运筹学与控制论

DOI:10.11721/cqnuj20220601

一种超松弛的最优传输近似点算法*

吴 凡, 刘向阳
(河海大学 理学院, 南京 211106)

摘要:【目的】最优传输在实际应用中通常使用 Sinkhorn 算法求解熵正则化形式得到近似解,考虑 Sinkhorn 算法的效果容易受熵正则化参数影响,且难以收敛到最终精确解,提出了一种超松弛形式的近似点算法。【方法】针对原最优传输的近似点算法,为其中传输计划的迭代计算引入超松弛算子,并给出了超松弛参数计算方法。【结果】在保持算法对正则化参数具有鲁棒性及可收敛至精确解的优点的同时,所提算法能更快地收敛至精确解。【结论】数值实验表明,相较于原近似点算法,所提算法进一步提升了收敛速度,在有限的迭代步骤下能够达到更高精度,算法可更好地应用于机器学习。

关键词:最优传输;超松弛;近似点算法;熵正则化;矩阵缩放算法
中图分类号:O224;O29 **文献标志码**:A **文章编号**:1672-6693(2022)06-0020-08

最优传输是概率论、偏微分方程和凸优化的活跃交叉研究领域。在讨论两个可分度量空间的距离时,最优传输可从其中定义的任意概率测度角度给出度量方法,由此定义的距离称为 Wasserstein 距离。近些年最优传输在机器学习尤其深度学习中应用广泛,如在生成对抗网络中,WGAN 通过近似 Wasserstein 距离训练生成对抗网络(Generative adversarial network,GAN)[1],使训练稳定性和生成样本多样性均显著提升,而 OT-GAN 的损失函数则直接由 Wasserstein 距离构成[2];在自然语言处理领域中,最优传输可用于机器翻译任务[3];在医学图像分割领域,王生生等人[4]将最优传输用于特征选择。而文献[5]中介绍了更多最优传输的相关应用。

然而,最优传输的计算复杂度过高使得应用一直受限。最优传输问题可采用线性规划方法求解精确解[6],但计算复杂度为 $O(n^3)$。近些年来,Lee 等人[7]将计算复杂度改进至 $O(n^{5/2})$,但计算时间仍难以接受,因此在实际应用中很少采用此类算法也即线性规划方法进行最优传输问题的求解。第二类算法是最优传输设定下的标准优化算法,包括加速梯度下降法[8]和拟牛顿法[9]等。虽然该类算法十分有效,但在近几年才开始兴起,理论分析仍然有待发展。最后一类且最为普遍的算法涉及到通过传输计划对目标函数进行正则化,称为熵正则化方法,由 Cuturi[10]提出。熵正则化后的目标函数可通过 Sinkhorn 矩阵缩放算法[11]进行求解。之后,国内外学者在此基础上提出了多种算法。如:Altschuler[12]提出了 Greenkhorn 算法,它是 Sinkhorn 算法的贪婪形式;Alexis 等人[13]提出了 Sinkhorn 算法的超松弛形式,能进一步加速 Sinkhorn 算法的收敛速度;Xie 等人[14]受熵正则化方法启发提出了一种快速的近似点算法,使用 Sinkhorn 算法作为内迭代步骤进行传输计划计算的算法,最终算法能够收敛至精确的最优传输计划。

本文提出的算法是对 Xie 等人[14]提出的近似点算法的一种改进。该近似点算法具有能够收敛至最优传输精确解和对参数选择具有鲁棒性的优点,但实际应用时包括在生成对抗网络等的应用中对算法收敛速度有更高的要求,在调研了其他 Sinkhorn 算法的改进算法后,笔者认为该算法的收敛速度有进一步提升的空间。这是由于近似点算法在内迭代步骤相当于对一个不同形式的熵正则化最优传输问题求解,使用了 Sinkhorn 算法,而 Sinkhorn 算法的收敛速度相对于超松弛的 Sinkhorn 算法较慢。因此本文将为该近似点算法的内迭代步骤中传输计划的计算引入超松弛算子,并给出适应于近似点算法的超松弛参数的计算方法,由此得到超松弛的近似点算法。通过提升内迭代步骤的收敛速度,从而提升整个近似点算法的收敛速度。最后通过理论和实验验证算法收敛速度是否得到提升。

* **收稿日期**:2021-10-29 **修回日期**:2021-12-15 **网络出版时间**:2022-12-07 17:21
资助项目:国家自然科学基金(No. 61773152)
第一作者简介:吴凡,男,研究方向为计算最优传输,E-mail:fanwu_hhu@163.com;**通信作者**:刘向阳,男,副教授,博士,E-mail:liuxy@hhu.edu.cn
网络出版地址:https://kns.cnki.net/kcms/detail//50.1165.N.20221207.1459.005.html

1 最优传输理论与近似点算法

1.1 离散最优传输与 Sinkhorn 算法

最优传输问题最早由法国数学家蒙日在解决实际运输问题时提出，即在给定代价函数下寻求将一种分布变换成另一种分布的有效方式。本文所讨论的是离散形式的最优传输问题，它涉及到求解下述的 Wasserstein 距离：

$$W(\boldsymbol{\mu},\boldsymbol{\nu})=\min_{\boldsymbol{\Gamma}\in\Sigma(\boldsymbol{\mu},\boldsymbol{\nu})}\langle \boldsymbol{c},\boldsymbol{\Gamma}\rangle, \tag{1}$$

其中：$\boldsymbol{\mu}$、$\boldsymbol{\nu}$ 为两个概率向量，$W(\boldsymbol{\mu},\boldsymbol{\nu})$是 $\boldsymbol{\mu}$，$\boldsymbol{\nu}$ 间的 Wasserstein 距离，矩阵 $\boldsymbol{c}=[c_{ij}]\in\mathbf{R}_{+}^{m\times n}$ 为代价矩阵，m、n 取值为正整数，c_{ij} 为 $\boldsymbol{\mu}$ 的第 i 个和 $\boldsymbol{\nu}$ 的第 j 个支撑点间的距离，$\Sigma(\boldsymbol{\mu},\boldsymbol{\nu})=\{\boldsymbol{\Gamma}\in\mathbf{R}_{+}^{m\times n}:\boldsymbol{\Gamma}\mathbf{1}_m=\boldsymbol{\mu},\boldsymbol{\Gamma}^{\mathrm{T}}\mathbf{1}_n=\boldsymbol{\nu}\}$。$\mathbf{1}_n$ 表示元素均为 1 的 n 维向量，$\langle\cdot\rangle$为 Frobenius 点积。问题(1)的精确解 $\boldsymbol{\Gamma}^{*}$ 被称为最优传输计划，通过线性规划方法可求解问题(1)的精确解，但时间复杂度过高，效率低。

Cuturi[10] 则引入了问题(1)的正则化形式，转化为下述的熵正则化最优传输问题：

$$W_{\varepsilon}(\boldsymbol{\mu},\boldsymbol{\nu})=\min_{\boldsymbol{\Gamma}\in\Sigma(\boldsymbol{\mu},\boldsymbol{\nu})}\langle \boldsymbol{c},\boldsymbol{\Gamma}\rangle+\varepsilon h(\boldsymbol{\Gamma})。 \tag{2}$$

其中：熵正则化项 $h(\boldsymbol{\Gamma})=\Sigma_{ij}\Gamma_{ij}\ln\Gamma_{ij}$，$\varepsilon$ 为熵正则化参数，$\varepsilon>0$。(2)式可采用 Sinkhorn 矩阵缩放算法快速求解，首先给定标号：

$$A_1:\begin{cases}\mathbf{R}^{n_1n_2}\to\mathbf{R}^{n_1}\\(A_1\boldsymbol{x})_i=\sum\limits_j\boldsymbol{x}_{i,j}\end{cases},A_2:\begin{cases}\mathbf{R}^{n_1n_2}\to\mathbf{R}^{n_2}\\(A_2\boldsymbol{x})_j=\sum\limits_i\boldsymbol{x}_{i,j}\end{cases},$$

$$C_k=\{\boldsymbol{\Gamma}\in\mathbf{R}^{n_1n_2}\mid A_k\boldsymbol{\Gamma}=\boldsymbol{u}_k\},\boldsymbol{u}_1=\boldsymbol{\mu},\boldsymbol{u}_2=\boldsymbol{\nu},\boldsymbol{u}_k\in\mathbf{R}_{+*}^{n_k},k=1,2。$$

记 P_C 是在集合 C 上的 Bregman 投影，将它定义为 $P_C(\boldsymbol{\zeta}):=\operatorname{argmin}\mathrm{KL}(\boldsymbol{\Gamma},\boldsymbol{\zeta})$，$\boldsymbol{\zeta}$ 为任意矩阵，$\mathrm{KL}(\cdot)$表示计算 KL 散度，则 Sinkhorn 算法为：

$$\boldsymbol{\Gamma}^0=\boldsymbol{G}=\mathrm{e}^{-\boldsymbol{c}/\varepsilon},\boldsymbol{\Gamma}^{l+1}=P_{C_2}(P_{C_1}(\boldsymbol{\Gamma}^l)),\boldsymbol{\Gamma}^0=\mathbf{G}=\mathrm{e}^{-\boldsymbol{c}/\varepsilon},\boldsymbol{\Gamma}^{l+1}=P_{C_2}(P_{C_1}(\boldsymbol{\Gamma}^l)),\lim_{l\to+\infty}\boldsymbol{\Gamma}^l=\boldsymbol{\Gamma}^{*}。$$

离散形式下的具体迭代步骤为：$\boldsymbol{a}^{(l+1)}=\dfrac{\boldsymbol{\mu}}{\boldsymbol{G}\boldsymbol{b}^{(l)}},\boldsymbol{b}^{(l+1)}=\dfrac{\boldsymbol{\nu}}{\boldsymbol{G}^{\mathrm{T}}\boldsymbol{a}^{(l+1)}}$。$\boldsymbol{b}^{(0)}$ 初始化为 $\dfrac{1}{n}\mathbf{1}_n$，$\boldsymbol{G}=[G_{ij}]$，其中 $G_{ij}=\mathrm{e}^{-c_{ij}/\varepsilon}$，则 $\Gamma_{ij}=a_iG_{ij}b_j$，因此最终可求得矩阵 $\boldsymbol{\Gamma}$。

1.2 近似点算法

Sinkhorn 算法应用广泛，但计算效率受正则化参数 ε 的影响较大，且无法收敛到精确解。本节介绍一种非确定的近似点算法[14]（Inexact proximal point method，IPOT），以下简称为 IPOT 算法。

根据一般形式的近似点算法[15]，构造适用于问题(1)的近似点迭代格式：

$$\boldsymbol{\Gamma}^{(t+1)}=\arg\min_{\boldsymbol{\Gamma}\in\Sigma(\boldsymbol{\mu},\boldsymbol{\nu})}\langle \boldsymbol{c},\boldsymbol{\Gamma}\rangle+\beta^{(t)}D_h(\boldsymbol{\Gamma},\boldsymbol{\Gamma}^{(t)})。 \tag{3}$$

其中：$\beta^{(t)}$ 为系数，作用及取值范围同参数 ε。若：

$$D_h(\boldsymbol{X},\boldsymbol{Y})=\sum_{i=1}^{n}\boldsymbol{X}_i\log\frac{\boldsymbol{X}_i}{\boldsymbol{Y}_i}-\sum_{i=1}^{n}\boldsymbol{X}_i+\sum_{i=1}^{n}\boldsymbol{Y}_i。 \tag{4}$$

将(4)式代入(3)式可得：

$$\boldsymbol{\Gamma}^{(t+1)}=\arg\min_{\boldsymbol{\Gamma}\in\Sigma(\boldsymbol{\mu},\boldsymbol{\nu})}\langle \boldsymbol{c}-\beta^{(t)}\log\boldsymbol{\Gamma}^{(t)},\boldsymbol{\Gamma}\rangle+\beta^{(t)}h(\boldsymbol{\Gamma})。 \tag{5}$$

令 $\boldsymbol{c}'=\boldsymbol{c}-\beta^{(t)}\ln\boldsymbol{\Gamma}^{(t)}$，而在(5)式中，$\boldsymbol{\Gamma}^{(t)}$ 是一个与 $\boldsymbol{\Gamma}$ 无关的确定的矩阵。因此 $\boldsymbol{c}'$可被看成一个新的代价矩阵，则(5)式转变为形如(2)式的熵正则化最优传输问题。与问题(2)相比，只需将 Sinkhorn 算法中的 G_{ij} 换为 $G'_{ij}=\mathrm{e}^{-c'_{ij}/\beta^{(t)}}=\Gamma_{ij}^{(t)}\mathrm{e}^{-c_{ij}/\beta^{(t)}}$ 即可。因此，算法可理解为每一步对 $\boldsymbol{G}$ 进行修正和利用 Sinkhorn 算法进行(5)式的求解。同时 IPOT 算法采取在每一步只进行一次 Sinkhorn 迭代的策略，算法最终仍能收敛至问题(1)的精确解[14]。

2 超松弛的 IPOT 算法

IPOT 算法具有较高的计算效率，本文参考超松弛形式的 Sinkhorn 算法[13]，为 IPOT 算法的内迭代步骤中传输计划的计算引入超松弛算子，并为超松弛参数的计算给出新的策略，使收敛速度得到进一步提升。

2.1 超松弛 Sinkhorn 算法

首先介绍超松弛的 Sinkhorn 算法,在 1.1 节 Sinkhorn 算法中,给定下述标号:

$$P_{C_1}(\boldsymbol{\Gamma})=\mathrm{diag}(\boldsymbol{a})\boldsymbol{\Gamma},P_{C_2}(\boldsymbol{\Gamma})=\boldsymbol{\Gamma}\mathrm{diag}(\boldsymbol{b}),\tag{6}$$

其中:diag(·)表示由向量生成对角矩阵,再给定参数 $\omega\geqslant 0$,为上述的 Bregman 投影算子引入 ω 松弛投影算子 $P_{C_k}^{\omega}$,有 $\log P_{C_k}^{\omega}(\boldsymbol{\Gamma})=(1-\omega)\log \boldsymbol{\Gamma}+\omega\log P_{C_k}(\boldsymbol{\Gamma})$。

接下来的定理 1 则给出如何选择超松弛参数 ω。在进行每一步超松弛的 Sinkhorn 迭代时,都分别需要在 $\boldsymbol{a}$ 和 $\boldsymbol{b}$ 方向上分别给定超松弛参数 ω,则分别记为 Θ_1、Θ_2。

定理 1[13]　根据局部收敛性选定目标参数 $\theta_0\in[1;2)$(具体方式在 2.3 节给出),并给定小的安全距离 $\delta>0$,定义 $\Theta^*(\omega)=\sup\{\omega\in[1;2]\mid\varphi_\omega(\min\omega)\geqslant 0\}$,其中:

$$\varphi_\omega(x)=x(1-x^{-\omega})-\omega\log x,$$
$$\Theta(\omega)=\min(\max(1,\Theta^*(\omega)-\delta),\theta_0)。\tag{7}$$

则选择的 Θ_k 为:

$$\Theta_k(\boldsymbol{\Gamma})=\Theta((A_k\boldsymbol{\Gamma})\div\boldsymbol{u}_k)。\tag{8}$$

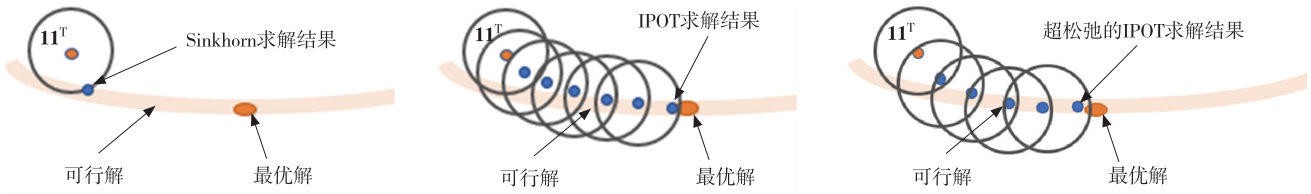
其中:÷表示对应元素相除,$k=1,2$,其余变量的符号说明已前文给出。

定理 1 选择的 Θ_k 即为每一步超松弛 Sinkhorn 迭代的超松弛参数 ω。

2.2 超松弛的 IPOT 算法

Xie 等人[14]提出的 IPOT 算法使用经典的 Sinkhorn 迭代作为内迭代步骤,并在每个内迭代步骤仅进行一次 Sinkhorn 迭代。而由于超松弛形式的 Sinkhorn 算法是对原 Sinkhorn 算法的一种加速,一步迭代能够比原 Sinkhorn 算法走得更远,而所提的算法采用超松弛形式的 Sinkhorn 算法,因此性能更优。

图 1a 展示了 Sinkhorn 算法的收敛结果,与最优解存在较大的误差。图 1b 则展示了原 IPOT 算法的收敛过程,它通过内外迭代步骤使得收敛结果逐渐逼近且最终能收敛至最优解。图 1c 则展示了超松弛的 IPOT 算法的收敛过程,与原 IPOT 算法迭代过程类似。但由于内迭代步骤引入了超松弛算子,能够在更少的迭代步骤达到同等的精度,也即算法的收敛更快。图中 3 个算法的初始化相同,本文提出的算法如算法 1 所示,基本框架继承于原 IPOT 算法[14]。



a　Sinkhon 算法　　b　IPOT 算法　　c　超松弛的 IPOT 算法

图 1　3 种算法迭代过程示意

Fig. 1　Iterative process of the three methods

算法 1　超松弛 IPOT 算法框架

输入:概率向量$\{\boldsymbol{\mu},\boldsymbol{\nu}\}$,支撑点分别为$\{g_i\}_{i=1}^m$,$\{h_j\}_{j=1}^n$,代价矩阵 $\boldsymbol{c}=[\|g_i-h_j\|]$。

输出:$\boldsymbol{\Gamma}^*$

$\boldsymbol{b}\leftarrow\frac{1}{m}\mathbf{1}_m$

$G_{ij}\leftarrow \mathrm{e}^{-c_{ij}/\beta}$

$\boldsymbol{\Gamma}^{(1)}\leftarrow\mathbf{1}\mathbf{1}^{\mathrm{T}}$

for $t=1,2,3,\cdots$ do

　$\boldsymbol{Q}\leftarrow\boldsymbol{G}\otimes\boldsymbol{\Gamma}^{(t)}$ %%⊗表示对应元素相乘

　　for $1=1,2,3,\cdots,L$ do%%通常设定 $L=1$

　　$\tilde{\boldsymbol{a}}=\frac{\boldsymbol{\mu}}{\boldsymbol{G}\boldsymbol{b}}$

$$\omega=\Theta(\boldsymbol{a}\div\tilde{\boldsymbol{a}})$$
$$\boldsymbol{a}=\boldsymbol{a}^{1-\omega}\otimes\tilde{\boldsymbol{a}}^{\omega}$$
$$\tilde{\boldsymbol{b}}=\frac{\boldsymbol{v}}{\boldsymbol{G}^{\mathrm{T}}\boldsymbol{b}}$$
$$\omega=\Theta(\boldsymbol{b}\div\tilde{\boldsymbol{b}})$$
$$\boldsymbol{b}=\boldsymbol{b}^{1-\omega}\otimes\tilde{\boldsymbol{b}}^{\omega}$$

end for

$$\boldsymbol{\Gamma}^{(t+1)}\leftarrow\mathrm{diag}(\boldsymbol{a})\boldsymbol{Q}\mathrm{diag}(\boldsymbol{b})$$

end for

2.3 超松弛参数选择

对于超松弛参数 ω 的选择尤为重要，而参数 θ_0 的选择直接影响超松弛参数 ω 的确定。本节主要给出 θ_0 的选择策略，之后 ω 的具体计算使用(7)、(8)式即可。首先给出超松弛 Sinkhorn 算法的局部收敛性，并由此给出参数 θ_0 选择方式。

定理 2[13]　Sinkhorn 算法的局部收敛性是线性的，为 $1-\xi$，而超松弛形式的 Sinkhorn 算法的最佳局部收敛性为 $(1-\sqrt{\xi})/(1+\sqrt{\xi})$，此时 $\theta_0=2/(1+\sqrt{\xi})$。若直接使用原超松弛的 Sinkhorn 算法中的 θ_0 选择策略，它会造成算法不收敛，见图 2。原因是 IPOT 算法在每步的 Sinkhorn 算法后进行了矩阵 $\boldsymbol{G}$ 的修正这一外迭代步骤，则每一步的 Sinkhorn 算法相当于是在不同的设定下进行的。因此考虑 IPOT 算法，根据所要求的精度设定算法停止条件，对每一次内迭代步骤的 Sinkhorn 算法进行局部收敛性估计，再经计算得到不同的 θ_0（每一步 θ_0 的计算与原超松弛的 Sinkhorn 算法中方法相同），从中选择最小的即可保证算法收敛性，且能尽量使算法具有较快的收敛速度。

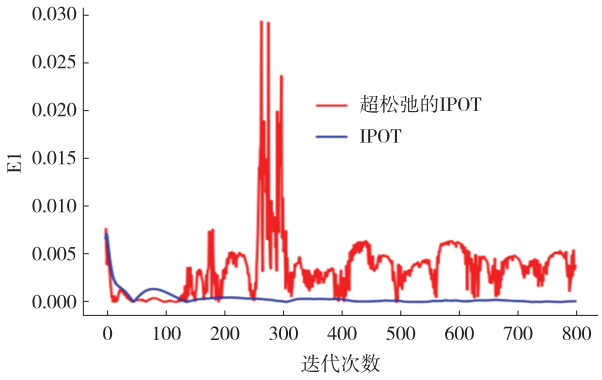


图 2　收敛性说明

Fig. 2　Indication of convergence

具体地，对于初始两个概率分布分别为 a_1、a_2，近似点算法过程中某一步 Sinkhorn 内迭代步骤结束后的暂时结果为 $\boldsymbol{a}$、$\boldsymbol{b}$。则此时误差为：

$$E_2=\|a_1-\boldsymbol{a}(\boldsymbol{Q}\mathrm{diag}(\boldsymbol{b}))\|+\|a_2-\boldsymbol{b}(\boldsymbol{Q}^{\mathrm{T}}\mathrm{diag}(\boldsymbol{a}))\|。$$

根据上式可计算局部收敛性，对于第 k 个 Sinkhorn 迭代，局部收敛性估计[13]为

$$1-\xi_k=1-(E_{2_{k-1}}-E_{2_k})/E_{2_k}。$$

则由定理 2 可得 $\theta_0^k=2/(1+\sqrt{\xi_k})$，因此最终 $\theta_0=\min\{\theta_0^n\}$，此时 $n=1,2,3,\cdots$。

2.4 收敛性分析

定理 3　对于优化问题，$\arg\min\limits_{x\in\chi} f(\boldsymbol{x})$ 假设 $\{\boldsymbol{x}^{(t)}\}$、$\{\boldsymbol{x}_*^{(t)}\}$ 分别是 IPOT 算法和超松弛的 IPOT 算法的迭代序列，定义误差序列 $\boldsymbol{e}^{(t+1)}\in\beta^{(t)}[\nabla f(\boldsymbol{x}^{(t+1)})+\partial_{\iota_\chi}(\boldsymbol{x}^{(t+1)})]+[\nabla h(\boldsymbol{x}^{(t+1)})-\nabla h(\boldsymbol{x}^{(t)})]$，其中：$\iota_\chi$ 为集合 χ 的示性函数，$\partial_{\iota_\chi}(\cdot)$ 为它的次微分。若 $\{\boldsymbol{e}^k\}$ 满足 $\sum\limits_{k=1}^{\infty}\|\boldsymbol{e}^k\|<\infty$，且 $\sum\limits_{k=1}^{\infty}\langle\boldsymbol{e}^k,\boldsymbol{x}^{(t)}\rangle$ 存在并有限，则 $\{\boldsymbol{x}^{(t)}\}$ 收敛至 $\boldsymbol{x}^{\infty}$，且 $f(\boldsymbol{x}^{\infty})=f^*$。而若对于 $\rho\in(0,1)$，有 $\|\boldsymbol{e}^{(t)}\|\leqslant\rho^t$，$\langle\boldsymbol{e}^{(t)},\boldsymbol{x}^{(t)}\rangle\leqslant\rho^t$，则 $\{\boldsymbol{x}^{(t)}\}$ 线性收敛，$\{\boldsymbol{x}_*^{(t)}\}$ 性质同上。具体地，令 $d^{(t)}=d(\boldsymbol{\Gamma},\boldsymbol{\Gamma}^{(t)})=h(\boldsymbol{\Gamma})-h(\boldsymbol{\Gamma}^*)-\langle\nabla h(\boldsymbol{\Gamma}^*),\boldsymbol{\Gamma}-\boldsymbol{\Gamma}^*\rangle$，则：

$$d^{(t+1)}\leqslant\lambda d^{(t)}+\lambda(L\|\boldsymbol{e}^{(t+1)}\|+\langle\boldsymbol{x}^{(t+1)},\boldsymbol{e}^{(t+1)}\rangle)。\tag{9}$$

$\{\boldsymbol{x}^{(t)}\}$ 和 $\{\boldsymbol{x}_*^{(t)}\}$ 的收敛速度均由上式给出。其中 $L=\sup\limits_{x\in\chi}\{\|\boldsymbol{x}\|\}$，对于 $\{\boldsymbol{x}^{(t)}\}$，λ 取 $\lambda_1=1/(1+\beta\eta_1)$，而对于 $\{\boldsymbol{x}_*^{(t)}\}$，$\lambda$ 取 $\lambda_2=1/(1+\beta\eta_2)$，$\eta$ 由 $f(\boldsymbol{x})-f(\boldsymbol{x}^*)\leqslant\eta d(\boldsymbol{x})$ 给出。且有 $\eta_1<\eta_2$，所以 $\lambda_1>\lambda_2$。因此超松弛 IPOT 算法进一步提升了收敛速度。

证明　直到(9)式均由文献[14]中定理 2 给出证明，而要证明 $\eta_1<\eta_2$，因为 η_1，η_2 描述了内迭代步骤分别采

取的 Sinkhorn 算法和超松弛的 Sinkhorn 算法的收敛性,大小关系与两算法的局部收敛性的大小关系保持一致,此处为了简便起见不妨直接令 $\eta_1=1-\xi,\eta_2=(1-\sqrt{\xi})/(1+\sqrt{\xi})$,则有:

$$\eta_1-\eta_2=1-\xi-(1-\sqrt{\xi})/(1+\sqrt{\xi})=\frac{(1-\xi)^2-(1-\sqrt{\xi})^2}{1-\xi}。$$

而 $\xi\in(0,1)$,则 $1-\xi<1-\sqrt{\xi}$,则 $\eta_1-\eta_2<0$,即 $\eta_1<\eta_2$。 证毕

3 实验结果及分析

本节将通过实验验证本文算法的收敛性能、对参数的鲁棒性及结果的稀疏性,也即将超松弛的 IPOT 算法与原 IPOT 算法、Sinkhorn 算法及其改进算法进行对比分析。实验中使用 python 开源的最优传输计算库 POT[16] 采用线性规划方法进行最优传输精确解的求解。

3.1 收敛性能

本节将应用算法求解如图 3 所示的两个高斯混合分布间的 Wasserstein 距离及最优传输计划来证明本文算法的收敛性能提升。红线为 $a_1=0.4\varphi(\cdot|60,8)+0.6\varphi(\cdot|40,6)$,蓝线为 $a_2=0.5\varphi(\cdot|35,9)+0.5(\cdot|70,9)$,其中 $\varphi(\cdot|\mu,\sigma^2)$表示均值为 μ,方差为 σ^2 的一维高斯分布。

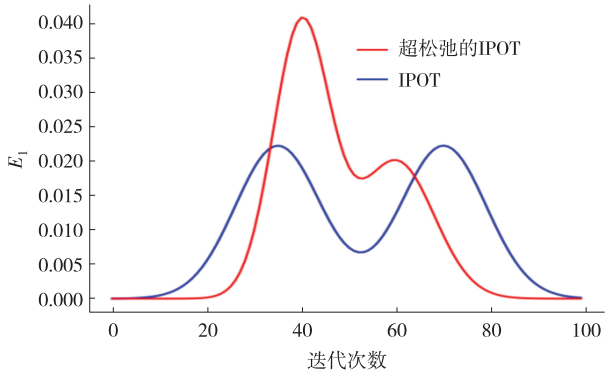


图 3 两个概率分布输入

Fig. 3 Two probability distributions as input

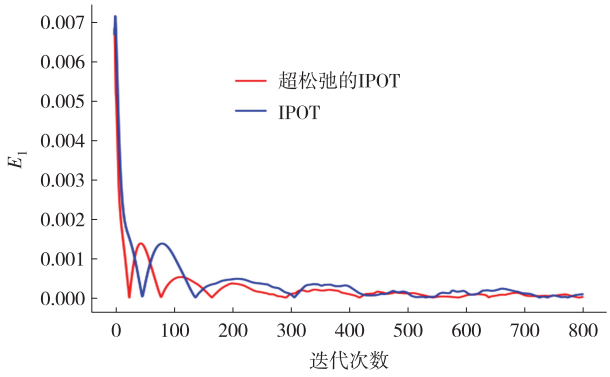


图 4 $\beta=0.005$ 时两个算法表现

Fig. 4 The iterative process with parameter of 0.005

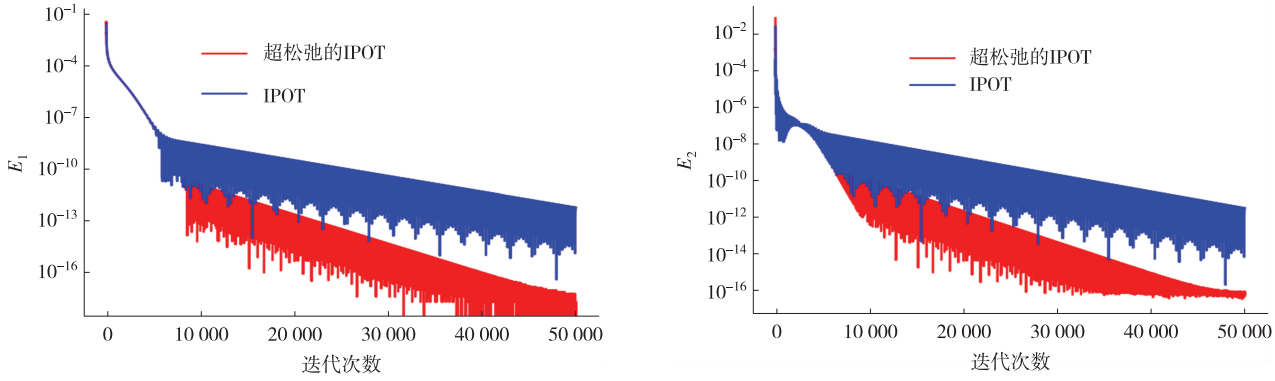
对于较小的 β,如 $\beta=0.005$ 时,IPOT 算法与超松弛的 IPOT 算法收敛情况如图 4 所示,图中纵轴的 $E_1=|W-W_{lp}|$ 表示算法得到的 Wasserstein 距离与精确 Wasserstein 距离间的误差,其中 W_{lp} 表示精确 Wasserstein 距离,横轴表示迭代次数。从图中可以看到超松弛的 IPOT 算法在同样的迭代步骤下误差基本上都小于原 IPOT 算法,也即为了达到相同的精度,超松弛的 IPOT 算法可以在更少的迭代步骤完成,收敛速度得到了一定的提升。

而对于较大的 β,如 $\beta=0.1$ 时,更多迭代次数下 IPOT 算法与超松弛的 IPOT 算法收敛情况如图 5 所示。其中 E_1 和横轴等定义同图 4,而 $E_2=\|a_1-\boldsymbol{a}(\boldsymbol{Q}\mathrm{diag}(\boldsymbol{b}))\|+\|a_2-\boldsymbol{b}(\boldsymbol{Q}^{\mathrm{T}}\mathrm{diag}(\boldsymbol{a}))\|$,从两个输入向量角度描述算法迭代过程中传输计划的误差。从图 5a 中可以看出,虽然在 β 较大时,在开始时的一些迭代步骤两个算法收敛速度几乎没有差别,但在达到一定迭代步骤后,超松弛的 IPOT 算法相比原 IPOT 算法在同样的迭代步骤能够达到更小的误差。图 5b 从另一个角度也佐证了在同样的迭代步骤下超松弛的 IPOT 算法误差更小。可以看到在 β 较大时,超松弛的 IPOT 算法在越逼近精确解时表现效果越优于原 IPOT 算法。

上述实验可以验证,超松弛的 IPOT 算法相对于原 IPOT 算法有一定的收敛速度提升,超松弛的 IPOT 算法可在更少的迭代步骤达到同等精度,收敛性能更佳。

进一步,将超松弛的 IPOT 算法、原 IPOT 算法与熵正则化方法的 Sinkhorn 算法[10] 及改进算法包括 greekhorn[12] 和超松弛的 Sinkhorn 算法[13] 进行直接对比,5 种算法的收敛性对比见图 6。横轴、竖轴表示同图 4。

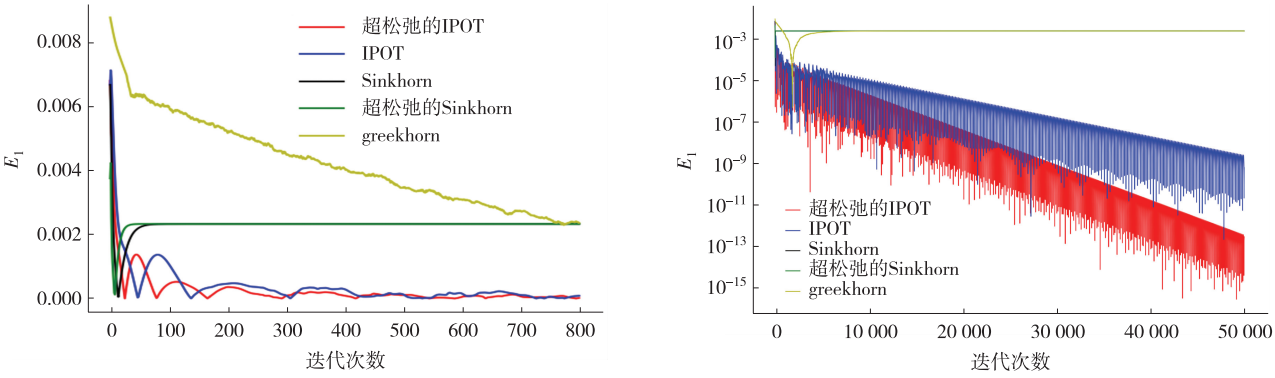
从图 6a 可以看到,熵正则化方法包括 Sinkhorn 算法及改进算法在迭代初期收敛速度都快于 IPOT 算法和本文所提的超松弛的 IPOT 算法,但这类算法在达到一定迭代步骤后误差就不再变化,具体如图 6b 所示,也即这类算法无法收敛至最优传输精确解,只能达到有限的精度。而原 IPOT 算法和超松弛的 IPOT 算法在迭代中误差能够持续下降,且超松弛的 IPOT 算法相对于原 IPOT 算法收敛速度更快。



a　以最优传输计划的准确性为度量　　b　以最优传输计划的行列和误差为度量

图 5　$\beta=0.1$ 时两算法在两种度量下的表现

Fig. 5　Algorithm performance under two metrics with parameter of 0.1



a　算法在较早迭代过程的收敛性对比　　b　算法在整体迭代过程的收敛性对比

图 6　$\beta=0.005$ 时 5 种算法的表现

Fig. 6　Algorithm performance with parameter of 0.005

3.2 对正则化参数选择的鲁棒性

本节输入分布同 3.1 节，选取正则化参数 β 分别为 1、0.1、0.01 和 0.005，采取所提出的超松弛 IPOT 算法进行 Wasserstein 距离的求解，各参数下的算法收敛情况如图 7 所示。

从图 7 可以看出，不同的参数 β 下算法均能够鲁棒地朝着最优传输精确解收敛，均可在有限迭代步骤下达到所要求的精度，只是收敛速度有所不同。因此所提的超松弛 IPOT 算法保留了原 IPOT 算法的优点，对正则化参数选择也具有鲁棒性。

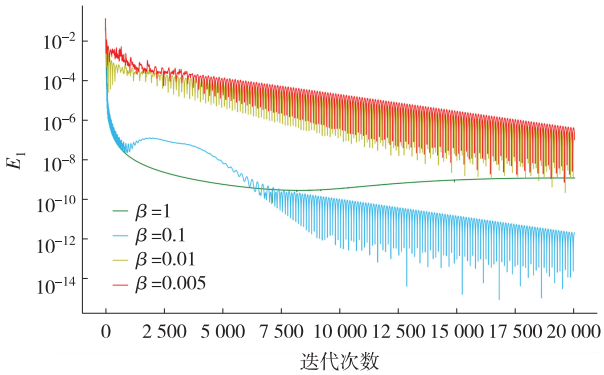


图 7　不同正则化参数下的迭代过程

Fig. 7　Iterative process under different regularization parameters

3.3 稀疏性

在诸如颜色传递[17]等图像处理任务中，往往需要精确而稀疏的最优传输计划。本节将直接对比分析 Sinkhorn 算法，IPOT 算法和超松弛的 IPOT 算法在各迭代步骤的最优传输计划图，如图 8 所示，输入分布同 3.1 节，Sinkhorn 算法中参数 $\varepsilon=0.01$，后两个算法中参数 $\beta=0.01$。图中黑色线为精确最优传输计划，红色为各算法在该迭代步骤下求得的最优传输计划。

从图 8 中可以看出，IPOT 算法和超松弛的 IPOT 算法均能在有限迭代步骤有效收敛，得到与精确最优传输计划误差较小的结果。而作为对比，Sinkhorn 算法则在很大的迭代步骤下仍然无法得到更精确的最优传输计划。因此在求解更为精确的最优传输计划时，所提的超松弛 IPOT 算法和原 IPOT 算法均优于 Sinkhorn 算法。

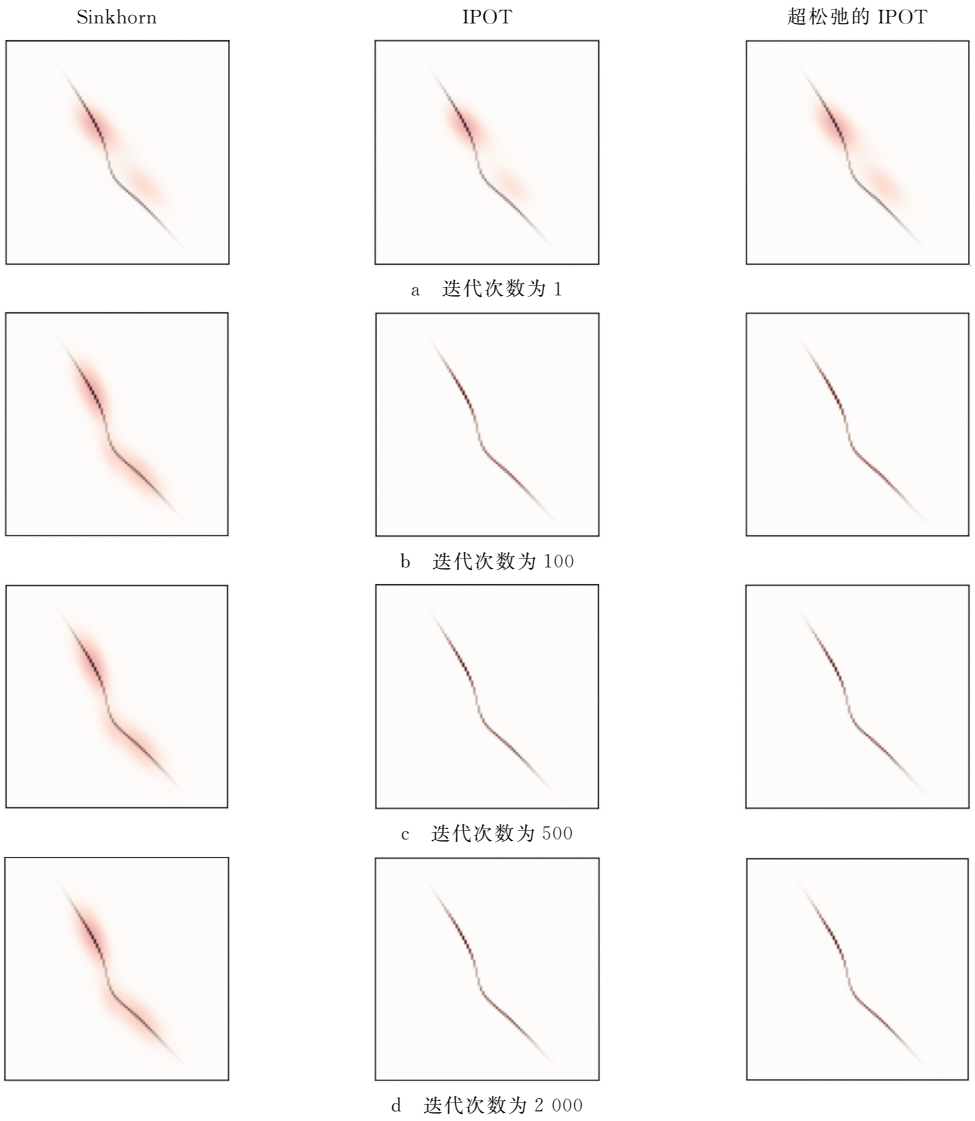


图8 各迭代步骤下最优传输计划

Fig. 8 Optimal transport plans under different iteration step

4 结束语

本文提出超松弛的IPOT算法,为原IPOT算法的内迭代步骤中传输计划的计算引入超松弛算子,并给出了超松弛参数的计算方法。所提算法,相较于原IPOT算法,提升了算法的收敛速度,在相同的迭代步骤下,能够达到更高的精度。未来将在近似点算法这个方向继续优化内迭代步骤,进一步提升整个算法的收敛速度。

参考文献:

[1] ARJOVSKY M, CHINTALA S, BOTTOU L. Wasserstein generative adversarial networks[C]//Proceedings of the 34th International Conference on Machine Learning. Sydney:PMLR,2017:214-223.

[2] SALIMANS T,ZHANG H,RADFORD A,et al. Improving GANs using optimal transport[EB/OL]. (2018-03-15)[2021-10-25]. https://arxiv.org/pdf/1803.05573.pdf.

[3] XU J J,ZHOU H,GAN C,et al. Vocabulary learning via optimal transport for neural machine translation[C]// Meeting of the Association for Computational Linguistics,2021.

[4] 王生生,姜林延,杨永波. 基于最优传输特征选择的医学图像分割迁移学习[J]. 吉林大学学报(工学版),2022,52(7):1626-1638.
WANG S S,JIANG L Y,YANG Y B. Transfer learning of medical image segmentation based on optimal transport feature selection[J]. Journal of Jilin University (Engineering and Technology Edition),2022,52(7):1626-1638.

[5] 马丽涛,边伟.最优传输理论及其在图像处理中的应用[J].运筹学学报,2019,23(3):109-125.
MA L T,BIAN W. A review of optimal transport in image processing[J]. Operations Research Transactions,2019,23(3):109-125.
[6] PELE O,WERMAN M. Fast and robust earth mover's distances[C]//2009 IEEE 12th International Conference on Computer Vision. Piscataway:IEEE,2009:460-467.
[7] LEE Y T,SIDFORD A. Path finding methods for linear programming:solving linear programs in iterations and faster algorithms for maximum flow[C]//2014 IEEE 55th Annual Symposium on Foundations of Computer Science. Piscataway:IEEE,2014:424-433.
[8] DVURECHENSKY P,GASNIKOV A,KROSHNIN A. Computational optimal transport:complexity by accelerated gradient descent is better than by Sinkhorn's algorithm[C]//International Conference on Machine Learning. Stockholm:PMLR,2018:1367-1376.
[9] CUTURI M, PEYRÉ G. A smoothed dual approach for variational Wasserstein problems[J]. SIAM Journal on Imaging Sciences,2016,9(1):320-343.
[10] CUTURI M. Sinkhorn distances:lightspeed computation of optimal transport[J]. Advances in Neural Information Processing Systems,2013,26:2292-2300.
[11] KNIGHT P A. The Sinkhorn-Knopp algorithm:convergence and applications[J]. SIAM Journal on Matrix Analysis and Applications,2008,30(1):261-275.
[12] ALTSCHULER J,WEED J,RIGOLLET P. Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration[J]. Advances in Neural Information Processing Systems,2017,2017:1965-1975.
[13] THIBAULT A,CHIZAT L,DOSSAL C,et al. Overrelaxed Sinkhorn-Knopp algorithm for regularized optimal transport[J]. Algorithms,2021,14(5):143.
[14] XIE Y J,WANG X F,WANG R J,et al. A fast proximal point method for computing exact wasserstein distance[C]//Uncertainty in Artificial Intelligence. PMLR,2020:433-453.
[15] ECKSTEIN J. Approximate iterations in Bregman-function-based proximal algorithms[J]. Mathematical Programming,1998,83(1):113-123.
[16] FLAMARY R,COURTY N,GRAMFORT A,et al. Pot:python optimal transport[J]. Journal of Machine Learning Research,2021,22(78):1-8.
[17] 谢斌,刘轲,张抗.结合图像签名和最优传输的颜色传递方法[J].液晶与显示,2021,36(8):1151-1165.
XIE B,LIU K,ZHANG K. Color transfer method for joint image signature and optimal transport[J]. Chinese Journal of Liquid Crystals and Displays,2021,36(8):1151-1165.

Operations Research and Cybernetics

An Overrelaxed Proximal Point Method for Optimal Transport

WU Fan, LIU Xiangyang
(School of Science, Hehai University, Nanjing 211106, China)

Abstract: [**Purposes**]In practical applications, Sinkhorn algorithm is usually used to solve the entropy regularization form of the optimal transport to obtain its approximate solution. Considering that the Sinkhorn algorithm is easily affected by the entropy regularization parameter, and it is difficult to converge to the final accurate solution, an overrelaxed form of proximal point method is proposed. [**Methods**]The overrelaxed operator is introduced to the iterative calculation of the transport plan in the original proximal point method of optimal transport, and the calculation strategy of the overrelaxed parameter is given. [**Findings**]While maintaining the robustness to the regularization parameter and the advantages of converging to the accurate solution, the proposed method can converge towards the accurate solution faster. Through theoretical and experimental analysis, the method is not only robust to the selection of entropy regularization parameter, but can also converge to an accurate optimal transport plan. [**Conclusions**]Compared with the original form of the proximal point method, the proposed method further improves its convergence speed, it can achieve higher accuracy within limited iteration, and the method can be better applied to machine learning.

Keywords: optimal transport; overrelaxed; proximal point method; entropy regularized; matrix scaling method

(责任编辑 黄 颖)