

# CtoC Bike Shareにおける マッチングモデルの提案と実装と検証

---

知能モデリング研究室  
風折晃輝

## 自由なタイミング・場所・期間で利用可能な自転車共有サービス

### 利用方法

- 01 ポートの検索・ポートにて自転車の選定  
↓
- 02 スマートロックにて認証・解錠  
↓
- 03 目的地を設定し利用開始  
↓
- 04 目的地のポートに自転車を返却し利用終了

※ポート：自転車を駐輪するための場所。

**ポート設置数の偏り**

都市部と地方の格差  
地方ではポート数が少ない

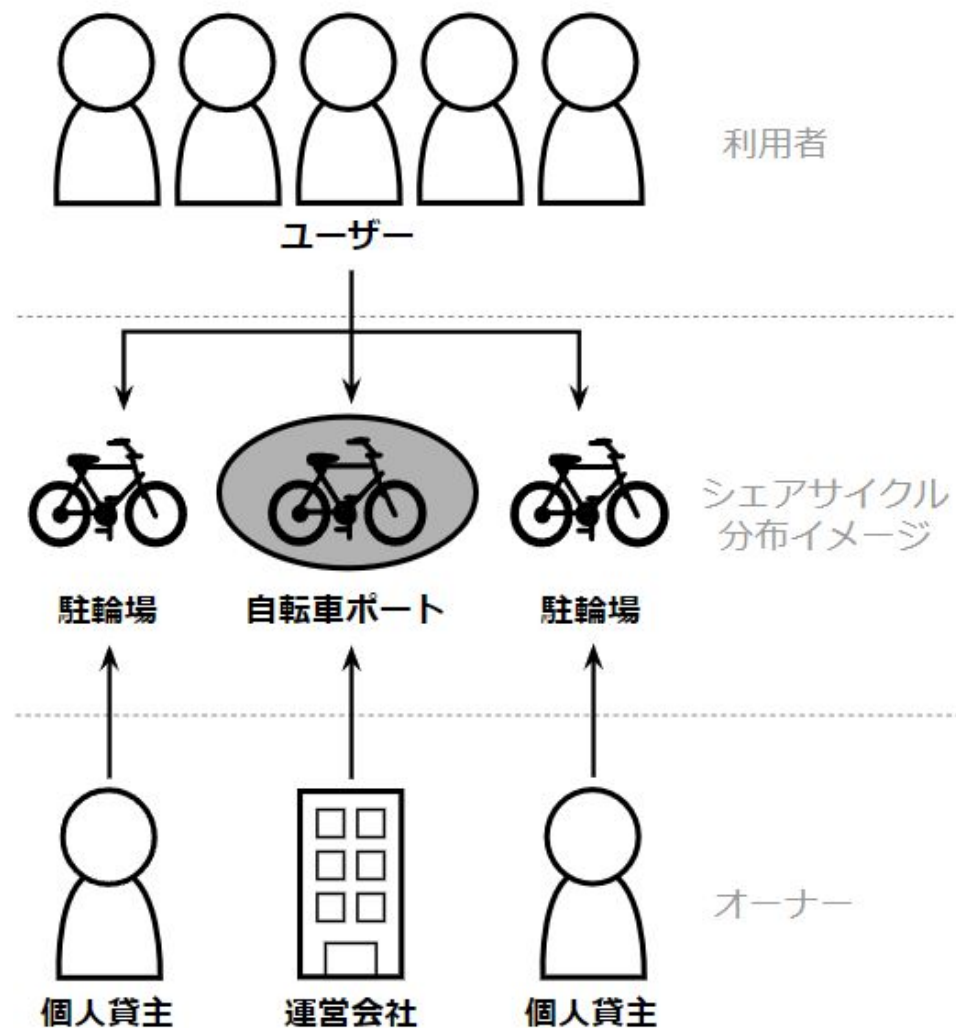
**ポートの存在による  
移動自由度の制限**

ポートが無い場所で乗り捨てできない

**自転車の調達**

サービスを提供するために  
大量の自転車を確保する必要性

## 個人間(C2C)ドックレスシェアサイクルシステムの実現

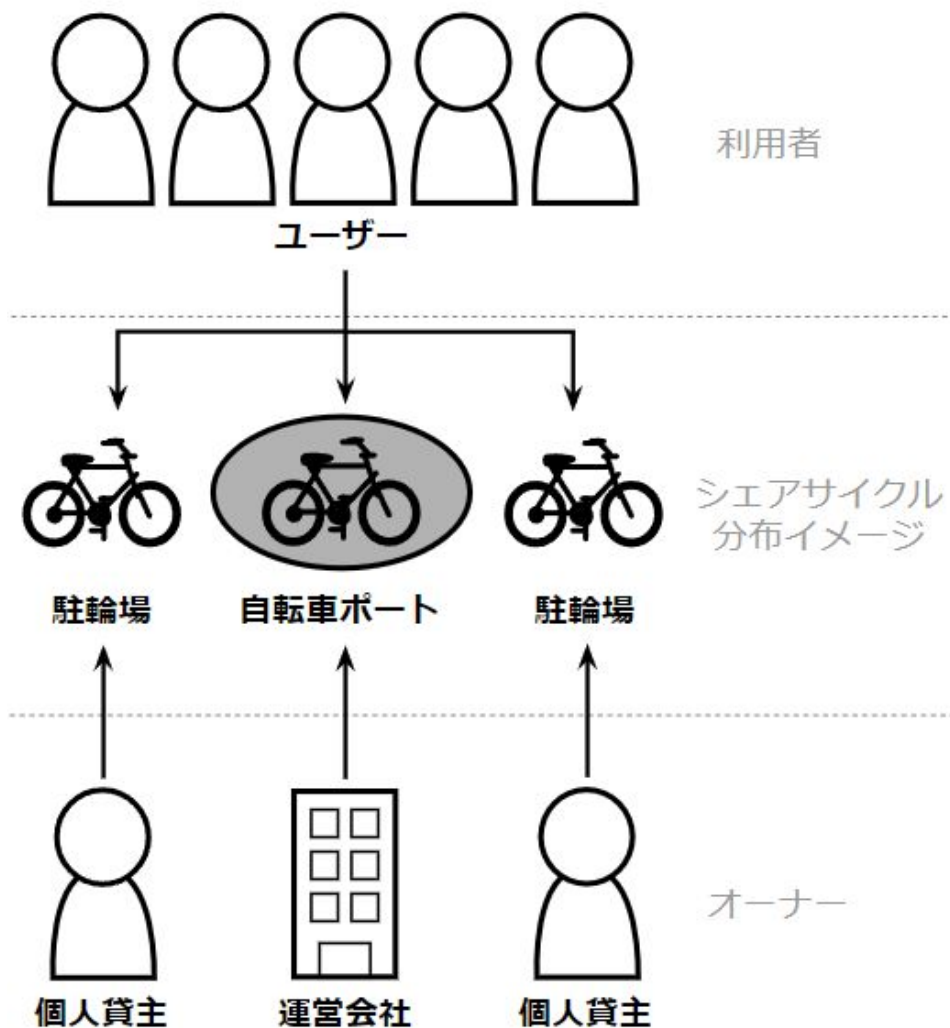


### 特徴

- 自転車を調達する必要がない
- 未使用自転車の有効活用
- ポート設置数の偏りに依存しない
- 乗り捨て可能な範囲の拡大

※ドックレス: 自転車を駐輪・固定するための特定のドックが不要。

## 個人間(C2C)ドックレスシェアサイクルシステムの実現



### 特徴

- 自転車を調達する必要がない
  - 未使用自転車の有効活用
  - ポート設置数の偏りに依存しない
  - 乗り捨て可能な範囲の拡大
- 実現するためのモデルを構築する

※ドックレス: 自転車を駐輪・固定するための特定のドックが不要。

## 前提

複数のリクエストを最適化処理するため  
1分間のユーザーリクエストをストックし、バッチ処理する.

### 対象

- 個人所有の自転車
- ドックレスで乗り捨て可能なシステム

### 対象外

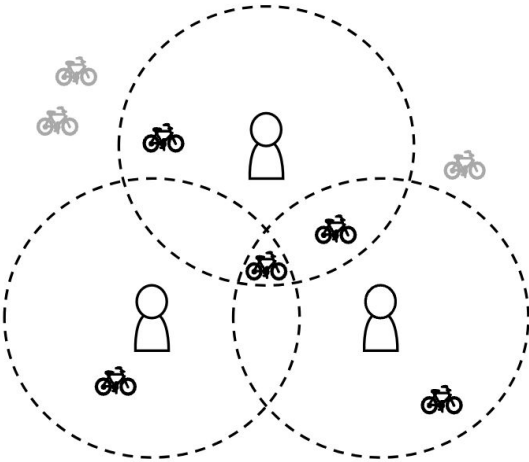
- 乗り捨てによる不法駐輪などの法的な側面  
→どこでも乗り捨てられることとする

制約条件

| 記号                      | 説明   |
|-------------------------|--|
| $R$                     | ユーザーリクエストの集合   |
| $B$                     | シェアリングされる自転車の集合                                      |
| $d_{b,r}^{\text{init}}$ | ユーザー $r(r \in R)$ と自転車 $b(b \in B)$ の割り当て前の距離        |
| $d_{b,r}$               | ユーザー $r(r \in R)$ に自転車 $b(b \in B)$ が割り当てられて移動した後の距離 |
| $x_{b,r}$               | ユーザー $r(r \in R)$ に自転車 $b(b \in B)$ が割り当てられたか否かの二値変数 |

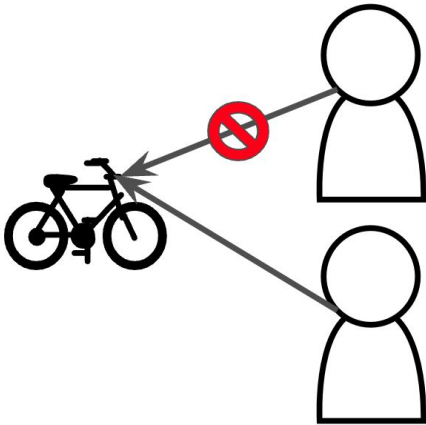
ユーザーから半径250m以内の  
自転車のみを割り当てる

$$x_{b,r} \leq \mathbb{I}(d_{b,r}^{\text{init}} \leq 250), \forall b \in B, \forall r \in R$$



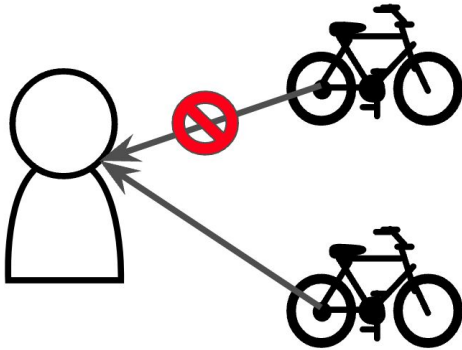
自転車に割り当てられる  
ユーザーは1人以下

$$\sum_{r \in R} x_{b,r} \leq 1, \forall b \in B$$



ユーザーに割り当てられる  
自転車は1台以下

$$\sum_{b \in B} x_{b,r} \leq 1, \forall r \in R$$



目的関数

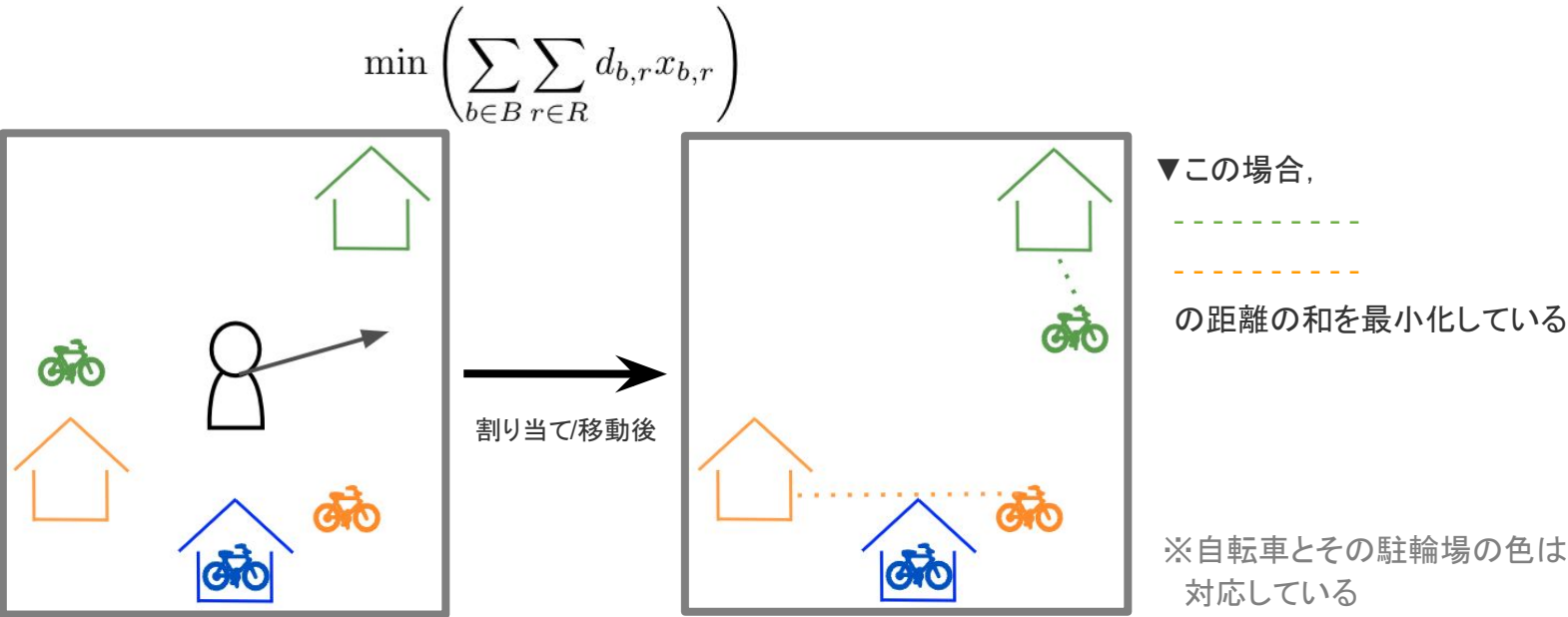
$$\min \left( \sum_{b \in B} \sum_{r \in R} d_{b,r} x_{b,r} - \alpha \sum_{b \in B} \sum_{r \in R} x_{b,r} \right)$$

| 記号               | 説明   |
|------------------|--|
| $R$              | ユーザーリクエストの集合   |
| $B$              | シェアリングされる自転車の集合                                      |
| $d_{b,r}^{init}$ | ユーザー $r(r \in R)$ と自転車 $b(b \in B)$ の割り当て前の距離        |
| $d_{b,r}$        | ユーザー $r(r \in R)$ に自転車 $b(b \in B)$ が割り当てられて移動した後の距離 |
| $x_{b,r}$        | ユーザー $r(r \in R)$ に自転車 $b(b \in B)$ が割り当てられたか否かの二値変数 |

※ $\alpha(>0)$ はトレードオフの調整を担う重み

割り当て移動後の自転車とその自転車の駐輪場までの距離を最小化

可能な限り多くのユーザーに  
自転車を割り当てる(最大化)



$$\max \left( \sum_{b \in B} \sum_{r \in R} x_{b,r} \right)$$



### 3種類の自転車割り当てモデルを別途構築して比較する

以上で説明したモデルを「**バッチ最適化割り当てモデル**」と呼ぶ

#### **ランダム割り当てモデル**

リクエストに対してランダムに自転車を割り当てる

#### **最近傍割り当てモデル**

リクエストに対してユーザーに最も近い自転車を割り当てる

#### **逐次最適化割り当てモデル**

1リクエストに対して数理最適化を適用する

### 自転車の再配置コストを比較評価指標とする

目的関数の一部である「割り当て移動後の自転車とその自転車の駐輪場までの距離」の総和を再配置コストと定義する

ニューヨーク市のタクシーデータを利用し、構築した自転車割り当てモデルを検証する。

### データの概要

- 提供元: タクシー・リムジン委員会(TLC)
- 2023年1月1日のNYCのイエロータクシー利用履歴
- 合計約7.7万のリクエスト数

自転車割り当てモデルは最適化処理を提供する  
Bikeying APIを利用する。



[bikeying.co](https://bikeying.co)



m

- ある1分間にストックされたリクエスト位置と自転車の分布をプロット。

## Dispatch ユーザーのリクエストに応じた最適な自転車の割り当てを行うサービス

POST

/bikes/dispatch Dispatch the best bike to the user



Parameters

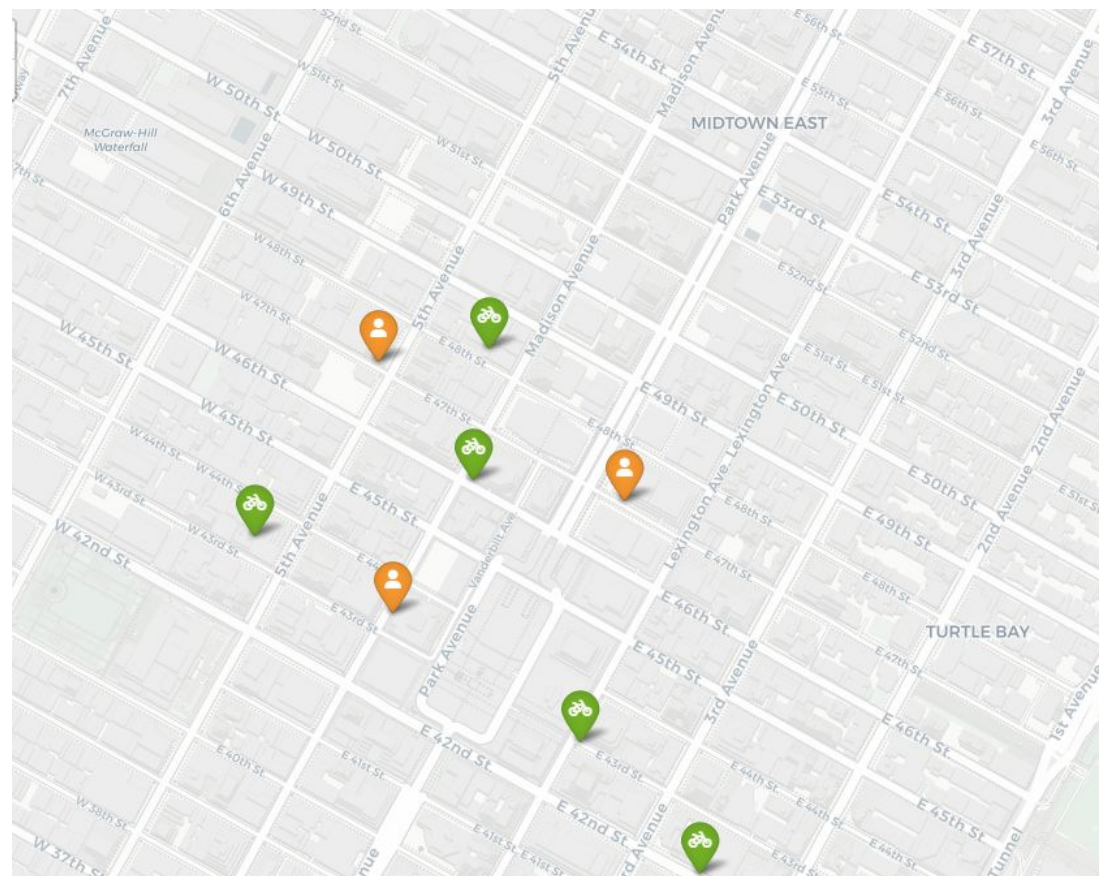
No parameters

Request body required

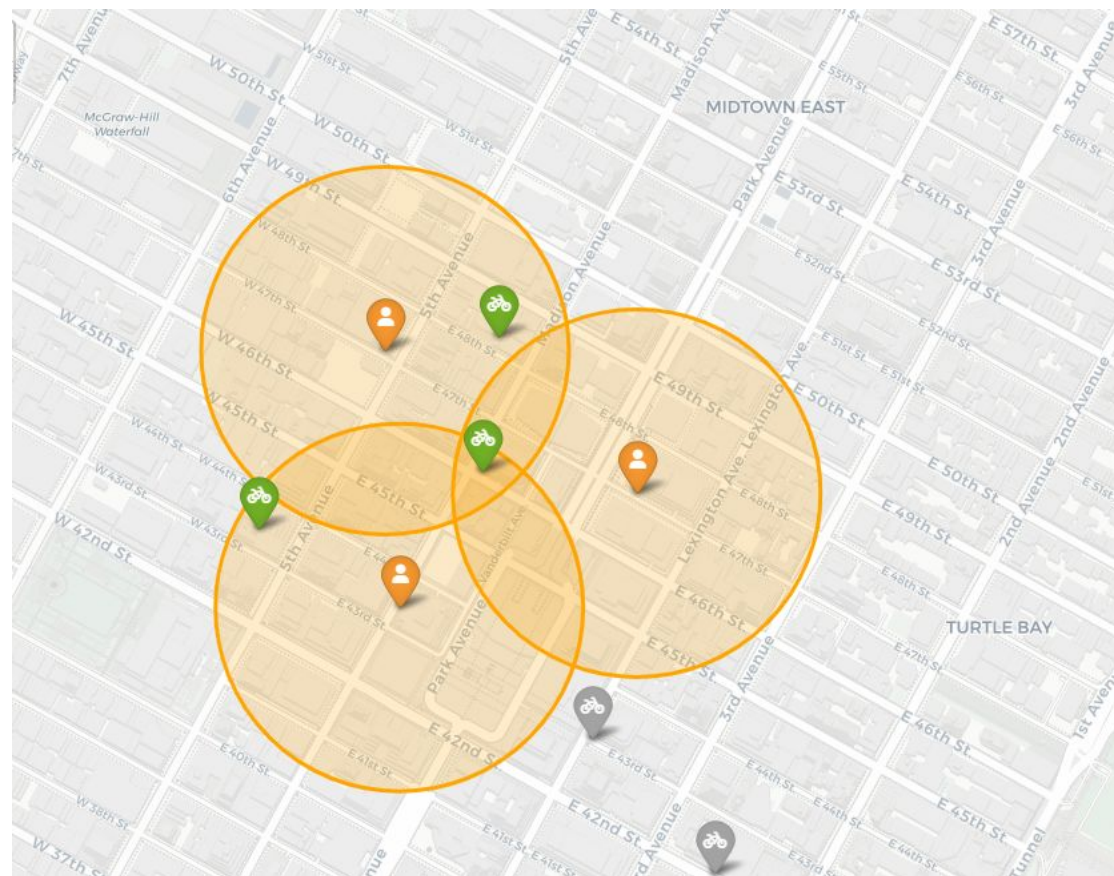
application/json

```
{
  "df_requests": [
    {
      "tpep_pickup_datetime": "2023-10-01T10:00:00Z",
      "tpep_dropoff_datetime": "2023-10-01T12:00:00Z",
      "PULocation": {
        "latitude": 90,
        "longitude": 180
      },
      "DOLocation": {
        "latitude": 90,
        "longitude": 180
      }
    },
    {
      "tpep_pickup_datetime": "2023-10-01T10:00:00Z",
      "tpep_dropoff_datetime": "2023-10-01T12:00:00Z",
      "PULocation": {
        "latitude": 90,
        "longitude": 180
      },
      "DOLocation": {
        "latitude": 90,
        "longitude": 180
      }
    }
  ]
}
```

Execute

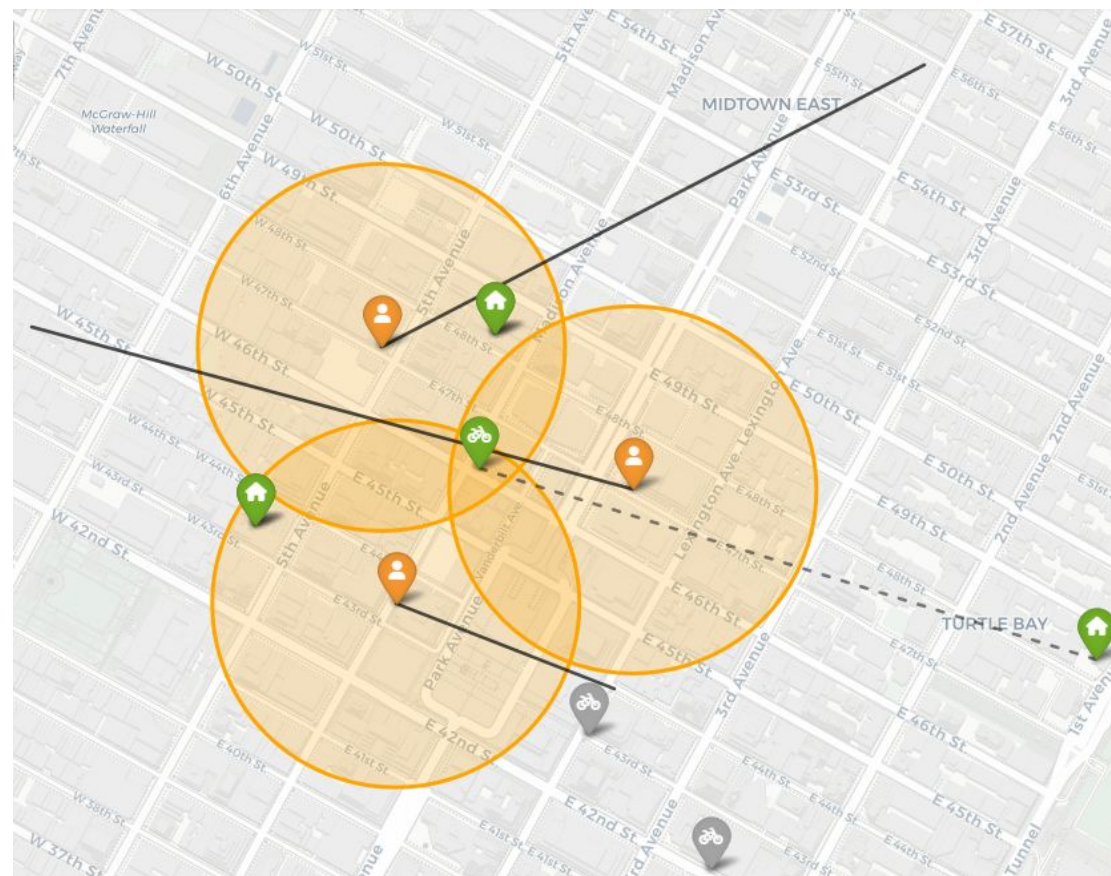


- ある1分間にストックされたリクエスト位置と自転車の分布をプロット。
- 制約条件の1つであるユーザーから半径 250mの自転車を対象とする。





- ある1分間にストックされたリクエスト位置と自転車の分布をプロット。
- 制約条件の1つであるユーザーから半径 250mの自転車を対象とする。
- **最適化処理に必要なユーザーの進行方向と自転車オーナーまでの位置関係を取得する。**




- ある1分間にストックされたリクエスト位置と自転車の分布をプロット。

Request URL

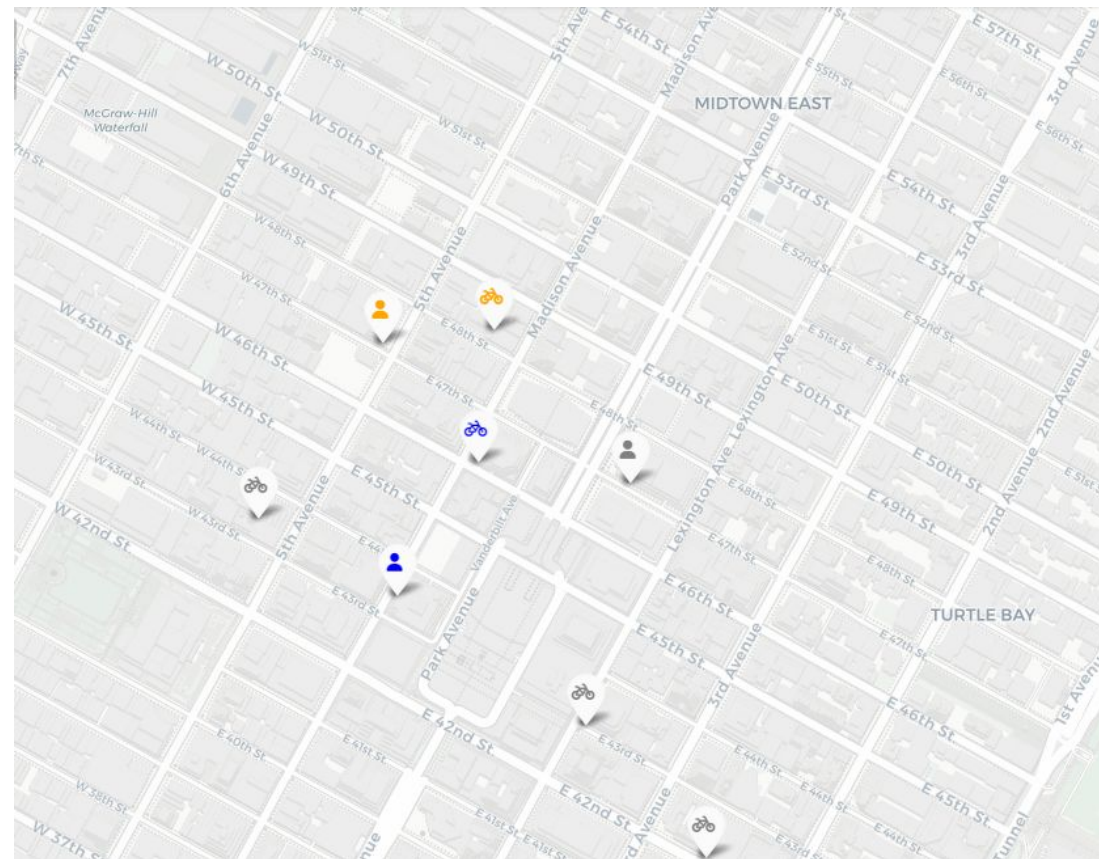
```
https://bikeying.com/bikes/dispatch
```

Server response

| Code | Details  |
|------|--|
| 200  | <div><p>Response body</p><pre>{   "results": [     {       "bike_id": "9781b731-b249-4cea-b953-9d1bf4673b9c",       "user_queue_index": "0"     },     {       "bike_id": "87543bdd-9693-4e6d-9e6e-75783eef6c2c",       "user_queue_index": "1"     }   ] }</pre><div> Download</div></div> |

- 2台の自転車の割り当てが成功する。

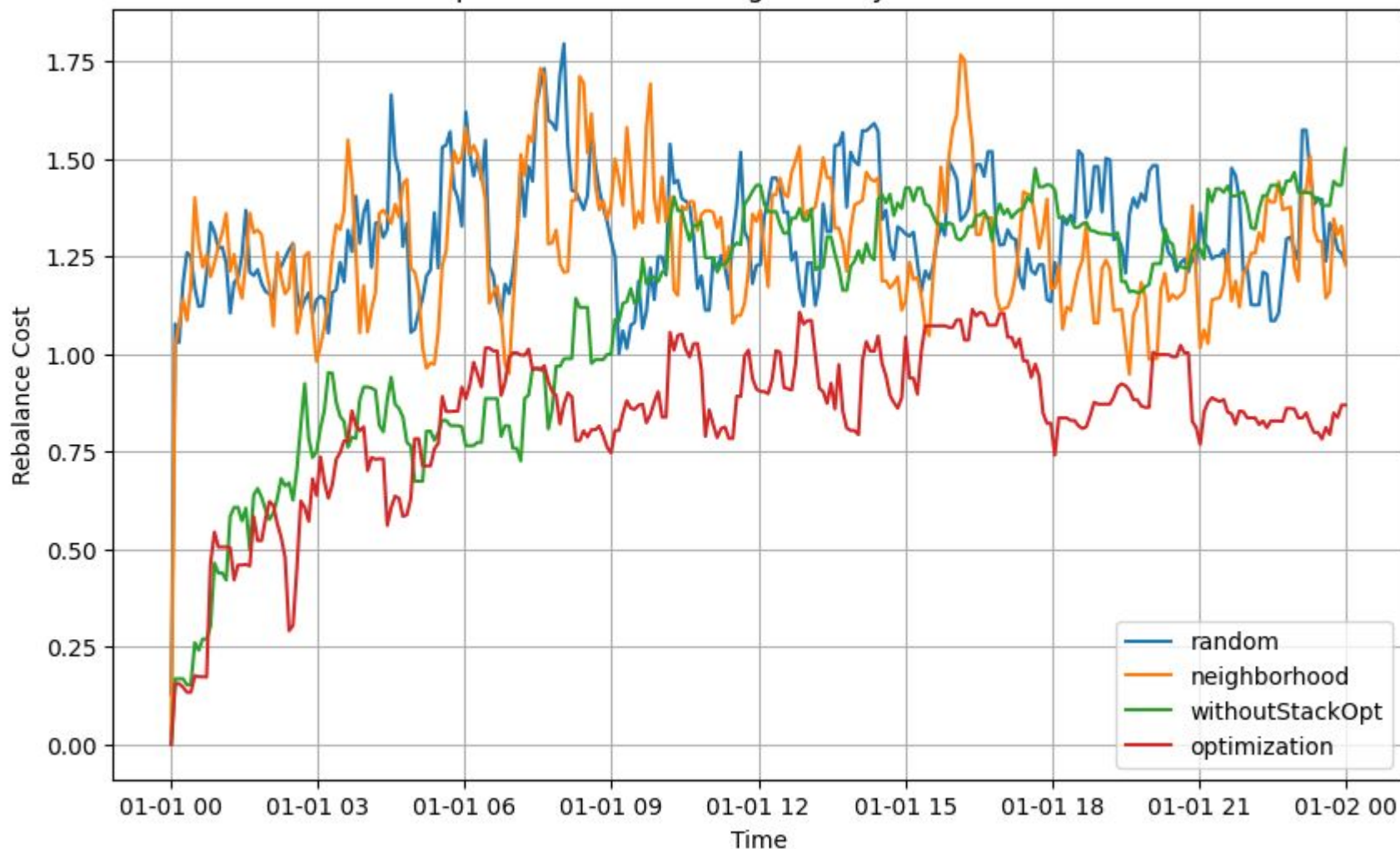
同様の処理を24時間分繰り返す。



※自転車とユーザーの色は割り当て成功後のペアに対応している

## 再配置コストはバッチ最適化割り当てモデルが最も良い結果となった

Comparison of rebalancing costs by model over time



ランダム割り当てモデル

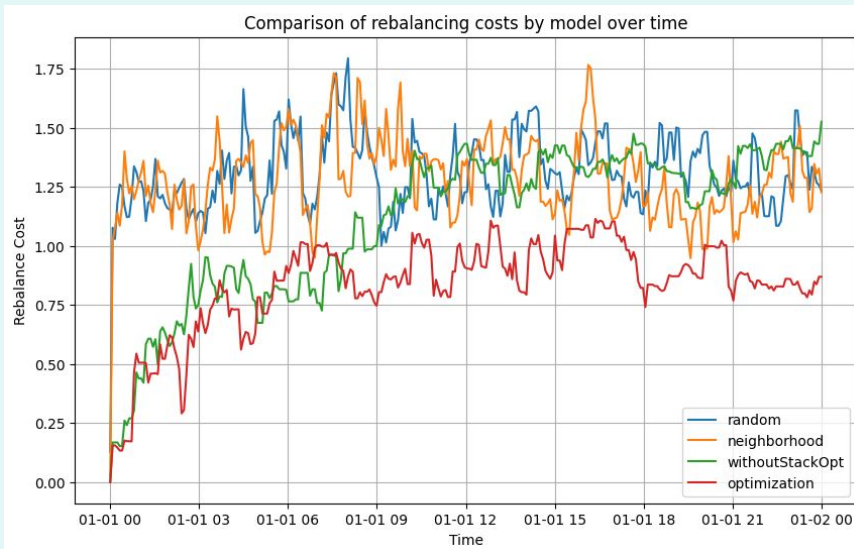
最近傍割り当てモデル

逐次最適化割り当てモデル

バッチ最適化割り当てモデル



## 考察



- 24時間継続して割り当て処理を試行した場合、提案したモデルの再配置コストの推移が有意.
- 割り当て処理開始後8時間程度までは、リクエストを1分間ストックするかどうかの影響は見られない.
- 長時間運用することになる場合はリクエストをストックするバッチ最適化割り当てモデルが適しているが、短時間の運用である場合は逐次最適化割り当てモデルが適している傾向にあると考えられる.
- ユーザーリクエストの時間帯ごとの目的地の偏りによる影響は否定できない

## 今後の課題

時間幅を拡張した検証 / リクエストをストックする時間幅の最適解検証



# C2Cドックレスシェアサイクル実現に向けた 数理最適化ベースの自転車割り当てモデルの構築

## 検証方法

1分間のユーザーリクエストをストックして最適化処理するバッチ最適化割り当てモデルに加え, 3つのベースラインとなるモデルを構築して比較.

## 結果

1分間のリクエストをストックして数理最適化処理を行ったバッチ最適化割り当てモデルが再配置コストを最小化することができた.

## 今後の課題

ユーザーリクエストの偏りによる影響は否定できないため, シミュレーションやリクエストをストックする時間幅を増減させて検証する必要がある.