

Содержание

- 1 Введение в предметную область
- 2 Мотивация и постановка задачи
- 3 Обзор
- 4 Система
- 5 Заключение

Мотивация и постановка задачи

Мотивация

- Крупномасштабные задачи, например, глубокое обучение, могут быть решены градиентными методами.
- Для стационарной детерминированной оптимизации быстрый градиентный метод (Нестеров, 1983) является эталоном. На практике часто применяется метод Адама.
- Онлайн обучение, мета-обучение / обучение мета-переносу – являются задачами нестационарной оптимизации.
- Нет системы для моделирования алгоритмов оптимизации для задач трекинга.

Постановка задач

- Исследовать существующие системы для моделирования алгоритмов оптимизации.
- Составить требования к системе.
- Разработать архитектуру системы.
- Реализовать классические методы стохастической оптимизации и новый метод «Рандомизированный быстрый квазиградиентный метод для задач трекинга» (RFQGT) в системе.
- Разработать прототип системы, включающий в себя возможность визуализации невязок алгоритмов.
- Сделать графический интерфейс в системе.
- Провести апробацию системы.

Системы

■ ПО

- Optimization Tool Matlab
- ADMB (AD Model Builder)
- CUTEr (Constrained and Unconstrained Testing Environment, revisited)
- TOMLAB
- GAMS (General Algebraic Modeling System)

■ Библиотеки

- GEKKO
- ALGLIB
- IMSL Numerical Libraries
- MIDACO
- IPOPT (Interior Point OPTimizer)

Сравнение систем

Система	Графич. Интерф.	Пользов. алгоритм	Модель измерений	Модель шума	Модель дрефта	Итерации и шаги
Optimization Tool Matlab	+	—	+	—	—	+
ADMB	—	+	+	+	—	+
GAMS	—	+	+	—	—	+
CUTEr	—	+	+	—	—	+
Новая система	+	+	+	+	+	+

Система

Требования к системе

- Возможность задания модели измерений.
- Возможность задания модели шума.
- Возможность задания модели дрефта.
- Возможность добавления пользовательских алгоритмов.
- Возможность задания количества шагов и количества итераций алгоритма.

Графический интерфейс системы

Model

Function:

Gradient:

Dim:

L: Mu:

Algorithms

SG FG SFGT RFQGT

Custom:

Noise

Scale:

Noise Model

Random

Custom:

Drift

Scale: Init point:

Drift Model

No drift

Linear

Nonlinear

Rand

Custom:

Iterations: Steps:

Iterations optpars: Steps optpars:

Estimation Error

Mean error norm

Step number

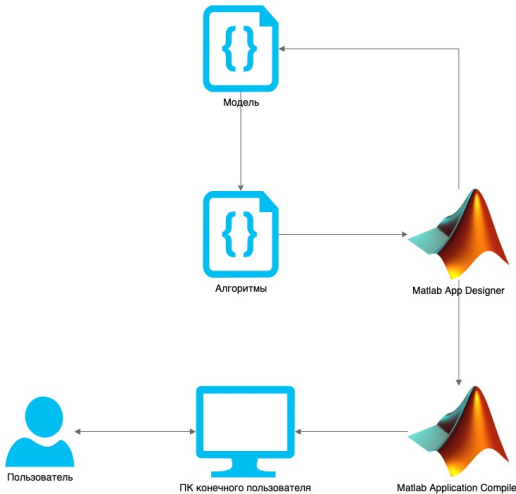
Start Save

	h	eta	alpha_x	other
SG				
FG				
SFGT				
RFQGT				

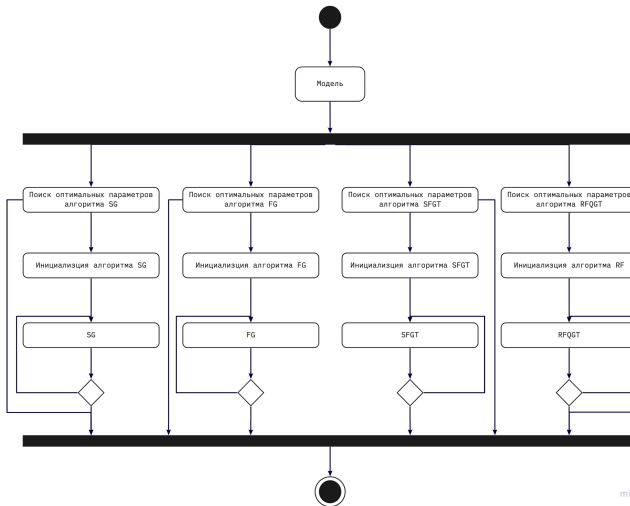
Используемые инструменты

- Matlab
- Matlab App Designer
- Matlab Application Compiler

Архитектура системы



Архитектура системы



Детали реализации

Модель в системе

- Константы алгоритма: константа Липшица, константа строгой выпуклости и т.д.
- Измерения функций
- Измерения градиента
- Точки минимума θ_n

Поиск оптимальных параметров в системе

- Шаг алгоритма $h < \frac{1}{L}$, $h = [\frac{0.2}{L}, \frac{0.4}{L}, \dots]$
- $\eta = [\frac{0.1}{\mu}, \frac{0.2}{\mu}, \dots]$
- ...

Схема работы алгоритмов в системе

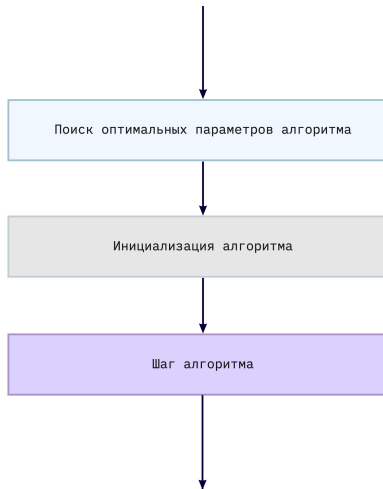


Схема работы алгоритмов в системе: RFQGT

Фиксируются значения параметров $h > 0$, $\beta > 0$, $\eta \in (0, \mu)$, $\alpha_x \in (0, 1)$

Начинаем с начальной оценки $\hat{\theta}_0 \in \mathbb{R}^q$; Выбираем $\gamma_0 > 0$, $v_0 = \hat{\theta}_0$

На n -ой итерации ($n \geq 0$):

- 1 Найти $\alpha_n \in [\alpha_x, 1)$ т. ч.: $\alpha_n < \sqrt{2H\gamma_{n+1}}$.
- 2 Присвоить $\gamma_{n+1} = (1 - \alpha_n)\gamma_n + \alpha_n(\mu - \eta)$.
- 3 Вычислить $x_n = \frac{1}{\gamma_n + \alpha_n\gamma_{n-1}} (\alpha_n\gamma_{n-1}v_{n-1} + \gamma_n\hat{\theta}_{2n})$.
- 4 Сгенерировать Δ_n по распределению Бернулли.
- 5 Найти значения 2 наблюдений $y_n^\pm = y_{2n-1} + \frac{1}{2}(1 \pm 1)(x_n \pm \beta\Delta_n)$.
- 6 Найти новую оценку $\hat{\theta}_{n+1}$:

$$\hat{\theta}_{2n-1} = \hat{\theta}_{2(n-1)}, \quad \hat{\theta}_{2(n)} = x_n - h\bar{Y}_n, \quad \bar{Y}_n = \Delta_n \frac{y_n^+ - y_n^-}{2\beta}.$$

- 7 Найти $v_{n+1} = \gamma_{n+1}^{-1} ((1 - \alpha_n)\gamma_n v_n + \alpha_n(\mu - \eta)x_n - \alpha_n\bar{Y}_n(x_n) + \alpha_n L\beta)$.

Апробация системы

- Модели измерений
 - Квадратичная функция различных размерностей
- Модели шума
 - Отсутствие шума
 - Аддитивный шум
 - Константный шум
- Модели дрефта
 - Отсутствие дрефта
 - Случайный
 - Линейный
 - Нелинейный
 - Пользовательский
- Алгоритмы
 - Стохастический градиентный спуск (SG)
 - Быстрый градиентный метод Нестерова (FG)
 - Быстрый стохастический градиентный метод для задач трекинга (SFGT)
 - Рандомизированный быстрый квазиградиентный метод для задач трекинга (RFQGT)
 - Пользовательский алгоритм

Апробация системы

Model

Function:

Gradient:

Dim:

L: Mu:

Algorithms

SG FG SFGT RFQGT

Custom: */Users/munira/diploma/algo/fg

Noise

Scale:

Noise Model

Random

Custom:

Drift

Scale: Init point:

Drift Model

No drift

Linear

Nonlinear

Rand

Custom:

Iterations: Steps:

Iterations optpars: Steps optpars:

Estimation Error

Step number

Start Save

	h	eta	alpha_x	other
SG	0.8000	NaN	NaN	
FG	0.5000	NaN	NaN	
SFGT	0.6000	0.1000	0.0580	
RFQGT	1	0.1000	0.5692	

Апробация системы

Model

Function:

Gradient:

Dim:

L: Mu:

Algorithms

SG FG SFGT RFQGT

Custom: */Users/munira/diploma/algo/fg

Noise

Scale:

Noise Model

Random

Custom:

Drift

Scale: Init point:

Drift Model

No drift

Linear

Nonlinear

Rand

Custom:

Iterations: Steps:

Iterations optpars: Steps optpars:

Estimation Error

Step number

Mean error norm

Start Save

	h	eta	alpha_x	other
SG	1	NaN	NaN	
FG	1	NaN	NaN	
SFGT	1	0.1000	0.0632	
RFQGT	1	0.9000	0.0211	

Нагрузочное тестирование систем

№	Шаги	Итер.	Шаги пар.	Итер. пар.	SG (сек.)	FG (сек.)	SFGT (сек.)	RFQGT (сек.)
1	10	100	10	1	0.03	0.04	0.24	0.26
2	10	1000	10	1	0.32	0.44	0.61	0.61
3	10	1000	10	10	0.33	0.48	1.62	1.65
4	300	10	300	1	0.08	0.19	3.43	3.49
5	300	100	300	1	0.43	0.92	4.22	4.35
6	300	100	300	10	0.75	1.52	33.19	35.70
7	300	1000	300	1	3.84	8.50	12.96	14.44

Среднее время работы алгоритмов

Заклучение

- Исследованы существующие системы для моделирования алгоритмов оптимизации.
- Составлены требования к системе.
- Разработана архитектура системы.
- Реализованы классические методы стохастической оптимизации и новый метод «Рандомизированный быстрый квазиградиентный метод для задач трекинга» (RFQGT) в системе.
- Разработан прототип системы, включающий в себя возможность визуализации невязок алгоритмов.
- Добавлен графический интерфейс в систему.
- Проведена апробация системы.
- Доклад на XXII конференции молодых ученых «Навигация и управление движением»
- Статья в Journal on Optimization Theory and Application

Приложения

Проект на GitHub