带次模惩罚和随机需求的设施选址问题*

王 星[1] 徐大川[2,†]

摘要 考虑带次模惩罚和随机需求的设施选址问题，目的是开设设施集合的一个子集，把客户连接到开设的设施上并对没有连接的客户进行惩罚，使得开设费用、连接费用、库存费用、管理费用和惩罚费用之和达到最小. 根据该问题的特殊结构，给出原始对偶3-近似算法. 在算法的第一步，构造了一组对偶可行解；在第二步中构造了对应的一组原始整数可行解，这组原始整数可行解给出了最后开设的设施集合和被惩罚的客户集合. 最后，证明了算法在多项式时间内可以完成，并且算法所给的整数解不会超过最优解的3倍.

关键词 次模惩罚, 随机需求，近似算法，原始对偶算法

中图分类号 O221.7

2010 数学分类号 90C27, 68W25

Facility location problem with submodular penalties and stochastic demands*

WANG Xing[1] XU Dachuan[2,†]

Abstract In this paper, we consider the facility location problem with submodular penalties and stochastic demands. The objective is to open a subset of facilities, to connect a subset of clients to open facilities, and to penalize the remaining clients such that the total cost of opening cost, connection cost, inventory cost, handling cost and penalty cost is minimized. Based on the special structure of the problem, we propose a primal-dual 3-approximation algorithm. In the algorithm, we construct a dual feasible solution in the first step followed by constructing the corresponding primal integer feasible solution in the second step. This primal integer feasible solution indicates the final opening facility set and penalty client set. We prove that the proposed algorithm can be implemented in polynomial time and the cost of the incurred primal integer feasible solution is no more than 3 times of the optimal value.

Keywords submodular penalties, stochastic demands, approximation algorithm, primal-dual algorithm

Chinese Library Classification O221.7

2010 Mathematics Subject Classification 90C27, 68W25

收稿日期：2012年5月10日

* 基金项目：北京市教育委员会科技计划面上项目 (No. KM201210005033)

1. 杭州电子科技大学理学院，杭州 310018；School of Science, Hangzhou Dianzi University, Hangzhou 310018, China

2. 北京工业大学数理学院，北京 100124；Department of Applied Mathematics, Beijing University of Technology, Beijing 100124, China

† 通讯作者 Corresponding author, Email: xudc@bjut.edu.cn

0 引 言

在无容量约束设施选址问题中，给定设施集合和客户集合，开设一些设施满足所有客户的要求并使得设施开设费用和连接费用之和达到最小. 无容量约束设施选址问题是经典的设施选址问题，它的第一个常数近似比是由Shmoys等[1]给出，近似比为3.16. 到目前为止，Li[2] 给出了关于此问题最好的近似比1.488. 另一方面，Guha和Khuller[3]给出了此问题的下界为1.463，除非$P=NP$.

在带次模惩罚的设施选址问题中，未被服务的客户集合$S\subseteq C$存在一个惩罚费用$h(S)$, 其中$h(S)$为单调增次模函数. 给定有限集N和定义在2^N上的一个实值函数h, 如果对于子集$X,Y\subseteq N$，有

$$h(X\cap Y)+h(X\cup Y)\leqslant h(X)+h(Y)$$

成立，则称h为次模函数；更进一步，如果当$X\subseteq Y$，有$h(X)\leqslant h(Y)$, 则称函数h是单调增的. 当$h(S)=\sum\limits_{j\in S}p_j$，其中$p_j$为客户$j\in C$不被服务时产生的惩罚费用，这种情形为带次模惩罚的设施选址问题的特例. 关于这个特例，Charikar等[4]第一次提出，并给出一个基于原始对偶的3-近似算法 (组合算法). [5, 6]分别使用线性规划舍入和对偶拟合加贪婪增广的技巧，把近似比改进到$2+\frac{2}{\mathrm{e}}$和1.8526. Hayrapetyan等[7]第一次提出带次模惩罚的设施选址问题，并给出一个具有常数近似比的近似算法，且当无容量约束设施选址问题的近似比为γ (与线性规划松弛比较得到) 时，他们的近似比为$1+\gamma$. 随后，Chudak和Nagano[8]提供了一个更有效的算法，但是近似比略微差些，为 $(1+\varepsilon)(1+\gamma)$. 最近，Du等[9]给出一个基于原始对偶的3-近似算法.

随机运输-库存网络设计问题 (本文中，我们称之为带随机需求的设施选址问题) 在供应链管理中有重要应用[10,11]. Shu等[12]给出了此问题的集覆盖规划形式，并利用列生成的办法来求解线性规划松弛. 最近，Li等[13]首次给出了原始对偶3-近似算法.

本文我们考虑带次模惩罚和随机需求的设施选址问题：给定设施集合F，客户集合C，客户集合中的每个客户$j\in C$有一个需求量d_j, d_j是随机变量，它的均值为μ_j，方差为σ_j^2. 每个设施$i\in F$都有开设费用f_i，把客户$j\in C$连接到设施$i\in F$的单位连接费用为c_{ij}，$F\cup C$上两点间的距离定义在度量空间上. 对每个设施i有一个库存总费用函数为$H_i\left(\sum_{j\in C_i}\sigma_j^2\right)$, 它是关于$\sum_{j\in C_i}\sigma_j^2$的单调增凹函数, 其中$C_i$为由设施$i$服务的所有客户组成的集合，且此函数的左导数存在. 设施$i\in F$服务所有连接到它上面的客户时，有一个管理费用$G_i\left(\sum_{j\in C_i}\mu_j\right)$, 它是关于$\sum_{j\in C_i}\mu_j$的单调增凹函数，其中$C_i$为由设施$i$服务的所有客户组成的集合. 对于每个$S\subseteq C$，有一个惩罚费用函数$h(S)$, $h(S)$是关于S的单调增次模函数. 我们要开设一些设施来满足每个客户的需求，即把客户连接到一个开设的设施上或者接受惩罚, 使得设施开设费用、连接费用、库存费用、管理费用和惩罚费用的总和达到最小. 根据该问题的特殊结构，沿用[9,13,14]的技巧，我们给出了原始对偶3-近似算法.

论文的结构如下：第1节，给出带次模惩罚和随机需求的设施选址问题的模型和预备知识; 第2节，给出原始对偶算法; 第3节，给出算法的理论分析，证明了近似比为3; 最后在第4节进行了讨论.

1 模 型

令 (i,S) 表示一个星形，它由设施i 和客户子集$S\subseteq C$组成. 引入二维变量$X_{i,S}$，表示星形 (i,S) 是否被选中，如果$X_{i,S}=1$, 则 (i,S) 被选中，即开设设施i且服务客户集合S；否则为0. 同时，引入二维变量Z_S, 如果S被选为惩罚的客户集合，则为1；否则为0. 下面给出带次模惩罚和随机需求的设施选址问题的整数规划形式：

$$\begin{aligned}
\min \quad & \sum_{i\in F}\sum_{S\subseteq C} C_{i,S}X_{i,S}+\sum_{S\subseteq C} h(S)Z_S \\
\text{s. t.} \quad & \sum_{i\in F}\sum_{S\subseteq C:j\in S} X_{i,S}+\sum_{S\subseteq C:j\in S} Z_S\geqslant 1, \qquad \forall j\in C, \\
& X_{i,S},\ Z_S\in\{0,1\}, \qquad \forall i\in F, S\subseteq C,
\end{aligned} \tag{1.1}$$

其中$C_{i,S}=f_i+\sum\limits_{j\in S}c_{ij}\mu_j+G_i\Big(\sum\limits_{j\in S}\mu_j\Big)+H_i\Big(\sum\limits_{j\in S}{\sigma_j}^2\Big)$. 第一组约束条件表示每个客户$j\in C$至少存在于一个被选中的星形$(i,S)$中或者被惩罚.

松弛整数约束，我们得到 (1.1) 的线性松弛规划，其对偶规划如下：

$$\begin{aligned}
\max \quad & \sum_{j\in C}\alpha_j \\
\text{s. t.} \quad & \sum_{j\in S}\alpha_j\leqslant C_{i,S}, \qquad \forall S\subseteq C, i\in F, \\
& \sum_{j\in S}\alpha_j\leqslant h(S), \qquad \forall S\subseteq C, \\
& \alpha_j\geqslant 0, \qquad \forall j\in C.
\end{aligned} \tag{1.2}$$

在对偶规划 (1.2) 中, α_j可以看做每个客户$j\in C$ 的预算，用来支付开设设施费用、连接费用、库存费用和管理费用的总费用或者用来支付惩罚费用.

引理 1.1[12] 令$v:\mathbb{R}\to\mathbb{R}$ 为凹函数，$x\in\mathbb{R}^{|X|}$ 为非零向量. 对每个$S\subseteq X$, 定义$u(S)\equiv v\left(\sum_{j\in S}x_j\right)$, 则$u$为$2^X$上的次模函数.

由$G_i(\cdot)$ 和$H_i(\cdot)$是凹函数和引理1.1可得下面引理.

引理 1.2[12] 对任意的$i\in F$, $S\subseteq C$, $C_{i,S}$ 是$2^{\mathbb{R}}$上的次模函数.

2 原始对偶算法

本节给出带次模惩罚和随机需求的设施选址问题的具体算法，算法的思想主要建立在Jain和Vazirani[14]和Du等[9]的原始对偶算法基础上.

先给出算法中使用的一些符号.

S_i: 表示连接到设施$i\in F$的客户集合；

l_i: 表示设施i在算法中被开设的次数；

S_{i,l_i}: 表示设施i在算法中第l_i次被开设时, 新增加的连接到i的客户集合；
$\tilde{F}$: 表示暂时开设的设施集合；
$\bar{F}$: 表示最终开设的设施集合；
S_p: 表示被惩罚的客户集合；
U: 表示未被连接且未被惩罚的客户集合；
t_j: 表示客户j首次连接到某个开设设施或者被惩罚的时间.

定义 2.1 对某一设施$i \in \widetilde{F}$, 如果存在$i' \in \widetilde{F}$, 使得$S_i \cap S_{i'} \neq \emptyset$, 则称$i'$为$i$的邻居. 记$N_i$为$i$的邻居组成的集合.

算法 2.1 原始对偶算法

步骤1 构造对偶可行解

步骤1.1 引入时间t, 初始化$t := 0$. 对任意的$j \in C$, $i \in F$, $\alpha_j := 0$, $U := C$, $S_i := \emptyset$, $l_i := 0$, $S_{i,l_i} := \emptyset$, $\widetilde{F} := \emptyset$, $\widetilde{F_j} := \emptyset$, $S_p := \emptyset$.

步骤1.2 顾客$j \in U$的预算以速率μ_j与时间t同步增长, 即$\alpha_j := \mu_j t$. 增长t直到下面的事件发生：

事件1 存在集合$S \subseteq U$, 集合$Q \subseteq C \backslash U$和设施$i \in F$, 使得

$$\sum_{j\in S}\mu_j t + \sum_{j\in Q}\alpha_j = C_{i,S\cup Q}$$

成立. 如果$i \notin \widetilde{F}$, 令$\widetilde{F} := \widetilde{F} \cup \{i\}$. 对$j \in S$, 令$\alpha_j := \mu_j t$, $t_j := t$, $i(j) := i$. 令$U := U \backslash S$. 对设施i, 令$S_{i,l_i} := S \cup Q$, $S_i := S_i \cup S_{i,l_i}$, $l_i := l_i + 1$.

事件2 存在集合$S \subseteq U$和集合$Q \subseteq C \backslash U$, 使得

$$\sum_{j\in S}\mu_j t + \sum_{j\in Q}\alpha_j = h(S\cup Q)$$

成立. 对$j \in S$, 令$\alpha_j := \mu_j t$, $t_j := t$. 令$U := U \backslash S$, $S_p := S_p \cup S \cup Q$.

当事件1和事件2同时发生时，任意执行其中一个.

步骤1.3 如果$U \neq \emptyset$, 转步骤1.2. 否则，转步骤2.

步骤2 构造原始整数可行解

步骤2.1 惩罚客户 把步骤1结束时的S_p作为最终被惩罚的客户集合，即S_p中所有的客户都不被服务，支付惩罚费用.

步骤2.2 开设设施

首先，令$\overline{F} := \emptyset$, $D := C \backslash S_p$表示未被惩罚的客户集合. $\widetilde{S_i}$表示最后连接到设施i上的客户集合. 接下来我们将迭代地执行下面的聚类过程，确定最终开设设施集合.

步骤2.2.1 取$\widetilde{F}$中使得$(H_i)'_-\left(\sum\limits_{j\in S_i}{\sigma_i}^2\right)$ $((H_i)'_-(\cdot)$ 表示$H_i(\cdot)$左导数)最小的设施, 记为i_0, 令$\overline{F} := \overline{F} \cup \{i_0\}$.

步骤2.2.2 对任意的设施$i' \in N_{i_0}$, 即$S_{i_0} \cap S_{i'} \neq \emptyset$. 我们假设设施$i'$被开设$m$次, 计算

$$l_{i'}^{i_0} := \min\left\{l_{i'} \in \{1, 2, \cdots, m\} : S_{i', l_{i'}} \cap S_{i_0} \neq \emptyset\right\}.$$

定义$S_{i'}^{i_0}(l_{i'}^{i_0}) := \{j : j \in S_{i'} \backslash S_{i', l_{i'}^{i_0}-1}\}$.

步骤2.2.3 令$\widetilde{F} := \widetilde{F} \backslash (\{i_0\} \cup \{i' \in N_{i_0} | l_{i'}^{i_0} = 1\})$, $D := D \backslash \cup_{i \in \overline{F}} S_i$. 对任意的$i' \in N_{i_0}$, 令$S_{i'} := S_{i', l_{i_0}^{i'}-1}$. 对每个$i \in \widetilde{F}$, 重新计算现在的$N_i$.

步骤2.2.4 如果$\widetilde{F} \neq \emptyset$, 转步骤2.2.1. 否则，转步骤2.3.

步骤2.3 指派客户

按照下面方式指派客户. 对任意的$i \in \overline{F}$, 令$\widetilde{S}_i := S_i \backslash S_p$, 其中$S_i$为$i$被选为最后开设的设施时连接到其上的客户集合. 对任意的$j \notin (\cup_{i \in \overline{F}} \widetilde{S}_i) \cup S_p$, 根据步骤1.2事件1，它第一次连接到的设施是$i(j)$, 这时存在$i \in \overline{F}$和$l_{i(j)}^i$, 使得$j \in S_{i(j)}^i(l_{i(j)}^i)$, 此时令$\widetilde{S}_i := \widetilde{S}_i \cup \{j\}$.

注 2.1 算法2.1的步骤2.2选取开设设施的原则是：按照某种顺序使得N_i不交的极大独立集. 注意N_i在每次迭代时都要更新.

3 算法分析

首先，我们证明算法2.1 在多项式时间内可以终止.

引理 3.1 已知步骤1中当前事件(两个事件之一)发生的时刻t_c, 则可以在多项式时间内找到下一个事件(两个事件之一)发生的时刻t_n.

证明 在算法2.1中，令t_c表示当前时刻，令U表示此时未被连接且未被惩罚的客户集合，令$\widetilde{F}$ 表示此时暂时开设的设施集合. 当$U \neq \emptyset$时，我们继续步骤1直到事件1或者事件2发生. 事件1的发生和事件2的发生是不同的情形，我们分这两种情形来估计下一个时刻t_n.

情形1 下一个发生事件为事件1, 其发生时刻为t_{n1}.

在算法2.1中，在时刻t_c之后，对任意的设施$i \in F$, 若有事件1发生，则存在集合$S \subseteq U$, 集合$Q \subseteq C \backslash U$ 使得等式

$$\sum_{j \in S} \mu_j t + \sum_{j \in Q} \alpha_j = C_{i, S \cup Q}$$

成立，整理得$t = \frac{C_{i,S \cup Q} - \sum_{j \in Q} \alpha_j}{\sum_{j \in S} \mu_j}$. 我们希望寻找到下一个最早的时刻$t_{n1}$, 即求

$$t_{n1} = \min_{\substack{i \in F, S \subseteq U, \\ Q \subseteq C \backslash U}} \frac{C_{i,S \cup Q} - \sum_{j \in Q} \alpha_j}{\sum_{j \in S} \mu_j}.$$

情形2 下一个发生事件为事件2, 其发生时刻为t_{n2}.

在算法2.1中，在时刻t_c之后，对任意的设施$i \in F$，若有事件2发生, 则存在集合$S \subseteq U$，集合$Q \subseteq C \backslash U$使得等式

$$\sum_{j \in S} \mu_j t + \sum_{j \in Q} \alpha_j = h(S \cup Q)$$

成立，整理得$t = \frac{h(S\cup Q)-\sum_{j\in Q}\alpha_j}{\sum_{j\in S}\mu_j}$. 我们希望寻找到下一个最早的时刻$t_{n2}$, 即求

$$t_{n2} = \min_{\substack{i\in F, S\subseteq U,\\ Q\subseteq C\setminus U}} \frac{h(S\cup Q)-\sum_{j\in Q}\alpha_j}{\sum_{j\in S}\mu_j}.$$

计算t_{n1}和t_{n2}可以归结为极小化一个次模函数与模函数的比值问题，从而可以在多项式时间内计算[15]. 令$t_n := \min\{t_{n1}, t_{n2}\}$, 所以我们可以在多项式时间内找到下一个事件发生时刻t_n.

算法2.1步骤1至多有m个事件发生，由引理3.1可知，算法2.1步骤1的运行时间为多项式时间; 算法2.1步骤2至多迭代n次. 于是，可得算法2.1在多项式时间内终止.

与[9]中引理3.2证明相似，由次模函数的性质可以证明下面两个引理(我们略去证明).

引理 3.2[9] *在算法2.1中步骤1的任何时刻t, 连接到设施$i\in\widetilde{F}$上的客户集合S_i和被惩罚的客户集合S_p都有下面的等式成立*

$$\sum_{j\in S_i}\alpha_j(t) = C_{i,S_i}, \quad \forall i\in\widetilde{F},$$
$$\sum_{j\in S_p}\alpha_j(t) = h(S_p),$$

其中$\alpha_j(t)$ 表示t时刻的α_j, 它随着时间t增长直到客户j被连上或者被惩罚，然后保持不变.

由引理3.2和算法2.1步骤1得到以下引理.

引理 3.3[9] *在算法2.1的步骤1, 连接到设施$i\in\widetilde{F}$上的客户集合S_i和被惩罚的客户集合S_p 都有下面的等式成立*

$$\sum_{j\in S_i}\alpha_j = C_{i,S_i}, \quad \forall i\in\widetilde{F},$$
$$\sum_{j\in S_p}\alpha_j = h(S_p).$$

在算法2.1步骤1结束时得到一组对偶可行解$\{\alpha_i\}$. 在算法2.1步骤2结束时，得到(1.1)的一组可行解. 回顾$\overline{F}$的定义，我们构造一组原始可行解如下,

$$X_{i,S} = \begin{cases} 1, & \text{如果 } i\in\overline{F},\ S=\widetilde{S}_i,\\ 0, & \text{否则;}\end{cases} \qquad Z_S = \begin{cases} 1, & \text{如果 } S=S_p,\\ 0, & \text{否则.}\end{cases} \tag{3.1}$$

上述解对应的目标函数值为$\sum_{i\in\overline{F}} C_{i,\widetilde{S_i}} + h(S_p)$.

下面我们给出每个客户$j\in C$的预算α_j的下界. 由引理3.3并用反证法可以得到(我们略去证明)以下引理.

引理 3.4 *在算法2.1的步骤1任意时刻, 对任意的$i\in\widetilde{F}$ 和$j\in S_i$, 我们有*

$$\alpha_j \geqslant c_{ij}\mu_j + G_i\left(\sum_{k\in S_i}\mu_k\right) - G_i\left(\sum_{k\in S_i\setminus\{j\}}\mu_k\right) + H_i\left(\sum_{k\in S_i}\sigma_k^2\right) - H_i\left(\sum_{k\in S_i\setminus\{j\}}\sigma_k^2\right).$$

引理 3.5 假设$v(\cdot)$ 是$(0,+\infty)$上的凹函数. 给定x_i $(i=1,2,3)$, 且$0<x_1<x_2<x_3$, 有

$$\frac{v(x_2)-v(x_1)}{x_2-x_1}\geqslant v'_-(x_2)\geqslant\frac{v(x_3)-v(x_2)}{x_3-x_2},$$

其中$v'_-(\cdot)$ 是$v(\cdot)$的左导数.

然后，我们给出$i\in\overline{F}$和在算法2.1结束时连接到i上的客户集合$\widetilde{S}_i$形成的星形$(i,\widetilde{S}_i)$的费用的上界. 下面的证明与[13]的引理3.3类似，只给出简要证明.

引理 3.6 在算法2.1的步骤2结束时, 对任意的$i\in\overline{F}$, 得

$$C_{i,\widetilde{S}_i\setminus S_p}\leqslant\sum_{j\in S_{i1}}\alpha_j+3\sum_{j\in S_{i2}}\alpha_j,$$

其中$\widetilde{S}_i=S_{i1}\bigcup S_{i2}$, S_{i1}表示在步骤2中i被选为中心时的连接到设施i上的客户集合, S_{i2}表示在步骤2中转连到设施i上的客户集合, $S_{i1}\cap S_{i2}=\emptyset$, $S_{i2}\cap S_p=\emptyset$.

证明 由引理3.3可知，对于$i\in\overline{F}$和S_{i1}, 有$C_{i,S_{i1}}=\sum\limits_{j\in S_{i1}}\alpha_j$ 成立. 由$C_{i,S}$的单调性，得$C_{i,\widetilde{S}_i\setminus S_p}\leqslant C_{i,S_{i1}\cup S_{i2}}$. 为了证明本引理，我们只需证

$$C_{i,S_{i1}\cup S_{i2}}-C_{i,S_{i1}}\leqslant 3\sum_{j\in S_{i2}}\alpha_j.$$

注意到$C_{i,S}$的定义，只需证

$$\begin{aligned}&\sum_{j\in S_{i2}}c_{ij}\mu_j+\left(G_i\left(\sum_{j\in S_{i1}\cup S_{i2}}\mu_j\right)-G_i\left(\sum_{j\in S_{i1}}\mu_j\right)\right)\\&\quad+\left(H_i\left(\sum_{j\in S_{i1}\cup S_{i2}}{\sigma_j}^2\right)-H_i\left(\sum_{j\in S_{i1}}{\sigma_j}^2\right)\right)\leqslant 3\sum_{j\in S_{i2}}\alpha_j.\end{aligned}\tag{3.2}$$

对每个$j\in S_{i2}$, 根据算法2.1, 存在$i(j)\in N_i$, 其中$i(j)$为j第一次连接的开设设施, $i(j)$在i被选为中心时有$j\in S^i_{i(j)}(l^i_{i(j)}):=\{j:j\in S_{i(j)}\backslash S_{i(j),l^i_{i(j)}-1}\}$, 使得$S_{i1}\cap S_{i(j),l^i_{i(j)}}\neq\emptyset$. 设$j'\in S_{i1}\cap S_{i(j),l^i_{i(j)}}$, 假设$j'$在设施$i(j)$第$l$次开设的时候连接到$i$上, 由$j\in S^i_{i(j)}(l^i_{i(j)}):=\{j:j\in S_{i(j)}\backslash S_{i(j),l^i_{i(j)}-1}\}$, 则$l\geqslant l^i_{i(j)}$, 即$j$连接到设施$i(j)$上的时间要比$j'$连接到设施$i(j)$上晚. 因为$i(j)$是客户$j$第一次连接的开设设施，所以$j$连接到设施$i(j)$上的时间为$t_j$. 但是$j'$第一次连接的时间$t_{j'}$要比它连接到设施$i(j)$上的时间要早，于是得$t_j\geqslant t_{j'}$. 由引理3.4, 引理3.5, 以及算法2.1步骤2.2.1, 得

$$\alpha_j=t_j\mu_j\geqslant c_{i(j)j}\mu_j+H_i\left(\sum_{k\in S_{i1}}\sigma_k^2+\sigma_j^2\right)-H_i\left(\sum_{k\in S_{i1}}\sigma_k^2\right),$$

$$\alpha_j=t_j\mu_j\geqslant t_{j'}\mu_j\geqslant c_{i(j)j'}\mu_j,$$

$$\alpha_j\geqslant t_{j'}\mu_j\geqslant c_{ij'}\mu_j+G_i\left(\left(\sum_{k\in S_{i1}}\mu_k\right)+\mu_j\right)-G_i\left(\sum_{k\in S_{i1}}\mu_k\right).$$

把上述三个式子相加, 并利用三角不等式，我们得

$$3\alpha_j \geqslant c_{ij}\mu_j + H_i\left(\sum_{k\in S_{i1}} \sigma_k^2 + \sigma_j^2\right) - H_i\left(\sum_{k\in S_{i1}} \sigma_k^2\right) + G_i\left(\sum_{k\in S_{i1}} \mu_k + \mu_j\right) - G_i\left(\sum_{k\in S_{i1}} \mu_k\right).$$

上式对S_{i2}中所有的客户求和，结合$H_i(\cdot)$和$G_i(\cdot)$的凹性，得到(3.2).

最后, 给出算法2.1的近似比.

定理 3.1 *算法2.1是带次模惩罚和随机需求的设施选址问题的原始对偶3-近似算法.*

证明 首先，算法2.1得到一组原始可行解(3.1), 目标函数值为$\sum\limits_{i\in\overline{F}} C_{i,\widetilde{S}_i\backslash S_p} + h(S_p)$. 其次, 考虑所有的$(i,\widetilde{S}_i\backslash S_p)$, $i\in\overline{F}$, 由引理3.6可得

$$\begin{aligned}\sum_{i\in\overline{F}} C_{i,\widetilde{S}_i\backslash S_p} &\leqslant \sum_{i\in\overline{F}}\left(\sum_{j\in S_{i1}}\alpha_j + 3\sum_{j\in S_{i2}}\alpha_j\right)\\ &= \sum_{i\in\overline{F}}\left(\sum_{j\in S_{i1}\backslash S_p}\alpha_j + \sum_{j\in S_{i1}\cap S_p}\alpha_j + 3\sum_{j\in S_{i2}}\alpha_j\right).\end{aligned}$$

由引理3.3, 得 $h(S_p)=\sum\limits_{j\in S_p}\alpha_j$. 于是,

$$\sum_{i\in\overline{F}} C_{i,\widetilde{S}_i\backslash S_p} + h(S_p) \leqslant \sum_{i\in\overline{F}}\left(\sum_{j\in S_{i1}\backslash S_p}\alpha_j + 3\sum_{j\in S_{i2}}\alpha_j\right) + 2\sum_{j\in S_p}\alpha_j \leqslant 3\sum_{j\in C}\alpha_j.$$

4 讨 论

在文中，我们考虑了带次模惩罚和随机需求的设施选址问题，给出了原始对偶3-近似算法. 下一步可以研究利用其他技巧，例如线性规划舍入，局部搜索等来改进该问题的近似比.

参 考 文 献

[1] Shmoys D B, Tardos E, Aardal K I. Approximation algorithms for facility location problems [C]//*Proceedings of STOC*, New York: Association for Computing Machinery, 1997, 265-274.

[2] Li S. A 1.488-approximation algorithm for the uncapacitated facility location problem [J]. *Information and Computation*, 2013, **222**: 45-58.

[3] Guha S, Khuller S. Greedy strike back: improved facility location algorithms [J]. *Journal of Algorithms*, 1999, **31**: 228-248.

[4] Charikar M, Khuller S, Mount D M, et al. Algorithms for facility location problems with outliers [C]//*Proceedings of SODA*, 2001, 642-651.

[5] Xu G, Xu J. An LP rounding algorithm for approximating uncapacitated facility location problem with penalties [J]. *Information Processing Letters*, 2005, **94**: 119-123.

[6] Xu G, Xu J. An improved approximation algorithm for uncapacitated facility location problem with penalties [J]. *Journal of Combinatorial Optimization*, 2009, **17**: 424-436.

[7] Hayrapetyan A, Swamy C, Tardos E. Network design for information networks [C]//*Proceedings of SODA,* 2005, 933-942.

[8] Chudak F A, Nagano K. Efficient solutions to relaxations of combinatorial problems with submodular penalties via the Lovasz extension and non-smooth covex optimization [C]//*Proceedings of SODA*, 2007, 79-88.

[9] Du D, Lu R, Xu D. A primal-dual approximation algorithm for the facility location problem with submodular penalties [J]. *Algorithmica*, 2012, **63**: 191-200.

[10] Daskin M S, Coullard C R, Max Shen Z J. An inventory-location model: formulation, solution algorithm and computational results [J]. *Annals of Operations Research*, 2002, **110**: 83-106.

[11] Max Shen Z J, Coullard C R, Daskin M S. A joint location-inventory model [J]. *Transportation Science*, 2003, **37**: 40-55.

[12] Shu J, Teo C P, Max Shen Z J. Stochastic transportation-inventory network design problem [J]. *Operations Research*, 2005, **53**: 48-60.

[13] Li Y, Shu J, Wang X, Xiu N, Xu D, Zhang J. Approximation algorithms for integrated distribution network design problems [J]. *INFORMS Journal on Computing*, DOI: 10.1287/ijoc.1120.0522.

[14] Jain K, Vazirani V V. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and Lagrangian relaxation [J]. *Journal of the ACM*, 2001, **48**: 274-296.

[15] Fujishige S. Submodular functions and optimization (2nd ed.) [M]. Amsterdam: Elsevier, 2005.