Сравнение результатов алгоритма поиска равновесного распределения транспортных потоков в двухстадийной модели с ускоренным и обычным алгоритмом Синхорна

Северилов П.А., Ивченков Я.П., Ревар В.И.,
Сотников А.Д., Емельянов А.В.

1 Введение

Равновесное распределение транспортных потоков можно искать при помощи следующей двухстадийной модели: на первой стадии на вход подается матрица затрат T, а на выходе получается матрица корреспонденций $d(T)$. Во второй стадии наоборот, модели на вход подается матрица корреспонденций $d(T)$, по которой рассчитывается матрица затрат $T(d) = T(t(f(d)))$.

Под равновесием в двухстадийной транспортной модели понимается такая пара (f, d), что

$$d = d(T(f(t))), f = f(d)$$

то есть (f, d) – есть неподвижная точка описанных двух блоков моделей. Собственно, часто на практике так и ищут равновесие последовательно (друг за другом) прогоняя описанные два блока.
Одним из основных методов решения задачи восстановления матрицы корреспонденций является ее сведение к задаче энтропийно-линейного программирования и переход к двойственной задаче, которая решается с помощью алгоритма Синхорна. Кроме этого, существует ускоренный алгоритм Синхорна.
В данной работе мы сравнили работоспособность алгорима Синхорна и ускоренного алгоритма Синхорна в задаче восстановления матрицы корреспонденций. Сравнили количество итераций обычного и ускоренного Синхорна в зависимости от итерации двухстадийной модели, сравнили сходимость при работе ускоренного и обычного алгоритмов Синхорна (зазор двойственности).

2 Постановка задачи

Пусть в некотором городе имеется n районов. Общее число жителей города постоянно и равно N, при это выполняется $N \gg n^2$. Пусть $L_i \geq 0$ это число жителей, выезжающих в типичный день за рассматриваемый промежуток времени из района i, а $W_j \geq 0$ число жителей, приезжающих на работу в район j в типичный день за рассматриваемый промежуток времени.

Обозначим через $d_{ij}(t) \geq 0$ – число жителей, живущих в i-м районе и работающих в j-м в момент времени t. Определим следующее множество:

$$Q = \left\{ d_{ij} \geq 0 : \sum_{i,j=1,1}^{n,n} d_{ij}(t) = N, \sum_{j=1}^{n} d_{ij}(t) = L_i, \sum_{i=1}^{n} d_{ij}(t) = W_j, i, j = 1, ..., n \right\}$$

Таже пусть $T > 0$ - затраты на путь от дома до работы, которые определяются как временем так и расстоянием, а $\gamma > 0$ – настраиваемый параметр модели, который можно интерпретировать как цену единицы затрат на путь от работы до дома. При $N \gg 1$ распределение $p(\{d_{ij}\}_{i,j=1,1}^{n,n})$ определяется, как решение задачи энтропийно-линейного программирования (ЭЛП):

$$\min_{d_{ij} \in Q} f(d_{ij}) := \gamma \sum_{i,j=1,1}^{n,n} d_{ij} T_{ij} + \sum_{i,j=1,1}^{n,n} d_{ij} \ ln(d_{ij})$$

Перепишем задачу, как задачу минимизации с точностью до знака:

$$\min_{\lambda^l, \lambda^\omega} \varphi(\lambda^l, \lambda^\omega) := ln(E^T B(\lambda^l, \lambda^\omega) E) - < \lambda^l, l > - < \lambda^\omega, \omega >,$$

где $B_{ij}(\lambda^l, \lambda^\omega) = exp(-\gamma T_{ij} + \lambda_i^l + \lambda_j^\omega)$
Итак, в качестве задачи оптимизации предлагается решать:

$$\min_{t_e \in dom \ \sigma_e^*, e \in E} F(t) := D(t, \lambda(t), \mu(t)) + \sum_{e \in E} \sigma_e^*(t_e)$$

При этом

$$\nabla F(t) = \sum_{(i,j) \in OD} d_{ij}(\lambda(t), \mu(t)) \nabla T_{ij}(t) + (\{\tau_e^{-1}(t_e)\}_{e \in E})^T$$

В качестве критерия оценки скорости сходимости в экспериментах используется величина, оценивающая "зазор двойственности"

$$\Delta(t^k, B_k) = \max_{t \in B_k \bigcap dom \ \sigma^*} < \nabla F(t^k), t^k - t >$$

Для решения задачи можно использовать алгоритм Синхорна, который в литература чаще называют методом балансировки или методом Брэгмана. Можно использовать и ускоренные варианты этого метода. Важной особенностью этой линейки методов является высокая (линейная) скорость сходимости при немалых значениях $\gamma > 0$.

3 Алгоритм Синхорна

Двойственную задачу оптимизации, описанную в Разделе 2 предлагается решать с помощью алгоритма Синхорна и его ускоренного варианта. Опишем мотивацию для использования описанных методов. Для этого рассмотрим следующую теорему.

Теорема Синхорна утверждает, что каждую квадратную матрицу с положительными элементами можно записать в определенной стандартной форме. Если

A - матрица размера $n \times n$ со строго положительными элементами, то существуют диагональные матрицы D_1 и D_2 со строго положительными диагональными элементами такие, что $D_1 A D_2$ является дважды стохастическим. Матрицы D_1 и D_2 уникальны по модулю умножения первой матрицы на положительное число и деления второй на такое же число.

Простой итерационный метод приближения к двойной стохастической матрице состоит в том, чтобы поочередно масштабировать все строки и все столбцы матрицы A, чтобы сумма была равна 1. Ниже приведено описание алгоритма Синхорна:

Input: $x^0 = [[\lambda^l]^0, [\lambda^w]^0] = (0, \ldots, 0) \in \mathbb{R}^{2n}$ – starting point.
for $k \geq 0$ do
  if $k \bmod 2 = 0$ then
    Compute
$$\begin{aligned} [\lambda^l]^{k+1} &= [\lambda^l]^k + \ln(l) - \ln\left(B([\lambda^l]^k, [\lambda^w]^k)\mathbf{1}\right), \\ [\lambda^w]^{k+1} &= [\lambda^w]^k. \end{aligned}$$
  else
    Compute
$$\begin{aligned} [\lambda^l]^{k+1} &= [\lambda^l]^k, \\ [\lambda^w]^{k+1} &= [\lambda^w]^k + \ln(w) - \ln\left(B^T([\lambda^l]^k, [\lambda^w]^k)\mathbf{1}\right). \end{aligned}$$
  end if
end for
Output: $x^k = [[\lambda^l]^k, [\lambda^w]^k] \in \mathbb{R}^{2n}$.

---

4 Ускоренный Алгоритм Синхорна

Согласно исследовательским работам по задаче ЭЛП для восстановления матрицы корреспонденций, более быстрое решение должен гарантировать ускоренный вариант алгоритма Синхорна. В качестве основы используется традиционный адаптивный ускоренный градиентный метод. Для этой задачи предполагается, что этот вариант алгоритма ускорит сходимость. Ниже приведено описание ускоренного алгоритма Синхорна.

Input: $x^0 := [[x^l]^0, [x^w]^0] = (0, \dots, 0) \in \mathbb{R}^{2n}$ – starting point, $L_0 = 1$, $a_0 = 0$.
repeat
  Set $y^0 := [[y^l]^0, [y^w]^0] = x^0$.
  Set $v^0 := [[v^l]^0, [v^w]^0] = x^0$.
  $L_{k+1} = L_k/2$
  while True do
    Set $a_{k+1} = \frac{1}{2L_{k+1}} + \sqrt{\frac{1}{4L_{k+1}^2} + a_k^2 \frac{L_k}{L_{k+1}}}$
    Set $\tau_k = \frac{1}{a_{k+1}L_{k+1}}$
    Set $y^k = \tau_k v^k + (1-\tau_k)x^k$
    Choose $i_k = \underset{i \in \{1,2\}}{\operatorname{argmax}} \|\nabla_i \varphi(y^k)\|^2$
    if $i_k = 1$ then
      Compute
$$\begin{aligned} [x^l]^{k+1} &= [y^l]^k + \ln(l) - \ln\left(B([y^l]^k, [y^w]^k)\mathbf{1}\right), \\ [x^w]^{k+1} &= [y^w]^k. \end{aligned}$$
    else
      Compute
$$\begin{aligned} [x^l]^{k+1} &= [y^l]^k, \\ [x^w]^{k+1} &= [y^w]^k + \ln(w) - \ln\left(B^T([y^l]^k, [y^w]^k)\mathbf{1}\right). \end{aligned}$$
    end if
    Set $v^{k+1} = v^k - a_{k+1}\nabla\varphi(y^k)$
    if $\varphi(x^{k+1}) \leqslant \varphi(y^k) - \frac{\|\nabla\varphi(y^k)\|^2}{2L_{k+1}}$ then
      Set $\hat{d}^{k+1} = \frac{a_{k+1}d^k(y^k) + L_k a_k^2 \hat{d}^k}{L_{k+1}a_{k+1}^2}$
      break
    end if
    Set $L_{k+1} = 2L_{k+1}$.
  end while
until $|f(\hat{d}^{k+1}) + \varphi(x^{k+1})| \le \varepsilon_f$, $||\hat{d}^{k+1}\mathbf{1} - l||_2 \le \varepsilon_{eq}$, $||(\hat{d}^{k+1})^T\mathbf{1} - w||_2 \le \varepsilon_{eq}$
Выход: $\hat{d}^{k+1}$, x^{k+1}.

5 Вычислительный эксперимент

Эксперименты ставились с целью сравнить скорость сходимости и точность решений, определяемых с помощью обычного и ускоренного варианта алгоритма Синхорна. Ниже приведены соответствующие графики.

Эксперименты проводились на данных города Владивосток.

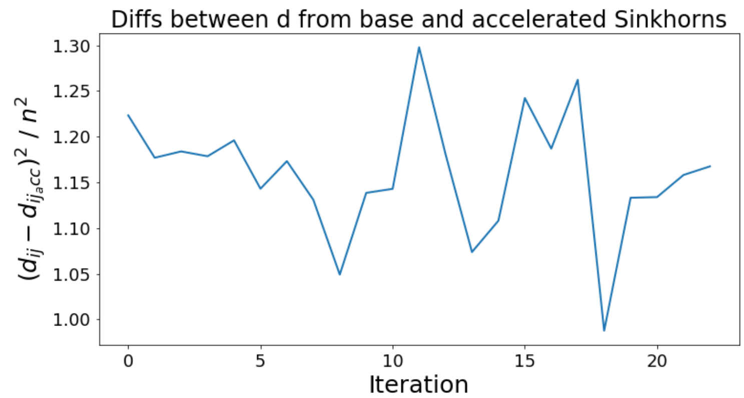

Рис. 1: Невязка между матрицами корреспонденций, восстановленными обычным и ускоренным алгоритмом Синхорна

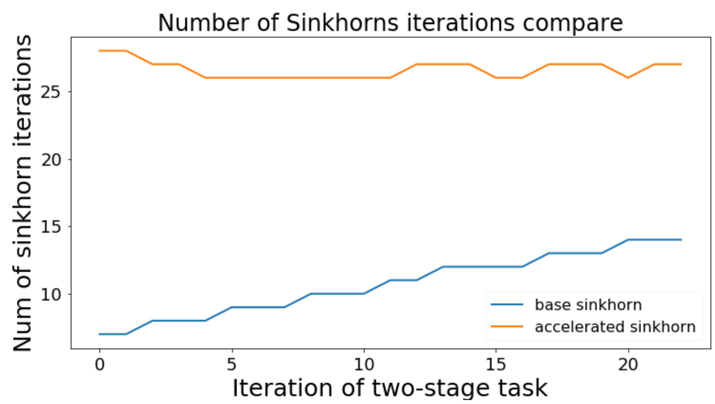

Рис. 2: Сравнение количества итераций обычного и ускоренного Синхорна в зависимости от итерации двухстадийной задачи

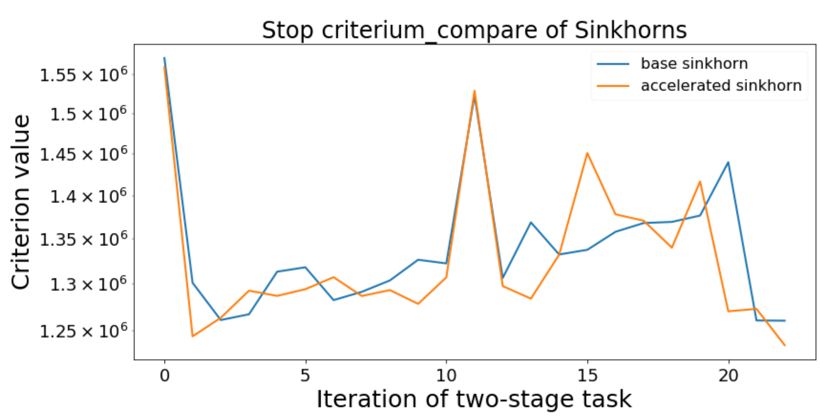

Рис. 3: Сравнение сходимости при работе ускоренного и обычного Синхорна (зазор двойственности: $\|\nabla f(t^k)\|_2 * 2\|t^0 - t^k\|_2$)

Видно, что количество итераций в ускоренном Синхорне устойчиво и практически не меняется, в то время как количество итераций в обычном Синхорне растет с каждой итерацией двухстадийной задачи. Посмотрим на маленькой подвыборке данных Владивостока тот же эксперимент:

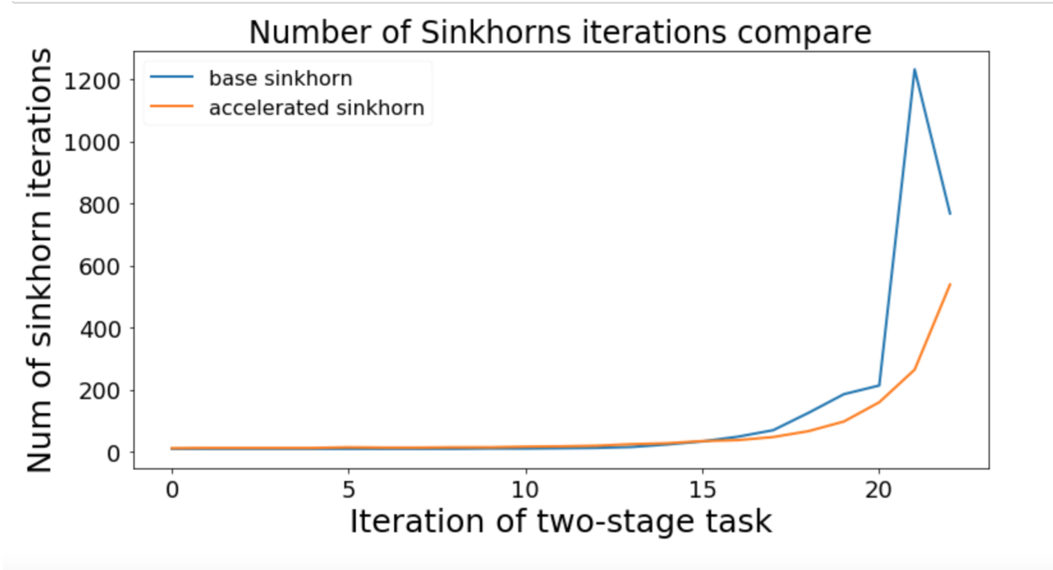

Рис. 4: Сравнение количества итераций обычного и ускоренного Синхорна в зависимости от итерации двухстадийной задачи на маленькой части данных

Видно, что с увеличением номера итерации двухстадийной модели количество итераций работы Синхорна растет быстрее, чем количество итераций ускоренной версии.