Vol.2015-AL-151 No.5
2015/1/13

Assigning proximity facilities for gatherings

Shin-ichi Nakano[1,a)]

Abstract:
In this paper we study a recently proposed variant of the problem, called the r-gathering problem.
Given a set C of customers, a set F of facilities, and a connecting cost $co(c, f)$ for each pair of $c \in C$ and $f \in F$, then r-gathering problem choose a subset $F' \subset F$ of facilities and find an assignment A from C to F' so that the cost $\max\{\max_{i\in C}\{co(i, A(i))\}, \max_{j\in F'}\{op(j)\}\}$ is minimized. The *the proximity r-gathering problem* finds an assignment with one more additional constraint, that is each customer should be assigned to a closest open facility. Armon gave a 9-approximation algorithm for the problem.
In this paper we give a simple 3-approximation algorithm for the proximity r-gathering problem.

1. Introduction

The facility location problem and many of its variants are studied[3], [4].

In the basic facility location problem we are given (1) a set C of customers, (2) a set F of facilities, (3) an opening cost $op(f)$ for each $f \in F$, and (4) a connecting cost $co(c, f)$ for each pair of $c \in C$ and $f \in F$, then we open a subset $F' \subset F$ of facilities and find an assignment A from C to F' so that a designated cost is minimized. A typical *max* version of the cost of an assignment A is $\max\{\max_{i\in C}\{co(i, A(i))\}, \max_{j\in F'}\{op(j)\}\}$. We assume that co satisfies the triangle inequality.

In this paper we study a recently proposed variant of the problem, called the r-gathering problem[1].

An *r-gathering* of customers C to facilities F is an assignment A of C to *open facilities* $F' \subset F$ such that r or more customers are assigned to each open facility. (Each open facility needs enough number of customers.) We assume $|C| \geq r$ holds. Then max version of the cost of an r-gathering is $\max\{\max_{i\in C}\{co(i, A(i))\}, \max_{j\in F'}\{op(j)\}\}$. Then the min-max version of the *r-gathering problem* finds an r-gathering having the minimum cost. (For the min-sum version see the brief survey in [1].)

Assume that F is a set of locations for emergency shelters, $op(f)$ is the time needed to prepare a shelter $f \in F$, and $co(c, f)$ is the time needed for a person $c \in C$ to reach assigned shelter $A(c) \in F$. Then an r-gathering corresponds to an evacuation plan such that each opened shelter serves r or more people, and the r-gathering problem finds an evacuttion plan minimizing the evacuation time span.

Armon[1] gave a simple 3-approximation algorithm for the problem and proved that with assumption $P \neq NP$ the problem cannot be approximated within a factor of less than 3 for any $r \geq 3$.

However in a solution above some person may be assigned to a farther open shelter although it has some closer open shelter. It may be difficult for the person to accept such an assignment for an emergency situation. Therefore Armon[1] also considered the problem with one more additional constraint, that is, each customer should be assigned to a closest open facility, and gave a 9-approximation algorithm for the problem. We call the problem *the proximity r-gathering problem*.

In this paper we give a simple 3-approximation algorithm for the proximity r-gathering problem.

The remainder of this paper is organized as follows. Section 2 contains our main algorithm for the proximity r-gathering problem. Section 3 considers a case with outliers. Section 4 gives a slightly improved algorithm for the original r-gathering problem. Section 5 contains a conclusion and an open problem.

2. Algorithm

We need some preparation.

A lower bound $lb(i, j)$ of the cost assigning $i \in C$ to $j \in F$ in any r-gathering is derived as follows. Let $N(j)$ be the set of r customers having up to r-th smallest connection costs to facility $j \in F$. If $i \in N(j)$ then define $lb(i, j) = max\{op(j), co(k, j)\}$, where k is the customer having the r-th smallest connection cost to j. Otherwise $lb(i, j) = max\{op(j), co(i, j)\}$. Then a lower bound $lb(i)$ of the cost for $i \in C$ in any r-gathering is derived as $lb(i) = \min_{j\in F}\{lb(i, j)\}$. Since we need to assign $i \in C$ to some facility, $lb(i)$ is also a lower bound for the cost of the solution of the proximity r-gathering problem. Let $bestf(i)$ for $i \in C$ be a facility $j \in F$ attaining cost $lb(i)$. Let $mates(i)$ for $i \in C$ be $N(bestf(i))$ if $i \in N(bestf(i))$, and $N(bestf(i)) \cup \{i\} - \{k\}$ otherwise. Thus if we assign $mates(i)$ to $bestf(i) \in F$ then the cost of the part is $lb(i)$.

We regard $co(f, f') = min_{i\in C}\{co(i, f) + co(i, f')\}$ for $f, f' \in F$. We define by opt the cost of the solution, that is

1 Gunma University, Kiryu, Gunma 376–8515, Japan
a) nakano@cs.gunma-u.ac.jp

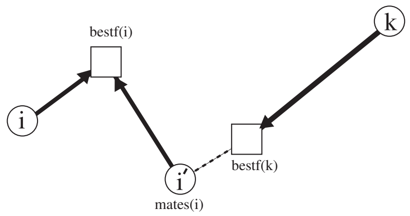

Fig. 1 Illustration for the proof of Lemma2.1 .

$\min_A \max\{\max_{i \in C}\{co(i, A(i))\}, \max_{j \in F'}\{op(j)\}\}$, where $F' \subset F$ is the set of opened facilities. Clearly $opt \geq lb(i)$ holds for any $i \in C$.

Now we give our algorithm to solve the proximity r-gathering problem.

Algorithm Best-or-factor3

for all $i \in C$ **do**
  Compute $lb(i)$, $bestf(i)$ and $mates(i)$
end for
Sort C in the non-increasing order of $lb(i)$
for all $i \in C$ in the non-increasing order of $lb(i)$ **do**
  if $bestf(i)$ is not assigned to yet, and none of $mates(i)$ has been assigned yet **then**
    Open $bestf(i)$
    for all $k \in mates(i)$ **do**
      Assign k to $bestf(i)$ /* Best-Assignment */
    end for
    for all f such that $co(f, bestf(i)) \leq 2 \cdot lb(i)$ **do**
      Shut down f
    end for
  end if
end for
for all unassigned $k \in C$ **do**
  Assign k to a closest open facility /* Factor3-Assignment */
end for

Clearly Algorithm **Best-or-factor3** finds an r-gathering. (Since whenever we newly open a facility we always assign r customers.)

The algorithm is similar to algorithm **Best-or-rest** in [1] for the original r-gathering problem, except (1) our algorithm has the "shut down f " operation, and (2) sorts C in the non-increasing order, while **Best-or-rest**[1] sorts C in the non-decreasing order. Actually we can modify algorithm **Best-or-rest** for the original r-gathering problem so that it does not need the sort. We show this in the later section.

We have the following lemma.

Lemma 2.1 Algorithm **Best-or-factor3** finds an r-gathering such that each customer is assigned to a closest open facility.

Proof. Assume otherwise for a contradiction. Since Factor3-Assignment never open any facility and always assign a customer to a closest open facility, we only consider for Best-Assignment. Then some $i' \in mates(i)$ assigned to facility $bestf(i)$ has a closer open facility, say $bestf(k)$ for some $k \in C$. We have two cases based on the opening order of $bestf(i)$ and $bestf(k)$.

If $bestf(k)$ opens earlier than $bestf(i)$ then $lb(k) \geq lb(i)$ holds, then

$$\begin{aligned} co(bestf(i), bestf(k)) &\leq co(i', bestf(i)) + co(i', bestf(k)) \\ &< co(i', bestf(i)) + co(i', bestf(i)) \\ &\leq 2 \cdot lb(i) \\ &\leq 2 \cdot lb(k) \end{aligned}$$

which contradicts to the fact that after we open facility $bestf(k)$ we shut down every surrounding facility with connection cost at most $2 \cdot lb(k)$. We need the sort for the last inequality.

Otherwise $bestf(i)$ opens earlier than $bestf(k)$ and $lb(i) \geq lb(k)$ holds, so

$$\begin{aligned} co(bestf(k), bestf(i)) &\leq co(i', bestf(k)) + co(i', bestf(i)) \\ &\leq co(i', bestf(i)) + co(i', bestf(i)) \\ &\leq 2 \cdot lb(i) \end{aligned}$$

which contradicts to the fact that after we open facility $bestf(i)$ we shut down every surrounding facility within connection cost at most $2 \cdot lb(i)$. $\mathcal{Q.E.D.}$

We have the following two theorems.

Theorem 2.2 The cost of an r-gathering found by Algorithm **Best-or-factor3** is at most $3 \cdot opt$.

Proof. Consider the cost for each assignment of $i \in C$. For Best-Assignment the cost is $lb(i) \leq opt$. So we need to consider only for $Factor3\text{-}Assignment$.

Each $i \in C$ assigned in $Factor3\text{-}Assignment$ was not assigned to $bestf(i)$ in Best-Assignment but later assigned to its closest already opened facility. So we consider only for connection costs.

Assume we assign $i \in C$ in $Factor3\text{-}Assignment$. We show that i always has an open facility with the connection cost at most $3\ opt$. We have two cases based on the reason why i was not assigned in $Best\text{-}Assignment$.

Case 1(a): Some $i' \in mate(i)$ is already assigned to some $bestf(k)$ since $i' \in mates(k)$ also holds. See Fig. 2(a).

The connection cost $co(i, bestf(k))$ is at most $co(i, bsetf(i)) + co(i', bestf(i)) + co(i', bestf(k)) \leq lb(i) + lb(i) + lb(k)$ $\leq 3\ opt$Thus $i \in C$ has an open facility with a connection cost at most $3\ opt$.

Case 1(b): $bestf(i)$ is already shut down just after some $bestf(k)$ is opened. See Fig. 2(b).

The connection cost $co(i, bestf(k))$ is at most

$$\begin{aligned} co(i, bsetf(i)) + co(bestf(i), bestf(k)) &\leq lb(i) + 2 \cdot lb(k) \\ &\leq 3\ opt \end{aligned}$$

Thus $i \in C$ has an open facility with the connection cost at most $3\ opt$. $\mathcal{Q.E.D.}$

Theorem 2.3 Algorithm **Best-or-factor3** runs in time $O(r|C| + |C||F|^2 + |C| \log |C|)$.

Proof. For each $j \in F$ by using a linear time selection algorithm[2], p.220 find the r-th closest customer to j, then choosing closer customers we can compute the set of up to (r-1)-th closest customers to j in $O(|C|)$ time. Thus we need $O(|C||F|)$ time in total to compute such a customer and a set of customers for all

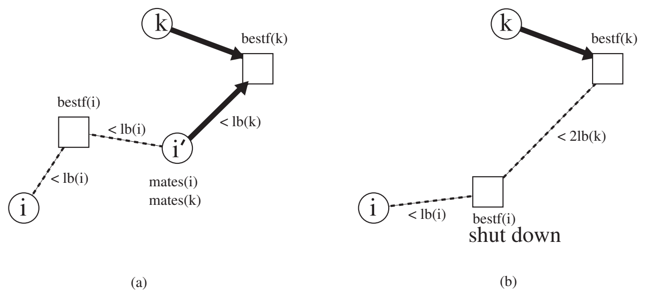

Fig. 2 Illustration for the proof of Theorem 2.2

$j \in F$.

Then we can compute $lb(i), bestf(i), mates(i)$ for all $i \in C$ in $O(|C||F|)$ time. We also compute $co(f, f')$ for every $f, f' \in F$ in $O(|C||F|^2)$ time.

We need $O(|C| \log |C|)$ time for the sort.

Then **Best-Assignment** part runs in $O(r|C| + |C||F|^2)$ time, and **Factor3-Assignment** part runs in $O(|C||F|)$ time.

Thus in total the algorithm runs in $O(r|C| + |C||F|^2 + |C| \log |C|)$ time. $\mathcal{Q.E.D.}$

3. Outlier

An (r, ϵ)-gathering of C to F is an r-gathering of $C - C'$ to F, where C' is any subset of C with size at most $\epsilon|C|$. Intuitively we can ignore at most $\epsilon|C|$ (outlier) customers for the assignment. The cost of an (r, ϵ)-gathering A is defined naturally, that is $\max\{\max_{i \in C-C'}\{co(i, A(i))\}, \max_{j \in F'}\{op(j)\}\}$, where $F' \subset F$ is the set of opened facilities. An (r, ϵ)-gathering problem finds an (r, ϵ)-gathering having the minimum cost.

By slightly modifying algorithm **Best-or-factor3** we can solve the problem as follows. (The modification is similar to Corollary 3.4 of [1] for the r-gathering problem, not for the proximity r-gathering problem.)

After sorting C with respect to $lb(i)$, let i' be the customer having the $\lceil \epsilon|C| \rceil$-th largest lb. We remove all customer i with $lb(i) > lb(i')$ from C. Since we need to assign at least one customer i with $lb(i) \geq lb(i')$ to some open facility, $opt \geq lb(i')$ holds.

Let C' be the set of the removed customers. This removal never affects $mates(i)$ for any remaining $i \in C - C'$, (because assuming $k \in C'$ is in $mates(i)$ for $i \in C - C'$ means $lb(k) \leq lb(i)$, contradicts to the choice of C'). So the removal also never affects $lb(i)$ and $bestf(i)$ for any remaining $i \in C - C'$.

Thus for the remaining customers algorithm **Best-or-factor3** computes an r-gathering with cost at most $3\ lb(i') \leq 3\ opt$. Now we have the following theorem.

Theorem 3.1 One can find an (r, ϵ)-gathering with cost at most $3 \cdot opt$ in $O(r|C| + |C||F|^2 + |C| \log |C|)$ time.

4. *r*-gathering without sort

The following algorithm **Best-or-rest** is a 3-approximate algorithm for the original r-gathering problem which is basically derived from [1] by just removing the sort of C.

We have the following theorems.

Theorem 4.1 The cost of an r-gathering found by Algorithm

Algorithm Best-or-rest

for all $i \in C$ **do**
  Compute $lb(i)$, $bestf(i)$ and $mates(i)$
end for
for all $i \in C$ **do**
  if $bestf(i)$ is not assigned to yet, and all $mates(i)$ are not assigned yet
  then
    Open $bestf(i)$
    for all $k \in mates(i)$ **do**
      Assign k to $bestf(i)$ /* Best-Assignment */
    end for
  end if
end for
for all unassigned $k \in C$ **do**
  Assign k to a closest open facility /* Rest-Assignment */
end for

Best-or-rest is at most $3 \cdot opt$.

Proof. The proof is just a subset of the proof of Theorem2.2.

Consider the cost for each assignment of $i \in C$. For *Best-Assignment* the cost is $lb(i) \leq opt$. So we need to consider only for *Rest-Assignment*.

Each $i \in C$ assigned in *Rest-Assignment* was not assigned to $bestf(i)$ but later assigned to its closest already opened facility. So we consider only for connection costs.

Assume we assign $i \in C$ in *Rest-Assignment*. The reason why i was not assigned in *Best-Assignment* is some $i' \in mate(i)$ is already assigned to some $bestf(k)$ since $i' \in mates(k)$ also holds.

The connection cost $co(i, bestf(k))$ is at most $co(i, bsetf(i)) + co(i', bestf(i)) + co(i', bestf(k)) \leq lb(i) + lb(i) + lb(k)$
$\leq 3\ opt$ Thus $i \in C$ has an open facility with a connection cost at most $3\ opt$. $\mathcal{Q.E.D.}$

We can prove the running time of the algorithm is $O(|C||F| + r|C|)$, by a similar way to the proof of Theorem 2.3. While in [1] the running time was $O(|C||F| + r|C| + |C| \log |C|)$ since it needs a sort of $|C|$.

5. Conclusion

In this paper we provided a simple approximation algorithm to solve the proximity r-gathering problem. The approximation ratio is 3, which improve the former result[1] of 9.

The algorithm can solve a slightly more general problem in which each $f \in F$ has a distinct minimum number r_f of customers needed to open. The algorithm also runs in $O(r|C| + |C||F|^2 + |C| \log |C|)$ time. We assuume $r > r_f$ holds for all $f \in F$.

Can we design an approximation algorithm for the min-sum version of the proximity r-gathering problem?

References

[1] A Armon, On min-max r-gatherings, Theoretical Computer Science, 412, pp.573-582 (2011)
[2] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, Third Edition, The MIT Press (2009)
[3] Z. Drezner, Facility Location: A Survey of Applications and Methods, Springer (1995)
[4] Z. Drezner and H.W. Hamacher, Facility Location: Applications and Theory, Springer (2004)