

Figure 1. Request routing in Base Station based Mobile-CDN.

mechanism in this architecture, the mobile client sends a content requests to the attached base station through back-haul links. If this requested content is included by the attached base station, then it is sent back to the client (*Cache Hit*). On the other hand, in the case of a Cache Miss in the storage unit of the attached base station, this content request is directed to the other neighbor base stations until it is found. Therefore, here, there are two different possibilities which are determined according to the cache hit/miss situation in the attached base station. In the next step, the requested content can be received from any neighbor base station. Additionally, if the requested content cannot be received from any base station, it is finally taken from the origin server. As a result of these steps, the based station based Mobile-CDN architecture has two different types of the content request according to the originating points. These can be called as mobile client and base station originating content requests as shown in Fig. 1.

To emphasize the performance defects, we adopt the above steps to more general Mobile-CDN architecture with a specific scenario as shown in Fig. 3a. In this scenario, we assume that the base stations 1, 2, and 4 receive content requests from different mobile clients. These three base stations direct the requests to neighbor base stations in the case of storage misses. With these redirections, the base station 3 can receive content requests from the base stations 1, 2, and 4. These content requests increase the load on the base station 3. On the other hand, the base station 5 can only

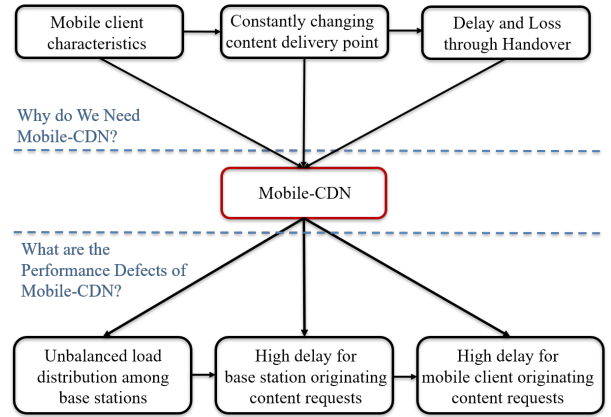


Figure 2. Mobile-CDN Summary.

receive base station originating content request from the base station 4. This situation leads to unbalanced load distribution among the base stations. Therefore, the base station originating requests in base station 3 experience a high delay because of these cache misses and request redirections. Also, the latency of mobile client originating requests in base station 3 increases due to the density of the waiting base station originating requests in the queue. We are also summarized the requirement reasons (Section 1) and performance defects (Section 1.1) of Mobile-CDN in Fig. 2.

1.2. Related Works

In the current literature, there are different Mobile-CDN solutions. The authors in [8] manage the transmission queues which receive requests from other base stations to guarantee the queuing delay. An optimization algorithm for the content placement and request redirection problems is proposed in [9]. In [10], the deployment of Mobile-CDN serving points is motivated with their benefits for mobile network operators. The LinkNYC urban communications network is shifted to the mobile content-delivery network from traditional IP network in [11]. A distributed solution to content replication by considering dynamic environments and load balancing is proposed in [12]. The authors in [13] uses the special mobile devices as caching servers in Mobile-CDN architecture. A user-operated content distribution network for the mobile smartphone is proposed in [14] to achieve widespread distribution of heavy content. On the other hand, the effects of the base station originating content requests on the load distributions and request latencies are not investigated by any work. Through this article, we aim to implement the drones in Mobile-CDN architecture as aerial content delivery points. In this way, we can balance the load and reduce the latencies causing from base station originating content requests.

As summarized in [15], drones can be used in the cellular network to support the small cells during intensive traffic. But, the drones are not yet implemented in a Mobile-CDN architecture for content delivery by any work

1.3. Article Contributions

In this article, we aim to create aerial content delivery points at the sky by deploying drones in Mobile-CDN architecture. This offers a more effective and inexpensive solution without changing the Mobile-CDN infrastructure. The drone deployment also enables load balancing by taking over requests from base stations. Additionally, the mobile client and base station originating latencies can be reduced with the drone-aided content delivery. With a more detailed description, in this work, we model each base station according to the M/M/c queuing system to determine the queuing loads. After the base station load determination, we can detect the heavily-loaded base station to transfer the drones by taking advantage of the Barabasi-Albert model. On the other hand, we model the drones based on the M/M/c/c queuing system. Thanks to this model, we can utilize the Erlang-B parameter to determine the blocking probability of drones due to the fact that they have less tolerance and capacity compared to the base stations. We manage these operations through the controller in three modules as load determination, assistant drone requirement determination, and additional drone transfer probability as explained in the next sections.

We organize the rest of the paper as follows: we give the proposed drone-aided Mobile-CDN architecture in Section II. The performance of the proposed approach is evaluated in Section III. The Section IV concludes the paper.

2. Drone-Aided Mobile-CDN Architecture

We consider a network architecture where base stations, mobile clients, origin server, assistant drones, and Ground Control Center (GCC) are deployed as shown in Fig. 3b. Each *base station* include a storage unit to replicate the content from the origin server. The *mobile clients* send a request to the attached base station to reach the required content. Also, each base station is represented with a 6-bit Base Station Identity Code (BSIC). In this model, we use this code also to specify the base stations to which drones will be transferred. The *origin server* includes the main web contents and these contents are replicated to the storage units of the base stations according to the push-based mechanism. The *assistant drones* are considered as aerial content delivery points in addition to the base stations. These assistant drones help to related base stations by replying to the content requests. In this

way, drones can take over the load on base stations to obtain a load-balanced Mobile-CDN architecture with reduced latencies. These drones use the microSD card as a storage unit like commercial drones. With SD cards, we can obtain scalable capacity with less cost. The assistant drones should be registered to the network before adding. The IP addresses, MAC addresses, and placement IDs are added during the registration process of drones. Here, the placement IDs are the BSICs of the base stations. Also, note that the drones are connected to the mobile clients through the radio interface. The GCC is the main control point for the drones with two modules as load determination and assistant drone requirement determination as shown in Fig. 3b.

In the proposed model, the GCC first observes the queuing loads of base stations [16], [17]. In this way, we can decide the loaded base stations to sent assistant drones as aerial content delivery points. For this aim, we also utilize the *Barabasi-Albert Model*. After the base station determination, we analyze the *Erlang-B* parameter of the drones to decide the request number to be serviced without blocking and transfer additional drones. The details of these procedures are explained in the following subsections.

2.1. Load Determination Module

In this module, we model the base stations based on the M/M/c queuing system by only considering the base station originating content requests. Here, a base station receives the content requests from the n neighbor base station with arrival rates as $\lambda_1, \lambda_2, \dots, \lambda_n$ according to the Poisson distribution. Thus, the total arrival rate for this base station is represented as λ . But, the service rate of the base station equals for all requests and is represented as μ . Here, the steady state system size probabilities can be found by using Eq. 1.

$$p_n = \begin{cases} \frac{\lambda^n}{n! \mu^n} p_0, & (0 \leq n < c) \\ \frac{\lambda^n}{c^{n-c} c! \mu^n} p_0, & (n \geq c), \end{cases} \quad \forall i = 1, \dots, N \quad (1)$$

In Eq. 1, p_0 is the probability that the queue of the base station is empty and it is found as given in Eq. 2.

$$p_0 = \left(\sum_{n=0}^{c-1} \frac{\lambda^n}{n! \mu^n} + \sum_{n=c}^{\infty} \frac{\lambda^n}{c^{n-c} c! \mu^n} \right)^{-1}, \quad \forall i = 1, \dots, N \quad (2)$$

Then, we can find the queue load of the corresponding base station (L_q) by using Eq. 3.

$$L_q = \left(\frac{r^c \rho}{c!(1-\rho)^2} \right) p_0, \quad r = \lambda/\mu \quad (3)$$

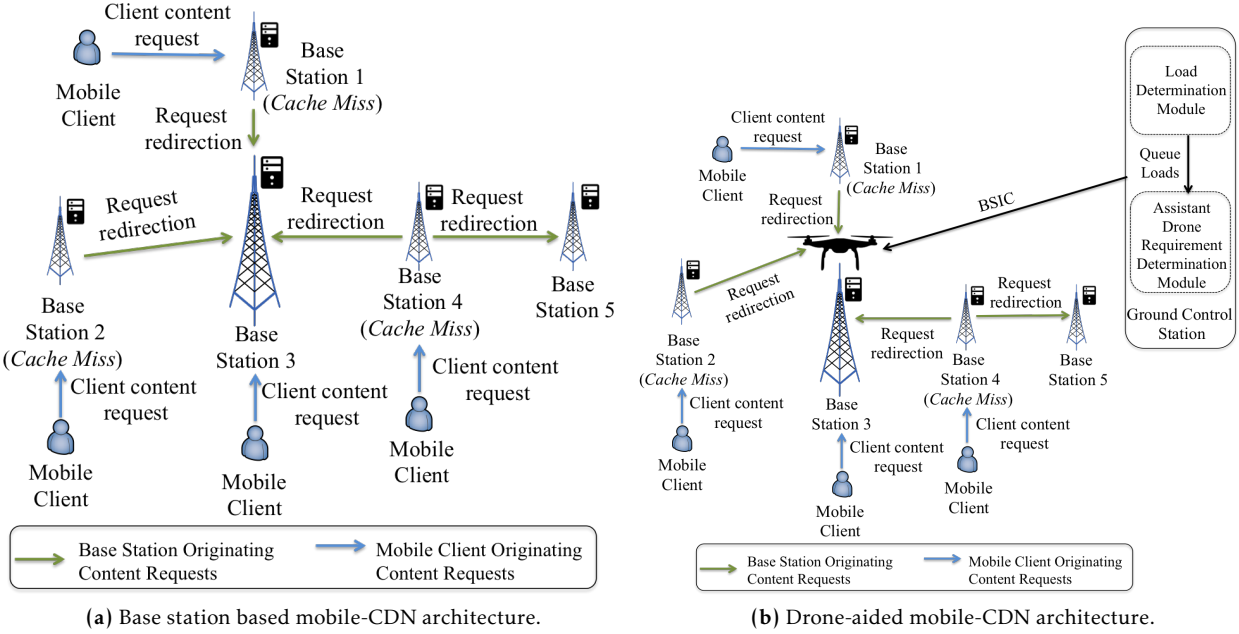


Figure 3. Mobile-CDN architectures for the base station based and proposed approaches

In this module, we can determine the queue loads of base stations by observing L_q parameter. In the next step, we select the loaded base stations among them sequentially by utilizing the Barabasi-Albert Model as explained in the upcoming subsection.

2.2. Assistant Drone Requirement Determination Module

The *Barabasi-Albert* is one of the important models to explain the scale-free networks. If a network is inhomogeneous and some nodes experience more load through connections, then it is evaluated as a scale-free network instead as random.

By considering these, we can represent the base stations as nodes on a graph. In this graph, the degree of a node represents the number of base stations neighbor to it. Therefore, some nodes have higher degrees compared to others. Accordingly, the created graph follows the *power-law degree distribution*. Also, as explained above, the base stations can accept the content requests as client and base station originating. Here, we only consider the base station originating content requests. According to these requests, some base stations can accept more request from other base stations with an increasing degree. As a result, we can represent the base station graph as a scale-free network and we can utilize the Barabasi-Albert Model.

By keeping these explanations in mind, the Barabasi-Albert Model uses two mechanisms as *growth* and *preferential attachment*. In this paper, we use these two mechanisms to determine the loaded base stations to which assistant drones will be transferred. The

Growth refers that the number of nodes on the network increases over time. Here, we consider the increasing nodes as assistant drones to diminish the load of the corresponding base station. The *Preferential Attachment* explain that the newly added nodes ordinarily connect to nodes with large *degrees*. Here, we consider that the assistant drones are transferred to the base stations which have large *loads* instead of *degrees*. Therefore, we calculate that the base station with load L_i requires an assistant drone with the P_{L_i} probability as $P_{L_i} = \frac{L_i}{\sum_{i=1}^{\infty} L_i}, \forall i = 1, \dots, N$.

How do We Handle Low Capacity Drones? As explained above, we utilize the Barabasi-Albert Model to determine the base stations which require assistant drones. The Barabasi-Albert Model adds a node to the network one at a time. But here, we determine the additional assistant drones according to their capacities instead of continuous insertion. To achieve this, we model the drones with the M/M/c/c queuing system.

The M/M/c/c queuing system does not include a queue. Therefore, if all sections of the drones are already in use, then the new content request is blocked. The aborted requests are not kept in the queuing system. The main reason for choosing this queuing system is that the drones have lower capacity and energy compared to the base stations. In this model, λ_d and μ_d represent the arrival and service rates of the drones, respectively. Also, the request flows of the drones can be divided into three components as carried, offered, and blocked requests. Here, the carried requests are actually processed by the drone. On the other hand,

the offered requests are the amount of request to be handled by the drone in the absence of any restrictions. Then, the blocked requests are the difference between the carried and offered requests.

In this model, our aim is to find the blocked request probabilities of the assistant drones. Because these values give us the possibility of transferring a new assistant drone to the corresponding base station. Accordingly, the blocking probability of the incoming content requests is equal to find the *Erlang-B* parameter as given with Eq. 4.

$$p_b = \frac{r^c}{c!} \left/ \sum_{i=0}^c \frac{r^i}{i!} \right., \quad r = \lambda/\mu, \quad \forall i = 1, \dots, N \quad (4)$$

As a summary, if the queuing load of the base station and *Erlang-B* of drone are still high then we transfer a new drone to this base station according to the Barabasi-Albert Model until obtaining a load-balanced Mobile-CDN architecture. This methodology also enable us to obtain an energy efficient Mobile-CDN architecture with the help of drones. Here, the drones are transferred according to the Barabasi-Albert Model. This leads that the drones use their energy effectively without being too loaded.

3. Performance Evaluation

3.1. Simulation Environment

The performance of the proposed method is evaluated through MATLAB-Simulink environment. In these simulations, we take the radius, Tx power, and carrier frequency of base stations as 1km, 46dBm, and 2GHz, respectively [18]. Also, the bandwidth is taken as 10MHz with 6dB noise rise. Moreover, the users move according to the Gauss-Markov mobility model. The mobile clients transfer the content requests to the attached base station. The content request is transferred to the other base stations when storage miss is encountered. As explained above, the base stations and drones are modeled with the M/M/c and M/M/c/c queuing systems, respectively. Also, we use the MAVProxy to control the drones and Cygwin software to emulate the UNIX commands of this controller. Additionally, the SITL simulator is used to create ArduCopter instance as a content delivery point at the sky.

Also, to evaluate the proposed approach, we investigate the delays of the base station and mobile client originating requests according to the arrival request number per unit time (λ). To find these delays,

we use queuing delay formula as $W_q = \left(\frac{r^c \rho}{c! c \mu (1-\rho)^2} \right) p_0$, $r =$

λ/μ . In this equation, p_0 is calculated by using Eq. 2. Additionally, in these simulations, we compare the

proposed drone-aided Mobile-CDN architecture with the Base Station based Mobile-CDN architecture.

3.2. Simulation Results

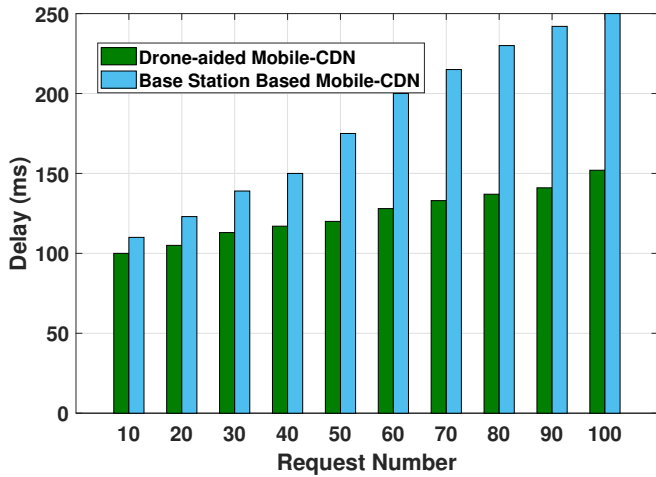
We first observe the delay of the base station originating content requests according to the arrival request rate. The request of the mobile client is redirected to other base stations during a storage miss. The base station observes higher queuing load as given in Eq. 3 if it receives a more redirected request from other base stations. Therefore, the delay of the base station originating requests increases with the storage misses and request redirections. In our proposed approach, we can reduce this delay by deploying drones for content delivery based on Eq. 4. Thus, as shown in Fig. 4a, we reduce this delay as 18% compared to the conventional Base Station based Mobile-CDN architecture. We also investigate the delay of the mobile client originating requests according to the arrival request rate. The high-density of the base station originating client requests in the queue increases the delay of mobile client originating requests. In the proposed drone-aided approach, we reduce this delay 25% compared to the Base Station based Mobile-CDN architecture as shown in Fig. 4b.

In this article, we define the unanswered requests of the mobile clients as request failures. These requests wait more in the queue of the base stations due to excessive load. Accordingly, the validity time of the requests expires and the requests are dropped. Here, we investigate the number of request failures which are observed in the busy hour of a server. The load of a server is high during a busy hour in the Base Station based Mobile-CDN architecture. But, in the proposed approach, we can reduce this load by deploying drones for content delivery. Therefore, as shown in Fig. 4c, we observe 22% less request failure compared to the Base Station based Mobile-CDN architecture.

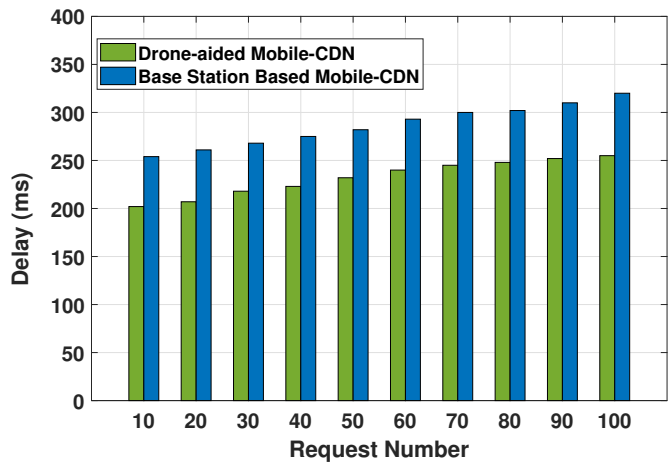
Lastly, we observe the packet losses for the proposed drone-aided and Base Station based Mobile-CDNs. In the Base Station based Mobile-CDN, we observe high packet losses and request drops due to queue overload. On the other hand, we can balanced load distribution among the base stations thanks to the drone deployment. Correspondingly, the proposed drone-aided Mobile-CDN has %31 less packet loss compared to the Base Station based Mobile-CDN architecture.

4. Conclusion

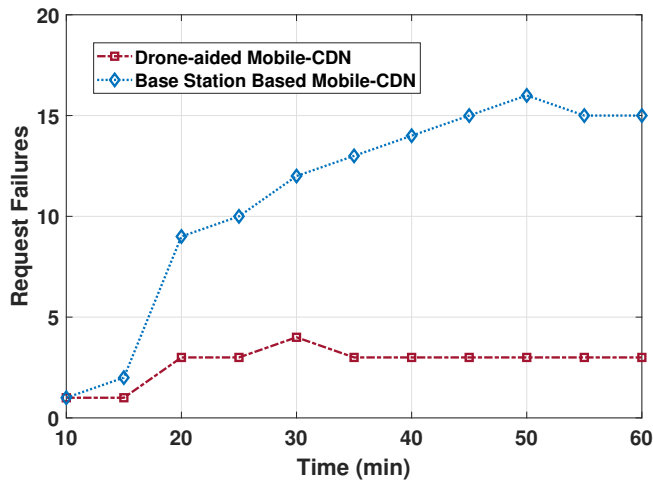
In this paper, we propose deploying the drones as aerial content delivery points in the Mobile-CDN architecture. This enables us to obtain a load balanced Mobile-CDN architecture with reduced latencies. We modeled the base stations with the M/M/c queuing



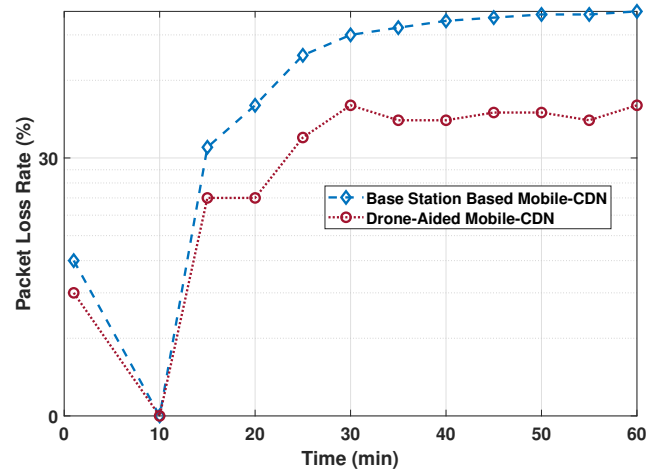
(a) Delay of base station originating content requests.



(b) Delay of mobile client originating content requests.



(c) Request failure analysis.



(d) Packet loss analysis.

Figure 4. Performance Evaluation Results.

system to obtain the queuing loads. Then, we utilize the Barabasi-Albert model to detect the loaded base stations gradually. Also, we model the drones according to the $M/M/c/c$ queuing system to calculate the Erlang-B parameters which are used in the determination of additional drones. The capacity and energy constraints of drones are the main reasons to select different models from the base stations.

References

- [1] BILEN, T., AHMADI, H., CANBERK, B. and DUONG, T.Q. (2022) Aeronautical networks for in-flight connectivity: A tutorial of the state-of-the-art and survey of research challenges. *IEEE Access* **10**: 20053–20079. doi:10.1109/ACCESS.2022.3151658.
- [2] (Cisco Visual Networking Index: Forecast and Trends, 2017-2022).
- [3] BILEN, T. and CANBERK, B. (2019) Deliver the content over multiple surrogates: A request routing model for high bandwidth requests. *Computer Communications* **146**: 39–47. doi:https://doi.org/10.1016/j.comcom.2019.07.009, URL <https://www.sciencedirect.com/science/article/pii/S0140366418309885>.
- [4] BILEN, T. and CANBERK, B. (2019) Handover-aware content replication for mobile-cdn. *IEEE Networking Letters* **1**(1): 10–13. doi:10.1109/LNET.2018.2873982.
- [5] BILEN, T. and CANBERK, B. (2015) Binary context tree based middleware for next generation context aware networks. In *2015 3rd International Conference on Future Internet of Things and Cloud*: 93–99. doi:10.1109/FiCloud.2015.67.
- [6] HAN, T., ANSARI, N., WU, M. and YU, H. (2013) On accelerating content delivery in mobile networks. *IEEE Communications Surveys Tutorials* **15**(3): 1314–1333. doi:10.1109/SURV.2012.100412.00094.

- [7] J.LIU, Q.YANG and G.SIMON (2017) Delay oriented content placement and request redirection for mobile-cdn : 498–501doi:10.1109/LCN.2017.28.
- [8] LIU, J., YANG, Q. and SIMON, G. (2017) Joint optimization of content placement and request redirection in mobile-cdn : 169–176doi:10.23919/INM.2017.7987277.
- [9] YOUSAF, F.Z., LIEBSCH, M., MAEDER, A. and SCHMID, S. (2013) Mobile cdn enhancements for qoe-improved content delivery in mobile operator networks. *IEEE Network* 27(2): 14–21. doi:10.1109/MNET.2013.6485091.
- [10] SINKY, H. and HAMDAROU, B. (2016) Cloudlet-aware mobile content delivery in wireless urban communication networks : 1–7doi:10.1109/GLOCOM.2016.7841664.
- [11] LA, C.A., MICHIARDI, P., CASETTI, C., CHIASSERINI, C.F. and FIORE, M. (2012) Content Replication in Mobile Networks. *IEEE Journal on Selected Areas in Communications* 30(9): 1762–1770. doi:10.1109/JSAC.2012.121021.
- [12] KANG, H.J. and KANG, C.G. (2014) Mobile device-to-device (d2d) content delivery networking: A design and optimization framework. *Journal of Communications and Networks* 16(5): 568–577. doi:10.1109/JCN.2014.000095.
- [13] PSARAS, I., SOURLAS, V., SHTEFAN, D., RENE, S., ARUMAITHURAI, M., KUTSCHER, D. and PAVLOU, G. (2017) On the feasibility of a user-operated mobile content distribution network. In *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*: 1–9. doi:10.1109/WoWMoM.2017.7974295.
- [14] ZHANG, L., FAN, Q. and ANSARI, N. (2018) 3-d drone-base-station placement with in-band full-duplex communications. *IEEE Communications Letters* 22(9): 1902–1905. doi:10.1109/LCOMM.2018.2851206.
- [15] BILEN, T., AYVAZ, K. and CANBERK, B. (2018) Qos-based distributed flow management in software defined ultra-dense networks. *Ad Hoc Networks* 78: 24–31. doi:https://doi.org/10.1016/j.adhoc.2018.05.002, URL <https://www.sciencedirect.com/science/article/pii/S1570870518301963>.
- [16] BILEN, T., DUONG, T.Q. and CANBERK, B. (2016) Optimal enodeb estimation for 5g intra-macrocell handover management. In *Proceedings of the 12th ACM Symposium on QoS and Security for Wireless and Mobile Networks, Q2SWinet '16* (New York, NY, USA: Association for Computing Machinery): 87–93. doi:10.1145/2988272.2988284, URL <https://doi.org/10.1145/2988272.2988284>.
- [17] BILEN, T. and CANBERK, B. (2020) Overcoming 5g ultra-density with game theory: Alpha-beta pruning aided conflict detection. *Pervasive and Mobile Computing* 63: 101133. doi:https://doi.org/10.1016/j.pmcj.2020.101133, URL <https://www.sciencedirect.com/science/article/pii/S1574119220300213>.