

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)

Факультет: Механико-математический факультет

Кафедра: Математического моделирования

Направление подготовки: 01.04.02 - Прикладная математика и информатика

## КУРСОВАЯ РАБОТА МАГИСТРА

Герасименко Дмитрий Владимирович

Интервальный подход к повышению стабильности частоты  
термокомпенсированного кварцевого генератора

**«К защите допущена»**

Заведующий кафедрой,

д.ф.-м.н, проф.

Шарый С. П. / \_\_\_\_\_

(фамилия, И., О.) / (подпись, МП)

«\_\_\_\_\_» \_\_\_\_\_ 2017г.

**Научный руководитель**

Заведующий кафедрой,

д.ф.-м.н, проф.

Шарый С. П. / \_\_\_\_\_

(фамилия, И., О.) / (подпись, МП)

«\_\_\_\_\_» \_\_\_\_\_ 2017г.

Дата защиты: «\_\_\_\_\_» \_\_\_\_\_ 2017г.

Новосибирск, 2017

# Оглавление

Введение . . . . .	3
<b>1 Постановка задачи</b>	<b>4</b>
1.1 Описание цифровых моделей . . . . .	6
1.2 Частотно-тепловая модель . . . . .	9
<b>2 Постановка задачи и анализ методов её решения</b>	<b>12</b>
2.1 Задача подбора коэффициентов . . . . .	12
2.2 Описание алгоритма . . . . .	14
2.3 Результаты . . . . .	17
Заключение и дальнейшие планы . . . . .	19

# Введение

Термокомпенсированный кварцевый генератор — это полупроводниковая микросборка, состоящая из кварцевого резонатора и микросхемы-генератора. Собственная частота резонатора достаточно стабильна, но зависит от температуры. Зависимость от температуры различается для разных экземпляров генератора.

Для работы с кварцевым генератором запускается USB программатор, предназначенный для тестирования и программирования микросхем генераторов. Программатор обеспечивает вывод частоты с тестируемого генератора на внешний измерительный прибор. Протокол работы программатора сохраняется во внешнем файле на компьютере. Эти данные позволяют анализировать работу генератора по трём различным моделям: идеализированной, производственной, модели спецификации.

Целью задачи является нахождение и верификация оптимального набора коэффициентов для стабилизации частоты генератора, подаваемых ему на вход. Особенностью моделей производственной и спецификации является их разрывность и несуществование производных. Данные обстоятельства ограничивают набор алгоритмов и инструментов для решения поставленной задачи.

# Глава 1

## Постановка задачи

Кварцевый резонатор генерирует собственные механические колебания и влияет на их частоту с помощью варикапа — ёмкости, управляемой напряжением, которое создаёт цифроаналоговый преобразователь (ЦАП). В состав сборки входит датчик температуры микросхемы, напряжение на котором линейно зависит от цифровой температуры, безразмерной целочисленной величины, характеризующей температуру на резонаторе. Назначение варикапа — компенсировать температурную зависимость так, чтобы уменьшить дрейф частоты колебаний. Кроме того, в микросхеме есть магазин постоянных ёмкостей, каждая из которых также может сдвинуть частоту. Частота, выдаваемая микросборкой, представляется в виде функции  $f(t, u, c)$ . Параметрами этой функции являются:

- $t$  — температура резонатора;
- $u$  — цифровое значение на входе ЦАП, представляющееся в виде 12-разрядного целого числа от 0 до 4095, вычисляемого по температуре и заданному вектору коэффициентов;

- $c$  — ёмкость, выбранная из магазина ёмкостей, не зависящая от температуры.

Микросхема содержит внутри себя:

- датчик температуры микросхемы;
- аналогово-цифровой преобразователь (АЦП), переводящий аналоговое показание датчика в 12-разрядный цифровой код  $T$ , диапазон которого используется не полностью, а приблизительно от 1000 до 3000;
- цифровой вычислитель  $u = u(T, K)$ , где  $K$  — вектор коэффициентов;
- программируемая память коэффициентов  $K$ ;
- магазин ёмкостей  $c$ .

Напряжение, подаваемое цифроаналоговым преобразователем на варикап, зависит от вектора  $K$ . Перед нами стоит задача найти такой набор коэффициентов, при котором выдаваемое на варикап напряжение стабилизирует частоту  $f$  собственных колебаний микросборки к заданному значению частоты  $f_0$ .

Вектор коэффициентов  $K$ , используемый для вычисления  $u$ , состоит из компонент  $INFBIT, SBIT, K1BIT, K2BIT, K3BIT, K4BIT, K5BIT$ . Компоненты вектора — целые числа в следующих диапазонах, представленные в виде двоичного числа заданной разрядности:

Бит	Диапазон	Разрядность
INFBIT	[0, 63]	6 бит
SBIT	[0, 31]	5 бит
K1BIT	[1, 255]	8 бит
K2BIT	[0, 127]	7 бит
K3BIT	[0, 31]	5 бит
K4BIT	[0, 31]	5 бит
K5BIT	[0, 15]	4 бит

Если все компоненты вектора  $K$  записать друг за другом, то получим 40-битное двоичное число, просуммировав разрядность всех компонент.

Магазин ёмкостей включает в себя 16 ёмкостей грубой настройки и 64 ёмкости точной настройки. Выбранные ёмкости грубой и точной настройки соединены параллельно.

Выбор ёмкости точной и грубой настройки из магазина задаётся следующими параметрами, представляющими собой номер выбранной ёмкости соответствующей настройки:

Бит	Диапазон	Разрядность
CDACC	[0, 15]	4 бит
CDACF	[0, 63]	6 бит

Значение ёмкости, выбранной из магазина, равно  $c = CDACC + CF2CC \cdot CDACF$ , где  $CF2CC = 0.034$  в условных единицах.

## 1.1 Описание цифровых моделей

Опишем оговоренные во введении модели, представляющие расчёты, проводимые цифровым вычислителем  $u$ .

Модель спецификации:

$$\begin{aligned}xd &= T - (1535 + 8 \cdot INFBIT), \\xs &= \text{floor} \left( \frac{xd \cdot (16 + SBIT)}{32} + 0.5 \right), \\pr2 &= (K4BIT - 25) \cdot 512 + K5BIT \cdot xs, \\res3 &= (K3BIT \cdot 512 + 4096) + \text{floor} \left( \frac{pr2 \cdot xs}{1024} + 0.5 \right), \\res4 &= (K2BIT \cdot 128 + 2560) + \text{floor} \left( \frac{res3 \cdot xs}{1024} + 0.5 \right), \\res5 &= (-K1BIT \cdot 128 - 12480) + \text{floor} \left( \frac{res4 \cdot xs - 1}{1024} \right), \\res6 &= 1032 + \text{floor} \left( \frac{res5 \cdot xs}{16384} + 0.5 \right), \\u &= \max(0, \min(4095, res6)).\end{aligned}$$

Изначально цифровой вычислитель был задуман проводить вычисления по модели спецификаций. Однако в процессе изготовления микросхемы умножитель был реализован с ошибкой, добавляющей к некоторым произведениям дополнительный перенос на некоторое количество бит. Учитывающую данную ошибку модель назовём моделью аппаратуры.

Модель аппаратуры:

$$xd = T - (1535 + 8 \cdot INFBIT),$$

$$\begin{aligned}
xs &= \text{floor} \left( \frac{xd \cdot (16 + SBIT) + p0(T, K)}{32} + 0.5 \right), \\
pr2 &= (K4BIT - 25) \cdot 512 + K5BIT \cdot xs + p1(T, K), \\
res3 &= (K3BIT \cdot 512 + 4096) + \text{floor} \left( \frac{pr2 \cdot xs + p2(T, K)}{1024} + 0.5 \right), \\
res4 &= (K2BIT \cdot 128 + 2560) + \text{floor} \left( \frac{res3 \cdot xs + p3(T, K)}{1024} + 0.5 \right), \\
res5 &= (-K1BIT \cdot 128 - 12480) + \text{floor} \left( \frac{res4 \cdot xs + p4(T, K) - 1}{1024} \right), \\
res6 &= 1032 + \text{floor} \left( \frac{res5 \cdot xs + p5(T, K)}{16384} + 0.5 \right), \\
u &= \max(0, \min(4095, res6)).
\end{aligned}$$

где  $p0, p1, p2, p3, p4, p5$  — целочисленные функции коэффициентов и температуры, содержащиеся в интервалах:

$$\begin{aligned}
p0(T, INFBIT, SBIT) & \quad [0, 1], \\
p1(T, INFBIT, SBIT) & \quad [0, 1], \\
p2(T, INFBIT, SBIT, K5BIT, K4BIT) & \quad [0, 2], \\
p3(T, INFBIT, SBIT, K5BIT, K4BIT, K3BIT) & \quad [0, 1], \\
p4(T, INFBIT, SBIT, K5BIT, K4BIT, K3BIT, K2BIT) & \quad [0, 1], \\
p5(T, INFBIT, SBIT, K5BIT, K4BIT, K3BIT, K2BIT, K1BIT) & \quad [0, 2].
\end{aligned}$$

Если в модели спецификации приближённо заменить разрывную функцию  $\text{floor}(z + 0.5)$  на непрерывную  $z$  и масштабировать некоторые переменные, то получим идеализированную модель.

Модель идеализированная:

$$INF = 1535 + 8 \cdot INFBIT,$$

$$SCALE = \frac{SBIT + 16}{32},$$

$$K0 = 1032,$$

$$K1 = -195 - 2 \cdot K1BIT,$$

$$K2 = \frac{20 + K2BIT}{2},$$

$$K3 = \frac{8 + K3BIT}{2},$$

$$K4 = \frac{K4BIT - 25}{8},$$

$$K5 = \frac{K5BIT}{16},$$

$$xs = (T - INF) \cdot SCALE,$$

$$x = xs \cdot 2^{-8},$$

$$u = \max(0, \min(4095, K0 + K1 \cdot x + K2 \cdot x^2 + K3 \cdot x^3 + K4 \cdot x^4 + K5 \cdot x^5)).$$

## 1.2 Частотно-тепловая модель

Описанная выше цифровая часть одинакова для всех экземпляров кварцевого генератора, тем не менее конкретные экземпляры могут отличаться в следующем:

- различия собственной частоты резонатора и её температурной зависимости;
- различия переходной характеристики термодатчика;
- различия вольт-фарадной характеристики варикапа;

- различия собственной ёмкости резонатора, то есть чувствительность частоты к ёмкости варикапа;
- различия тепловых характеристик микросборки, приводящих к неодинаковой температуре резонатора и термодатчика.

Тем самым мы можем подобрать цифровое значение  $u$  для заданной цифровой температуры  $T$ , заставляющее генератор выдавать требуемую частоту  $f_0$ .

Поэтому проводят обмер каждого экземпляра генератора. Партию генераторов помещают в термокамеру и меняют их температуру. В процессе измерения мы можем отключить цифровой вычислитель и вместо этого устанавливать значение  $u$  через отладочный интерфейс микросхемы. Во время измерений читают цифровые показания термодатчика  $T$  и подбирают числовое значение  $u$  на входе ЦАП при котором генератор выдаёт требуемую частоту  $f_0$ . Кроме того, измеряют изменение частоты при небольшом изменении  $u$  для оценки чувствительности частоты к  $u$ .

Температура термодатчика, находящегося в микросхеме, может отличаться от температуры резонатора по следующим причинам:

- из-за нагрева микросхемы;
- при меняющейся внешней температуре, из-за разных тепловых сопротивлений от внешней среды до резонатора и до термодатчика в микросхеме.

Техническое задание на генератор предполагает ограниченную скорость изменения внешней температуры. Тем не менее, если мы обмеряем микросхему при изменении температуры от холодной к горячей и от горячей к

холодной, то получаем своего рода гистерезис - разные значения  $u$ , при котором генератор выдаёт требуемую частоту  $f_0$  на прямом и обратном проходе. Этот гистерезис имеет место даже и при очень медленной скорости изменения температуры.

В результате получаем частотно-тепловую модель экземпляра генератора — две функции  $f_{\text{inf}}$  и  $f_{\text{sup}}$ , оценивающих частоту:

$$f_{\text{inf}}(T, u, c) \leq f \leq f_{\text{sup}}(T, u, c). \quad (1.1)$$

$f_{\text{inf}}$  и  $f_{\text{sup}}$  — кусочно-линейные или кусочно-полиномиальные функции, являющиеся интерполяцией проведённых измерений.  $f_{\text{inf}}$  и  $f_{\text{sup}}$  монотонны по  $u$  и антимонотонны по  $c$ .

*Информация, которая была приведена в данной главе является достаточно для составления общего представления о математических моделях рассматриваемых в данной работе. В ходе повествования будут предоставлены дополнительные сведения, которые могут понадобиться для уточнения постановки задачи, понимания действий и результатов, представленных далее.*

## Глава 2

# Постановка задачи и анализ методов её решения

### 2.1 Задача подбора коэффициентов

Техническое задание задаёт относительную стабильность частоты  $ppm$ , которое определяется соотношением, выполненным во всём температурном диапазоне:

$$f_0(1 - ppm) \leq f \leq f_0(1 + ppm), \quad (2.1)$$

где  $f$  — выдаваемая частота,  $f_0$  — требуемая частота. Критерием выбора того или иного набора коэффициентов  $K$  является значение величины  $ppm$  — выбирается результат с наименьшим значением. Кроме того, мы гарантируем выполнение (2.1), если оценка  $[f_{\text{inf}}, f_{\text{sup}}] \subseteq [f_0 \cdot (1 - ppm), f_0 \cdot (1 + ppm)]$ . Тогда

$$f_0 \cdot (1 - ppm) \leq f_{\text{inf}}(T_i, u_i) \leq f \leq f_{\text{sup}}(T_i, u_i) \leq f_0 \cdot (1 + ppm). \quad (2.2)$$

Из (2.2) вытекает, что в случае тепловой и цифровых моделей минималь-

ное значение  $ppm$  вычисляется следующим образом соответственно:

$$ppm_i(T_i, K) = \frac{\max(f_0 - f_{\inf}(T_i, u_i), f_{\sup}(T_i, u_i) - f_0)}{f_0}, \quad (2.3)$$

$$ppm_i(T_i, K) = \frac{|u - u_i|}{f_0}. \quad (2.4)$$

Сформулируем стоящую перед нами задачу: необходимо для каждого заданного показания цифровой температуры  $T$  найти такой вектор коэффициентов  $K$ , что значение относительной стабильности частоты  $ppm$  будет минимальным на множестве возможных значений. Согласно формулам (2.3)–(2.4) поставленная задача, в случае тепловой и цифровых моделей, сводится к задаче поиска глобального минимума целевых функций  $\Delta f = \max(|f_0 - f_{\inf}|, |f_0 - f_{\sup}|)$  и  $\Delta u = |u - u_i|$ , соответственно.

Важно отметить, что разрабатываемый программный комплекс, использующий средства интервального анализа, в том числе, служит своего рода средством для проверки результатов глобальной оптимизации, полученных с помощью предоставленного эвристического алгоритма. Учитывая что эвристический алгоритм проводит оптимизацию в терминах каждой из представленных моделей, стоит начать с рассмотрения постановки в терминах цифровых моделей и провести процедуру поиска глобального оптимума с помощью простейшего интервального алгоритма глобальной оптимизации. После, стоит проанализировать полученные результаты и выбрать методы, применение которых сократит время расчётов и позволит получить более точную оценку глобального оптимума.

Входные данные представляются в виде текстового файла, где каждая микросборка характеризуется следующим набором значений:

- Номер рассматриваемой микросборки,

- Значение температуры  $T$  на датчике,
- Значение выдаваемое цифровым вычислителем  $u$ ,
- Значение частоты  $f_0$ .

Работая с постановкой задачи в терминах цифровых моделей, предоставленное значение  $f$  нам не требуется непосредственно для построение приближённого решения задачи глобальной оптимизации, в данном случае она представляет из себя значение ассоциируемое с поданным на вход «эталонным»  $u$ .

## 2.2 Описание алгоритма

В этой работе для решения поставленной задачи будут использоваться известные инструменты интервального анализа. Опишем преимущества, которые даёт интервальный подход. Оперирование интервалами вместо точечных величин даёт преимущество при получении внешней оценки области значений функции  $\mathbf{F}(\mathbf{X})$ , являющейся интервальным расширением самой  $F(x)$ . При стремлении бруса области определения  $\mathbf{X}$  к нулю, погрешность такой оценки уменьшается [1]. Так как для сложных функций, имеющих множественные вхождения переменных или вхождение других функций, оценка, полученная с помощью естественного интервального расширения, зачастую является грубой, то возникает необходимость в уменьшении её погрешности. Это приводит к идее использования интервального расширения отличного от естественного, имеющего меньшую погрешность оценки, либо же к применению процедуры, позволяющей сужать область определения. Средства, помогающие добиться желаемых результатов, разработаны и уже существуют в рамках интервального анализа [1], [2].

Говоря о процедуре сужения области значений и вычисления соответствующих интервальных оценок, стоит сказать, что их основоположником является Р. Е. Мур [3]. В настоящее время предложенная им идея носит скорее теоретическое значение из-за своей высокой трудоёмкости, с подробным описанием его идей можно ознакомиться так же в [1].

Перед представлением псевдокода алгоритма стоит отметить, что весь программный код был написан на языке Java. Данный выбор обусловлен несколькими факторами: наличием соответствующей библиотеки, реализующей интервальные вычисления, гибкостью предоставляемых ею инструментов, возможностью запуска написанной программы независимо от операционной системы и архитектуры.

Данная библиотека является российской разработкой и реализована в соответствии со стандартом «IEEE P1788» [5] группой математиков-энтузиастов, занимающихся интервальными вычислениями. Главными разработчиками на данный момент являются Жилин С.И. и Надёжин Д.Ю., с полным же списком участников можно ознакомиться на сайте самого проекта [4] в подразделе «Интервалы в Java – проект JInterval». Принципы работы библиотеки описаны в статье [6].

Среди ключевых особенностей алгоритма 1 можно выделить способ получения интервальной оценки целевой функции  $\Delta u$  — применение естественного интервального расширения, имеющего первый порядок сходимости по ширине области значений [2]. Основным недостатком такого подхода является сокращающееся изменение нормы брусков при увеличении размерности задачи, что отрицательно сказывается на времени проведения расчётов.

Представленный далее алгоритм представляет из себя модификацию алгоритма `GlobalOpt` подробное описание которого можно найти в [1].

**Вход:**

Брус  $\mathbf{K} \in \mathbb{R}^n$ ;

Интервальная оценивающая функция  $\Delta \mathbf{u}$  для целевой функции  $\Delta u$ ;

Точность  $\varepsilon > 0$ .

**Выход:**

Оценка снизу глобального минимума  $u(\mathbf{K}, T)$  с точностью  $\varepsilon$ , где  $\mathbf{K} \in X$ .

---

$\mathbf{Y} \leftarrow \mathbf{K}$ ;

вычислим интервальную оценку  $\Delta \mathbf{u}(\mathbf{Y})$ ;

присваиваем  $y \leftarrow \Delta \mathbf{u}(\mathbf{Y})$ ;

инициализируем *рабочий список*  $\mathcal{L} \leftarrow \{(\mathbf{Y}, y)\}$ ;

присваиваем  $Z_1 \leftarrow \infty$ ;

присваиваем  $Z_2 \leftarrow y$ ;

**пока**  $(Z_1 - Z_2 \geq \varepsilon)$  **выполнять**

    присваиваем  $Z_1 \leftarrow y$ ;

    выбираем максимальную по ширине компоненту  $w$  бруса  $\mathbf{X}$ ;

    дробим брус  $\mathbf{Y}$  на потомки  $\mathbf{Y}'$ ,  $\mathbf{Y}''$ , где

$$\mathbf{Y}' = (\mathbf{Y}_1, \dots, [\underline{\mathbf{Y}}_w, \text{mid } \mathbf{Y}_w], \dots, \mathbf{Y}_n),$$

$$\mathbf{Y}'' = (\mathbf{Y}_1, \dots, [\text{mid } \mathbf{Y}_w, \overline{\mathbf{Y}}_w], \dots, \mathbf{Y}_n);$$

    вычисляем  $\mathbf{u}(\mathbf{Y}')$ ,  $\mathbf{u}(\mathbf{Y}'')$ ;

    присваиваем  $y' \leftarrow \underline{\mathbf{u}(\mathbf{Y}')}$ ,  $y'' \leftarrow \underline{\mathbf{u}(\mathbf{Y}'')}$ ;

    удаляем из списка  $\mathcal{L}$  ведущую запись  $(\mathbf{Y}, y)$ ;

    помещаем записи  $(\mathbf{Y}', y')$ ,  $(\mathbf{Y}'', y'')$  в список  $\mathcal{L}$  в порядке  
        возрастания второго поля;

    обозначаем новую ведущую пару как  $(\mathbf{Y}, y)$ ;

    присваиваем  $Z_2 \leftarrow y$ ;

**конец**

возвращаем интервал  $[y, y + \varepsilon]$ , как содержащий глобальный оптимум.

**Алгоритм 1:** Модифицированный алгоритм GlobOpt.

## 2.3 Результаты

Приведём результаты поиска глобального оптимума для цифровых моделей, полученных с помощью алгоритма 1 и сравним их с результатами работы эвристического алгоритма. Для удобства приведём результаты лишь для двух микросборок при значении  $\varepsilon = 10^{-5}$ :

Модель	$\max p_{prt}$	$K$
Спецификации	1.832883971279708E-6	(16, 18, 136, 8, 29, 13, 1)
Производственная	1.832883971279708E-6	(16, 18, 136, 8, 29, 13, 1)
Идеализированная	1.7987219728543962E-6	(16, 18, 138, 6, 31, 14, 0)

Таблица 2.1: Интервальный алгоритм. Микросборка №4

Модель	$\max p_{prt}$	$K$
Спецификации	1.832883971279708E-6	(16, 24, 102, 0, 16, 19, 0)
Производственная	1.832883971279708E-6	
Идеализированная	1.844920794349421E-6	

Таблица 2.2: Эвристический алгоритм. Микросборка №4

Модель	$\max p_{prt}$	$K$
Спецификации	2.1661372266261925E-6	(15, 19, 89, 0, 16, 4, 6)
Производственная	2.1661372266261925E-6	
Идеализированная	2.188766184262254E-6	

Таблица 2.3: Интервальный алгоритм. Микросборка №8

Модель	$\max p_{prt}$	$K$
Спецификации	2.1661372266261925E-6	(15, 19, 89, 0, 16, 4, 6)
Производственная	2.1661372266261925E-6	
Идеализированная	2.188766184262254E-6	

Таблица 2.4: Эвристический алгоритм. Микросборка №8

Как видно из приведённых данных, для микросборки №4 максимально значение  $p_{prt}$  для эвристического и интервального алгоритмов отличается

лишь в случае идеализированной модели, для микросборки №8 результаты абсолютно идентичны, это верно и при проведении расчётов с меньшим эпсилон:  $\varepsilon = 10^{-8}$ ,  $\varepsilon = 10^{-15}$ . Из представленных данных можно сделать вывод, что в рамках цифровых моделей, результаты полученные интервальным и эвристическим алгоритмом, преимущественно, отличаются лишь для идеализированной модели микросборки, которая не применяется при производстве устройств. Таким образом, хоть с помощью интервального алгоритма глобальной оптимизации и не был результат, который давал бы меньшую оценку глобального минимума, описанный интервальный алгоритм глобальной оптимизации пригоден для использования как простейшее средство верификации: с его помощью мы можем найти интервал в котором гарантированно лежит глобальный минимум, и удостовериться в результатах работы эвристического алгоритма.

Т.к. в случае рассмотрения тепловой модели диапазон цифровых температур  $T$  и соответствующих им значений, поданных на вход существенно шире, проведение расчётов с помощью простейшего интервального алгоритма требует значительных временных затрат — примерно около 1 часа 20 минут на поиск оптимального значения  $ppm$  для каждого экземпляра микросборки. Поэтому на данный момент приведём результаты расчётов лишь для одного экземпляра.

Модель	$\max ppm$	$K$
Спецификации	0.2398045504956933E-6	(45, 18, 47, 71, 17, 31, 12)
Производственная	0.2398045504956933E-6	
Идеализированная	0.24007377096629873E-6	

Таблица 2.5: Интервальный алгоритм. Микросборка №8

Модель	$\max p_{pt}$	$K$
Спецификации	0.2398045504956933E-6	(45, 18, 47, 71, 17, 31, 12)
Производственная	0.2398045504956933E-6	
Идеализированная	0.24007377096629873E-6	

Таблица 2.6: Эвристический алгоритм. Микросборка №8

Выводы, которые можно сделать по представленным результатам не будут отличаться от сделанных ранее для цифровых моделей, но стоит отметить, что в данном случае заметен главный недостаток применения простейшей вариации интервального алгоритма глобальной оптимизации.

## Заключение и дальнейшие планы

Подведём итоги. В ходе данной работы было рассмотрено решение задачи подбора коэффициентов методами интервального анализа, начато построение программного комплекса для решения поставленной задачи, были получены первые результаты показывающие уместность применения интервальных техник. Поставленная задача была решена в терминах цифровых моделей — написанная программа для заданной микросборки находит глобальный оптимум максимального значения  $p_{pt}$  и подтверждает результаты предоставленные эвристическим алгоритмом. В случае рассмотрения тепловой модели проявляются известные недостатки применения простейшего алгоритма `GlobalOpt` — первый порядок сходимости по ширине области значений оцениваемой, с помощью естественного интервального расширения, функции. Поэтому, в рамках этой модели, задача не была решена полностью — требуется существенное увеличение скорости работы алгоритма для анализа всех данных, поданных на вход.

В виду сказанного к дальнейшим планам можно отнести:

- реализацию и подробное описание интервальной арифметики наклонов;
- применение техник интервального анализа, позволяющих сузить рассматриваемую область определения;
- применение техник для верификации существования решения на заданном интервале, как дополнительное средство проверки.

Стоит отметить, что весь код проекта находится в открытом доступе и может быть найден по адресу: [github.com/nadezhin/thermocompensation](https://github.com/nadezhin/thermocompensation).

# Литература

- [1] ШАРЫЙ С. П. *Конечномерный интервальный анализ*.  
– Электронная книга, доступная по адресу  
<http://www.nsc.ru/interval/Library/InteBooks/SharyBook.pdf>
- [2] ELDON HANSEN, G. WILLIAM WALSTER *Global optimization using interval analysis*. — New York: Marcel Dekker, 2004.
- [3] MOORE R.E. *Interval analysis*. — Englewood Cliffs: Prentice Hall, 1966.
- [4] *JInterval*. <http://www.nsc.ru/interval/?page=Programing>.
- [5] *IEEE P1788*.  
–<http://standards.ieee.org/findstds/standard/1788-2015.html>.
- [6] NADEZHIN D.YU., ZHILIN S.I. *JInterval Library: Principles, Development, and Perspectives*. — *Reliable Computing*, 21.01.2014.