Improved Bounds for Online Routing and Packing Via a Primal-Dual Approach

In proceeding of: Foundations of Computer Science, 2006. FOCS

Zhang Xiaoxi zxx1121xiaoxi@gmail.com

Online Computation

- Input is revealed in parts
- Oline algorithms respond to each new input upon arrival
- Competitive Ratio c:

Competitive Ratio c:

- A minimization problem I.
- For each instance of I, there is a set of feasible solutions
- For each feasible solution, there is a cost
- OPT(I) is the optimal cost **for an instance** of I

$$\min \sum_{i=1}^{n} c_i x_i$$

$$\sum_{i=1}^{n} a_{ij} x_i \geq b_j, \forall 1 \leq i \leq n, x_i \geq 0$$

c-competitive algorithm:

For every instance of I, the cost is at most c*OPT(I)+a

Competitive Ratio c:

- A minimization optimization problem I.
- For each instance of I, there is a set of feasible solutions
- For each feasible solution, there is a cost
- OPT(I) is the optimal cost for an instance of I

c-competitive algorithm

Minimization Problem:

For every instance of I, the **cost** is at most c*OPT(I)+a

Maximization Problem:

For every instance of I, the **profit** is at least OPT(I)/c-a

How to give the bound of competitive ratio?

We will discuss later

The Online Packing-Covering Framework

(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \leq j \leq m$:	$\sum_{i \in S(j)} x_i \geq 1$	$\forall 1 \leq i \leq n$:	$\sum_{j \mid i \in S(j)} y_j \leq c_i$
$\forall 1 \leq i \leq n$:	$x_i \geq 0$	$\forall 1 \leq j \leq m$:	$y_j \geq 0$

- Online Covering Problem
  - Cost function is known in advance

The Online Packing-Covering Framework

(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \le j \le m$:	$\sum_{i \in S(j)} x_i \ge 1$	$\forall 1 \le i \le n$:	$\sum_{j \mid i \in S(j)} y_j \le c_i$
$\forall 1 \le i \le n$:	$x_i \ge 0$	$\forall 1 \le j \le m$:	$y_j \ge 0$

- Online Covering Problem
  - Cost function is known in advance
  - Linear constraints are given to the algorithm one-by-one

$$\sum_{i \in S(j)} a_{ij} x_i \ge 1$$

$$\begin{matrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{matrix}$$

The Online Packing-Covering Framework

(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \le j \le m$:	$\sum_{i \in S(j)} x_i \ge 1$	$\forall 1 \le i \le n$:	$\sum_{j \mid i \in S(j)} y_j \le c_i$
$\forall 1 \le i \le n$:	$x_i \ge 0$	$\forall 1 \le j \le m$:	$y_j \ge 0$

- Online Covering Problem
  - Cost function is known in advance
  - Linear constraints are given to the algorithm one-by-one
  - The algorithm increase the variables without decreasing any previously increased variables

The Online Packing-Covering Framework

(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \leq j \leq m$:	$\sum_{i \in S(j)} x_i \geq 1$	$\forall 1 \leq i \leq n$:	$\sum_{j \mid i \in S(j)} y_j \leq c_i$
$\forall 1 \leq i \leq n$:	$x_i \geq 0$	$\forall 1 \leq j \leq m$:	$y_j \geq 0$

- Online Packing Problem
  - Values ci are known in advance
  - Profit function and exact packing constraints are not known in advance

The Online Packing-Covering Framework

(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \leq j \leq m$:	$\sum_{i \in S(j)} x_i \geq 1$	$\forall 1 \leq i \leq n$:	$\sum_{j \mid i \in S(j)} y_j \leq c_i$
$\forall 1 \leq i \leq n$:	$x_i \geq 0$	$\forall 1 \leq j \leq m$:	$y_j \geq 0$

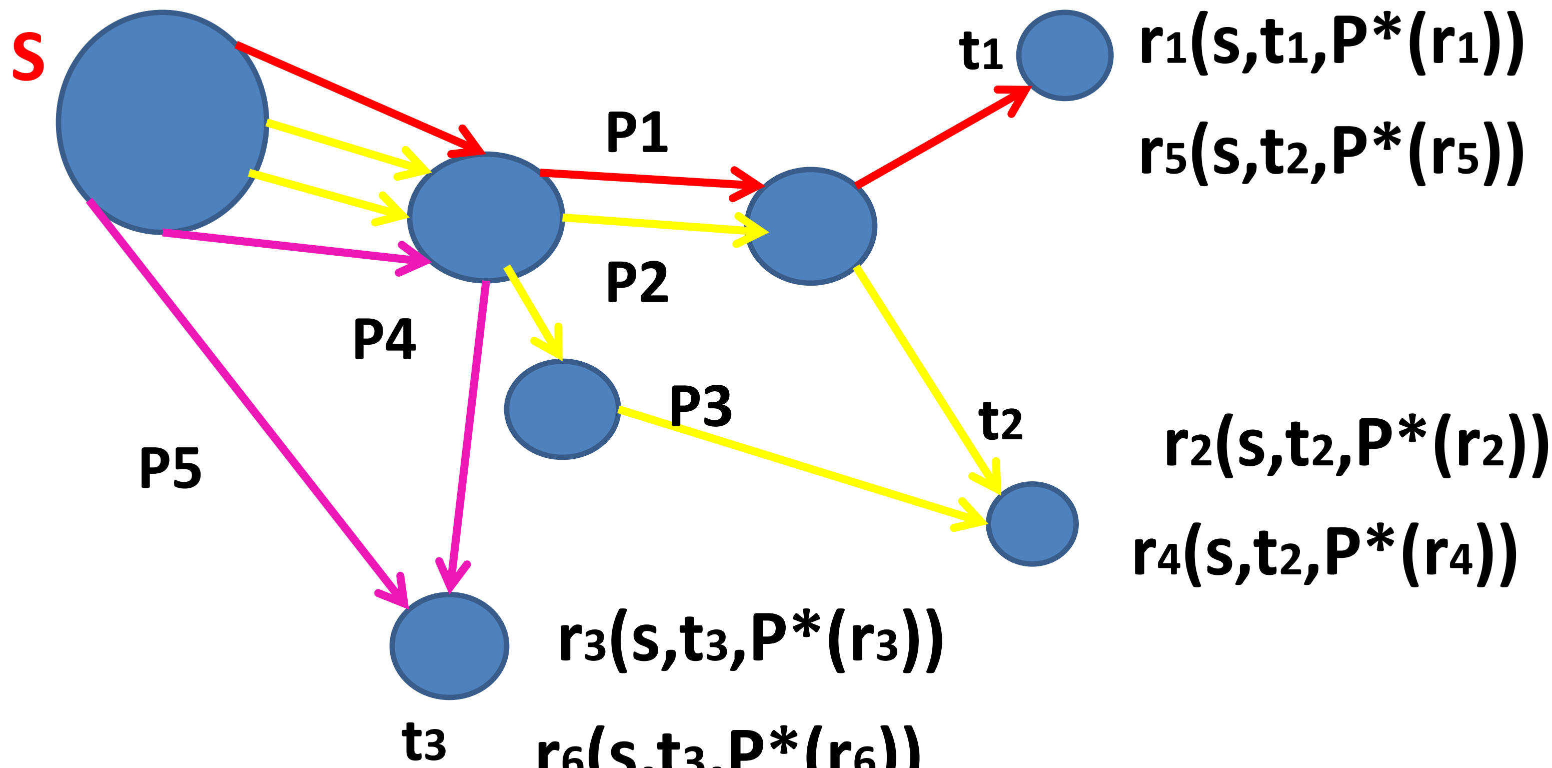
- Online Packing Problem
    - Values ci are known in advance
    - Each packing constraint is revealed gradually

The Online Packing-Covering Framework

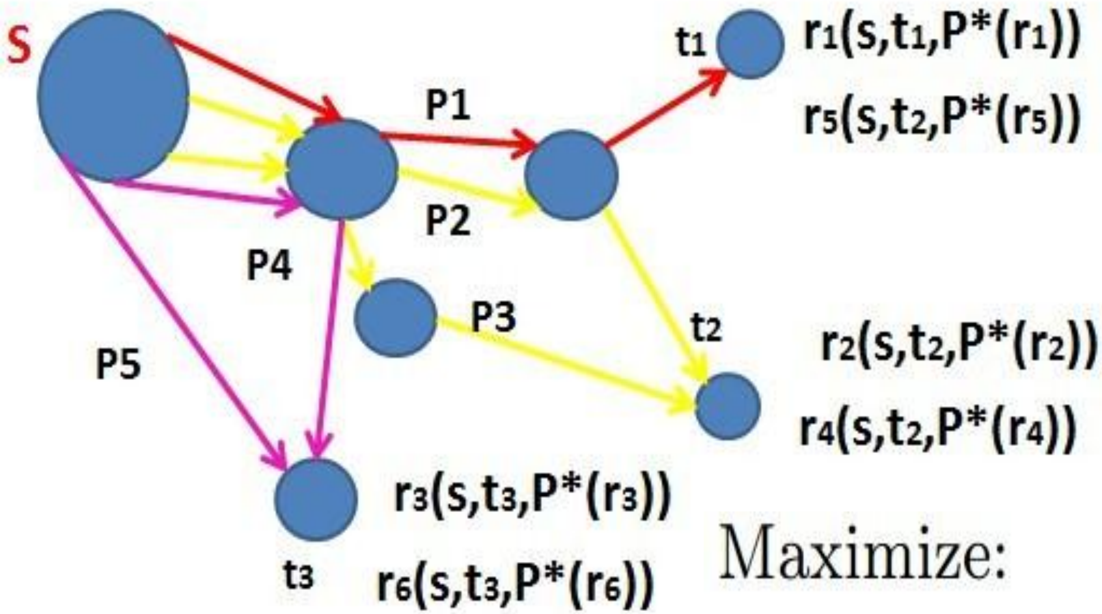
(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \le j \le m$:	$\sum_{i \in S(j)} x_i \ge 1$	$\forall 1 \le i \le n$:	$\sum_{j \mid i \in S(j)} y_j \le c_i$
$\forall 1 \le i \le n$:	$x_i \ge 0$	$\forall 1 \le j \le m$:	$y_j \ge 0$

- Online Packing Problem
  - Values ci are known in advance
  - Each packing constraint is revealed gradually
  - Each y_j is increased in its round without decreasing any previously given variables

Online Routing Model



The splittable routing problem



Maximize: $\sum_{r_i} \sum_{P \in \mathbb{P}(r_i)} f(r_i, P)$

subject to:

$\forall r_i \in \mathbb{R}$: $\sum_{P \in \mathbb{P}(r_i)} f(r_i, P) \leq 1$

$\forall e \in E$: $\sum_{r_i \in \mathbb{R}, P \in \mathbb{P}(r_i) | e \in P} f(r_i, P) \leq u(e)$

$\forall r_i, P : f(r_i, P) \geq 0$

The splittable routing problem (dual) and its corresponding primal problem

Primal	Dual
Minimize: $\sum_{e\in E} u(e)x(e) + \sum_{r_i} z(r_i)$	Maximize: $\sum_{r_i} \sum_{P\in\mathbb{P}(r_i)} f(r_i,P)$
subject to:	subject to:
$\forall r_i \in \mathbb{R}, P \in \mathbb{P}(r_i)$: $\sum_{e\in P} x(e) + z(r_i) \geq 1$	$\forall r_i \in \mathbb{R}$: $\sum_{P\in\mathbb{P}(r_i)} f(r_i,P) \leq 1$
	$\forall e \in E$: $\sum_{r_i\in\mathbb{R}, P\in\mathbb{P}(r_i) \mid e\in P} f(r_i,P) \leq u(e)$
$\forall r_i, z(r_i) \geq 0, \forall e, x(e) \geq 0$	$\forall r_i, P : f(r_i,P) \geq 0$

Primal	Dual
Minimize: $\sum_{e\in E} u(e)x(e) + \sum_{r_i} z(r_i)$	Maximize: $\sum_{r_i}\sum_{P\in\mathbb{P}(r_i)} f(r_i,P)$
subject to:	subject to:
$\forall r_i \in \mathbb{R}, P \in \mathbb{P}(r_i)$: $\sum_{e\in P} x(e) + z(r_i) \geq 1$	$\forall r_i \in \mathbb{R}$: $\sum_{P\in\mathbb{P}(r_i)} f(r_i,P) \leq 1$
	$\forall e \in E$: $\sum_{r_i\in\mathbb{R}, P\in\mathbb{P}(r_i) \mid e\in P} f(r_i,P) \leq u(e)$
$\forall r_i, z(r_i) \geq 0, \forall e, x(e) \geq 0$	$\forall r_i, P: f(r_i,P) \geq 0$

(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \leq j \leq m$:	$\sum_{i\in S(j)} x_i \geq 1$	$\forall 1 \leq i \leq n$:	$\sum_{j \mid i\in S(j)} y_j \leq c_i$
$\forall 1 \leq i \leq n$:	$x_i \geq 0$	$\forall 1 \leq j \leq m$:	$y_j \geq 0$

$$
\begin{aligned}
&\text{Maximize:} \quad \textstyle\sum_{r_i} \sum_{P \in \mathbb{P}(r_i)} f(r_i, P) \\
&\text{subject to:} \\
&\forall r_i \in \mathbb{R}\text{: } \textstyle\sum_{P \in \mathbb{P}(r_i)} f(r_i, P) \leq 1 \\
&\forall e \in E\text{: } \textstyle\sum_{r_i \in \mathbb{R}, P \in \mathbb{P}(r_i) | e \in P} f(r_i, P) \leq u(e) \\
&\forall r_i, P : f(r_i, P) \geq 0
\end{aligned}
$$

Routing algorithm 1:

When a new request $r_i = (s_i, t_i)$ arrives:

(1) if there exists a path $P \in \mathbb{P}(r_i)$ such that $\sum_{e \in P} x(e) < 1$:

   (a) Route the request on P and set $f(r_i, P) \leftarrow 1$.

   (b) Set $z(r_i) \leftarrow 1$.

   (c) For each $e \in P$: $x(e) \leftarrow x(e)(1 + 1/u(e)) + 1/(|P| \cdot u(e))$, where $|P|$ is the length of the path P.

Routing algorithm 1:

When a new request $r_i = (s_i, t_i)$ arrives:

(1) if there exists a path $P \in \mathbb{P}(r_i)$ such that $\sum_{e \in P} x(e) < 1$:

(a) Route the request on P and set $f(r_i, P) \leftarrow 1$.

(b) Set $z(r_i) \leftarrow 1$.

(c) For each $e \in P$: $x(e) \leftarrow x(e)(1 + 1/u(e)) + 1/(|P| \cdot u(e))$, where $|P|$ is the length of the path P.

- Initially, x(e)=0,z(ri)=0
- This algorithm is 3-competitive
- It violates the capacity of each edge by at most a factor of O(log d) (i.e. the load on each edge is at most O(log d) exceeding the capacity of the edge)

Routing algorithm 2:

Initially: $x(e) \leftarrow 0$.

When a new request $r_i = (s_i, t_i, \mathbb{P}(r_i))$ arrives:

(1) If there exists a path $P(r_i) \in \mathbb{P}(r_i)$ of length < 1 with respect to $x(e)$:

  (a) Route the request on "any" path $P \in \mathbb{P}(r_i)$ with length < 1.

  (b) $z(r_i) \leftarrow 1$.

  (c) For each edge e in $P(r_i)$:

$$x(e) \leftarrow x(e) \exp\left(\frac{\ln(1+n)}{u(e)}\right) + \frac{1}{n}\left[\exp\left(\frac{\ln(1+n)}{u(e)}\right) - 1\right].$$

- This algorithm is O(u(min)[exp(ln(1+n)/u(min))-1])-competitive
- If u(min)>=log n then it's O(log n)-competitive
- It does not violate the capacity constraints

(a) Route the request on "any" path $P \in \mathbb{P}(r_i)$ with length < 1.

(b) $z(r_i) \leftarrow 1$.

(c) For each edge e in $P(r_i)$:

$$x(e) \leftarrow x(e) \exp\left(\frac{\ln(1+n)}{u(e)}\right) + \frac{1}{n}\left[\exp\left(\frac{\ln(1+n)}{u(e)}\right) - 1\right].$$

Primal		Dual	
Minimize:	$\sum_{e \in E} u(e)x(e) + \sum_{r_i} z(r_i)$	Maximize:	$\sum_{r_i} \sum_{P \in \mathbb{P}(r_i)} f(r_i, P)$
subject to:		subject to:	

◆Proof.

When an ri is routed, the **increase of the primal cost** is at most:

$$1 + \sum_{e \in P} u(e)\left(x(e)\left[\exp(\ln(1+n)/u(e)) - 1\right] + \frac{1}{n}\left[\exp(\ln(1+n)/u(e)) - 1\right]\right)$$

$$\leq$$

$$2(u(\min)\left[\exp(\ln(1+n)/u(\min)) - 1\right]) + 1$$

While the **increase of the dual profit** is 1

- This algorithm is O(u(min)[exp(ln(1+n)/u(min))-1])-competitive
- If u(min)>=log n then it's O(log n)-competitive
- It does not violate the capacity constraints

- Proof.

When an ri is routed, the **increase of the primal cost** is at most:

$$1 + \sum_{e \in P} u(e)\left(x(e)\left[\exp(\ln(1+n)/u(e)) - 1\right] + \frac{1}{n}\left[\exp(\ln(1+n)/u(e)) - 1\right]\right)$$

$$\leq$$

$$2(u(\min)\left[\exp(\ln(1+n)/u(\min)) - 1\right]) + 1$$

While the **increase of the dual profit** is 1

Thus the ratio between the primal and dual solutions is at most O(u(min)[exp(ln(1+n)/u(min))-1])

How to compute or give a bound for the competitive ratio?

- (1)Compute the ratio B between primal and dual(routing)
- (2)Compute the increment of the cost and profit in primal and dual

Proof.

Primal:=Minimizing problem Dual:=Maximizing problem

Minimum=X (offline) Maximum=Y(offline)

Using weak duality: X (offline) >=Y (offline)

Online version : X(online) & Y(online)

(1)Then if we have X(online)<=BY(online), we'll get:

Y(online)>=X(online)/B>=X(offline)/B>=Y(offline)/B

(2)It's just because the values of the primal and dual solutions are all zero initially.

A New Generic Online Algorithm Using Primal-Dual Approach

- Definition of a (c1,c2)-competitive routing algorithm:
- Routes at least 1/c1 of the maximum possible bandwidth

A New Generic Online Algorithm Using Primal-Dual Approach

- Definition of a (c1,c2)-competitive routing algorithm:
- Routes at least 1/c1 of the maximum possible bandwidth——competitive ratio

A New Generic Online Algorithm Using Primal-Dual Approach

- (c1,c2)-competitive routing algorithm:
- Routes at least 1/c1 of the maximum possible bandwidth——competitive ratio
- Guarantees that the load on each edge is at most c2

A New Generic Online Algorithm Using Primal-Dual Approach

- Guarantees that the load on each edge is at most c2——violate the capacity constraint

(P): Primal (Covering)		(D): Dual (Packing)	
Minimize:	$\sum_{i=1}^{n} c_i x_i$	Maximize:	$\sum_{j=1}^{m} y_j$
subject to:		subject to:	
$\forall 1 \leq j \leq m$:	$\sum_{i \in S(j)} x_i \geq 1$	$\forall 1 \leq i \leq n$:	$\sum_{j \mid i \in S(j)} y_j \leq c_i$
$\forall 1 \leq i \leq n$:	$x_i \geq 0$	$\forall 1 \leq j \leq m$:	$y_j \geq 0$

A New Generic Online Algorithm Using Primal-Dual Approach

- Routes at least 1/c1 of the maximum possible bandwidth
- Guarantees that the load on each edge Is at most c2

- Our Algorithm is (1,O(logn))-competitive
- Advantage : competitive ratio is small

A Generic Online Routing Algorithm

Initially, $\forall j$: $x(e,j) \leftarrow u(\min,j)/m \cdot u(e,j)$.
When new request $r_i = (s_i, t_i, \mathbb{P}(r_i))$ arrives:

(1) Consider all copies of G from G_k to G_0. In each copy G_j:

   (a) Let $P(r_i,j) \in \mathbb{P}(r_i,j)$ be the shortest path with respect to $x(e,j)$ and let α be the length of $P(r_i,j)$.

   (b) If $\alpha < 1$:

      (i) Route the request on $P(r_i,j)$.

      (ii) For each edge e in $P(r_i,j)$: $x(e,j) \leftarrow x(e,j)(1 + 1/u(e,j))$.

      (iii) $z(r_i,j) \leftarrow 1 - \alpha$.

   (c) Else ($\alpha > 1$):

      (i) If the total bandwidth routed in this step in G_j is less than $u(\min,j)$, and the current request can be routed in G_j, route the request in an arbitrary feasible path $P \in \mathbb{P}(r_i,j)$.

   (d) If the request is routed — finish.

(2) Reject requests that got rejected from all copies.

When new request $r_i = (s_i, t_i, \mathbb{P}(r_i))$ arrives:

(1) Consider all copies of G from G_k to G_0. In each copy G_j:

  (a) Let $P(r_i, j) \in \mathbb{P}(r_i, j)$ be the shortest path with respect to $x(e, j)$ and let α be the length of $P(r_i, j)$.

  (b) If $\alpha < 1$:

    (i) Route the request on $P(r_i, j)$.

    (ii) For each edge e in $P(r_i, j)$: $x(e, j) \leftarrow x(e, j)(1 + 1/u(e, j))$.

    (iii) $z(r_i, j) \leftarrow 1 - \alpha$.

Decompose G(V,E) into graphs G0,G1,…,Gk
For each j of 0~k, the vertex set of Gj is V
The edges in Gj are those of G having capacity at least m^j

(b) If $\alpha < 1$:

(i) Route the request on $P(r_i, j)$.

(ii) For each edge e in $P(r_i, j)$:

$x(e, j) \leftarrow x(e, j)(1 + 1/u(e, j))$.

(iii) $z(r_i, j) \leftarrow 1 - \alpha$.

When a request is routed in jth copy,
the total primal value in jth copy increases by

$$(1 - \alpha) + \sum_{e \in P(r_i, j)} x(e, j) = 1 - \alpha + \alpha = 1$$

And the dual profit increase 1 too.
So its competitive ratio is 1

- One weakness:
- the duration of the request it doesn’t consider
- ——B. Awerbuch, Y. Azar, and S. Plotkin. Thoughtput-competitive online routing. In Proc. Of 34^{th} FOCS, page 32-40, 1993

My future work

- Study some details to construct ‘half’ problem based on the other half
- Study how to adjust my algorithm(competitive ratio is not good) if I even don’t know the physical meaning of the variables in the other half problem
- How to make the math model more delicate if the constraints in real problem are so many

- Thank you!
- Q&A