Throughput Maximisation by maximizing data & check to wait :

Begin

S ← { S_0 }

Max_data ← 0 ;

/* *next_soj_location* with the maximum volume of data collected . */

Repeat

For each feasible location sj ∈ F **do**

- Sort in survival time sequence in decreasing order
- Find the max term from the **Data** sequence

Data = { (t_1.1.r_g) , (t_2.2.r_g) , …. , (t_k.k.r_g) , ….. , (t|N(sj)|.|N(sj)|.r_g) }

$D(s_j)$ ← (t_k.k.r_g) ; /*assumed maximum*/

If Max_data < $D(s_j)$

Temp_soj_location ← s_j ;

temp_soj_time ← t_k ;

Max_data = $D(s_j)$;

Prev_loc = S(end);

For location Prev_loc **do**

- Calculate time to travel from Prev_loc to *Temp_soj_location* and harness energy for all sensors during this time.
- Sort in survival time sequence in decreasing order
- Find the max term from the **Data** sequence

Data = { (t_1.1.r_g),(t_2.2.r_g),….,(t_k.k.r_g),…..,(t|N(Prev_loc)|.|N(Prev_loc)|.r_g) }

D(Prev_loc) ← (t_k.k.r_g) ; /*assumed maximum*/

Time_prev = t_k ;

If Max_data <= D(Prev_loc)

Next_soj_location ← Prev_loc ;

Next_soj_time ← Time_prev ;

Else

Next_soj_location ← *Temp_soj_location* ;

Next_soj_time ← *temp_soj_time* ;

S ← S U { *Next_soj_location* } ;

Max_data ← 0 ;

Update the energy of sensors ;

Until there is no more feasible sojourn location ;

End

Throughput Maximisation by maximizing gain & check to wait :

Begin

$S \leftarrow \{ S_0 \}$

Max_data ← 0 ;

/* *next_soj_location* with the maximum volume of data collected . */

Repeat

For each feasible location $s_j \in F$ **do**

- Sort in survival time sequence in non-increasing order
- Find the max term from the **Data** sequence

Data = { ($t_1.1.r_g$) , ($t_2.2.r_g$) , ….. , ($t_k.k.r_g$) , ….. , ($t|N(sj)|.|N(sj)|.r_g$) }

$D(s_j) \leftarrow (t_k.k.r_g)$; /*assumed maximum*/

$\Delta t(s_j) \leftarrow T_j + t_{i,j} + t_{i,0} - t_{j,0}$ /*time cost factor*/

$G(s_j) \leftarrow D(s_j) \div \Delta t(s_j)$

If Max_Gain < $G(s_j)$ /*Choosing location with **maximum data gain per time-unit***/

Temp_soj_location ← s_j ;

Temp_soj_time ← t_k ;

Max_data = $D(s_j)$;

Prev_loc = S(end);

For location Prev_loc **do**

- Calculate time to travel from Prev_loc to s_j and harness energy for all sensors during this time.
- Sort in survival time sequence in decreasing order
- Find the max term from the **Data** sequence

Data = {($t_1.1.r_g$),($t_2.2.r_g$),….,($t_k.k.r_g$),…..,($t|N(Prev\_loc)|.|N(Prev\_loc)|.r_g$)}

$D(Prev\_loc) \leftarrow (t_k.k.r_g)$; /*assumed maximum*/

Time_prev = t_k ;

If Max_data <= D(Prev_loc)

Next_soj_location ← Prev_loc ;

Next_soj_time ← Time_prev ;

Else

Next_soj_location ← *Temp_soj_location* ;

Next_soj_time ← *temp_soj_time* ;

S ← S U { *Next_soj_location* } ;

Max_Gain ← 0 ;

Update the energy of sensors ;

Until there is no more feasible sojourn location ;

End

Last Location algorithm:

Begin

Max_data ← 0 ;

For each location sj ∈ F **do**

Compute Δt_j ;

If $\Delta t_j > 0$ **then**

Generate the value sequence :

$(t_{i1.j}.1.r_g) , (t_{i2.j}.2.r_g) , ... , (t_{ik.j}.k.r_g), ... ,(t_{i|N(sj)|.j}.|N(sj)|.r_g)$;

Identify a maximum term from the terms whose survival times are less than Δt_j , and let i_l be the index of the maximum term ;

Identify the terms from the sequence whose survival times are strictly greater than Δt_j , and let i_k be the maximum index of all such terms;

If $(t_{il.j}.|N(sj,i_l)|.r_g >= \Delta t_j.k.r_g)$ **then**

$T_j \leftarrow t_{il.j}.N_s(sj) \leftarrow N(sj,i_l)$;

$D(s_j) \leftarrow t_{il.j} . l . r_g$;

Else

$T_j \leftarrow \Delta t_j$ **;**

$N_s(s_j) \leftarrow N(s_j , i_k)$;

$D(s_j) \leftarrow \Delta t_j . k . r_g$;

If *Max_data* < $D(s_j)$ **then**

Next_soj_location ← s_j ;

Next_sensor_set ← $N_s(s_j)$;

Next_soj_time ← T_j ;

Max_data ← $D(s_j)$;

Return *Next_soj_location , Next_soj_time and Next_sensor_set* ;

End