  - выбора значений параметров
  - ошибок замеров
  - выбора метрики качества
- Объединение выводов нескольких методов анализа
- Анализ зависимостей параметров
- Возможность анализа параметров по неполным данным
- Разработка других методов анализа



# Вопросы

---





# Литература

---

1. R. Vanam, E. Riskin, S. Hemami, R. Ladner. Distortion-Complexity Optimization of the H.264/MPEG-4 AVC Encoder using the GBFOS Algorithm. Proceedings of the IEEE Data Compression Conference, Snowbird, UT, March 2007.
2. Zhang, Huipin; Cote, Guy. Determining optimal configuration of video encoding parameters using numerical search algorithms. Proceedings of the SPIE, Volume 6822, pp. 682211-682211-7 (2008).
3. Лотов А.В., Поспелова И.И. Конспект лекций по теории и методам многокритериальной оптимизации. – М.: МГУ, 2008.
4. Sangyoun Lee; Madisetti, V.K. Parameter optimization of robust low-bit-rate video coders. IEEE, 1999.
5. D.Saupe, M.Ruhl, R.Hamzaoui, L.Grandi, D.Marini. Optimal hierarchical partitions for fractal image compression. IEEE, 1998.