ECHO: Enhanced Conditional Handover boosted by Trajectory Prediction

Anna Prado, Hansini Vijayaraghavan, Wolfgang Kellerer
Chair of Communication Networks, Technical University of Munich, Germany.
{anna.prado, hansini.vijayaraghavan, wolfgang.kellerer}@tum.de

***Abstract*—Conditional handover (CHO) has been introduced in 5G to improve mobility robustness, namely, to reduce the number of handover failures by preparing target Base Stations (BSs) in advance and allowing the user to decide when to make a handover. This algorithm constantly prepares and releases BSs, thereby adapting to the fast changing radio condition. A user might make a handover to a distant BS that has a favorable channel only for a short time due to signal fluctuations. This increases the handover rate and might result in a Radio Link Failure (RLF) afterwards. Moreover, the constant preparation and release of BSs leads to an increased exchange of control messages between the user, the serving BS and all target BSs. Hence, there is a need to carefully select the target BSs. Therefore, we propose the Enhanced CHO (ECHO) scheme that uses trajectory prediction to prepare the BSs along the user's path. To achieve this, we also propose a Sequence to Sequence (Seq2Seq) mobility prediction model. ECHO with only one prepared BS (ECHO-1) outperforms CHO with three prepared BSs. ECHO-1 reduces the handover rate by 23 percent and the RLF rate by 77 percent, while also reducing the number of control messages in the network by 69 percent.**

***Index Terms*—mobility, conditional handover, trajectory prediction**

I. Introduction

In dense cellular networks, mobile users often switch from one Base Station (BS) to another necessitating handovers which result in an interruption time during which a user cannot transmit or receive data. Therefore, their number should be minimized, while also satisfying the user's Quality of Service (QoS). 5G uses GHz frequencies to provide very low latency, so that users can enjoy services like Augmented Reality (AR) with delay requirements of 13 ms [1] making mobility management challenging due to blockages and high attenuation at high frequencies. In the traditional LTE/5G handover algorithm, mobile users experience many handovers and connection failures. To improve mobility robustness, Conditional handover (CHO) was introduced by 3GPP in 5G Release 16, which decouples BS preparation and handover execution phases [2]. In the following, we use the words BS and cell interchangeably.

During handover preparation, the serving BS sends the user's context to all candidate BSs and, depending on the implementation, resources at the candidate BSs might be reserved for the user. In CHO, the threshold for preparing the target BSs (preparation offset) is often smaller than the execution offset to ensure that information about prepared cells can be signalled to the user while radio conditions to the serving BS are still favorable. The serving BS prepares multiple target BSs and signals the list of prepared candidates to the user. Many cells might satisfy the preparation condition in a dense scenario and deciding which cells will be most suitable for the user in the future is not trivial. One could prepare all cells that satisfy the preparation condition, however, this approach leads to a significantly increased overhead in control signaling. Since the channel fluctuates significantly in dense networks, the prepared BSs are often removed or replaced by better ones, which again requires signaling between the user, the serving BS and the target BSs. Furthermore, in dense scenarios, users waste resources on an increased number of measurements since they have to constantly monitor the channel of all prepared cells. Moreover, when a user has multiple prepared BSs with similar Reference Signal Received Power (RSRP) values, it is not obvious which one should be selected for a handover. A prepared BS may seem to have a good channel condition currently, however, it might degrade rapidly in the near future. This might result in an Radio Link Failure (RLF) with an unacceptable delay for AR, hence, to satisfy the user's QoS and reduce signaling/measurement overhead, an intelligent CHO algorithm is required which also minimizes the number of prepared cells to reduce signaling and monitoring overhead.

One possible solution to improve the CHO scheme is to predict the user's location in the near future and prepare only the cells along the user trajectory, to which the user is more likely to make a handover. Artificial neural networks have recently proven to be useful in learning complex patterns from data [3], thus, we enhance CHO with a Deep Learning (DL) model to predict the user trajectory.

A. Related Work

There are works that have used trajectory prediction for handover management such as [4], where a service migration schemes based on group trajectory prediction using DL is proposed. There are few works on CHO, and most of them focus on optimization of handover thresholds, as in [5]. Differently, in [6] they use predicted future RSRP values to decide which cells should be prepared for CHO. However, they only consider geometric blockages, and many handovers happen due to sudden blockages and shadowing. These fluctuations are challenging to predict based on the channel measurements. Thus, some BSs that have high RSRP values at the current time

This work was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy: 6G Future Lab Bavaria.

step seem to be a good candidate for a handover although they are very far away and might stay favourite only for a short time due to signal fluctuations. Many BSs can be prepared simultaneously with CHO to provide certain guarantees. This, however, results in wasted signaling and resource reservation since a user performs a handover to only one BS. There are also works on signaling, like [7], which proposes a signaling mechanism for handover preparation to enhance 3GPP handover methods. However, large modifications of the existing handover procedures are required. Knowing the user trajectory allows the serving BS to prepare the closest BSs that are more likely to be good candidates for a handover in the future.

B. Contribution

In this work, we propose a sequence-to-sequence DL model for trajectory prediction and show its superiority over other models. Then, we introduce a handover management scheme called Enhanced CHO (ECHO) that uses the proposed model to perform online predictions. ECHO reduces handover and radio link failure rate, while at the same time reduces signaling overhead. We achieve this by preparing BSs that are along the user trajectory and are, thus, more likely to be favorable for the user in the near future. We limit the number of prepared cells without degrading the performance, so the number of cells that are prepared/released and are constantly monitored by the user is reduced. To the best of our knowledge, this is the first work that enhances CHO with trajectory prediction.

C. Organization

The rest of the paper is organized as follows. Section II evaluates the performance of different trajectory prediction models and introduces a Sequence to Sequence (Seq2Seq) model for trajectory prediction. Section III explains the CHO procedure and introduces ECHO that uses this Seq2Seq model. Section IV provides information about the system and evaluates the CHO and ECHO algorithms. Finally, Section V concludes the paper.

II. Deep Learning Based Trajectory Prediction

The prediction of the position of mobile users in the near future is a trajectory prediction problem. The trajectory of a mobile user is a time series of user positions. Therefore, the prediction of the trajectory for a few time steps into the future using past time steps as input, is actually a multi-step, multivariate time series prediction problem. In this section, we explore different models to solve this and propose a Seq2Seq model which is then evaluated on a model-based dataset.

A. Self-Similar Least-Action Walk (SLAW) Dataset

The trajectory of a user is the sequence of (x,y) coordinates of the user position over time, sampled at regular intervals. In order to evaluate the prediction models developed, a mobility model-based dataset of user trajectories is generated. In this work, we assume only pedestrian traffic so, we require a mobility model that can accurately describe human walk behavior. For this purpose, we use the SLAW [8] model which is effective in representing patterns among the trajectories of users sharing common interests such as in a university campus or theme park, where AR can be used. The common popular spots visited by all users are modelled as pause point clusters. Using this model we collect various datasets for evaluating the trajectory prediction models. Specifically, we collect the trajectories of 200 users in an area of 1000 m x 1000 m with an average user speed of $1\ \mathrm{m\,s^{-1}}$. This is repeated for various sampling time intervals of 1 s, 10 s and 60 s.

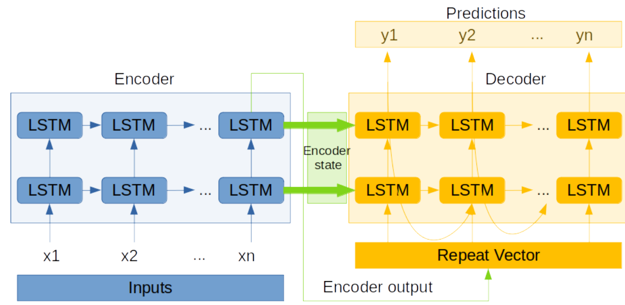


Fig. 1. Architecture of proposed Stack_Seq2Seq Model

B. Trajectory Prediction Models

To evaluate various prediction models, we select the dataset collected with a sampling interval of 10 s. Before developing a trainable model we need a performance baseline to compare with the more complex models. A simple baseline is to repeat the input position of the last time step for the required number of output time steps. We name this the baseline_last model. Another baseline_window approach is to repeat the previous window of time steps for the future time steps, assuming the future positions will be a repetition of the past. A simple trainable approach that is expected to perform better than the baseline models is a linear model to discover any linear relationships among the input positions. Another approach is to use a single dense layer to process the input positions. Some works also use a Convolutional Neural Network (CNN) [9] which predicts the trajectory based on a history of input steps, which could lead to better performance than the dense model. A better model to learn from a history of inputs is a Recurrent Neural network which could be a Long Short-Term Memory (LSTM) [10] model or a Gate Recurrent Unit (GRU) [11] model. Another promising model is the Seq2Seq model which is specifically designed for predicting sequences [12]. This model has 2 neural networks, an encoder and decoder network to map the input sequence to an output sequence. These networks can be a simple layer of LSTM.

In order to boost the performance of the Seq2Seq model we propose the stack_seq2seq model, in which we stack the LSTM layers. This is the model that will be used in the rest of this work. The architecture of the proposed stack_seq2seq model is described in Fig. 1. Apart from these models we also use a model with a fully connected layer followed by an LSTM layer (fc_lstm).

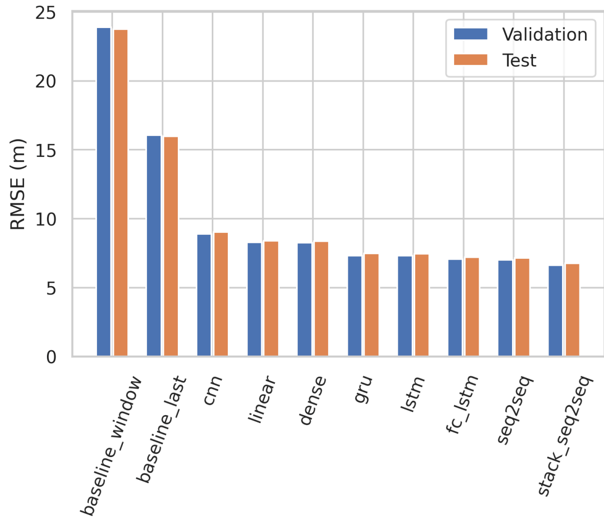


Fig. 2. Comparison of prediction performance for different trajectory prediction models over the validation and test set. Performance is measured in terms of average error over all time steps and coordinates

C. Evaluation of various prediction models

We evaluate the various trajectory prediction models described in the previous section using the dataset collected with a sampling interval of 10 s. The users in this dataset can pause for a maximum of 15 minutes at a popular spot. Hence, the user does not have a constant trajectory but rather has multiple long pauses. This makes the trajectory prediction challenging. The dataset is split such that different users are in the training and test dataset as would be the case in a real environment. The trajectory of a user is split to predict 5 time steps into the future using positions of the past 5 time steps. The prediction Root Mean Squared Error (RMSE) in meters is averaged over the x and y coordinates and over the 5 future time steps and compared in Fig. 2.

The baseline models do not perform well, but they provide a baseline for the performance. Among the Recurrent Neural networks (RNNs), the stack_seq2seq displays the best performance, although only marginally better than the rest. Another important observation is that the performance of the linear model is seemingly only a little worse than the stack_seq2seq. The linear model works well to predict the linear relationship in the trajectory of the users but would fail in predicting the non-linearity such as changes in direction or the pausing of users. This is where the proposed stack_seq2seq model outperforms other models. This seemingly small gain in the distance gap between the true and predicted position becomes very important in current heterogeneous cellular networks with dense deployments of small cells, small range Light-Fidelity (LiFi) [13] cells and beamformed networks where a small difference in position can result in a different BS association.

The time taken for training the models and making predictions are given in Table I. All models were trained using a GPU GeForce GTX 1650 and the predictions were made with an Intel Core i5 CPU. Although the training complexity is high for the stack_seq2seq model, the prediction performance is in the order of a few tens of milliseconds which is acceptable for online prediction in cellular networks.

TABLE I
TRAINING TIME PER EPOCH AND PREDICTION TIME PER USER FOR VARIOUS MODELS

Model	Train Time (s)	Prediction Time (s)
baseline_window	-	0.036
baseline_last	-	0.035
cnn	101	0.030
linear	72	0.035
dense	74	0.032
gru	130	0.032
lstm	137	0.034
fc_lstm	154	0.027
seq2seq	194	0.033
stack_seq2seq	310	0.029

D. Stack_Seq2Seq Model Evaluation

The stack_seq2seq model has proved to be a high performing model as seen in Sec. II-C. So this model is chosen as the trajectory prediction model in the handover management simulations. We also select the dataset collected with a sampling interval of 1 s. Fig. 3 shows the performance of the proposed prediction model on this dataset. As can be expected, a small error accumulating effect can be observed over time. But this model is able to minimize this effect and the resulting maximum error for the fifth time step is less than 2 m with a probability of 90%.

III. HANDOVER MANAGEMENT

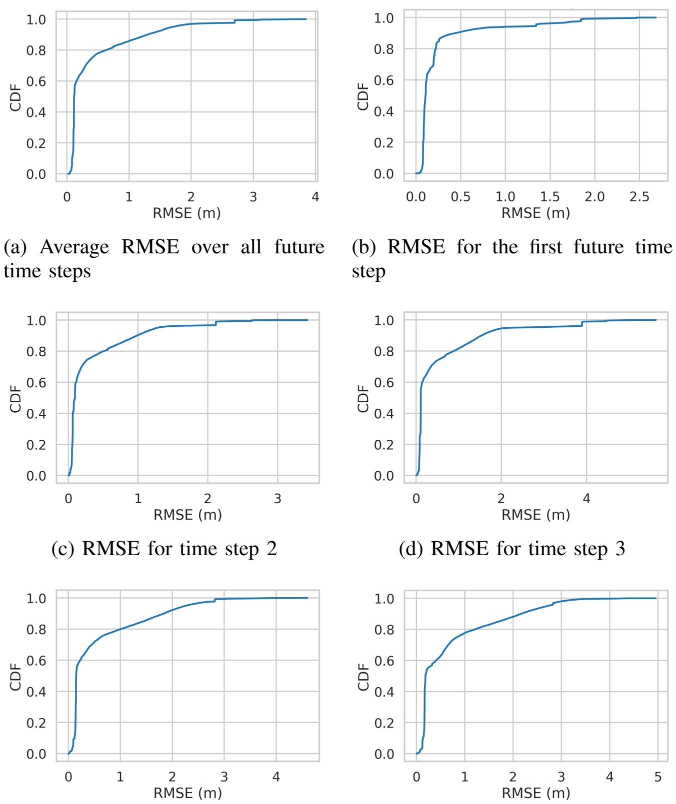
In this section, we explore the differences between CHO [2] and the proposed ECHO which uses a stack_seq2seq model from Sec. II. We explain how the set of next candidate BSs are selected based on trajectory prediction.

A. Conditional Handover (CHO)

We consider the Make-Before-Break (MBB) CHO algorithm, which terminates the connection with the serving BS only after establishing the connection to the target BS during handover to reduce Handover Interruption Time (HIT). The user measures the channel and sends the Measurement Report (MR) to its serving BS periodically. The BS applies Layer-3 filtering and averages RSRP values over 200 ms [2]. Based on these measurements, the serving BS selects candidate BSs that should be prepared for a potential handover. CHO uses an A3 event for handover preparation, execution and cell release. The A3 event is triggered when a neighboring BS becomes better than the serving BS by a preparation or execution offset. In CHO, preparation offset can be set negative to prepare cells in advance and increase robustness.

B. Enhanced Conditional Handover (ECHO)

We assume that the user's position is available at the user device for location-based services when the user is in



(a) Average RMSE over all future time steps

(b) RMSE for the first future time step

(c) RMSE for time step 2

(d) RMSE for time step 3

(e) RMSE for time step 4

(f) RMSE for time step 5

Fig. 3. CDF of Prediction RMSE for 5 future time steps for a dataset with a sampling interval of 1 s using the stack_seq2seq model

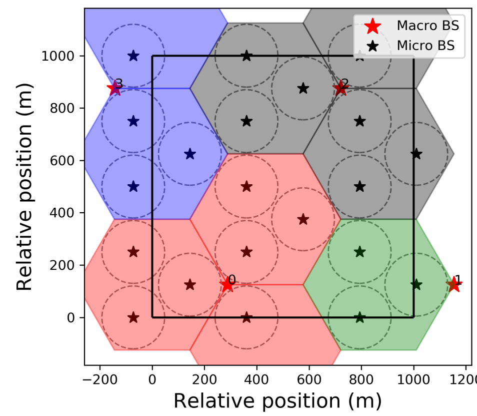


Fig. 4. Two-tiered system architecture, where dashed circles show the coverage of micro BSs.

TABLE II
SIMULATION PARAMETERS

Parameter	Value
Carrier frequency macro	0.5 GHz
Carrier frequency micro	4 GHz
Channel measurement periodicity	100 ms
HIT (assuming MBB)	14.5 ms
Handover preparation time	28.5 ms
RLF T_out	-8 dB
T310	600 ms
CHO preparation offset	-3 dB
CHO/ECHO execution offset	3 dB

CONNECTED_MODE [14]. Furthermore, other positioning methods such as trilateration can be used to determine the user's position at the BS or controller [15]. Differently from CHO, in ECHO, the BS decides which cells should be prepared and released based on the user's predicted proximity to the cells. The user reports its current position to the serving BS together with the RSRP values in the MR or other signals to avoid adding extra signalling messages.

After obtaining the user's position, the serving BS or the controller predicts the user's future position in the next multiple steps and computes the Euclidean distance to all BSs that cover the user. Since the resources in the network are finite, we limit the number of prepared cells to N. If the serving BS belongs to the micro-tier, then it prepares at least one closest macro BS and N-1 closest micro cells. Otherwise, N micro cells are prepared. Users are more likely to be close to many micro cells than to a macro cell. Thus, by ensuring that there is always one prepared macro cell, the user will not be out of coverage. Finally, when the serving BS receives an ACK from the candidate BSs, it sends the list with prepared BSs to the user. The user stores this list, and as in CHO, the A3 event is also used for handover execution in ECHO. The serving BS releases prepared cells in ECHO when they do not belong to the list with the closest candidates anymore.

For resource optimization, it is useful to know which BS the user will be near/connect to. Then this BS can plan its resource allocation in an optimal way. Thus, by predicting the user trajectory a few steps into the future, we provide some additional time to be able to run a complex optimization algorithm.

IV. Results and Evaluation

The channel model with line of sight blockages, macro and micro scenario parameters are taken from 3GPP 5G Release 14 [16]. The simulation has 25 BSs (macro and micro) and 30 users, who move within the area shown as a black square in Fig. 4. Other simulation parameters are provided in Table II. We assume frequency reuse, so there is no interference. We simulate the system for 2.5 hours for every algorithm.

To estimate the impact of imperfect predictions, we compare ECHO with the true trajectory of the user (ECHO_known) and predicted (ECHO_pred) trajectories. We also evaluate the algorithms' performance for different number of prepared BSs $N \in \{1, 2, 3\}$, which are the trailing numbers in the algorithm abbreviation states in Fig. 5 - 8.

In the box plots (Fig. 7, Fig. 9), the mean and standard deviation are shown in orange and black, respectively. Moreover, we evaluate ECHO with 5-step predictions into the future (ECHO_look_ahead) and observe that ECHO outperforms CHO, and ECHO_look_ahead achieves similar results

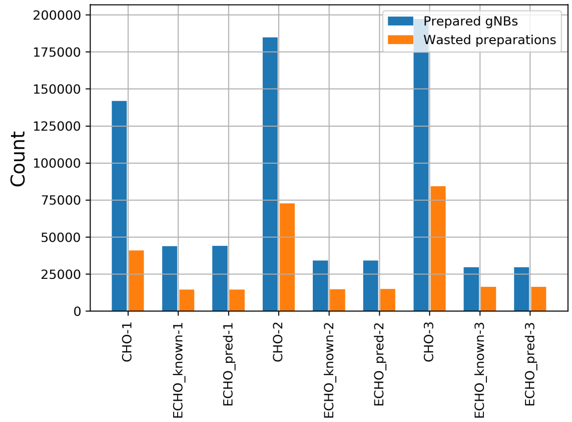


Fig. 5. Total number of prepared and wasted BSs in the system.

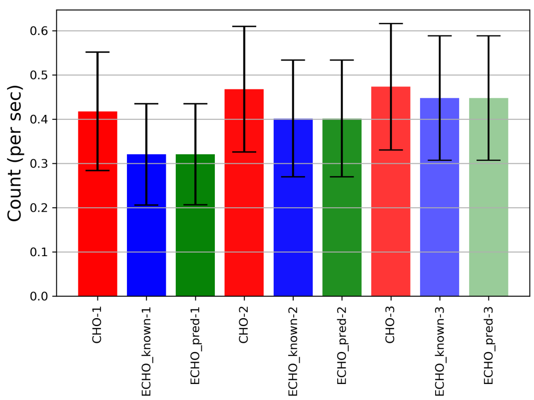


Fig. 6. Handover rate per user.

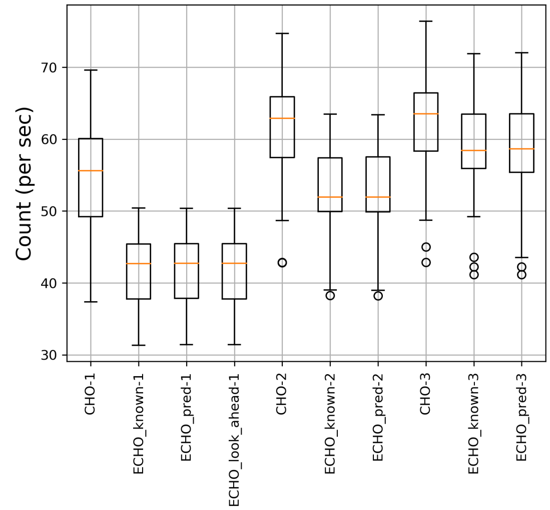


Fig. 7. Total HIT per user over the simulation.

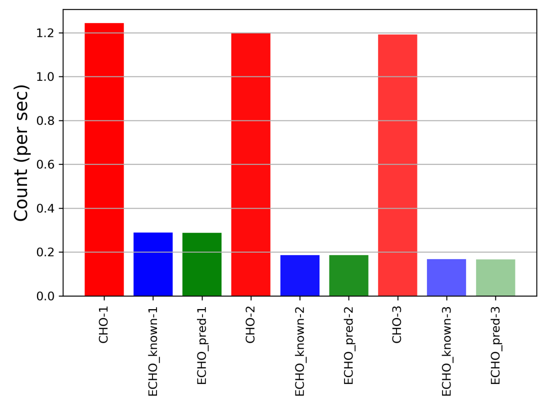


Fig. 8. RLF rate per user.

as ECHO with 1-step prediction. So, time to run DL models or optimization algorithms can be gained without trading-off the performance.

As shown in Fig. 5, ECHO reduces the number of cell preparations by 69%, 81% and 84% for one, two and three prepared BSs, respectively. Wasted preparations, when a cell was prepared and released, but the user did not connect to it, also decrease by 64%, 79% and 80% with ECHO. CHO prepares distant cells that, for a short period of time, have a higher RSRP value due to the line of sight presence. The users either do not connect to these prepared BSs or connect and suffer from poor radio conditions that result in an RLF as shown in Fig. 8. Although the number of wasted prepared BSs with CHO increases significantly with the increase of prepared BSs, it stays almost the same with ECHO because users still connect to most of the prepared cells and benefit from the channel diversity. Interestingly, the number of preparations reduces for ECHO when N increases. This happens because when multiple cells are prepared, only seldom does a new cell become closer to the user's position and replace an already prepared cell. CHO updates and releases BSs frequently according to signal fluctuations, while ECHO keeps the closest BSs prepared longer (since they stay the same for some period of time), thus reducing the signalling.

In Fig. 6, one can see that ECHO reduces the handover rate, thus, HIT reduces as well by 23%, 14% and 5% depending on N as shown in Fig. 7. HIT is the time when the user cannot exchange data with any BS. Fig. 7 also presents the HIT for ECHO with 5-step prediction into the future, which achieves a very similar performance as ECHO with known and predicted trajectories. Thus, we conclude that the error of Seq2Seq model is acceptable for its performance.

RLFs happen in the simulation due to line of sight blockages and shadowing in the channel, which makes the Signal to Noise Ratio (SNR) drop below the out of sync threshold T_out. No handover failures happen since the preparation and execution phases are decoupled in CHO and ECHO. Fig. 8 shows that ECHO reduces RLF rate from 1.24 to 0.28 RLFs per second on average when $N = 1$. Furthermore, the RLF rate obviously reduces when more BSs are prepared, e.g., for CHO-3 and ECHO-3 the values are 1.19 and 0.16 since we trade resources for a better performance. However, CHO with even three prepared BS has a high RLF rate. ECHO with only

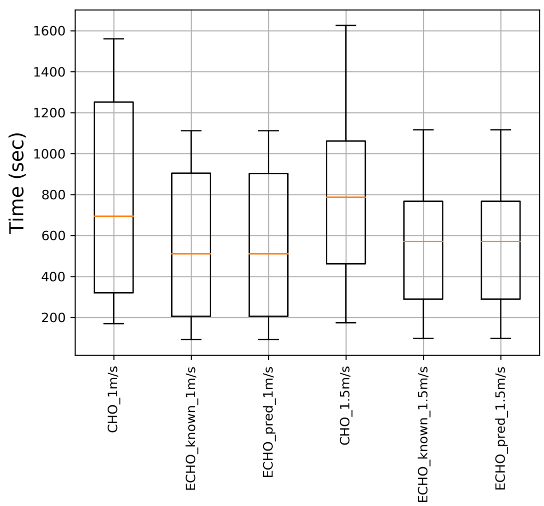


Fig. 9. The time a user had poor SNR within 150 min of simulation time for three handover algorithms with one gNB prepared and two speeds ($1\,m/s$ and $1.5\,m/s$).

one prepared BS achieves a significantly lower RLF rate than CHO-3, namely, 76% less.

Although it seems like ECHO might force users to connect to a BS with worse RSRP values, it is the opposite. Fig. 9 shows the total time during which a user moving at 1 and 1.5 $\mathrm{m\,s^{-1}}$ can neither transmit nor receive due to a low SNR received from the serving BS. As expected, faster users have a poor channel more often, hence, they experience more handovers and RLFs and benefit more from ECHO. For faster moving users, the throughput increases by over 4.5% and decreases the RLF rate 8 times, from 1.28 to 0.16.

CHO increases robustness by preparing cells in advance, however, it often prepares wrong cells and wastes resources for signaling and useless cell preparations. ECHO prepares the BSs to which the user is heading to, thus, the user is more likely to have a favourable channel with the closest BSs and connect to them in the near future. ECHO with just one prepared BS outperforms even CHO with three prepared BSs in terms of handover rate, RLF rate and the network sum throughput, while at the same time significantly reducing the number of cell preparations and releases. As a result, the network sum throughput also increases in the range of 3.5% to 4.5% with ECHO, while at the same time reducing signaling from 69% to 80%.

V. Conclusion

In this work, we evaluate different models for trajectory prediction and propose a Seq2Seq model using stacked LSTM to predict the location of pedestrian cellular users in the near future. We conclude that the main challenge is to predict when and how long the users will pause, and not the user's path itself. We enhance CHO with this prediction model and propose ECHO that utilizes the predicted user trajectory to decide which BSs should be prepared for a potential handover. We evaluate ECHO with the known trajectory, ECHO with the next position and ECHO with five next predicted positions. ECHO with only one prepared BS outperforms CHO with three prepared BSs. ECHO decreases handover and radio link failure rate, reduces the number of BS preparations, thus, reducing the signalling in the network. Not only does the network save the resources and does not have to prepare and release BSs back and forth, but the users also have to monitor fewer prepared cells. The idea of ECHO complies with 5G ultra-lean design and the goal to minimize the number of measurements to achieve higher data rates and higher energy efficiency.

References

[1] E. Bastug, M. Bennis, M. Médard, and M. Debbah, "Toward interconnected virtual reality: Opportunities, challenges, and enablers," *IEEE Communications Magazine*, vol. 55, no. 6, pp. 110–117, 2017.

[2] 3GPP, "NR; NR and NG-RAN Overall description; Stage-2," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.300, 3 2021, version 16.5.0. [Online]. Available: http://www.3gpp.org/DynaReport/38300.htm

[3] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3039–3071, 2019.

[4] Z. Zhao, M. Karimzadeh, L. Pacheco, H. Santos, D. Rosário, T. Braun, and E. Cerqueira, "Mobility management with transferable reinforcement learning trajectory prediction," *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 2102–2116, 2020.

[5] H. Martikainen, I. Viering, A. Lobinger, and T. Jokela, "On the basics of conditional handover for 5g mobility," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2018, pp. 1–7.

[6] C. Lee, H. Cho, S. Song, and J.-M. Chung, "Prediction-based conditional handover for 5g mm-wave networks: A deep-learning approach," *IEEE Vehicular Technology Magazine*, vol. 15, no. 1, pp. 54–62, 2020.

[7] A. Jain, E. Lopez-Aguilera, and I. Demirkol, "Improved handover signaling for 5g networks," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2018, pp. 164–170.

[8] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong, "Slaw: A new mobility model for human walks," in *IEEE INFOCOM 2009*, 2009, pp. 855–863.

[9] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in *Competition and cooperation in neural nets*. Springer, 1982, pp. 267–285.

[10] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.

[11] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[12] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *arXiv preprint arXiv:1409.3215*, 2014.

[13] H. Haas, L. Yin, Y. Wang, and C. Chen, "What is lifi?" *Journal of lightwave technology*, vol. 34, no. 6, pp. 1533–1544, 2015.

[14] M. Yassin and E. Rachid, "A survey of positioning techniques and location based services in wireless networks," in *2015 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*. IEEE, 2015, pp. 1–5.

[15] B. T. Fang, "Trilateration and extension to global positioning system navigation," *Journal of Guidance, Control, and Dynamics*, vol. 9, no. 6, pp. 715–717, 1986.

[16] 3GPP, "Study on channel model for frequencies from 0.5 to 100 GHz," 3rd Generation Partnership Project (3GPP), Technical Report (TR) 38.901, 1 2020, version 16.1.0. [Online]. Available: http://www.3gpp.org/DynaReport/38901.htm