Algorithm 2 Inner loop algorithm

Input : $EnaD, PS$

Output : $\theta_{i,j}, P_{i,j}^{D}, EE_{i,j}^{D}$

$for\ i \in EnaD\ do$

$for\ k \in PS_i^D\ do$

Calculate $P_{i,max}^{R}$ using (5), and obtain N_{max} using function A

$for\ j = 1\!:\!N_{max}\ do$

Initialize P_i^D as $P_i^D(0), G_i^D$ as $G_i^D(0)$

Initialize t=0, I, ϕ

obtain $\theta_i(t)$ by calculating (14) using $P_i^D(0)$

$while\ t < I$

Obtain $P_i^D(t+1)$ using $\theta_i(t)$ to calculate (15)

Obtain $\theta_i(t+1)$ using $P_i^D(t+1)$ to calculate (14)

$if\ T_i^D - G_i^D(t) EC_i^D < \phi\ then$

$\theta_{i,j} = \theta_i(t+1), P_{i,j}^D = P_i^D(t+1), EE_{i,j}^D = G_i^D(t)$

$break\ iteration$

$else$

Update all the Lagrange multipliers using (16)

$G_i^D(t+1) = \frac{T_i^D\left(\theta_i(t+1), P_i^D(t+1)\right)}{EC_i^D\left(\theta_i(t+1), P_i^D(t+1)\right)},\ continue$

$endif$

$t = t + 1$

$end\ while$

$end\ for$

$end\ for$

$end\ for$

Algorithm 1 Pre-Matching Algorithm

Input : $D, C, P_k^C, P_{thresold}^1, P_{max}, T_{min}^D$

Output : $PS, InfD, EnaD$

Initialize $EnaD = \emptyset, InfD = \emptyset$

for $i \in D$ do

  $PS_i^D = C$

  for $k \in C$ do

    obtain $\theta_{i,min}$ using (8), obtain $T_{i,max}^D$ using (7)

      if $\theta_{i,min} \geq 1$ or $T_{i,max}^D \leq T_{min}^D$ then

        Remove current k from PS_i^D

      end if

    end for

    if $PS_i^D = \emptyset$ then

      add i to $InfD$

    elseif $PS_i^D \neq \emptyset$ then

      add i to $EnaD$

    end if

    add PS_i^D to PS

end for

Algorithm 4 Preference list

Input : $P_i^{D*}, EE_i^{D*}, \theta_i^*, EnaD, C, PS$

Output : Ω_i^D, Ω_k^C

1: $for\ i \in EnaD\ do$

2: $for\ k \in PS_i^D\ do$

3: $get\ EE_i^{D*}$ **and place k into** Ω_i^D **in descending order of** EE_i^{D*}

4: $endfor$

5: $endfor$

6:

7: $for\ k \in PS_i^D\ do$

8: $for\ i \in EnaD\ do$

9: **obtain** $P_{interference}$ **caused by i using (17), and place i into** Ω_k^c **in ascending order of** $P_{interference}$

10: $endfor$

11: $endfor$

Simulation parameter	Value
Radius of Cellular Network	250m
Number of D2D links N	$10 \sim 70$
Number of CUE links M	$10 \sim 70$
D2D communication distance	$10 \sim 70m$
Path loss exponent	3
EH power segment $[Pth_1\ Pth_2\ Pth_3\ Pth_4]$	[10 100 230.06 57368] uw
EH linear function coefficient $[k_1\ k_2\ k_3\ k_4]$	[0 0.3899 0.6967 0.1427]
EH linear function intercept $[b_1\ b_2\ b_3\ b_4]$	[0 -1.6613 -19.1737 108.2778]
Maximum harvested power $P_{max}^{harvested}$	250uw
Max transmission power P_{max}	23dBm
Circuit power consumption P_{cir}	20dBm
CUE transmission power P_k^C	23dBm
Initial Lagrange multipliers $\alpha, \beta, \gamma, \delta, \sigma$	0.1
Noise power N_0, N_1	-100dBm
Throughput requirement of D2D link i T_{min}^D	2bits/Hz
Throughput requirement of CUE k T_{min}^C	1bit/Hz

Algorithm 5: One-to-one Stable Matching Algorithm

Input : $\boldsymbol{\Omega_i^D, \Omega_k^C, EnaD, C, PS}$

Output : $\boldsymbol{\phi, \phi^R}$

Step 1 : $\boldsymbol{while\ EnaD \neq \phi\ do}$

Step 2 : $\boldsymbol{for\ i \in EnaD\ do}$

Step 3 : **Let $\boldsymbol{i}$ propose its most preferred CUE $\boldsymbol{\Omega_i^D}$ which should be the first CUE in $\boldsymbol{\Omega_i^D}$**

Step 4 : $\boldsymbol{for\ k \in PS}$

Step 6 : $\boldsymbol{if}$ **k is the most preferred CUE for i** $\boldsymbol{then}$

Step 7 : $\boldsymbol{if\ k}$ **receives only one proposal from D2D link $\boldsymbol{i}$ then**

Step 8 : $\boldsymbol{i}$ **will be matched with $\boldsymbol{k}$,$\boldsymbol{\phi = (i, k)}$**

Step 9	:	The matched i will be removed form $EnaD$ $break$
Step 10	:	$elseif$ k receives not only one proposal from different D2D links $then$
Step 11	:	if one of the D2D links α has only one preferred CUE in its preference list $then$
Step 12	:	α will be matched with k, $\phi = (i, k)$, it will be removed from $EnaD$
Step 13	:	$elseif$ all the D2D links have more than one preferred CUE in their preference lists $then$
Step 14	:	The CUE k will be matched with its most preferred D2D link i' from its preference list Ω_k^C, and the matched D2D link will be removed form $EnaD$
Step 15	:	$end\ if$
Step 15	:	$end\ if$
Step 16	:	$elseif\ \ k\ is\ the\ most\ preferred\ CUE\ for\ i\ \ then$
Step 17	:	$continue$
Step 18	:	$end\ if$
Step 19	:	$end\ for$
Step 20	:	i deletes its most preferred CUE from Ω_i^D
Step 21	:	$end\ for$
Step 22	:	$end\ while$
Step 23	:	Gather all the unmatched CUEs in ϕ_R

Algorithm 3 Out loop Algorithm

Input :$EnaD, PS, \theta_{i,j}, P_{i,j}^D, EE_{i,j}^D$

Output :$P_i^{D*}, \theta_i^*, EE_i^{D*}$

1: $for\ i \in EnaD$

2: $for\ k \in PS_i^D$

3: Obtain $P_{i,max}^R$ using (5), and obtain N_{max}

$\boldsymbol{for\ j = 1: N_{max}}$

$\boldsymbol{j^* = \arg\max \{EE^D_{i,1}, EE^D_{i,2}, \dots, EE^D_{i,j}, \dots EE^D_{i,N_{max}}\}}$

Assign $\boldsymbol{P_i^{D*} = P^D_{i,j^*}\ EE_i^{D*} = EE^D_{i,j^*}, \theta_i^* = \theta_{i,j^*}}$

$\boldsymbol{end\ for}$

$\boldsymbol{end\ for}$

$\boldsymbol{end\ for}$