Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

A Clustering Approach to Edge Controller Placement in Software-Defined Networks with Cost Balancing

Reza Soleymanifar * **Amber Srivastava Carolyn Beck Srinivasa Salapaka**

* *Authors are with Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, 1308 W Main St, Urbana, IL 61801 U.S.A. (e-mail: $\{reza2, asrvstv6, beck3, salapaka\}$@illinois.edu).*

Abstract: In this work we introduce two novel maximum entropy based clustering algorithms to address the problem of Edge Controller Placement (ECP) in wireless edge networks. These networks lie at the core of the fifth generation (5G) wireless systems and beyond. Our algorithms, ECP-LL and ECP-LB, address the dominant leader-less and leader-based controller placement topologies and have linear computational complexity in terms of network size, number of clusters and dimensionality of data. Each algorithm places controllers close to edge node clusters and not far away from other controllers to maintain a reasonable balance between synchronization and delay costs. While the ECP problem can be expressed as a multi-objective mixed integer non-linear program (MINLP), our algorithms outperform state of the art MINLP solver, BARON both in terms of accuracy and speed. Our proposed algorithms have the competitive edge of avoiding poor local minima through a Shannon entropy term in the clustering objective function. Most ECP algorithms are highly susceptible to poor local minima and greatly depend on initialization.

Keywords: Clustering, deterministic annealing, 5G networks, software-defined networks, wireless edge networks, edge controller placement.

1. INTRODUCTION

Wireless networks are of high importance in modern telecommunication systems and in order to enhance these systems, Software-Defined Networks (SDN) have been introduced as an emerging paradigm whose primary advantage is giving network administrators greater control over the network traffic and administration (Alshamrani et al., 2018). Traditionally wireless networks have played both the role of administration and relay of data within the same infrastructure. One of the limitations of this architecture is that modifying these networks requires manually re-configuring nodes of the network to accommodate the new changes. Softwarization is a new trend in wireless communication networks that helps to automate this type of manual work.

One of the most studied research problems, on which SDN itself heavily relies, is the so-called edge controller placement problem (ECP) (Alshamrani et al., 2018). Controller placement is one of the most important components of software-defined networks (Kuang et al., 2018). This problem was first introduced in Heller et al. (2012) and is in general NP-hard (Singh and Srivastava, 2018). Controllers are network nodes which are designated to control other nodes of a network. ECP in an edge network essentially reduces to determining how many and which nodes in the network need to be designated as the controllers. This placement induces several costs including delays between edge nodes and the controllers they are assigned to, and synchronization delay between the controllers themselves which we refer to as delay and synchronization costs respectively throughout the paper. Our approach here to study ECP is based on viewing this problem in a data clustering sense. Many clustering based approaches in literature are hindered by naive initialization and are thus prone to poor local optima (He et al., 2004). This leads to multiple optimization attempts with varied initializations that increase total computation time needed to find an optimal placement. These approaches are also restricted to a single objective value which prevents the decision maker from simultaneously considering multiple criteria. In this paper, we leverage properties of the deterministic annealing (DA) algorithm (Rose, 1998), which is tailored to avoid these shortcomings, and introduce algorithms that iteratively minimize the costs associated with ECP. In order to evaluate our algorithms we compare the final costs incurred with those of the MINLP formulation.

We identify the core competences of our algorithms as being (1) scalable and fast, due to linear computational complexity in terms of problem size and number of controllers, (2) high quality in terms of near optimal solutions, (3) initialization independent as we always start with one controller in the mass center of dataset, (4) excellent at avoiding poor local minima due to the use of a Shannon entropy term in the clustering objective function and (5) able to address a multi-objective scheme.

The rest of the paper is organized as follows. In Section 2 we overview ECP and SDN related works from recent years. In Section 3 we formally define ECP and explain the subtleties of this problem. In Section 4 we describe our approach to the problem and explain how we adapt the DA to the ECP problem. The reader may refer to Section 5 to see the results of the simulations and finally Section 6 shows conclusions and avenues for future research.

2. LITERATURE REVIEW

The controller placement problem for SDN's was first introduced in Heller et al. (2012). Li and Xu (2018) implement the Cuckoo search algorithm for the problem of controller placement in SDN's. Lu et al. (2019) identify the main function of SDN's as decoupling the data plane and control plane and identify controller placement as one of the hottest topics in SDN literature.

Liao et al. (2017) propose a density based controller placement which uses a clustering algorithm to split the network into multiple sub-networks. Papa et al. (2018) consider ECP in the context of satellite networks and design an integer linear program to address this problem. Focusing on reliability aspects of ECP, Alshamrani et al. (2018) address maximizing fault-tolerance aspects of controller placement rather than performance. They show sacrificing latency for reliability is generally not a good trade-off except in special scenarios.

Das and Gurusamy (2018) use a multi-objective optimization model to derive a multi-period roll-out plan for controller placement. A similar problem to ECP, the satellite gateway placement problem, is addressed in detail in Liu et al. (2018). Zhiyang Su and Hamdi (2015) propose a novel scheme to minimize measurement overhead, and formulate the Measurement-aware Distributed Controller Placement (MDCP) problem as a quadratic integer programming problem.

Zhang et al. (2018) design a multi-objective controller placement scheme that simultaneously addresses reliability, load balance and latency. They use the heuristic adaptive bacterial foraging optimization to solve this problem.

Dvir et al. (2018) study the wireless controller placement problem using a multi-objective optimization problem and measure the sensitivity of this placement to a variety of metrics.

In this paper we present the first maximum entropy based clustering algorithm to address ECP in wireless edge networks. A tutorial on deterministic annealing for the unfamiliar reader may be found in Rose (1998). We distinguish our algorithms from previous clustering approaches in that it is the first multi-objective clustering approach to the ECP problem and it does not require initialization. We found previous algorithms in literature that typically enjoy a fast speed such as Cuckoo search, GA, BFO, and other heuristics suffer from susceptibility to poor local optima solutions. On the other hand exact approaches like quadratic integer programming are too slow to be practical for real-world scenarios. Our algorithms address these shortcomings by leveraging their ability to sense and escape poor local minima and at the same time enjoy fast speed due to linear computational complexity in terms of parameters of the problem.

3. PROBLEM STATEMENT

Wireless networks can be illustrated by a graph as shown in Figure 1, in which the vertices are the network nodes and the edges represent the connection between them. One or multiple numbers of these vertices can be designated as a controller and ECP reduces to finding the optimal placement, and assignment of these controllers.

Leader-less and leader-based topologies are two popular schemes for placement of controllers (Qin et al., 2018). The distinction between the two is that in the former all pairs of controllers in the network directly communicate with each other while in the latter controllers only communicate with a leader controller.

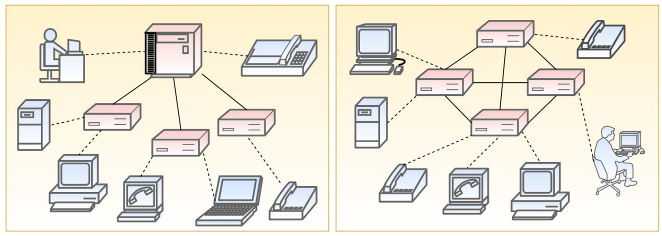


Fig. 1. Leader-based (left) and leader-less (right) schemes. Solid and dashed lines are respectively controller and node connections.

To cast this problem as a mathematical program we define $\mathcal{N}$ as the set of all edge nodes with $\mathbf{Card}(\mathcal{N}) = N$ and $\mathcal{N}_h$ as the set of edge nodes that can serve as controllers with $\mathcal{N}_h \subseteq \mathcal{N}$. Additionally, $\mathcal{X} = (x_i \in \mathbb{R}^d, i \in \mathcal{N})$ determines the position of edge nodes in the wireless network. We use $\overline{\mathcal{X}} = (\overline{x}_i \in \{0,1\}, i \in \mathcal{N}_h)$ to represent the controller placement policy. If we choose node i to be a controller then $\overline{x}_i = 1$ otherwise $\overline{x}_i = 0$. Similarly $\mathcal{Q} = (q_{ij} \in \{0,1\}, i \in \mathcal{N}, j \in \mathcal{N}_h)$ determines the controller assignment policy where $q_{ij} = 1$ if node i is assigned to controller j otherwise $q_{ij} = 0$. $\mathcal{Z} = (z_j \in \{0,1\}, j \in \mathcal{N}_h)$ determines the leader assignment policy in the leader-based scheme. $z_j = 1$ if controller j is the leader and $z_j = 0$ otherwise. $d_{ij} = d(x_i, x_j)$ encodes the communication delay between nodes i and j which we assume to be proportional to the squared Euclidean distance, i.e. $d_{ij} = \|x_i - x_j\|_2^2$.

3.1 Leader-less Case

In this setting all controllers communicate not only with edge nodes but also with each other. Thus we incur a controller synchronization cost between *all* pairs of controllers. We can express the optimal assignment as the solution of the following integer program:

$$\min_{\mathcal{Q},\overline{\mathcal{X}}} \quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_h} q_{ij}d_{ij} + \gamma \sum_{i,j\in\mathcal{N}_h} \overline{x}_i\overline{x}_j d_{ij} \sum_{k\in\mathcal{N}} q_{kj} \tag{1}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{N}_h} q_{ij} = 1 \quad \forall i \in \mathcal{N} \tag{2}$$

$$q_{ij} \leq \overline{x}_j \qquad \forall i,j \in \mathcal{N} \tag{3}$$

$$\overline{x}_i \in \{0,1\}, \quad i \in \mathcal{N}_h \tag{4}$$

$$q_{ij} \in \{0,1\}, \quad i \in \mathcal{N}, j \in \mathcal{N}_h, \tag{5}$$

The first term in the objective function corresponds to communication delay across all node-controller pairs. The second term shows the synchronization delay between controllers. Note that synchronization delay also depends on how many nodes are assigned to a certain controller. Constraint (2) ensures that each edge node is only assigned to one controller and constraint (3) ensures node assignments to a controller are only made to designated controller nodes. Parameter $\gamma \geq 0$ shows the relative importance of synchronization cost compared to delay cost.

3.2 Leader-based Case

The leader-based case is similar to the previous one, except that controllers synchronize only with the leader. We can express the optimal assignment in this setting as the solution to the following integer program:

$$\min_{(\mathcal{Q},\overline{\mathcal{X}},\mathcal{Z})} \quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_h} q_{ij}d_{ij} + \gamma \sum_{i\in\mathcal{N}_h}\sum_{j\in\mathcal{N}_h} \overline{x}_i z_j \left(N d_{ij}\right) \tag{6}$$

$$\text{s.t.} \quad \text{constraints (2)-(5)}$$

$$\sum_{j\in\mathcal{N}_h} z_j = 1 \tag{7}$$

$$z_j \in \{0,1\}, \quad j \in \mathcal{N}_h. \tag{8}$$

Constraint (7) ensures that there is exactly one leader controller in the network. Finding globally optimal solutions for leader-less and leader-based cases is an NP-hard problem (Singh and Srivastava, 2018).

4. SOLUTION APPROACH

In the deterministic annealing clustering setting, the expected *distortion* [1] can be defined as

$$D = \sum_{i=1}^{N} p(x_i) \sum_{j=1}^{m} p(y_j \mid x_i) D(x_i, y_j).$$

$X = \{x_i\}_{i=1}^N$ are the data points and $Y = \{y_j\}_{j=1}^m$ are cluster centroids. $p(y_j \mid x_i)$ is called the association probability [2] of point x_i with centroid y_j and $D(x_i, y_j)$ is the distortion measure which is typically chosen to be the squared Euclidean distance. We interpret $p(x_i)$ as the relative importance given to ith node and assume, if not otherwise indicated that $p(x_i) = \frac{1}{N}$. System entropy can be defined as $H = -\sum_{i=1}^N p(x_i) \sum_{j=1}^m p(y_j \mid x_i) \log p(y_j \mid x_i)$. We also define the system free energy as $F = D - TH$ where T is the system's so-called temperature. [3] Note that F can be viewed as the Lagrangian for the primary objective of minimizing D, with T being the Lagrange multiplier. The central iteration of DA can be summarized as sequentially optimizing F with respect to association probabilities and centroid locations.

[1] Distortion is an average weighted distance term, between nodes and centroids, that serves as our basic cost function.

[2] The weighting indicating that a node belongs to a particular centroid. For each node the sum of these associations over all centroids must equal one.

[3] A coefficient scaling the entropy term which indicates how important the entropy term is compared to the distortion term. We typically reduce this coefficient from a high value to a value close to zero.

4.1 Leader-less Case

For the purpose of adapting the DA clustering to the leader-less ECP problem we define the distortion measure as $D(x_i, y_j) = d(x_i, y_j) + \gamma \sum_{j'=1}^m d(y_j, y_{j'})$. In order to observe the relation to integer program (1)-(5) notice we can write total distortion as

$$\begin{aligned} D =& \sum_{i=1}^{N}\sum_{j=1}^{m} p(y_j \mid x_i) d(x_i, y_j) \\ &+ \gamma \sum_{j'=1}^{m}\sum_{j=1}^{m} \left(d(y_j, y_{j'}) \sum_{i=1}^{N} p(y_j \mid x_i) \right) \end{aligned} \tag{9}$$

This is objective function (1) with hard assignments q_{ij} replaced by the soft association probabilities. Setting partial derivatives of the free energy term with respect to association probabilities to zero and solving, yields solution:

$$p(y_j \mid x_i) = \frac{\exp\left(\frac{-D(x_i,y_j)}{T}\right)}{Z_i}, \; Z_i = \sum_{j=1}^{m} \exp\left(\frac{-D(x_i, y_j)}{T}\right)$$

Thus association probabilities have the celebrated Boltzmann distribution. Similarly setting derivatives with respect to the centroids y_j to zero leads to the following linear system of equations:

$$\eta y_j - \gamma \sum_{j' \neq j} y_{j'} = C_j, \quad j = 1, \dots m \tag{10}$$

where $\eta = \gamma(m-1)+1$ and $C_j = \sum_{i=1}^N p(x_i \mid y_j) x_i$. We may compute $p(x_i \mid y_j)$ using Bayes' rule. This gives us a linear system of md variables and md equations with m and d being respectively the number of centroids and the dimensionality of data. It is essential for the convergence of our clustering algorithm that this linear system of equations always has a solution.

Theorem 1. Given the linear system of equations in (10) with η and C_j defined as above, if $\gamma \neq \frac{1}{n-m}, \frac{1}{n-2m}$ then there always exists a unique solution $\{y_j\}_{j=1}^m$, where the coefficient matrix associated with the system of the equations is non-degenerate with determinant $\left(\frac{(\gamma m+1)^m(\gamma(n-m)-1))}{\gamma(n-2m)-1}\right)^d$.

Proof. 1 We can write the coefficient matrix associated with (10) as the block matrix $\Theta \in \mathbb{R}^{md\times md}$ with diagonal blocks equal to ηI and non-diagonal blocks equal to $-\gamma I$ such that $I \in \mathbb{R}^{d\times d}$. Dividing all rows by constant $-\gamma$ we get $\det(\Theta) = (-\gamma)^{md}\det(\bar{\Theta})$. $\bar{\Theta}$ is a block diagonal matrix with diagonal elements equal to αI and non-diagonal blocks equal to I with $\alpha = -\frac{\eta}{\gamma}$. Using straightforward linear algebra we can transform $\bar{\Theta}$ to an upper triangular matrix:

$$\bar{\Theta} \times \begin{bmatrix} I & 0 & \dots & 0 \\ \frac{-1}{\alpha+n-2}I & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{-1}{\alpha+n-2}I & \frac{-1}{\alpha+n-3}I & \dots & I \end{bmatrix} = \begin{bmatrix} \beta_1 I & \times & \dots & \times \\ 0 & \beta_2 I & \dots & \times \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \beta_m I \end{bmatrix}$$

Where $\beta_i = \alpha - \frac{n-i}{\alpha+n-i-1}$ and $\det(\bar{\Theta}) = \prod_{i=1}^m (\beta_i)^d$. We can use telescoping to further simplify the product to $\left(\frac{(\alpha-1)^m(\alpha+n-1)}{\alpha+n-(m+1)}\right)^d$. This will give:

$$\det(\Theta) = \left(\frac{(\gamma m + 1)^m (\gamma(n-m) - 1))}{\gamma(n-2m) - 1} \right)^d$$

which is nonzero for $\gamma \neq \frac{1}{n-m}$ and well defined for $\gamma \neq \frac{1}{n-2m}$.

The resulting DA clustering algorithm for this case is given in Algorithm 1.

Algorithm 1: ECP-LL

Set max # of clusters K_{max} and min temperature T_{min};
Initialize: $T \leftarrow \infty, K = 1, y_1 = \sum_{i=1}^N x_i p(x_i)$;
while *Convergence test* **do**
  $p(y_j \mid x_i) \leftarrow \exp\left(-\frac{d(x_i,y_j)+\gamma\sum_{j'=1}^m d(y_j,y_{j'})}{T}\right)/Z_i, \ \forall i, j;$
  Solve: $\eta y_j^{new} - \gamma \sum_{j' \neq j} y_{j'}^{new} = \sum_{i=1}^N p(x_i \mid y_j) x_i, \ \forall j;$
  Update: $y_j \longleftarrow y_j^{new}, \ \forall j;$
  if $T \leq T_{min}$ **then**
    break;
  else
    Annealing: $T \longleftarrow \alpha T, \quad \alpha \in (0,1);$
    Perturb $y_j, \ \forall \ j;$
  end
end
$y_j \longleftarrow \arg\min_{x_i \in \mathcal{N}_h} d(x_i, y_j), \ \forall j;$

For the convergence test we stop at iteration τ if $\|F_\tau - F_{\tau-1}\| < \delta$ for some predetermined tolerance level $\delta > 0$. At perturbation step we replace y_j with $y_j + \epsilon$ and $y_j - \epsilon$ for some small and random perturbation vector $\epsilon \in \mathbb{R}^d$. Perturbed centroids will automatically merge back together if codebook Y needs not to expand, otherwise they separate further away (Rose, 1998). In the last line of Algorithm 1 we designate the closest valid node to each centroid as a controller.

The iteration complexity for this algorithm depends on (a) calculation of mutual squared Euclidean distances between x_i, y_j for $i \in \{1, \ldots, N\}, j \in \{1, \ldots, m\}$, (b) similar calculation of mutual distances between centroids, (c) calculation of association probabilities and (d) solving the linear system of equations. The complexities for these operations are respectively, $O(NK_{max}d)$, $O(K_{max}^2 d)$, $O(K_{max}N)$ and $O(K_{max}^3 d^3)$. For large N these terms are dominated by $O(NK_{max}d)$, thus for a maximum number of iterations τ the algorithmic computational complexity for the leader-less case is $O(\tau N K_{max} d)$ which is linear in data size, maximum number of clusters and dimensionality of data.

4.2 Leader-based case

In order to adapt DA to the leader-based ECP problem we define an appropriate distortion measure by:

$$D(x_i, y_j) = d(x_i, y_j) + \gamma \min_{j \in \{1,\ldots,m\}} \sum_{j'=1}^m d(y_j, y_{j'}) \quad (11)$$

Similarly we can consider the weighted total distortion as:

$$D = \sum_{i=1}^N \sum_{j=1}^m p(y_j \mid x_i) d(x_i, y_j) + \gamma \min_{j \in \{1,..,m\}} \sum_{j'=1}^m N d(y_j, y_{j'})$$

In order to observe its relation to MINLP objective function, notice (6) is equivalent to the following objective function:

$$\min_{(\mathcal{Q},\overline{X})} \quad \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_h} q_{ij} d_{ij} + \gamma \min_{j \in \{1,\ldots,m\}} \sum_{i \in \mathcal{N}_h} \overline{x}_i \left(N d_{ij} \right)$$

To establish this equivalence, we used the relationship that for $\mathcal{Z} = \{z_j \in \{0,1\} \mid \sum_{j=1}^m z_j = 1\}$ then $\min_{\mathcal{Z}} \sum_{j=1}^m z_j \psi_j = \min_{j \in \{1,\ldots,m\}} \psi_j$ where $\psi_j \in \mathbb{R} \ \forall j \in \{1, 2, \ldots, m\}$.

Following the steps in the leader-less case we obtain the association probabilities and centroid update rules as follows:

$$p(y_j \mid x_i) = \frac{\exp\left(\frac{-d(x_i,y_j)}{T}\right)}{Z_i}, \ Z_i = \sum_{j=1}^m \exp\left(\frac{-d(x_i,y_j)}{T}\right)$$

$$y_j = \frac{\gamma N y_{j^*} + \sum_{i=1}^N p(y_j \mid x_i) x_i}{\gamma N + \sum_{i=1}^N p(y_j \mid x_i)} \quad y_j \neq y_{j^*} \quad (12)$$

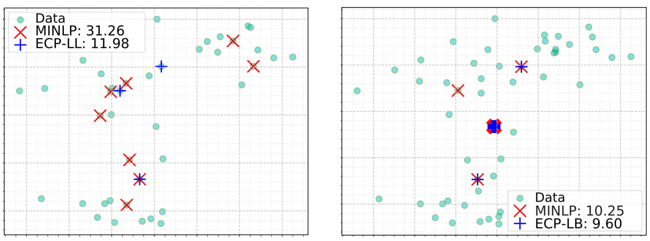
$$y_{j^*} = \frac{\gamma N \sum_{j' \neq j^*} y_{j'} + \sum_{i=1}^N p(y_{j^*} \mid x_i) x_i}{(m-1)\gamma N + \sum_{i=1}^N p(y_{j^*} \mid x_i)} \quad (13)$$

where $j^* = \arg\min_{\mathrm{j} \in \{1,\ldots,\mathrm{m}\}} \sum_{j'=1}^m d(y_j, y_{j'})$ is the index of the leader centroid. The resulting maximum entropy based clustering algorithm for the leader-based topology is similar to the leader-less case in Algorithm 1 except that update rules for association probabilities and centroids is replaced with equations above.

Using a similar procedure the computational complexity can be found as $O(NK_{max}d) + O(K_{max}^2 d) + O((N + K_{max})d) + O(NK_{max})$. For a maximum of τ iterations and large N this is again dominated by $O(\tau N K_{max} d)$.

5. RESULTS

In order to evaluate the performance of our algorithms we compare their final costs with the integer programs (1)-(5) and (6)-(8). We use the state of the art MINLP solver BARON to draw this comparison (Sahinidis, 1996). We used Gaussian distribution to generate our data with K as the number of Gaussian clusters within the data.



(a) (b)

Fig. 2. (a) ECP-LL vs. MINLP (b) ECP-LB vs. MINLP

Superior performance of our clustering algorithms can be observed even in small problem instances like in Figure 2. While BARON is stuck in a poor local optimum with an excessive number of controllers, ECP-LL has managed to achieve a considerably lower objective value with fewer controller placements. The slightly fatter markers in (b) indicate the leader controllers.

In Figure 3, as an immediate result of avoiding controller synchronization cost, placements become more packed as γ increases.

Figure 4 shows sensitivity to different hyper-parameters for ECP-LL algorithm. (a) shows as γ increases the optimal objective value also increases and stays relatively constant for very large values of γ. (b) also shows a similar pattern that as γ increases ECP-LL places fewer controllers in the network. (c) shows the optimal value for hyper-parameter K_{max}. We validate that the optimal value of K_{max} is the number of inherent clusters in the dataset. (d) Shows the the values of non-projected and projected [4] solutions versus the number of iterations. The projected solution is obtained by making association probabilities hard and then projecting the solution centroids onto the data set. We observed that the two converge to the same value across most scenarios.

Table 4 compares performance of ECP algorithms against BARON. While the former by far outperforms the latter in terms of total run time, the difference in accuracy is emphasized as problem size increases. Figure 5 illustrates how run time grows linearly as a function of data size and number of clusters.

6. CONCLUSION

In this work we introduced two multi-objective maximum entropy based clustering algorithms for the problem of Edge Controller Placement in wireless communication networks. Our algorithms address the shortcoming of current approaches in the SDN literature that are either fast at the expense of sub-par solution quality or provide high quality solutions with run times that are impractical for real-world scenarios. ECP-LL and ECP-LB each address a different controller placement topology, and their design is inspired by a Mixed Integer Nonlinear Program. We show that our algorithms outperform state of the art MINLP solver, BARON in both speed and accuracy. The iteration computational complexity for these algorithms is $O(NK_{max}d)$ which is linear in data size, maximum number of clusters and dimensionality of data.

REFERENCES

Alshamrani, A., Guha, S., Pisharody, S., Chowdhary, A., and Huang, D. (2018). Fault Tolerant Controller Placement in Distributed SDN Environments. In *2018 IEEE International Conference on Communications (ICC)*, 1–7. IEEE. doi:10.1109/ICC.2018.8422593. URL https://ieeexplore.ieee.org/document/8422593/.

Das, T. and Gurusamy, M. (2018). INCEPT: INcremental ControllEr PlacemenT in Software Defined Networks. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, 1–6. IEEE. doi:10.1109/ICCCN.2018.8487454. URL https://ieeexplore.ieee.org/document/8487454/.

[4] At the last step of algorithm we mapped centroids onto the closest edge node available. We call such a solution projected, otherwise we call the solution non-projected.

Dvir, A., Haddad, Y., and Zilberman, A. (2018). Wireless controller placement problem. In *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, 1–4. IEEE. doi:10.1109/CCNC.2018.8319228. URL http://ieeexplore.ieee.org/document/8319228/.

He, J., Lan, M., Tan, C.L., Sung, S.Y., and Low, H.B. (2004). Initialization of cluster refinement algorithms: A review and comparative study. In *IEEE International Conference on Neural Networks - Conference Proceedings*, volume 1, 297–302. doi:10.1109/ijcnn.2004.1379917.

Heller, B., Sherwood, R., and McKeown, N. (2012). The controller placement problem. In *Proceedings of the first workshop on Hot topics in software defined networks - HotSDN '12*, 7. ACM Press, New York, New York, USA. doi:10.1145/2342441.2342444. URL http://dl.acm.org/citation.cfm?doid=2342441.2342444.

Kuang, H., Qiu, Y., Li, R., and Liu, X. (2018). A Hierarchical K-Means Algorithm for Controller Placement in SDN-Based WAN Architecture. In *2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, 263–267. IEEE. doi:10.1109/ICMTMA.2018.00070. URL https://ieeexplore.ieee.org/document/8337381/.

Li, F. and Xu, X. (2018). A Discrete Cuckoo Search Algorithm for the Controller Placement Problem in Software Defined Networks. In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 292–296. IEEE. doi:10.1109/IEMCON.2018.8614785. URL https://ieeexplore.ieee.org/document/8614785/.

Liao, J., Sun, H., Wang, J., Qi, Q., Li, K., and Li, T. (2017). Density cluster based approach for controller placement problem in large-scale software defined networkings. *Computer Networks*, 112, 24–35. doi:10.1016/J.COMNET.2016.10.014. URL https://www.sciencedirect.com/science/article/abs/pii/S1389128616303620.

Liu, J., Shi, Y., Zhao, L., Cao, Y., Sun, W., and Kato, N. (2018). Joint Placement of Controllers and Gateways in SDN-Enabled 5G-Satellite Integrated Network. *IEEE Journal on Selected Areas in Communications*, 36(2), 221–232. doi:10.1109/JSAC.2018.2804019. URL http://ieeexplore.ieee.org/document/8286925/.

Lu, J., Zhang, Z., Hu, T., Yi, P., and Lan, J. (2019). A Survey of Controller Placement Problem in Software-Defined Networking. *IEEE Access*, 7, 24290–24307. doi:10.1109/ACCESS.2019.2893283. URL https://ieeexplore.ieee.org/document/8618449/.

Papa, A., De Cola, T., Vizarreta, P., He, M., Mas Machuca, C., and Kellerer, W. (2018). Dynamic SDN Controller Placement in a LEO Constellation Satellite Network. In *2018 IEEE Global Communications Conference (GLOBECOM)*, 206–212. IEEE. doi:10.1109/GLOCOM.2018.8647843. URL https://ieeexplore.ieee.org/document/8647843/.

Qin, Q., Poularakis, K., Iosifidis, G., Kompella, S., and Tassiulas, L. (2018). SDN Controller Placement With Delay-Overhead Balancing in Wireless Edge Networks. *IEEE Transactions on Network and Service Management*, 15(4), 1446–1459. doi:10.1109/TNSM.2018.2876064. URL https://ieeexplore.ieee.org/

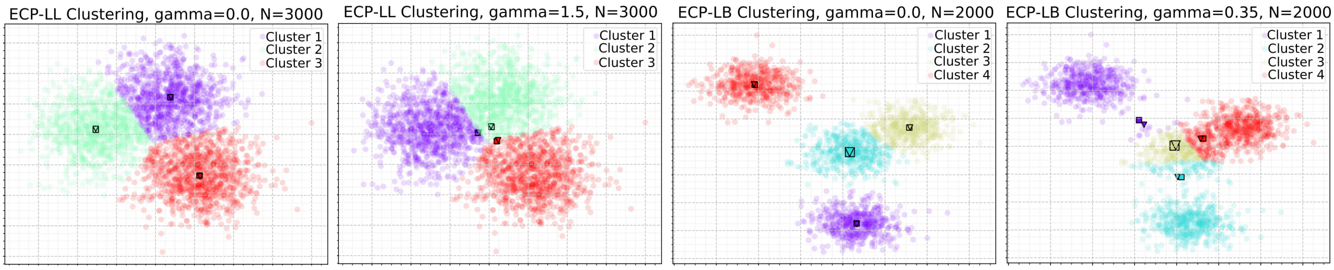


Fig. 3. Controller placement sensitivity to parameter γ

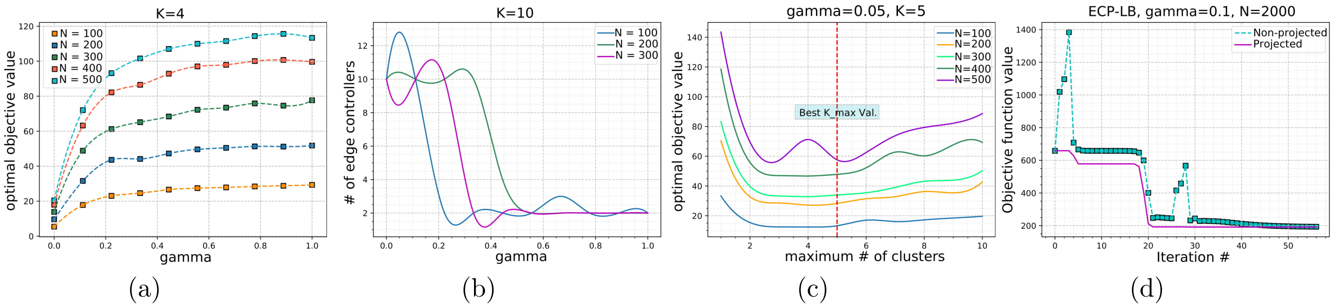


Fig. 4. (a) γ vs. optimal objective value, (b) γ vs. optimal number of controllers, (c) hyper-parameter K_{max} vs. optimal objective value and (d) Iteration number vs. projected and non-projected solutions objective function values

Table 1. Tuples (ECP vs. BARON) include run time (sec.), obj. value and number of controllers triplets as a function of size of dataset N and number of clusters K with $\gamma = 0.1$

LB	N=20	N=60	LL	N=20	N=60
K=2	(0.9,1.9,5), (2.1,6.2,4)	(2.3,6.2,2), (906.8,15.2,3)	K=2	(0.3,7.1,2), (622.2,15.3,4)	(0.7,17.50,2), (627.5,585,55)
K=4	(1.4,2.8,6), (14.8,4.3,5)	(4.5,6.7,7), (604.1,9.1,6)	K=4	(0.4,8.6,7), (614.5,12.9,5)	(1,16.8,7), (624.1,233.8,48)
K=6	(2.06,3.52,8), (63.59,4.88,3)	(5.23,9.10,8), (612.3,12.7,5)	K=6	(0.6,7.7,4), (605.2,13.7,6)	(1.6,23.2,4), (638.9,95.9,13)
K=8	(2.8,2.7,11), (16.8,3.7,4)	(7,11.7,11), (604.5,15,5)	K=8	(0.8,6.5,4), (1630.9,9.3,6)	(1.9,25.8,4), (618.6,390.7,54)
K=10	(3.4,3.1,8), (68.9,4.9,4)	(10.1,7.3,12), (607.6,12.2,5)	K=10	(1.1,6.8,2), (602,10.3,5)	(2.5,19.5,4), (621.6,81.9,15)

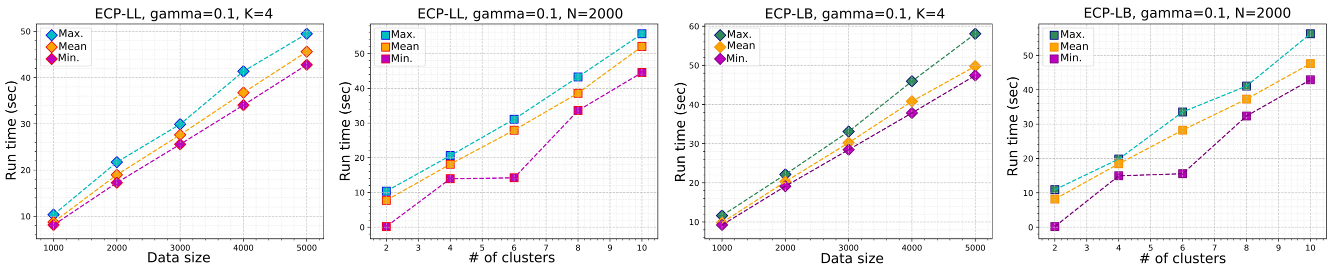


Fig. 5. ECP algorithms run time vs. # of clusters and data size.

document/8491378/.

Rose, K. (1998). Deterministic Annealing for Clustering, Compression, Classification, Regression, and Related Optimization Problems. Technical report, California Institute of Technology. URL https://pdfs.semanticscholar.org/4ce8/bc485df9ac987f18d99c7af1d95f9cbea6b2.pdf.

Sahinidis, N.V. (1996). BARON: A general purpose global optimization software package. *Journal of Global Optimization*, 8(2), 201–205. doi:10.1007/bf00138693.

Singh, A.K. and Srivastava, S. (2018). A survey and classification of controller placement problem in SDN. *International Journal of Network Management*, 28(3), e2018. doi:10.1002/nem.2018. URL http://doi.wiley.com/10.1002/nem.2018.

Zhang, B., Wang, X., and Huang, M. (2018). Multi-objective optimization controller placement problem in internet-oriented software defined network. *Computer Communications*, 123, 24–35. doi:10.1016/J.COMCOM.2018.04.008. URL https://www.sciencedirect.com/science/article/pii/S0140366416307241.

Zhiyang Su and Hamdi, M. (2015). MDCP: Measurement-Aware Distributed Controller Placement for Software Defined Networks. In *2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS)*, 380–387. IEEE. doi:10.1109/ICPADS.2015.55. URL http://ieeexplore.ieee.org/document/7384318/.