



Инструкция по использованию библиотеки qiot через ssh



Содержание

№	Раздел	Страница
I	Создание персонального SSH ключа	3
II	SSH соединение с qiopt из терминала	4
III	Соединение с qiopt из Visual Studio Code	6-7
IV	Запуск задач оптимизации с помощью qiopt	8-13

I. Создание персонального SSH ключа

1. Откройте терминал и используйте стандартный инструмент [OpenSSH](#) для генерации ключей с помощью команды:

```
$ ssh-keygen -t ed25519 -C "name_of_your_keypair"
```

Ваш публичный и приватный ключи будут сохранены по умолчанию:

Windows - C:\Users\username\.ssh

Linux - /home/username/.ssh

MacOS - /Users/username/.ssh

Для более детальной информации пройдите по ссылке [link](#)

II. SSH соединение с qiopt из терминала

1. Запустите команду в терминале :
`ssh -i <auth_private_key> username@host`
2. Запустите команду (в директории “examples”):
`./run_demo.sh`

Ожидаемый результат:

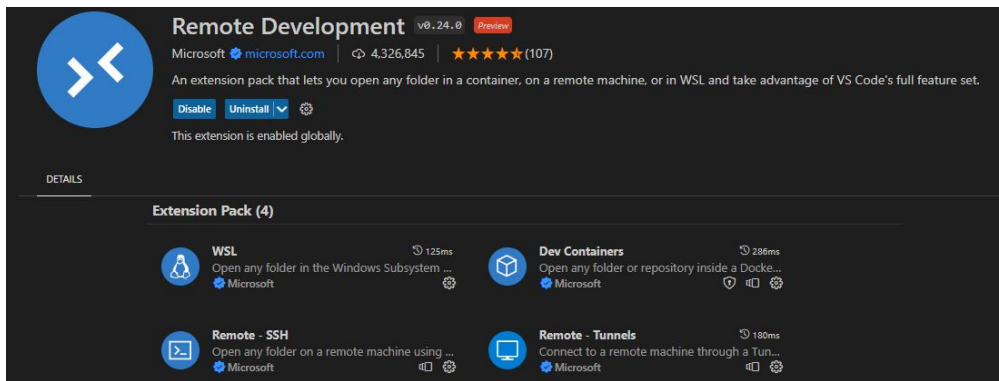
```
mushakova@laplace:~/examples$ ./run_demo.sh
Using config file: ~/examples/test_config.yaml
2023-11-16 15:42:28,597 - main - INFO - /app/qiopt/main.py - CURRENT DIR: /app/qiopt
2023-11-16 15:42:28,613 - main - INFO - /app/qiopt/main.py - Combinatorial solver SimCIMSolver loaded
2023-11-16 15:42:28,622 - main - INFO - /app/qiopt/main.py - ISING problem 'binomial_graph3' loaded
2023-11-16 15:42:28 71b510638ee6 model[6] INFO Loaded J: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_3.pkl
2023-11-16 15:42:28,970 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:42:28,970 - main - INFO - /app/qiopt/main.py - ISING problem 'binomial_graph4' loaded
2023-11-16 15:42:28 71b510638ee6 model[6] INFO Loaded J: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_4.pkl
2023-11-16 15:42:29,095 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:42:29,095 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph5' loaded
2023-11-16 15:42:29 71b510638ee6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_5.pkl
2023-11-16 15:42:29,216 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:42:29,217 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph6' loaded
2023-11-16 15:42:29 71b510638ee6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_6.pkl
2023-11-16 15:42:29,339 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:42:29,339 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph7' loaded
2023-11-16 15:42:29 71b510638ee6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_7.pkl
2023-11-16 15:42:29,436 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:42:29,437 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph8' loaded
2023-11-16 15:42:29 71b510638ee6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_8.pkl
2023-11-16 15:42:29,536 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:42:29,543 - main - INFO - /app/qiopt/main.py - Solution results saved to /home/mushakova/examples/output/solution.yaml
```

Используйте ваши данные

1. **auth key pair**
2. **Username@host**

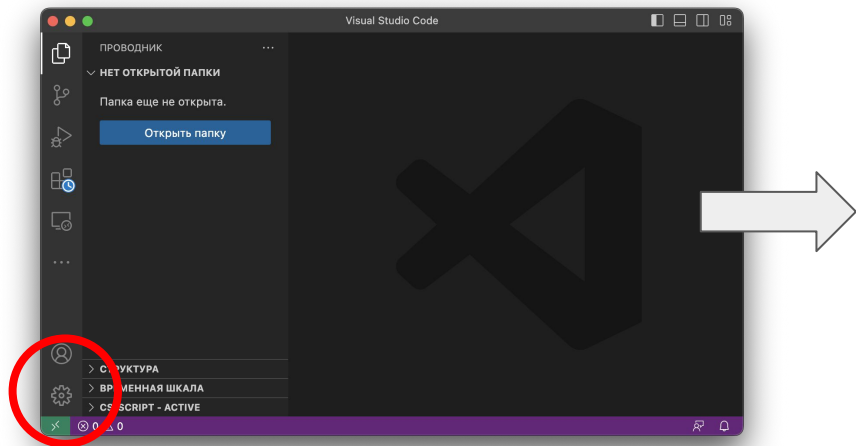
III. Запуск qiopt из Visual Studio Code

1. Установите [Visual Studio Code](#)
2. Установите расширение для Visual Studio Code: [Remote Development Extension Pack](#) (оригинальная инструкция тоже может быть использована)
3. Установите SSH соединение (смотрите стр. 6)
4. Откройте терминал и запустите команду `run_demo.sh` (`./run_demo.sh`) (см. стр. 7)

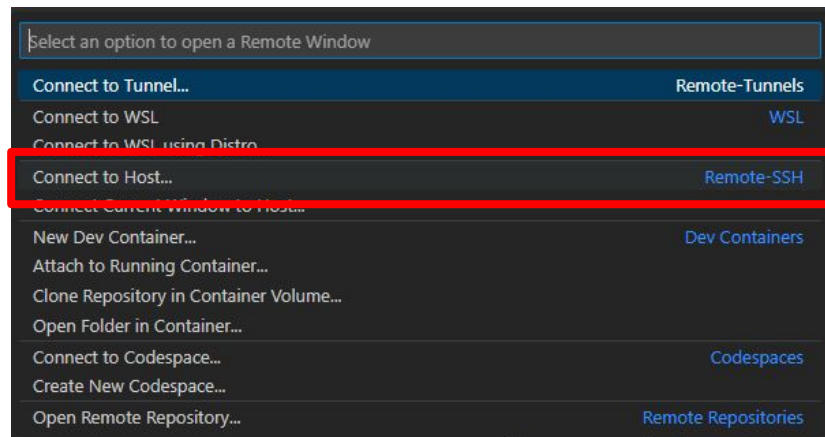


SSH соединение из Visual Studio Code

Нажмите зеленую кнопку слева внизу



Выберите Remote-SSH (см. vs code инструкцию пожалуйста), вставьте ваш hostname (формат username@host)



Запуск кода example из терминала VS Code

Откройте папку examples

```

✓ EXAMPLES [SSH: 89.208.85.236]
> binomial_graphs
> output
> output_model
display_output.py
$ run_demo.sh
! test_config.yaml
! test_model.yaml

```

Откройте Open terminal and run command: `run_demo.sh` file (или `./run_demo.sh`)

```

✓ EXAMPLES [SSH: 89.208.85.236]
> binomial_graphs
> output
> output_model
display_output.py
$ run_demo.sh
! test_config.yaml
! test_model.yaml

```

Open to the Side Ctrl+Enter

Open With...

Open in Integrated Terminal

Select for Compare

Open Timeline

Cut Ctrl+X

Copy Ctrl+C

Download...

Copy Path Shift+Alt+C

Copy Relative Path Ctrl+K Ctrl+Shift+C

Rename... F2

Delete Permanently Delete

Ожидаемый результат

```

mushakov@tplace:~/examples$ ./run_demo.sh
Using config file: ~/examples/test_config.yaml
2823-11-16 15:42:28.597 - main - INFO - /app/qlopt/main.py - CURRENT DIR: /app/qlopt
2823-11-16 15:42:28.613 - main - INFO - /app/qlopt/main.py - Combinatorial solver StarCINSolver loaded
2823-11-16 15:42:28.622 - main - INFO - /app/qlopt/main.py - ISING problem 'binomial_graph3' loaded
2823-11-16 15:42:28.710518638e6 model[6] INFO Loaded J: /home/mushakova/examples/binomial_graphs/pkl/binomial_graph_3.pkl
2823-11-16 15:42:28.978 - main - INFO - /app/qlopt/main.py - Problem completed
2823-11-16 15:42:28.978 - main - INFO - /app/qlopt/main.py - ISING problem 'binomial_graph4' loaded
2823-11-16 15:42:28.710518638e6 model[6] INFO Loaded J: /home/mushakova/examples/binomial_graphs/pkl/binomial_graph_4.pkl
2823-11-16 15:42:29.005 - main - INFO - /app/qlopt/main.py - Problem completed
2823-11-16 15:42:29.005 - main - INFO - /app/qlopt/main.py - QUBO problem 'binomial_graph5' loaded
2823-11-16 15:42:29.710518638e6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binomial_graph_5.pkl
2823-11-16 15:42:29.216 - main - INFO - /app/qlopt/main.py - Problem completed
2823-11-16 15:42:29.216 - main - INFO - /app/qlopt/main.py - QUBO problem 'binomial_graph6' loaded
2823-11-16 15:42:29.339 - main - INFO - /app/qlopt/main.py - Problem completed
2823-11-16 15:42:29.710518638e6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binomial_graph_6.pkl
2823-11-16 15:42:29.436 - main - INFO - /app/qlopt/main.py - Problem completed
2823-11-16 15:42:29.436 - main - INFO - /app/qlopt/main.py - QUBO problem 'binomial_graph7' loaded
2823-11-16 15:42:29.710518638e6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binomial_graph_7.pkl
2823-11-16 15:42:29.536 - main - INFO - /app/qlopt/main.py - Problem completed
2823-11-16 15:42:29.536 - main - INFO - /app/qlopt/main.py - QUBO problem 'binomial_graph8' loaded
2823-11-16 15:42:29.710518638e6 model[6] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binomial_graph_8.pkl
2823-11-16 15:42:29.543 - main - INFO - /app/qlopt/main.py - Problem completed
2823-11-16 15:42:29.543 - main - INFO - /app/qlopt/main.py - Solution results saved to /home/mushakova/examples/output/solution.yaml

```

IV Запуск задач оптимизации с помощью qiopt

Папка Examples содержит примеры config файлов для задач оптимизации. Используйте эту команду или подобную для запуска солвера для решения задач QUBO/Ising из формата pickle (вначале вы должны создать вначале директорию output):

```
$ qiopt run -c ~/examples/test_config.yaml -o ~/output/
```

```
mushakova@laplace:~$ qiopt run -c ~/examples/test_config.yaml -o ~/output/
2023-11-16 15:46:22,986 - main - INFO - /app/qiopt/main.py - CURRENT DIR: /app/qiopt
2023-11-16 15:46:23,003 - main - INFO - /app/qiopt/main.py - Combinatorial solver SimCIMSolver loaded
2023-11-16 15:46:23,012 - main - INFO - /app/qiopt/main.py - ISING problem 'binomial_graph3' loaded
2023-11-16 15:46:23 4be2b0a5166c model[7] INFO Loaded J: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_3.pkl
2023-11-16 15:46:23,362 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:46:23,362 - main - INFO - /app/qiopt/main.py - ISING problem 'binomial_graph4' loaded
2023-11-16 15:46:23 4be2b0a5166c model[7] INFO Loaded J: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_4.pkl
2023-11-16 15:46:23,493 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:46:23,493 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph5' loaded
2023-11-16 15:46:23 4be2b0a5166c model[7] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_5.pkl
2023-11-16 15:46:23,618 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:46:23,618 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph6' loaded
2023-11-16 15:46:23 4be2b0a5166c model[7] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_6.pkl
2023-11-16 15:46:23,738 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:46:23,738 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph7' loaded
2023-11-16 15:46:23 4be2b0a5166c model[7] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_7.pkl
2023-11-16 15:46:23,835 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:46:23,835 - main - INFO - /app/qiopt/main.py - QUBO problem 'binomial_graph8' loaded
2023-11-16 15:46:23 4be2b0a5166c model[7] INFO Loaded Q: /home/mushakova/examples/binomial_graphs/pkl/binominal_graph_8.pkl
2023-11-16 15:46:23,934 - main - INFO - /app/qiopt/main.py - Problem completed
2023-11-16 15:46:23,942 - main - INFO - /app/qiopt/main.py - Solution results saved to /home/mushakova/output/solution.yaml
```


Создание собственных задач оптимизации

Поддерживаемые задачи: ISING (SimCIM), QUBO (SimCIM)

Задача в транслятор (qiopt) загружается с помощью конфигурационного файла формата **yaml**.

Конфигурационный файл **yaml** должен быть создан вручную или лучше с помощью кода.

Запишите QUBO матрицу в *.pkl* файл в папку ``problems:data_dir`` (QUBO должна быть реализована в виде матрицы python *numpy* или *scipy.sparse.csr_matrix* matrix и сохранена в формат *pickle*).

Загрузка задачи производится из файла

```
problems:
- id: QUBO problem test from data
  type: QUBO
  data_dir: /data/qubo

objective:
  to: min
  filenames: Q.pkl
  target: -2.0

simcim:
  opts:
    num_steps: 1000
    sigma: 0.5
```

Предусмотренные ключи для конфигурационного файла `yaml`

1. `problems` – неизменяемый ключ
2. `id` - id задачи (требуется для всех типов)
3. `type` – тип задачи (QUBO, ISING)
4. `data_dir` – директория для загрузки данных (только для загрузки из файлов)
5. `objective` – целевой ключ, возможные ключи:
 - `to` – направление оптимизации (min, max)
 - `target` – целевое значение оптимизации (необязательно)
 - `filenames` – индексированный файл для целевой функции

Опции работы SimCIM

Различные опции решателя SIMCIM доступны через ключи `options` (должно быть на уровне ключа `problems`) :

1. `alpha` – параметры градиентного сглаживания, диапазон `[0..1]`
2. `sigma` – параметр пертурбации амплитуды
3. `q` – начало градиентной квантизации, диапазон `[0..1]`
4. `dt` – шаг оптимизации
5. `num_steps` – количество шагов решателя, диапазон `[0..5000]`
6. `num_runs` – количество запусков решателя, диапазон `[0..100]`
7. `seed` – установка состояния `random`

Получение решений

Запустите задачу с помощью команды (вы можете модифицировать также *run_demo.sh*):

```
$ qiopt run -c ~/config.yaml -o ~/output/
```

После успешного запуска задачи результат будет сохранен в *solution.yaml*, пример решения :

```
- id: QUBO problem test
objective:
  target: -2.0
  to: min
solution:
  energy: -2.0
  variables:
    b_0: 1.0
    b_1: 0.0
    b_2: 1.0
    b_3: 0.0
    b_4: 1.0
type: QUBO
```