This paper includes an errata for the Delay Efficient Sleep Scheduling (DESS) problem presented in [1]. This errata corrects our main complexity result mentioned in [1]. We first define the decision problem G-DESS which is a more general case of DESS.

***Definition** 1:* **G-DESS**(G, S, k, Δ): Given a graph $G = (V, E)$, a set S of pairs (s_i, t_i) such that $s_i, t_i \in V$, number of slots k, and a positive number Δ, does there exist a slot assignment function $f : V \to [0, \cdots k-1]$, such that $\max_i\{d_f(s_i, t_i)\} = \Delta$?

We now prove the main complexity result:

***Theorem** 1:* G-DESS(G, S, k, Δ) is NP-Complete.

Proof: Given the slot assignment function f, one can compute the shortest delay path from each node to all the other nodes in polynomial time. Moreover, there are only a polynomial number of such nodes. The maximum delay among all the pairwise $s_i \to t_i$ paths should then be compared against Δ. All these steps can be done in polynomial time. Thus G-DESS$(G, S, k, \Delta) \in NP$.

To prove that G-DESS(G, S, k, Δ) is NP-Hard, we propose the following polynomial time reduction from 3-CNF-SAT to G-DESS.

Consider a 3-CNF formula F consisting of n variables and m clauses i.e. $F = C_1 \bigwedge C_2 \bigwedge \cdots C_m$, where each $C_i = l_{i1} \bigvee l_{i2} \bigvee l_{i3}$ and $l_{ij} \in \{x_1, \bar{x_1}, \cdots x_n, \bar{x_n}\}$.

Construct the following graph $G = (V, E)$:

1) For each clause C_i, add nodes s_i, t_i to V. For each variable x_i, add nodes X_i, X_i' to V.
2) For each variable x_i, add (X_i, X_i') to E.
3) For each clause C_i, if literal x_j is present in clause C_i, add (s_i, X_j) and (X_j', t_i) to E. If literal $\bar{x_j}$ is present in clause C_i, add (s_i, X_j') and (X_j, t_i) to E.
4) $\forall i < n$ add an edge (t_i, t_{i+1}) to E.

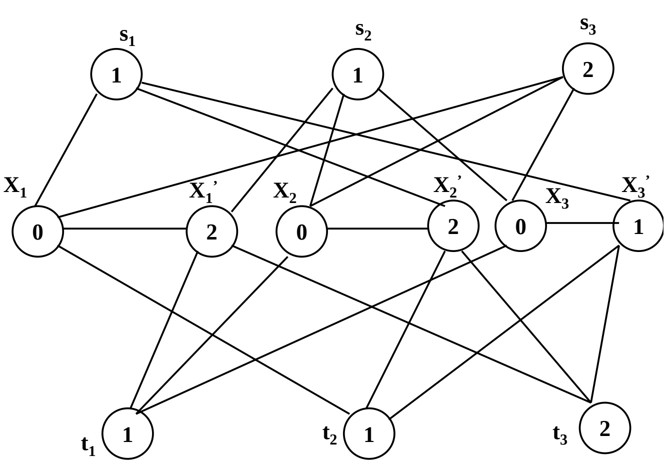

Fig. 1. Reduction from 3-CNF-SAT to G-DESS$(G, S, 3, 3)$. Here, $F = (x_1 \bigvee \bar{x_2} \bigvee \bar{x_3}) \bigwedge (\bar{x_1} \bigvee x_2 \bigvee x_3) \bigwedge (x_1 \bigvee x_2 \bigvee x_3)$. The satisfying assignment is $x_1 = 0$, $x_2 = 0$ and $x_3 = 1$.

Let number of slots $k = 3$, $\Delta = 3$ and $S = \{(s_1, t_1), (s_2, t_2), ...(s_n, t_n)\}$. Note that in the constructed graph, the number of hops between any s_i and t_i is also 3. An example reduction in shown in figure 1.

We first prove that if the given formula is satisfiable, there exists a slot assignment function f such that G-DESS$(G, S, 3, 3)$ is true. i.e. there exists an slot assignment function such that $\max_i\{d_f(s_i, t_i)\} = 3$.

For every variable, if x_j is true in the satisfying assignment, let $f(X_j) = 0$ and $f(X_j') = 1$. If x_j is false, let $f(X_j) = 0$ and $f(X_j') = 2$.

i.e. if x_j is true, $d_f(X_j, X_j') = 1$ and $d_f(X_j', X_j) = 2$, otherwise $d_f(X_j, X_j') = 2$ and $d_f(X_j', X_j) = 1$.

Since the formula is satisfiable each clause C_i will have at least one true literal. Pick one of these true literals. If this literal is x_j, let $f(s_i) = 2$ and $f(t_i) = 2$, then $d_f(s_i \to t_i) = d_f(s_i, X_j) + d_f(X_j, X_j') + d_f(X_j', t_i) = 1 + 1 + 1 = 3$. If the literal is $\bar{x_j}$, let $f(s_i) = 1$ and $f(t_i) = 1$, then $d_f(s_i \to t_i) = d_f(s_i, X_j') + d_f(X_j', X_j) + d_f(X_j, t_i) = 1 + 1 + 1 = 3$.

Thus there exists a slot assignment function such that $\max_i\{d_f(s_i, t_i)\} = 3$.

We next prove that if there exists an f such that G-DESS$(G, S, 3, 3)$ is true, then the given formula is satisfiable.

Consider each clause C_i. There exists a path $s_i \to t_i$, such that $d_f(s_i, t_i) = 3$. There are two possibilities for this path:

1) The path is $s_i \to X_j \to X_j' \to t_i$ for some variable x_j. Thus x_j is present in clause C_i. Moreover, since $d_f(s_i, t_i) = 3$, it must be the case that the delay along each of these edges is 1. In this case, let $x_j =$ true.
2) The path is $s_i \to X_k' \to X_k \to t_i$ for some x_k. In this case $\bar{x_k}$ is present in C_i. Moreover, since $d_f(s_i, t_i) = 3$, it must be the case that the delay along each of these edges is 1. In this case, let $x_k =$ false.

We now show that this proposed truth assignment is satisfying and consistent.

For each clause C_i, one of the above 2 possibilities exist. For the first possibility, a literal x_j is present in C_i and we assign x_j to be true. For the second possibility, a literal $\bar{x_k}$ is present in C_i and we assign x_k to be false. Thus in either case, each clause has at least one true literal and hence the proposed truth assignment is satisfying.

To prove consistency of the truth assignment, for a given variable x_i, consider any 2 source destination pairs (s_j, t_j) and (s_k, t_k) that use X_i and X_i' as intermediate vertices on their shortest delay path. We claim that both these shortest delay paths must traverse the edge (X_i, X_i') in the same direction. i.e. their shortest delay paths are either $s_j \to X_i \to X_i' \to t_j$ and $s_k \to X_i \to X_i' \to t_k$ respectively or $s_j \to X_i' \to X_i \to t_j$ and $s_k \to X_i' \to X_i \to t_k$ respectively. If this was not the case, without loss of generality assume that one of the shortest delay paths is $s_j \to X_i \to X_i' \to t_j$ and the other is $s_k \to X_i' \to X_i \to t_k$. Since both paths have a delay of 3, it must be the case that $d_f(X_i, X_i') = d_f(X_i', X_i) = 1$. However for $k = 3$ slots, there exists no slot assignment that gives a unit delay along each direction for any edge $(z, w) \in G$. Thus, all source destination pairs (s_j, t_j) that use the edge (X_i, X_i') for the shortest delay path must traverse the edge in the same direction i.e. either $s_j \to X_i \to X_i' \to t_j$ or $s_j \to X_i' \to X_i \to t_j$. In the first case, the variable x_i is assigned true and in the second, x_i is assigned false. Thus, the proposed truth assignment is consistent.

∎

I. Acknowledgements

We thank Matthew J. Miller from University of Illinois for pointing out errors in the proof of NP-Completeness in [1].

References

[1] G. Lu, N. Sadagopan, B. Krishnamachari and A. Goel, “Delay Efficient Sleep Scheduling in Wireless Sensor Networks,” in *IEEE Infocom*, 2005.