EDGE CONTROLLER PLACEMENT FOR NEXT GENERATION WIRELESS SENSOR NETWORKS

Mohamed S. Abouzeid, Heba A. El-khobby,
Mahmoud A. A. Ali and Mohamed E. Nasr

Electronics and Electrical Communication Department,
Faculty of Engineering-Tanta University, Tanta, 31527, Egypt

ABSTRACT

Nowadays, Fog architecture or Edge architecture is becoming a popular research trend to distribute a substantial amount of computing resources, data processing and resource management at the extreme edge of the wireless sensor networks (WSNs). Industrial communication is a research track in next generation wireless sensor networks for the fourth revolution in the industrial process. Adopting fog architecture into Industrial communication systems is a promising technology within sensor networks architecture. With Software Defined Network (SDN) architecture, in this paper, we address edge controller placement as an optimization problem with the objective of more robustness while minimizing the delay of network management and the associated synchronization overhead. The optimization problem is provided and modelled as submodular function. Two algorithms are provided to find the optimal solution using a real wireless network to get more realistic results. Greedy Algorithm and Connectivity Ranking Algorithm are provided. Greedy algorithm outperforms connectivity ranking algorithm to find the optimum balance between the different metrics. Also, based on the network operator preference, the number of edge controllers to be placed will be provided. This research paper plays a great role in standardization of softwarization into Industrial communication systems for next generation wireless sensor networks.

KEYWORDS

Fog Architecture, Submodularity, Software Defined Network, Controller Placement, Virtual Process Function.

1. INTRODUCTION

Recently, for next generation wireless sensor networks, there are critical requirements such as increasing the operational efficiency of industrial control process. In addition, improving the effectiveness of operational activities is an increasing demand for various industries today. Furthermore, more flexible operations should be supported with the reduction of capital expenditure [1]. Thus, softwarization is a key enabler to achieve these facilities. More specifically, Software Defined Network (SDN) is a promising architecture for next generation wireless sensor networks [2]. Healthcare systems and the automotive industry are simple examples of industrial communication systems in WSNs. Legacy industrial control process includes real time data collection form sensors, after that data processing is performed in hardware controllers and finally execution of control commands through actuators [3-5]. With SDN, softwarization can embrace the control process in an entity. Thus, the hardware controllers can be replaced by software instances [6]. There are some advantages of SDN. One of them is that the control process function can be placed in a commodity server which can be flexibly provisioned on demand [7]. In addition, they have a good ability of upgrading in amore simple way than hardware controller.

With the introduction of Mobile Edge Computing (MEC), next generation wireless sensor networks can distribute substantial amount of computing and storage resources to edge nodes at the extremes of the network with low latency and high bandwidth [8]. The critical aspects for adoption of MEC into next generation wireless sensor networks are the robustness of the wireless communication links, the minimized delay of network management and the associated overhead not only between controllers and data-plane nodes, but also, between the controllers themselves (Inter-controller communication) [9]. One way to improve the reliability of next generation wireless sensor networks is to place the virtual process function (VPF) of the controller on another edge node in case of node or link failure [10]. However, this process is limited because of the resource constraints of the edge nodes besides their limited power and storage resources [11].

In this paper, we address the problem of placing edge controllers in next generation wireless sensor networks with the objective of minimizing the network management delay, overhead control messages and invalid control paths which increases the robustness of the system. We formulate an optimization problem for placing the virtual process function into the edge nodes and taking into consideration the questions: How many controllers should be placed at the edge of the network in close proximity to the end sensors and actuators. Also, where exactly should be the controllers placed to improve the resilient of the system. The objective is to minimize the capital expenditure of next generation wireless sensor networks, while increasing its flexibility.

2. RELATED WORK

There has been a significant interest in cloud computing for industrial use cases, both concerning potential application areas and security aspects [12], [13], but the focus of these works is on the architectures and requirements, rather than resource management.

There are recent works on resource allocation in MEC for sensor networks [14]. The problem of allocating visual sensors with correlated measurements to computing resources to maximize system capacity is introduced in [8]. In [15], the authors propose the problem of allocating health sensors to health cloud servers to maximize system utility. The authors in [16] modelled the individual Virtual Network Function (VNF) placement problem as a generalized assignment problem where the controllers assigned to different nodes to minimize the total assignment cost. The placement of chains of VNF is modelled in [17] as an Integer Linear Problem (ILP). The problem of controller placement to maximize the resilience under link or node failure is introduced as ILP and numerical results are provided in [18].

To the best of our knowledge, our work is the first work to handle the edge controller placement problem as an optimization problem for industrial use case. The optimization problem considers resilience constraints besides the delay of network management and the associated overhead to achieve the optimal placement of the controller in SDN industrial communication system for next generation wireless sensor networks. This research work considers minimizing the delay and control overhead which plays a great role in minimizing the operating expenses. This work also reduces the down time of the network by maximizing the reliability performance of WSNs. The novelty of our work appears by applying submodularity conditions to the optimization performance of next generation wireless sensor networks.

3. SYSTEM MODEL

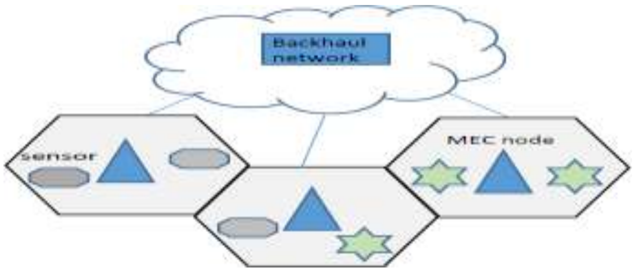

Figure 1. The communication infrastructure consists of BSs, MEC nodes and sensorsand backhaul network.

We consider a system that consists of a set of B Base stations (BSs). A subset N node are equipped with computational and storage resources and serves as edge nodes. We denote by K a subset of N that can host control process function such that K $\subseteq$ N .The base stations are interconnected by a backhaul network. Sensors and Internet of Things (IoT) devices communicate wirelessly within the base station. The failure of an edge node or the wireless communication link result in failure scenario. This failure leads to that the edge node becomes unsuitable to host the virtual process function (VPF) [19]. We assume that the network operator is able to estimate the occurrence probability of each failure scenario. We denote by πl the failure occurrence probability. It is assumed that Generalized pareto distribution is used to estimate such failure occurrence probability. Generalized pareto distribution is used because it provides Independent Identical Distributed (IID) probability for failure scenarios.

Each pair of controllers are communicated with each other, for synchronization purposes, using messages. These messages are exchanged at a constant rate besides other exchanged at a rate that depends on the controller load. When considering SDN enabled edge network, the interaction between a controller and data-plane nodes takes place through Openflow protocol. Heartbeat messages are periodically transferred between the controller and data plane nodes, besides static request/reply messages. Thus, as the network scales up, the overhead grows. For synchronization purposes between a cluster of controllers, there is a significant overhead that should be considered in modelling the edge controller placement problem for industrial use cases.

4. PROBLEM FORMULATION

In this section, we formulate the problem of edge controller placement as an optimization problem. We consider a network of a diverse set of N $\subseteq$ B . This subset supports edge nodes which can host Virtual Process Function (VPF) an act as an edge controller. Not all edge nodes are active all the time, thus, we assume that K $\subseteq$ N active per time. This results in that K edge nodes generate PacketIN messages with arrival rate λ packets/time. The probability of the generated traffic can be estimated by $P_{traff} = \frac{e^{-\lambda t}(\lambda t)^K}{K!}$. Thus, the expected value of the generated traffic is $\sum_{k \in K} v_k P_{traff}$ where $v_k \in \{0,1\}$ is a decision binary optimization variable to denote whether the edge node is available or not. This availability is justified by enough storage capacity to save virtual process image and enough power for execution.

One of the important metrics to be considered in edge controller placement for industrial use cases is the reliability of the communication system. The network may fail due to the wireless link goes down or the component failure. We consider basically a set of L failure scenarios based on the valid control paths, where the control paths are the logical links between the controllers or between the edge nodes and the controllers L {1, 2, , L}. We denote by l the failure probability of each link by using generalized pareto distribution as it produces Independent Identically Distributed (IID) scenarios. Let us denote by hnk {0,1} a binary optimization variable that denotes whether a control path between edge node n and a controller at k exists or not. Hence, the expected value of invalid control paths

$$J_R = \frac{1}{L} \sum_{n \in N} \sum_{k \in K} h_{nk} \cdot \pi_l \tag{1}$$

We consider a binary optimization variable $y_{nk} \in \{0,1\}$ to denote whether node n is assigned to a controller at node k ($y_{nk} = 1$) or not ($y_{nk} = 0$). Hence, the assignment policy can be expressed as

$$y = (y_{nk} \in \{0,1\} : n \in N, k \in K) \tag{2}$$

The assignment of an edge node to a controller induces a cost. This cost increases with the topological distance. We denote $d_{nk} \geq 0$ is the delay (in millisecond) when node n is assigned to a controller at node k. The total assignment cost is expressed as follows

$$J_a = \sum_{n \in N} \sum_{k \in K} y_{nk}\, d_{nk}\, v_k \cdot P_{traff} \tag{3}$$

The controllers exchange also, messages between them for synchronization purposes. There are two types of inter-controller associated overhead. One type of them is that messages are being exchanged at a constant rate. The other type are messages exchanged at a rate depends on the controller traffic load. $w_{zk}^{Const} \geq o$ denote messages exchanged at constant rate while $w_{zk}^{dep} \geq o$ denotes controllers load dependent messages. These messages exchanged between controllers placed at node z and node k. We consider also x_k as a binary optimization variable to denote whether a controller placed at node k ($x_k = 1$) or not ($x_k = 0$). Thus, the placement policy

$$X = (x_k \in \{0,1\} : k \in K) \tag{4}$$

The total overhead synchronization costs between a pair of controllers at node j and node k is given as follows

$$J_s = \sum_{z \in Z} \sum_{k \in K} X_z X_k (w_{zk}^{Const} + w_{zk}^{dep} \sum_{n \in N} y_{nk}\, v_k \cdot P_{traff}) \tag{5}$$

It should be considered that each node n should be assigned to only one controller such that

$$\sum_{k \in K} y_{nk} \geq 1, \forall n \in N \tag{6}$$

Also, the controller can be placed at node k if and only if the sum of computing resource requirement does not exceed the generated traffic from node k.

$$\sum_{k \in K} X_k \geq v_k . P_{traff} , \forall k \in K \tag{7}$$

In addition, the controller should be placed at node k, so that node n can be assigned to it, such that

$$y_{nk} \leq x_k , \forall k \in K , n \in N \tag{8}$$

Based on the network operator preferences, the network operator can perform trade-off between the different metrics. A weight value $\beta \geq 0$ is used to balance the optimization problem.

$$J_{tot} = J_a + J_R + \beta J_s \tag{9}$$

By increasing β , more priority is given to the synchronization overhead cost. The edge controller placement problem can be expressed as follows

$$\begin{aligned} &\min \quad J_{tot} \\ &s.t. \ (2),(4),(6),(7),(8) \end{aligned}$$

The above optimization problem is a challenge problem as it contains discrete variables with objective function with quadratic and cubic terms. This problem is NP-Hard problem, as the edge controller placement depends on the topological distance between the controllers.

5. OPTIMIZATION ALGORITHMS

We begin by showing that for a given controller placement x, the optimal assignment policy y can be found. Denote x_k is a controller placed at node k. The set of all possible locations for the controller, also called the ground set G

$$G = (X_k : k \in K) \tag{10}$$

A subset of elements X G corresponds to a controller placement policy X. Let X n the binary representation of the set of controllers positions such that X (x 1, x 2 ,..., x k) , the objective function f can be expressed as a set function such that $f : 2^G \rightarrow \mathbb{R}$:

$$f(x) = J_{tot}(x, y(x)) \tag{11}$$

For a given controller placement at X, the optimal assignment is y. This leads us to a definition of a set of functions. These functions called submodular function [20].

Definition: let G is the ground set, a set function $f : 2^G \rightarrow \mathbb{R}$ is said to be submodular function

if and only if, for two subsets A , B$\subseteq$G and A $\subseteq$ B and every element that, i $\in$G \ B , it holds that,

$$f(A \cup \{i\}) - f(A) \geq f(B \cup \{i\}) - f(B) \quad (12)$$

which means adding an element to a smaller set, resulting in that the respective gain expands. This is called the marginal value [21]. The marginal value increases with the smaller set function [22].

We will show in the appendix that the objective function f(x) under certain conditions on the cost value is a submodular function As shown from the appendix that the total balanced cost function is submodular, we use two optimization algorithms to balance the different metrics and to minimize the overall cost. One algorithm is a Greedy algorithm. The algorithm proceeds in K iterations which corresponds to arbitrary order r1 ,r2, …..,rk of the ground set G. At each iteration, two solutions are maintained A and B. Also, the reliability of valid control paths is checked. Initially, A is assigned to 0 and B is assigned to G. At Kth iteration, the algorithm is either adds rk to A or removes it from B. This decision is done randomly and greedily based on the marginal value of each of the two options. The two solutions coincide and A=B after Kth iterations. Thus, the algorithm returns the optimal placement correspond to the minimum cost value. With that algorithm, we get a solution to the edge controller placement problem with an approximation that w dep is constant and identical for all cases The other algorithm is the Connectivity Ranking Algorithm (CRA) for comparison purposes of the obtained results [23]. The connectivity ranking algorithm sort the nodes in descending order according to their connectivity. The idea is to repeatedly place a new controller at one of the k edge nodes and calculate the total cost Jtot until the next cost is higher than the current one. Then, we choose the current placement as the optimal one.

Greedy Algorithm

---

1. $G \leftarrow B, \varphi \leftarrow A$
2. For k=1 to K do
3. For l=1 to L do
4. $\Delta A \leftarrow f(A) - f(A \cup \{r_k\})$
$\Delta B \leftarrow f(B) - f(B \setminus \{r_k\})$
5. With $prob. \dfrac{\Delta A}{\Delta A + \Delta B} do$
6. $A \leftarrow A \cup \{r_k\}$
7. else
8. $B \leftarrow B \setminus r_n$
9. End
10. End
11. return A

Connectivity Ranking Algorithm

---

1. $V \leftarrow \varphi, \infty \leftarrow$ mincost
2. //sort nodes by degree in descending order
3. $K \leftarrow sort(k, \deg(k))$
4. For k=1 to K do
5. For l=1 to L do
6. If CommCost(k)<mincost then
7. mincost $\leftarrow$ CommCost(k)
8. $V \leftarrow k$
9. else
10. mincost $\leftarrow$ CommCost(k);break;
11. End
12. End
13. Return V, mincost

6. EVALUATION RESULTS

In this section, we provide the results by running the proposed algorithms using real wireless network topology on Matlab software. We use MANIAC mobile ad hoc network in [24]. The network contains 14 nodes which allows us to execute the proposed algorithms in a reasonable time. We define the delay cost as the aggregate delay of the links of the valid shortest path. We set the delay of each link with average value 12.2 msec. We set the average probability of failure is 0.15 and the probability of estimated traffic is 0.65. We set w w^{const} = 0.2**hops* and w^{dep} = 0.6**hops*, where hops indicate the number of hops of the shortest path between the respective nodes.

Figure 2 shows the relationship between the weight value β and the total balanced cost J_{tot}. As shown form the figure, the greedy algorithm outperforms connectivity ranking algorithm (CRA) along the values of β . Furthermore, the performance of Greedy algorithm exceeds connectivity ranking algorithm at larger value ofβwhere more priority is given to the delay and reliability than overhead synchronization cost. Figure 3 shows the number of controllers needed to be placed to optimize the controller placement problem. For lower values of β where minimum delay is preferred, the number of edge controllers is high to minimize the delay cost between data-plane nodes and edge controllers. While the number of edge controllers goes down when the synchronization overhead cost is preferred with large values of β , so that inter-controller communication is reduced.

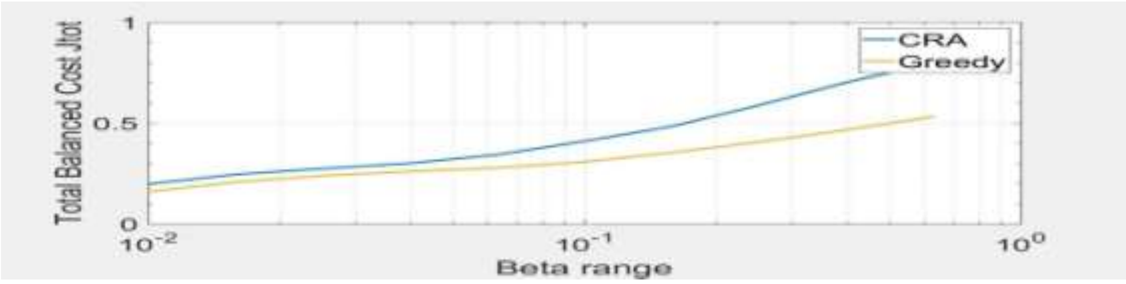

Figure 2. The total balanced cost against Beta range

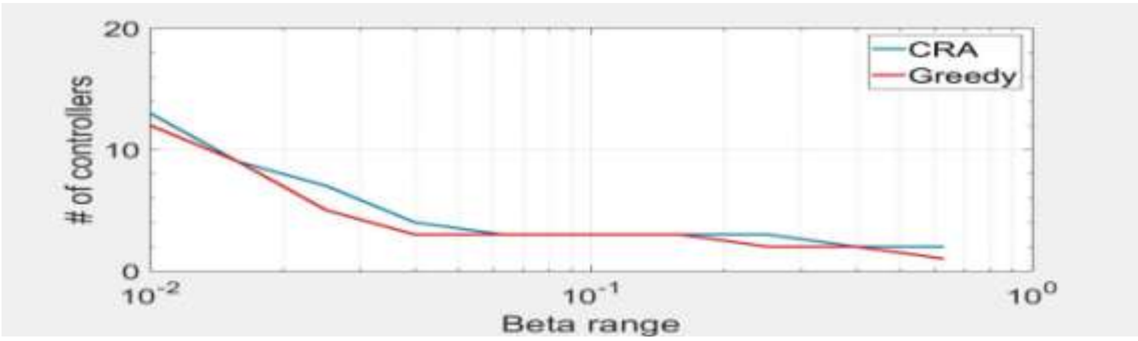

Figure 3. The number of controllers against Beta range

7. LIMITATIONS AND FUTURE WORK

The aforementioned work did not take into consideration the shape of traffic overheads. In the real world, different nodes generate diverse types of traffic that must be considered effectively. Also, the configured network has to support SDN configuration to be able to apply the optimization formula and submodularity conditions. In addition, the delay of each link isassumed to be constant which is not the case in the real time for heterogeneous wireless sensor networks. In the future, we plan to investigate different delays for disparate links. Additional mechanisms of forming a controller will be analysed. We will consider the interference and the congestion of the links in placing the edge controller in next generation WSNs.

8. CONCLUSIONS

In this paper, we addressed the edge controller placement problem for next generation wireless sensor networks. The problem was formulated as an optimization problem which is submodular function. We used Greedy algorithm and Connectivity Ranking Algorithm (CRA) for findingthe optimal solution for the balanced problem. The evaluation results were shown and provided that Greedy algorithm outperforms Connectivity Ranking Algorithm (CRA) along the value of weight value β. This paper provides an interesting optimization formula for industrial use cases that helps significantly in standardization of next generation wireless sensor networks.

REFERENCES

[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Edge Computing: Vision and Challenges”, IEEE Internet of Things Journal, vol. 3, no. 5, 2016.

[2] D. Kreutz, F. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, S. Uhlig, “Software-Defined Networking: A Comprehensive Survey”, in Proc. of the IEEE, vol. 103, no. 1, pp. 14-76, 2015.

[3] M. Peng, S. Yan, K. Zhang, C. Wang, “Fog Computing based Radio Access Networks: Issues and Challenges”, IEEE Network, vol. 30, no. 4, pp. 46-53, 2016.

[4] I. S. Acharyya and A. Al-Anbuky, “Towards wireless sensor network softwarization,” in Proc. of IEEE Conference on Network Softwarization, 2016, pp. 378–383.

[5] C. Alippi, R. Fantacci, D. Marabissi, and M. Roveri, “A cloud to the ground: The new frontier of intelligent and autonomous networks of things,” IEEE Comm. Mag., vol. 54, no. 12, pp. 14–20, 2016.

[6] A. Singh “A survey and classification of controller placement problem in SDN” International Journal of Network Management, 2018.

[7] H. Xu, Z. Yu, C. Qian, X. Li, Z. Liu, “Minimizing Flow Statistics Collection Cost of SDN Using Wildcard Requests”, in Proc. IEEE Infocom, 2017.

[8] E. Emil, G. Dn, and F. Viktoria, “Radio and computational resource management for fog computing enabled wireless camera netoworks,” in Proc. of IEEE GlobeCoWorkshop on Internet of Everything, 2016, pp.1–6.

[9] A. Muqaddas, A. Bianco, P. Giaccone, G. Maier, “Inter-controller Traffic in ONOS Clusters for SDN Networks”, in Proc. IEEE ICC, 2016.

[10] Zhao, P., & Dán, G. “Resilient placement of virtual process control functions in mobile edge clouds” in IFIP networking conference and workshops, 2017.

[11] R. Cohen, L. Lewin-Eytan, J. S. Naor, and D. Raz, “Near optimal placement of virtual network functions,” in Proc. of IEEE INFOCOM, 2015, pp. 1346–1354.

[12] M. Yigit, V. C. Gungor, and S. Baktir, “Cloud computing for smart grid applications,” Computer Networks, vol. 70, pp. 312–329, 2014.

[13] G. Dn, R. B. Bobba, G. Gross, and R. H. Campbell, “Cloud computing for the power grid: from service composition to assured clouds,” in Proc. of Usenix Workshop on Hot Topics in Cloud Computing (HotCloud), 2013.

[14] A. Sallahi, M. St-Hilaire, “Optimal Model for the Controller Placement Problem in Software Defined Networks”, IEEE Communications Letters, vol. 19, no.1, pp. 30-33, 2015.

[15] S. Das, S. Misra, M. Khatua, and J. J. Rodrigues, “Mapping of sensor nodes with servers in a mobile health-cloud environment,” in Proc. Of IEEE International Conference on e-Health Networking, Applications & Services (Healthcom), 2013, pp. 481–485.

[16] M. Ghaznavi, A. Khan, N. Shahriar, K. Alsubhi, R. Ahmed, and R. Boutaba, “Elastic virtual network function placement,” in Proc. Of IEEE CloudNet, 2015, pp. 255–260.

[17] H. Moens and F. De Turck, “VNF-P: A model for efficient placement of virtualized network functions,” in Proc. of IEEE International Conference on Network and Service Management (CNSM), 2014, pp. 418–423.

[18] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, “Network function placement for NFV chaining in packet/optical datacenters,” Journal of Lightwave Technology, vol. 33, no. 8, pp. 1565–1570, 2015.

[19] A. Hmaity, M. Savi, F. Musumeci, M. Tornatore, and A. Pattavina, “Virtual network function placement for resilient service chain provisioning,” in Proc. of IEEE International Workshop on Resilient Networks Design and Modeling (RNDM), 2016, pp. 245–252.

[20] U. Feige, V. Mirrokni, J. Vondrak, “Maximizing Non-monotone Submodular Functions”, in Proc. IEEE FOCS, 2011.

[21] N. Buchbinder, M. Feldman, J. Naor, R. Schwartz, “A Tight Linear Time (1/2)- Approximation for Unconstrained Submodular Maximization”, in Proc. IEEE FOCS, 2012.

[22] S.O. Gharan, J. Vondrak, “Submodular Maximization by Simulated Annealing”, in Proc. ACM/SIAM SODA, 2011.

[23] Z. Su, M. Hamdi “MDCP: Measurement-aware Distributed Controller Placement for Software Defined Networks” IEEE 21st International Conference on Parallel and Distributed Systems, 2015.

[24] A. Hilal, J.N. Chattha, V. Srivastava, M.S. Thompson, A.B. MacKenzie, L.A. DaSilva, P. Saraswati, CRAWDAD dataset vt/maniac: http://crawdad.org/vt/maniac/20081101, https://doi.org/10.15783/C78W2V

APPENDIX

Based on that the weighted sum of submodular functions is also submodular function.

$$J_{s\_Const} = \sum_{z \in Z} \sum_{k \in K} X_z X_k w_{zk}^{Const} \quad (13)$$

$$J_R = \frac{1}{L} \sum_{n \in N} \sum_{k \in K} h_{nk} \cdot \pi_l \quad (14)$$

$$J_a = \sum_{n \in N} \sum_{k \in K} y_{nk} d_{nk} v_k \cdot P_{traff} \quad (15)$$

$$J_{s\_dep} = \sum_{z \in Z} \sum_{k \in K} X_z X_k w_{zk}^{dep} \left(\sum_{n \in N} y_{nk} v_k \cdot P_{traff} \right) \quad (16)$$

Here, J_{s_Const} and J_{s_dep} denote the constant and the load dependent synchronization costs,respectively. J_R denote the expected value of invalid control paths as the reliability metric. For a given node $k \in K$, load cost. Ja denote the total assignment cost that includes
Let us consider two placement sets A and B where $A \subseteq B \subseteq G$, where G is the ground set that contains all possible values of controller placement. We add an element $x_k \in G \setminus B$ to both placement sets.

1) For the function J_{S_Const} and the placement set A, the marginal value of adding a controller at node m, x_m

$$\sum_{z \in K} w_{zm} - \sum_{k \in K} w_{mk} \quad (17)$$

If we replace A with placement set B, the marginal value decrease. Hence, the above function is submodular function.

2) For the invalid control paths function, by adding invalid control path between node n and m to the function, the marginal value decrease with placement set B.

$$\sum_{n \in N} h_{nm} - \sum_{k \in K} h_{mk} \quad (18)$$

Hence, the function is submodular.

3) For the assignment function, with the placement set A, where node n is assigned to controller at node m the marginal value is

$$d_{nm} v_n \leq d_{nk} v_k \qquad (19)$$

Thus, the marginal value decrease if we replace A with the placement set B.

4) For the function J_{s_dep} , with placement set A, the marginal value of adding a controller at node m, x_m is w^{dep} which is independent of the assignment policy with the assumption of that its value is constant. So, the function is also submodular.