Grey Wolf Aware Energy-saving and Load-balancing in Software Defined Networks Considering Real Time Traffic

Kuldeep Kurroliya
Department of Computer Science & Engineering
National Institute of Technology, Rourkela
Odisha, India
kuldeep.nitrkl2304@gmail.com

Sagarika Mohanty
Department of Computer Science & Engineering
National Institute of Technology, Rourkela
Odisha, India
sagarikam_23@yahoo.com

Khushboo Kanodia
Department of Computer Science & Engineering
National Institute of Technology, Rourkela
Odisha, India
kkanodia2307@gmail.com

Bibhudatta Sahoo
Department of Computer Science & Engineering
National Institute of Technology, Rourkela
Odisha, India
bdsahu@nitrkl.ac.in

***Abstract*—The energy depletion in Software Defined Network (SDN) is a primary issue for the operators and regulators of the network because of the ubiquitous implementation of SDN to satisfy the demands of bandwidth-consuming applications. While conventional optimization of edge weights technique is used in various load balancing mechanisms in IGP networks. In this paper, we present a new edge weight optimization algorithm known as Grey Wolf Aware Load balancing and Energy saving(GLE) which simultaneously optimizes energy consumption and load balance in SDN networks. This technique can be applied along with the current mechanisms which use the link active/inactive concept to achieve a significant improvement in energy efficiency. The proposed method in this paper is a new aspect to save energy in SDN networks without compromising conventional traffic-engineering realizations in the environment of plain IP routing. To evaluate the energy-saving by the proposed mechanism without loss of generality, we tested it on various currently proposed methods who have used the technique of link sleeping as described in the related works. The evaluation is tested on the GEANT network topology. GLE shows a noteworthy improvement in the energy-saving as compared to the recently proposed algorithms while achieving significant performance in load balance.**

***Index Terms*—Energy Consumption, Software-Defined Network (SDN), Green networks, Efficient Routing, Grey Wolf Aware Load balancing and Energy saving (GLE).**

I. Introduction

The techniques to save energy of modern networks have been explored more since the last few decades due to the huge power consumption and operation cost of the computer networks. The excessive increase in energy consumption causes the greenhouse gas effect because of the release of energy from nonrenewable resources. European-Telecoms absorbs approximately 22.4 TWh / year which is presumed to increase by about 36.8 TWh in 2021 if we fail to deploy improved energy-efficient networks [1]. ISP backbone network presently consumes approx. 11% of the overall energy demands which can increase to 41% by 2021 [2] if we do not adopt green strategies. This is because of the increased utilization of applications that consume high bandwidth, for example, Video On Demand (VOD), cloud computing on the internet, etc. In the past few years, various energy-based traffic engineering methods have been proposed to improve the energy-saving in IP networks [3] [4] [5] [6] [7]. In the this papers, energy-saving is improved by any one of the two mechanisms, that is, put the network equipment / edges into the sleep state or the transmission data rate of devices. To maximally improve the gain of energy-saving, ETE algorithms have designed various strategies to drift the overall traffic to the minimum number of network devices possible, so that other network devices can be put in an off state to conserve energy. Till the reduced network can satisfy the traffic requests, energy can be conserved without service degradation to the users.

Load balancing is the common aim of traffic-engineering in most of the networks and different types of techniques have been proposed about this objective [8]. The main target of load balancing is to reduce the MLU value by optimally satisfying traffic requests. This markdown in the MLU value guarantees Quality of Service (QoS) and it also helps in managing the unexpected problems that arise in satisfying traffic requests. So the load balancing scheme tries to distribute the traffic whereas energy-saving techniques aim to concentrate the traffic to active network devices. As a result, there is a tradeoff among the two objectives.

Load-balancing plays a significant role in SDN to acquire network performance and efficiency. From the literature review of the various articles mentioned in the references section, it can be noticed that the following parameters are responsible for load balancing in SDN. The parameters are the response time of the controller, load threshold, switch throughput, edge utilization in terms of bandwidth, and cost of the link, rate of the flow, etc. In this paper, we have considered the load balancing metric of edge utilization in terms of bandwidth.

In GLA, the optimization of IGP link weights affects the traffic dissemination in the network. Precisely, the sharp setting of link weights in IGP can simultaneously increase energy savings and improve load performance despite the residual topology. This contradicts the traditional schemes of link weight which do not perform efficiently operation of link sleeping.

The essential contributions of this article include: Certain number of routes are generated for every source, destination pair to satisfy switch-to-controller traffic requests. The GLE algorithm discovers the optimal set of link weights to satisfy the real-time traffic requests thereby, improving the energy-efficiency which is more than the GLA algorithm for a particular value of link utilization of any link. The principal purpose is to refine the energy-efficiency along with fulfilling the load-balancing aim. The traffic matrix considered in the paper is extracted from the internet-zoo topology site. The execution of algorithms is done on three different topologies and the analogous inter-related results are displayed in the figures.

The hierarchy of paper is depicted as follows: Section-I gives the introduction of SDN and discusses the tradeoff between energy-savings and load balancing. Section-II gives information about the related techniques proposed by different authors to achieve the above two aims in both isolated and combined way. Section-III depicts the network model to simultaneously optimize the two objectives while still maintaining a good accuracy of network performance. Section IV discusses the GLE technique to optimize the two objectives. SectioncV describes the comparison of GLE and GLA method on the Geant network. This also describes the performance of the GLE method on AT&T and Abilene network is shown. Section VI gives the conclusion of the paper.

II. Related Works

The various techniques to save energy in the field of traffic engineering are described in the article [5]. In the article, the author discussed how traffic requests can be satisfied optimally to improve energy efficiency in network equipment. Various solutions were developed and encapsulated in the articles [3]. ETE methods are categorized as online or offline. Offline methods are described in the article [9] [10] [11], uses the technique of global optimization to keep a view and information of the complete state of the network. They are implemented with the protocols that are long-established.

Online ETE methods are described in the article [12] [13] [14] [15]. In online methods, local information of the network state is stored for reconfiguration of the network. These methods are more susceptible to unpredicted problems that occur in satisfying dynamic traffic requests. These methods monitor the network more accurately than offline schemes. They can also handle extra challenges like network stability and protocol extensions. As a result deployment of online techniques is more difficult than offline schemes. Offline methods do not consider the case of modifying network link weights according to the current traffic requests to optimize the performance of the network.

Sampa Sahoo et al. in [16] proposed an learning automata-based and a game theory based scheduling in [17] technique for task scheduling in cloud. Those approach can be applied in Software Defined Networks for various applications. Our proposed algorithm can be applied in the architecture model proposed in [18] for improvement of the system.

III. Network Model

TABLE I
Description of Variables

Symbol	Description
$G(M, L)$	G is an undirected graph. M represent set of nodes. L denotes the set of edges.
LU	Maximum Edge Utilization
E	Energy Savings
c_{sd}	Capacity of bandwidth of edge from node s to d
t^{ij}	Traffic flow demand from vertex i to j
g_{sd}^{ij}	Traffic flow demand from i to j that passes through edge from s to d
g_{sd}	Overall traffic flow demand on edge from s to d.
α	Maximum usable link capacity to satisfy traffic demands.
S2C	Switch to controller traffic requests.

The main objective of the problem is to mutually optimize the load and energy savings in the network. Mathematically, the problem can be represented as:

$$maximize\ E \tag{1}$$

$$minimize\ LU \tag{2}$$

subjected to the constraints of:

$$\sum_{k=1}^{|M|} g_{jk}^{sd} - \sum_{k=1}^{|M|} g_{kj}^{sd} = \begin{cases} t^{sd} & \forall\ s,\, d,\, k = s \\ -t^{sd} & \forall\ s,\, d,\, k = d \\ 0 & \forall\ s,\, d,\, k \neq s,\, d \end{cases} \tag{3}$$

$$g_{jk} \leq \frac{\alpha}{100} * c_{jk}\ \forall\ j, k\ with\ \alpha \in\ [0, 100] \tag{4}$$

The first objective of maximizing the energy-saving gain is denoted by Equation (1). The interpretation of E is particular to existing ETE techniques. The second objective of minimizing Maximum link-utilization (MLU) is denoted by Equation (2). Minimizing the MLU of the network leads to attaining load balance. If the GLE is performed on one traffic matrix then the MLU corresponds to that one instant of traffic matrix only. The worst-case scheme represents the GLE involving different traffic matrices, then the MLU changes according to the traffic matrix at that instant. Equation (3) denotes the flow preservation constraint. Equation (4) denotes that if the topology with less than the total number of links is used then the utilization of active links should be less than a given threshold value. After putting a particular set of links in the sleep state, the maximum link utilization of all other links should not be more than α. One more constraint is that the given network should be fully connected after configuring some links in an inactive state so that there exists a route among any two nodes in the modified network.

IV. Heuristic

A. Grey-Wolf Optimization

Grey-wolfs are associated with the family of Canidae and can be assumed as apex prey. They belong to the highest level in the food chain. They are mostly found in a group whose avg. size is 6-13. They follow a rigid social hierarchy. The ”alpha” mainly has the responsibility of deciding hunt and sleep regions of target prey. The population of wolfs is divided into different types of search representatives, namely, α, β, δ, and ω according to the priority of the fitness values. Alpha plays the main role in making every decision. Beta help and advises alpha in taking decisions. The main job of the delta is to report alpha and beta and support omega. The rest of the population of wolves is considered as omega.

In the GLA algorithm, the author tried to find the optimal link weights for satisfying traffic requests through a genetic approach. In GLA, a specific combination of link weights is compressed as a chromosome. The genes in the chromosome are equal to the total links. A population is formed of such chromosomes. Then the operator of two segments has been adopted in the crossover operation followed by the mutation operator with a mutation probability of 50%. On the other side, in the GLE algorithm, we adopted a grey wolfs strategy to find the near-optimal link weights to satisfy real-time traffic requests. First, a random set of certain combinations of link weights is considered for initializing the traffic matrix to satisfy current traffic demands. Then the link weights are modified according to the top three solutions in the population to maximize the energy savings.

1) Communal Hierarchy: To define the arithmetic model of a communal chain of grey-wolf heuristic technique, we have considered a population of size n. Among all the individuals of the population, α is the principal solution, β is the next best, and δ is the next best after β. The remaining individuals of the population are ω and these works according to the guidance given by the top three best individuals in the optimization procedure.

2) Search operation: In search of the target prey, ω change their locations according to the locations of alpha-α, beta-β, and delta-δ. The corresponding equations to change the locations of ω are given by (8) (9) (10) (11) (12) (13) (14). While exploring for prey, wolfs forllow the mechanism of convergence and divergence to attack on it. If $|C| \geq 1$, then they diverges otherwise converges to find the near-optimal solution.

3) Encircle / Attack: The arithmetic equation to encompass the prey is given by the equations (8) (9) (10) (11) (12) (13) (14).

The below equations are used to initialize the variables C and F of the algorithm.

$$\vec{c} = 2 - \frac{2 * t}{maximum\ num\ of\ iterations} \tag{5}$$

where t is the iteration number

$$\vec{C} = 2.\vec{c}.\vec{r_1} - \vec{c} \tag{6}$$

where the value of c is linearly decremented from 2 to 0 during the course of iterations. r_1 is a random number in [0, 1].

$$\vec{F} = 2.\vec{r_2} \tag{7}$$

r_2 is a random number in [0, 1]
To update the members of an individual in the population following equations are used.

$$\vec{H_\alpha} = |\vec{F_1}.\vec{Z_\alpha} - \vec{Z}| \tag{8}$$

$$\vec{H_\beta} = |\vec{F_2}.\vec{Z_\beta} - \vec{Z}| \tag{9}$$

$$\vec{H_\delta} = |\vec{F_3}.\vec{Z_\delta} - \vec{Z}| \tag{10}$$

$$\vec{Z_1} = \vec{Z_\alpha} - \vec{C_1}.\vec{H_\alpha} \tag{11}$$

$$\vec{Z_2} = \vec{Z_\beta} - \vec{C_2}.\vec{H_\beta} \tag{12}$$

$$\vec{Z_3} = \vec{Z_\delta} - \vec{C_3}.\vec{H_\delta} \tag{13}$$

$$\vec{Z}(t+1) = \frac{\vec{Z_1} + \vec{Z_2} + \vec{Z_3}}{3} \tag{14}$$

where t is the iteration number

Algorithm 1 Fitness

Input : $T,\ p$
Output : E

$T \leftarrow Traffic\ Matrix$
$p \leftarrow random\ set\ of\ link\ weights$
$E \leftarrow Energy\ Savings$
$Set\ the\ bandwidth\ capacity\ of\ links$
$Update\ T\ according\ to\ p$
$Satisfy\ S2C\ traffic\ request\ with\ updated\ T$
$by\ calculating\ path\ for\ each\ pair\ considering$
$bandwidth\ utilization$
$Find\ the\ number\ of\ inactive\ edges\ and\ total\ edges$
$E \leftarrow \frac{total\ inactive\ edges}{total\ edges}$
return E

Algorithm 1 defines the objective function. Traffic matrix T and a random set of link weights p are given as input and energy savings E (%) is output. With the random set of link weights, the traffic matrix is updated and the bandwidth capacity of all links is set. Then according to the current traffic matrix(link weights) and bandwidth values, the S2C traffic requests are satisfied by calculating the certain number of paths between every pair of S2C traffic demands Among the certain number of paths, a particular path is selected as per the bandwidth utilization constraints. The energy savings(%) is calculated as :

$$Energy\ Savings\ (E) = \frac{Number\ of\ Inactive\ Edge}{Total\ number\ of\ Edges} * 100 \tag{15}$$

Algorithm 2 GLE Algorithm

Input : $T,\ itr\_max,\ n$
Output : Z_α

$T \leftarrow Topology\ Matrix$
$itr\_max \leftarrow Maximum\ number\ of\ iterations$
$n \leftarrow Population\ Size$
$Initialize\ the\ c$, C and F $using\ eq$ (5) (6) (7)
$wolfs \leftarrow \phi$
for $i \leftarrow 1$ to n **do**
  $t \leftarrow generate\ random\ list\ of\ link\ weights$
  $wolfs \leftarrow wolfs\ \cup t$
end for
$scores \leftarrow \phi$
for $i \leftarrow 1$ to n **do**
  $s \leftarrow Fitness(T, wolfs(i))$
  $scores \leftarrow scores \cup s$
end for
$sort\ wolfs\ according\ to\ scores$
$Z_{\alpha score} \leftarrow the\ best\ solution$
$Z_{\beta score} \leftarrow the\ second\ best\ solution$
$Z_{\delta score} \leftarrow the\ third\ best\ solution$
$t \leftarrow 0$
while $t \leq itr\_max$ **do**
  $c = 2 - t * \frac{2}{itr\_max}$
  for $i \leftarrow 4$ to n **do**
    $Update\ c,\ C\ and\ F\ using\ eq$ (5) (6) (7)
    $Update\ weights\ of\ remaining\ individuals$
    $according\ to\ first\ three\ best\ solutions$
    $using\ eq$ (8) (9) (10) (11) (12) (13) (14)
    $scores(i) \leftarrow Fitness(T, wolfs(i))$
  end for
  $sort\ wolfs\ according\ to\ scores$
  Update $Z_{\alpha score}$
  Update $Z_{\beta score}$
  Update $Z_{\delta score}$
  $t \leftarrow t + 1$
end while
return $Z_{\alpha score}$

In Algorithm 2(GLE), Traffic Matrix T, grey-wolf population size n, and maximum number of iterations max_itr are passed as input parameters. Then the algorithm parameters c, C, and F are initialized according to the equations (5) (6) (7). Wolfs population denoted by $wolfs$. The individuals in $wolfs$ are represented by lists of random weights assigned to links of the traffic matrix. The range for the random weights assigned to the links is set as (1, number of links). The upper bound for the range of random weights is not fixed and can be set as per the convenience of the working environment. The size of the individual list is equal to the total number of edges in the considered topology matrix. The fitness values, that is, energy saved(%) for each combination of link weights are calculated and stored in $scores$. Then the $wolfs$ are sorted in descending order according to the fitness values defined in $scores$. Then $Z_{\alpha score}$ stores the principal solution from the $wolfs$, $Z_{\beta score}$ stores the next best solution, and $Z_{\delta score}$ stores the next best solution after $beta$. The rest of the individuals of the $wolfs$ are considered as ω. In the final phase of the algorithm, for each individual in $wolfs$, the link weights are updated according to the equations (8) (9) (10) (11) (12) (13) (14) and $Z_{\alpha score}$, $Z_{\beta score}$, and $Z_{\delta score}$ are updated in each iteration of the algorithm. Finally, the solution corresponding to the best fitness value, that is, $Z_{\alpha score}$ is returned.

TABLE II
TOPOLOGY CONSIDERED

Name of Topology	Total nodes	Total Edges
Geant	37	58
At&T	25	56
Abilene	11	14

The time complexity of the GLA algorithm mainly depends on the number of iterations and population size and the time complexity of the fitness function. In the GLE algorithm also the time complexity depends on the number of iterations, wolfs population, and the complexity of the fitness function. Therefore, the time complexity of both the algorithms is comparably the same but the GLE algorithm shows a greater improvement in energy-efficiency along with load balance of edges.

V. EVALUATION

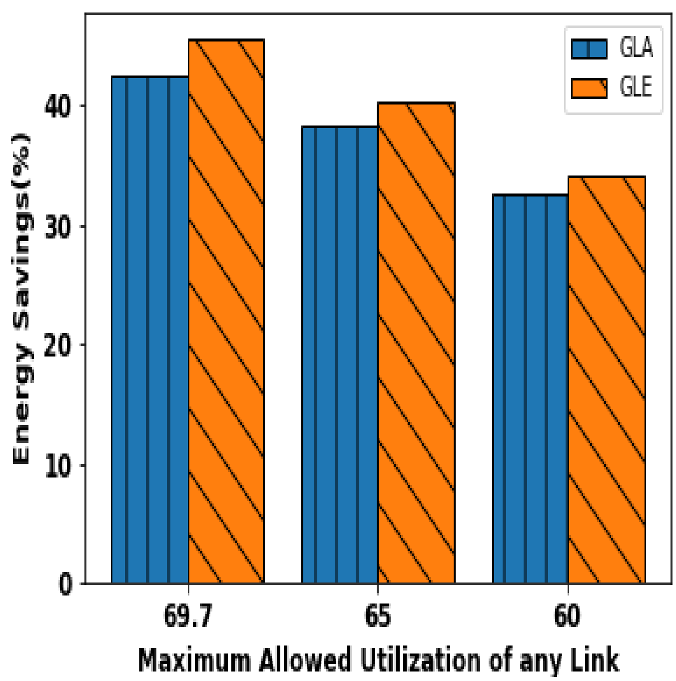

Fig. 1. Energy Efficiency of GLE using variable link weights on GEANT network.

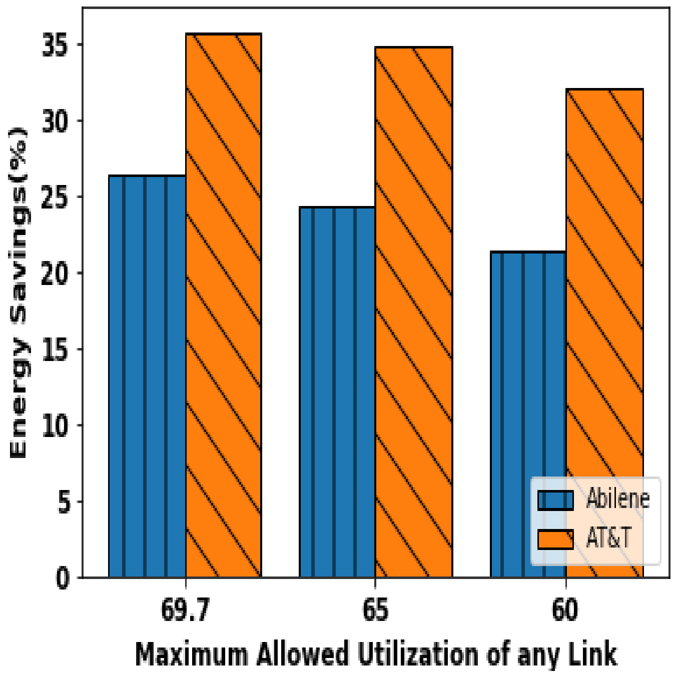

Fig. 2. Energy Efficiency of GLE using variable link weights on Abilene and At&T network.

The network topography considered for evaluating the algorithmic performance is AT&T(56 edges, 25 nodes), Abilene(14 edges, 11 nodes), and GEANT(58 edges, 37 nodes). The algorithms are executed in a $PYTHON - 3.7$ environment. The specifications of the system include $i7\ processor$ and 8GB RAM. For various experiments to be conducted, the topography of networks is extracted from the internet-zoo site. A random set of switch-to-controller traffic requests are considered for generating traffic traces. The algorithm calculates a certain number of paths for each traffic request and then tries to satisfy the demand of the path where the link utilization does not go beyond a maximum threshold value. The real-time traffic is considered in the sense that depending on the current traffic requests the GLE algorithm tries to modify the link weights to utilize the links of the network in an effective manner. If we lower the maximum allowed utilization of any link, then a large number of links should be put in the $active$ state to satisfy the traffic requests which results in low energy saving. Besides this, on the other aspect, if we increase the maximum allowed utilization of any link then the traffic requests can be satisfied with fewer links. As a result, we noticed that the energy efficiency is directly proportional to the maximum allowed utilization of the link.

Fig. 1 describes the variation of energy-efficiency with the maximum allowed utilization of any link in the traffic matrix in the case of the GEANT network. In Fig. 2 the corresponding variations of the two parameters are shown in case of AT&T and Abilene network.

VI. Conclusion

In our paper, we introduced an algorithm, GLE (Grey Wolf Aware Load Balancing and Energy Saving) which simultaneously optimize the two objectives of energy-saving and load balancing in the SDN network. GLE performs better than the existing energy-aware algorithms that use the IGP technique of routing and conserves energy by putting network equipment and link to sleep state.

To find the optimal set of edge weights is a NP-Hard problem. GLE approach balances the tradeoff between two objective functions and finds an optimal set of link weights. The performance of the algorithm has been improved by first finding the three best solutions and then updating the other solutions according to the positions of the first three solutions. The grading of the search agents is done in such a way to effectively search the solution space. GLE is compared with the GLA algorithm described in [19]. The results are evaluated on three network topologies Geant, Abilene, and AT&T.

The analysis of simulation experiments was performed on three topologies GEANT, Abilene, and AT&T. From the analysis, GLE shows better performance than GLA as depicted in the figures. Here, the improved energy-saving gain is attained along with optimal load balance. In future works, we will consider the parameters of a lifetime of the network, average energy of the network, etc. and some other related QoS(Quality of Service) parameters while saving energy.

References

[1] R. Bolla, F. Davoli, R. Bruschi, K. Christensen, F. Cucchietti, and S. Singh, "The potential impact of green technologies in next-generation wireline networks: Is there room for energy saving optimization?" *IEEE Communications Magazine*, vol. 49, no. 8, pp. 80–86, 2011.

[2] C. Lange, "Energy-related aspects in backbone networks," in *Proceedings of 35th European Conference on Optical Communication (ECOC 2009),(Wien, AU)*, 2009.

[3] A. P. Bianzino, C. Chaudet, D. Rossi, and J.-L. Rougier, "A survey of green networking research," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 1, pp. 3–20, 2010.

[4] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti, "Energy efficiency in the future internet: A survey of existing approaches and trends in energy-aware fixed network infrastructures," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 2, pp. 223–244, 2010.

[5] M. Gupta and S. Singh, "Greening of the internet," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, 2003, pp. 19–26.

[6] F. Francois, N. Wang, K. Moessner, and S. Georgoulas, "Optimization for time-driven link sleeping reconfigurations in isp backbone networks," in *2012 IEEE Network Operations and Management Symposium*. IEEE, 2012, pp. 221–228.

[7] L. Chiaraviglio, M. Mellia, and F. Neri, "Reducing power consumption in backbone networks," in *2009 IEEE International Conference on Communications*. IEEE, 2009, pp. 1–6.

[8] N. Wang, K. H. Ho, G. Pavlou, and M. Howarth, "An overview of routing optimization for internet traffic engineering," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 1, pp. 36–56, 2008.

[9] F. Francois, N. Wang, K. Moessner, and S. Georgoulas, "Optimizing link sleeping reconfigurations in isp networks with off-peak time failure protection," *IEEE Transactions on Network and service Management*, vol. 10, no. 2, pp. 176–188, 2012.

[10] A. Cianfrani, V. Eramo, M. Listanti, M. Polverini, and A. V. Vasilakos, "An ospf-integrated routing strategy for qos-aware energy saving in ip backbone networks," *IEEE Transactions on Network and Service Management*, vol. 9, no. 3, pp. 254–267, 2012.

[11] W. Fisher, M. Suchara, and J. Rexford, "Greening backbone networks: reducing energy consumption by shutting off cables in bundled links," in *Proceedings of the first ACM SIGCOMM workshop on Green networking*, 2010, pp. 29–34.

[12] N. Vasić, P. Bhurat, D. Novaković, M. Canini, S. Shekhar, and D. Kostić, "Identifying and using energy-critical paths," in *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, 2011, pp. 1–12.

[13] C. Panarello, A. Lombardo, G. Schembra, L. Chiaraviglio, and M. Mellia, "Energy saving and network performance: a trade-off approach," in *Proceedings of the 1st International Conference on Energy-efficient Computing and Networking*, 2010, pp. 41–50.

[14] A. P. Bianzino, L. Chiaraviglio, and M. Mellia, "Grida: A green distributed algorithm for backbone networks," in *2011 IEEE Online Conference on Green Communications*. IEEE, 2011, pp. 113–119.

[15] M. Charalambides, D. Tuncer, L. Mamatas, and G. Pavlou, "Energy-aware adaptive network resource management," in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*. IEEE, 2013, pp. 369–377.

[16] S. Sahoo, B. Sahoo, and A. K. Turuk, "A learning automata-based scheduling for deadline sensitive task in the cloud," *IEEE Transactions on Services Computing*, 2019.

[17] M. K. Patra, S. Sahoo, B. Sahoo, and A. K. Turuk, "Game theoretic approach for real-time task scheduling in cloud computing environment," in *2019 International Conference on Information Technology (ICIT)*. IEEE, 2019, pp. 454–459.

[18] M. K. Patra, "An architecture model for smart city using cognitive internet of things (ciot)," in *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*. IEEE, 2017, pp. 1–6.

[19] F. Francois, N. Wang, K. Moessner, S. Georgoulas, and K. Xu, "Green igp link weights for energy-efficiency and load-balancing in ip backbone networks," in *2013 IFIP Networking Conference*. IEEE, 2013, pp. 1–9.