Cloud Computing Problem Formulation

November 3, 2016

1 Problem Formulation

1.1 Center

$\vec{\mathbf{O}(t)}$ = all orders sequence in the queue in time slot t, e.g.$(o_i, o_{i+1}, ..., o_m)$
$\vec{a(t)}$ = new arrival orders sequence in time slot t, e.g.$(o_j, o_{j+1}, ..., o_m)$
$\mathbf{O}(\vec{t+1}) = \vec{\mathbf{O}(t)} + \vec{a(t)} - \sum_{i=1}^{n} \vec{r_i(t)}$

1.2 Locker

$\vec{\mathbf{u}_i}$ = available delivers sequence in locker i, e.g.$(u_i, u_{i+1}, ..., u_m)$
$\vec{\mathbf{o}_i(t)}$ = all orders sequence in the queue in time slot t for locker i, e.g.$(o_i, o_{i+1}, ..., o_m)$
$\vec{r_i(t)}$ = new arrival orders sequence for locker i in time slot t, e.g.$(o_j, o_{j+1}, ..., o_m)$
$\vec{h_i(t)}$ = orders delivered for locker i in time slot t
$\mathbf{o}_i(\vec{t+1}) = \vec{\mathbf{o}_i(t)} + \vec{r_i(t)} - \vec{h_i(t)}$
$\vec{h_i(t)} = \sum_{u \in \vec{\mathbf{u}_i}} \vec{b_u(t)}$

1.3 Deliver

$\lambda_u(t)$ = is deliver u active in time slot t, 0,1
l_u = locker position for deliver u
ρ_u = origin position for deliver u
$\vec{b_u(t)}$ = orders delivered by u in time slot t, e.g.$(o_i, o_{i+1}, ..., o_m)$
d_u^{max} = max distance for deliver u per time slot
c_u^{max} = max capacity of deliver u per time slot
$d_u(t)$ = distance for deliver u in time slot t
$c_u(t)$ = capacity for deliver u in time slot t
$\vec{v}_u(t)$ = paths for deliver u in time slot t, e.g. $(\rho_u, l_u, p_0, p_1, .., p_n, \rho_u)$, where $p_i \in \mathbf{V}$
$\mathbf{E}_u(t)$ = edges for deliver u in time slot t, and $\mathbf{E}_u(t) \subseteq \mathbf{E}$
$d_u(t) = \sum_{(u,v) \in \mathbf{E}_u(t)} w(u, v)$
$\sum_{o \in \vec{b_u(t)}} g_o \leq c_u^{max}$
$\sum_{o \in \vec{b_u(t)}} \mu_o \subseteq \vec{v}_u(t)$

1.4 Order

g_o = the capacity for order o
σ_o = deadline of order o
μ_o = destination address of order o

1.5 Map

$\mathbf{V}$ = nodes in the city
$\mathbf{E}$ = edges in the city
$w(u, v)$ = the distance between position u and position v

1.6 Payment

β = constant commision fee
γ = payment for unit distance $\delta(t)$ = payment for time slot t
Δ = the whole payment
$\delta(t) = \sum_{u=1}^{n} \lambda_u(t) \cdot (\gamma d_u(t) + \beta)$
$\Delta = \sum_{t=1}^{T} \delta(t)$

1.7 Constraints

$$\sum_{o \in b_u\vec{(t)}} g_o \leq c_u^{max}$$

$$\sum_{o \in b_u\vec{(t)}} \mu_o \subseteq \vec{v}_u(t)$$

$$\forall o \in h_i\vec{(t)}, t \leq \sigma_o$$