Multi-Agent Deep Reinforcement Learning For Distributed Handover Management In Dense mmWave Networks

Authors: Mohamed Sana; Antonio De Domenico; Emilio Calvanese Strinati; Antonio Clemente

Presenter: Stanley Wu

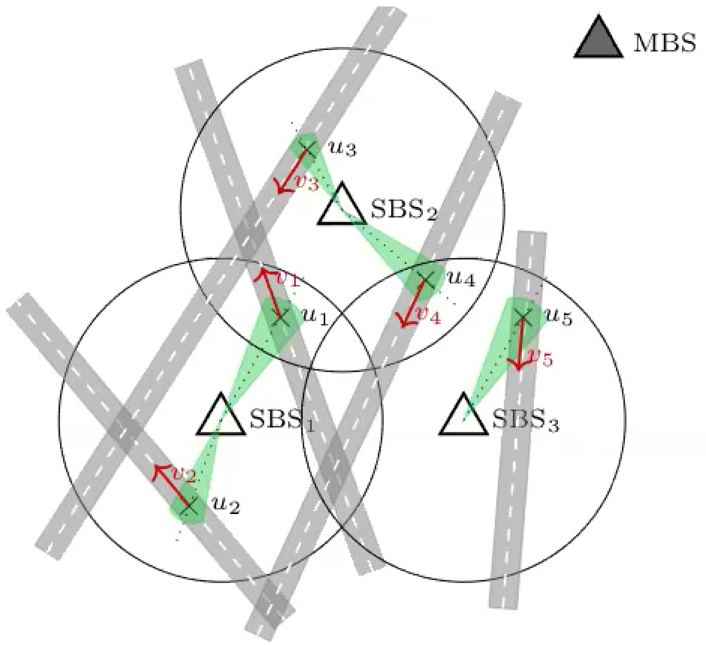
Jun. 25 2020

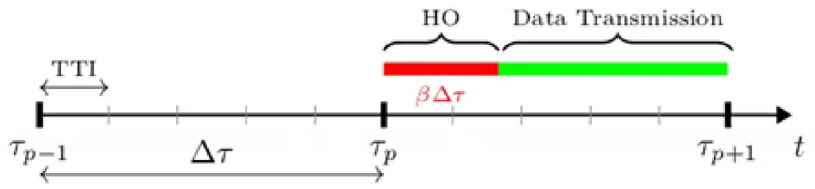
Outline

- Main Question
- Solution
- Simulation Configuration
- Simulation Result
- Discussion

Main Question

- Under a HetNet:
  - N_s small base Station (SBS), 1 macro base station (MBS)
  - To increase the network sum-rate
  - Via reducing the redundant HO





Handover process timeline.

- If $\Delta\tau$ is the HO time-to-trigger interval (TTI), a HO process can be triggered every $\tau_p = p\Delta\tau + \tau_0$.
- If a UE decides to handover at time τ_p, then **it spends (loses) $\beta\Delta\tau$ for initiating HO process.** → HO penalty

$\Rightarrow$ the effective data received by UE j between τ_p and τ_{p+1} is

$$\overline{R}_{i,j}(\tau_p, \beta) = \int_{\tau_p}^{\tau_p + (1-\beta\lambda_j(\tau_p))\Delta\tau} R_{i,j}(t)dt.$$

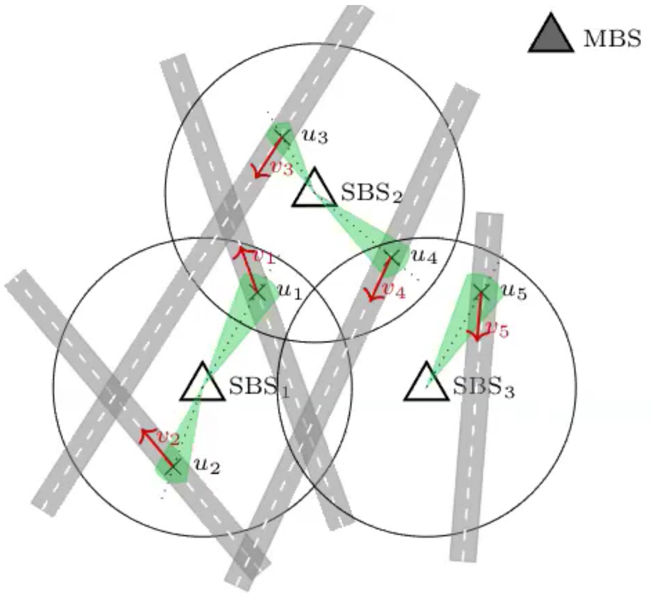
- $\mathcal{S} = \{0, 1, \ldots, N_s\}$: set of BSs.
- $\mathcal{U} = \{0, 1, \ldots, K\}$: set of K UEs.
- $\mathcal{U}_i$: set of UEs covered by BS i.
- $\mathcal{S}_j$: set of BSs at the reach of UE j.
- $R_{i,j} = B_{i,j}\log_2(1 + \mathrm{SINR}_{i,j})$ is the achievable rate, where
- $B_{i,j}$ the bandwidth allocated to UE j.
- τ_0 is an initial system delay.
- $\lambda_j(\tau_p)$ **indicates if the UE has Handover (=1) or not (=0).**

Main Question (cont.)

- The network sum-rate $R(\tau_p, \beta)$ between τ_p and τ_{p+1}

$$R(\tau_p, \beta) = \frac{1}{\Delta\tau} \sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{U}} \overline{R}_{i,j}(\tau_p, \beta).$$

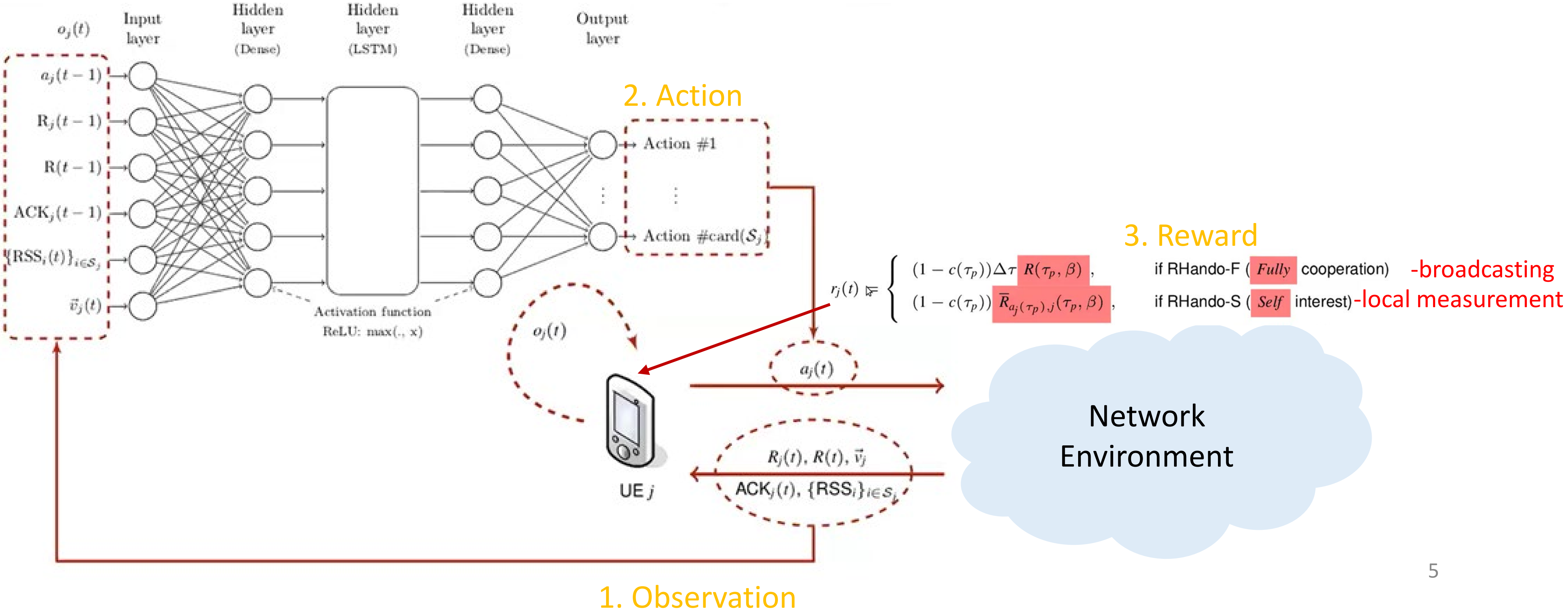
$$\begin{aligned} \underset{\{x_{i,j}\}}{\text{maximize}} \quad & R_T(\beta) = \frac{1}{P}\sum_{p=1}^{P} R(\tau_p, \beta), \quad P = \lfloor T/\Delta\tau \rfloor \\ \text{subject to} \quad & x_{i,j}(\tau_p) \in \{0, 1\}, \Rightarrow \text{Association variables are binary} \\ & \sum_{j\in\mathcal{U}_i} x_{i,j}(\tau_p) \le N_i,\ i \in \mathcal{S}, \Rightarrow \text{BS } i \text{ can serve at most } N_i \text{ UEs.} \\ & \sum_{i\in\mathcal{S}_j} x_{i,j}(\tau_p) = 1,\ j \in \mathcal{U}. \Rightarrow \text{A UE is associated to one BS.} \end{aligned}$$



- $\mathcal{S} = \{0, 1, \ldots, N_s\}$: set of BSs.
- $\mathcal{U} = \{0, 1, \ldots, K\}$: set of K UEs.
- $\mathcal{U}_i$: set of UEs covered by BS i.
- $\mathcal{S}_j$: set of BSs at the reach of UE j.
- $R_{i,j} = B_{i,j}\log_2(1 + \mathrm{SINR}_{i,j})$ is the achievable rate, where
- $B_{i,j}$ the bandwidth allocated to UE j.
- τ_0 is an initial system delay.
- $\lambda_j(\tau_p)$ **indicates if the UE has Handover (=1) or not (=0).**

Solution

- Use a deep multi-agent reinforcement learning (DRQN) for distributed handover management called RHando (Reinforced Handover)



Solution (cont.)

- Each UE maintains its own DRQN [1] and learns to maximize a long term reward by minimizing the loss function:

$$\mathcal{L}_j(\theta_j) = \mathbb{E}_{e_j^b(t) \sim \mathcal{B}_j} \left[(\boldsymbol{w_j^b} \delta_j^b(t))^2 \right]$$

Where subscript b indicates an entry in the mini batch β_j of experiences $e_j^b(t)$,

$\delta_j^b(t)$ is the TD error w.r.t the target value $y_j^b(t)$

$$\delta_j^b(t) = y_j^b(t) - Q_j(o_j^b(t), h_j^b(t-1), a_j^b(t) | \theta_j),$$

$$y_j^b(t) = r_j^b(t) + \gamma \max_{a'} Q_j(o_j^b(t+1), h_j^b(t), a' | \hat{\theta}_j).$$

- $h_j(t)$ represents the recurrent neural network parameters,
- θ_j is the DRQN weights. Note that $\hat{\theta}_j$ is the target DRQN weights (update less frequently).

Solution (cont.)

- The DRQN may end up in a local optimal state instead of in a global optimal state
- To approximate the global optimal state, using the hysteretic Q-learning algorithm, let DRQN be updated via a gradient decent algorithm with two distinct learning rates α and β [2]:

$$w_j^b = \begin{cases} \alpha, if\ \delta_j^b(t) \geq 0 \\ \beta, others \end{cases} \quad (\beta \ll \alpha \leq 1)$$

Simulation Configuration

- UEs use Random WayPoint Model v= [0, 10] m/s

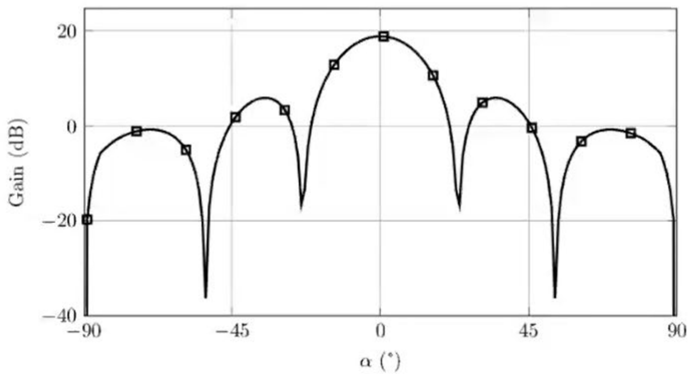
	Macro cell (3GPP TR 36.872)	Small cell
Parameters	Values	
Carrier frequency, f_c	2.0 GHz	28 GHz
Bandwidth, B	10 MHz	500 MHz
Thermal Noise, N_0	-174 dBm/Hz	
Noise figure	5 dB	
Shadowing, X	9 dB	12 dB
Transmit power	46 dBm	20 dBm
g_0 (TX/RX)	17 dBi / 0 dBi	-
Cell radius, r		50 m
Beam width, θ	$360°$	$20°$
Side lobe gain, ξ		-20 dBi
Inter-cell distance		$1.2 \times r$
Pathloss model	$128.1 + 36.7\log_{10}(d)$	Eq. (1), $d_0 = 5$ m
TTI	10ms	
$\Delta\tau$	1s	
T	2000s	

Path loss model

$$\mathrm{PL} = -20\log_{10}\left(\frac{4\pi d_0}{\lambda_i}\right) - 10\eta_i\log_{10}\left(\frac{d_{i,j}}{d_0}\right) - X_{i,j}. \quad (1)$$

d_0 is the reference distance, $\eta_i = 2.5$ the path loss coefficient, λ_i the wavelength.

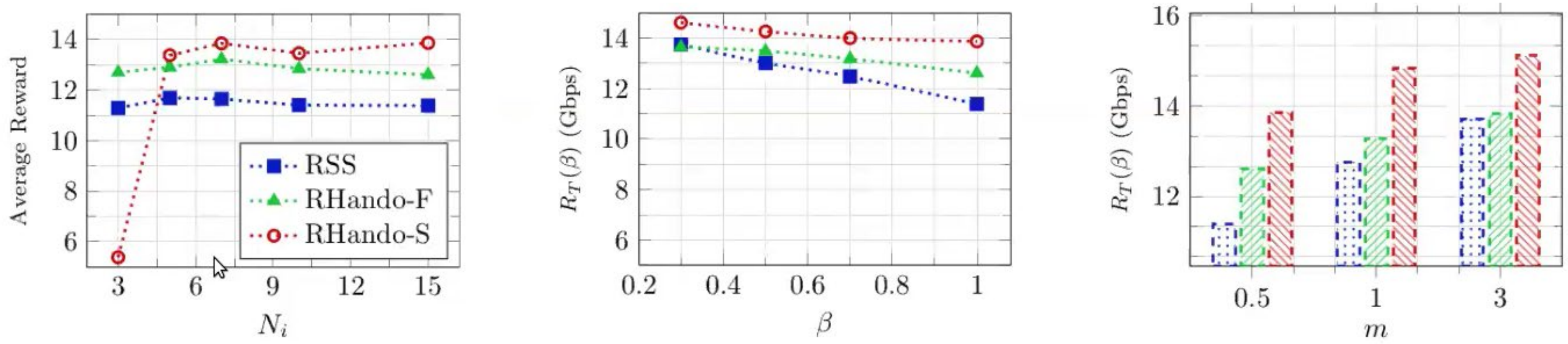
Simulated TX/RX antenna gain radiation pattern for an array of 5x5 elements operating at 28 GHz [3]



Simulation Result

- Reduce the HO frequency and increase the network sum-rate

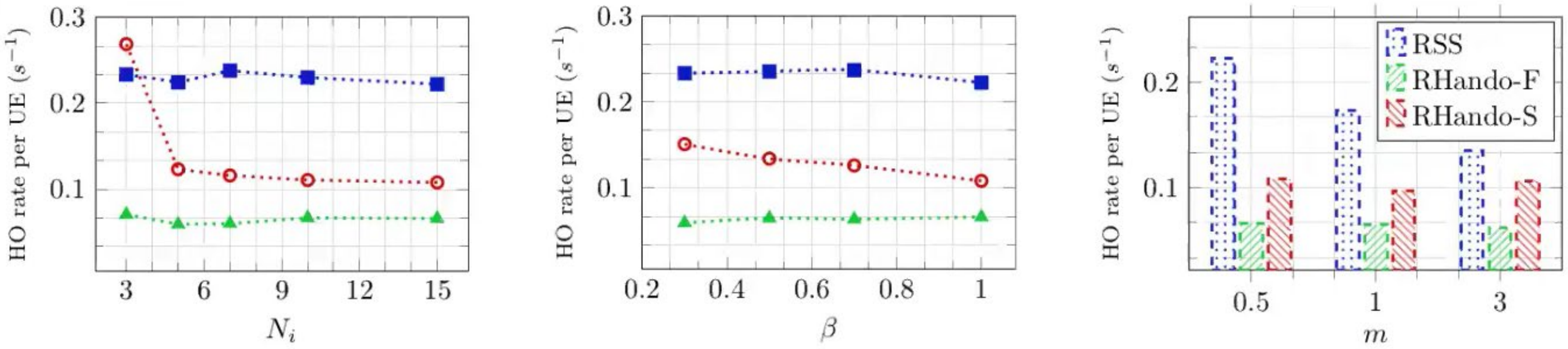
Nakagami fading scale factor: m



(a) Avg. reward w.r.t. N_i, $K = 15$, $m = 0.5$, $\beta = 1$.

(b) $R_T(\beta)$ w.r.t. β. $N_i = K = 15$, $m = 0.5$.

(c) $R_T(\beta)$ w.r.t. m. $N_i = K = 15$, $\beta = 1$.



(d) HO rate w.r.t. N_i. $K = 15$, $m = 0.5$, $\beta = 1$.

(e) HO rate w.r.t. β. $N_i = K = 15$, $m = 0.5$.

(f) HO rate w.r.t m. $N_i = K = 15$, $\beta = 1$.

Reference

- [1] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian, “Deep Decentralized Multi-task Multi-Agent Reinforcement Learning under Partial Observability,” in *Proc. International Conference on Machine Learning (ICML)*, 06–11 Aug 2017, vol. 70, pp. 2681–2690.
- [2] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, “Hysteretic Q-learning: an Algorithm for Decentralized Reinforcement Learning in Cooperative Multi-agent Teams,” in *Proc. International Conference on Intelligent Robots and Systems (IEEE/RSJ),* 2007, pp. 64–69.