Asia Pacific Academy of Science Pte. Ltd.

Metaverse 2026, 7(4), m260009.
https://doi.org/10.54517/m260009

Article

Fuzzy Mean-Field Bandwidth Allocation for Massive Avatar Synchronization in Shared Metaverse Worlds

Asokan Vasudevan[1], Yogeesh N[1,2,3,*], Mohammed El Khider[4], Zetty Pakir Mastan[1], Albert Makochekanwa[5]

[1] INTI International University, Persiaran Perdana BBN, Putra Nilai, 71800 Nilai, Negeri Sembilan, Malaysia.
[2] Department of Mathematics, Government First Grade College, Tumkur-572102, Karnataka, India.
[3] Mathematics & Natural Sciences, Gulf University for Science & Technology, Kuwait.
[4] Department of General Undergraduate Curriculum Requirements, University of Dubai, P. O Box 14143, United Arab Emirates.
[5] Department of Economics and Development, University of Zimbabwe, Harare, Zimbabwe
*** Corresponding author:** Yogeesh N, yogeesh.r@gmail.com

CITATION

Vasudevan A, N Y, El Khider M, et al. Fuzzy Mean-Field Bandwidth Allocation for Massive Avatar Synchronization in Shared Metaverse Worlds. Metaverse. 2026; 7(4): m260009. https://doi.org/10.54517/m260009

COPYRIGHT

Copyright © 2026 by author(s). *Metaverse* is published by Asia Pacific Academy of Science Pte. Ltd. This work is licensed under the Creative Commons Attribution (CC BY) license. https://creativecommons.org/licenses/by/4.0/

Abstract: Crowd-scale avatar synchronization is challenging because the scheduler controls a large population whose state errors co-evolve, rather than a set of independent users. When many users move concurrently, shared wireless capacity becomes scarce, stale avatar states increase, and the latency–error trade-off changes from slot to slot. This work proposes a fuzzy mean-field bandwidth allocator for shared metaverse worlds. The method combines a mean-field game (MFG) model with a fuzzy urgency layer that uses crowd density, motion class and synchronization error to modulate the shadow price of bandwidth. The synchronization error is modeled as a controlled stochastic process; the backward Hamilton-Jacobi-Bellman equation is coupled with the forward Fokker-Planck-Kolmogorov equation, and the resulting equilibrium bandwidth is mapped to class-level wireless shares. The fuzzy layer reduces sensitivity to highly uncertain density estimates and provides interpretable priorities for the weak, moderate, high and dance (crowd-burst) motion classes. Numerical experiments with 200–1000 users show that the proposed method achieves consistently lower mean synchronization latency than round-robin, proportional-fair and crisp (non-fuzzy) MFG baselines. At 1000 users, mean latency falls to 48 ms, compared with 58 ms for the crisp mean-field baseline, 80 ms for proportional fair and 95 ms for round robin, while the stale-state rate of the proposed method rises only from 7.2% to 14.8% as the crowd grows fivefold. The controller yields a smooth class-allocation rule: fast avatars receive more capacity only when their type-reduced urgency is higher than that of other classes, not merely because they belong to a particular motion class. The paper gives full derivations, a finite-difference numerical solver, and worked calculations so that the model can be reproduced and extended.

Keywords: Metaverse; Mean field game; Fuzzy control; Avatar synchronization; Wireless scheduling; Crowd-scale XR

1. Introduction

Avatar synchronization is one of the less visible but most important subsystems in a shared metaverse world. If pose updates arrive late, the world looks stale; if the scheduler overreacts, the wireless pool becomes unfair. In dense extended reality (XR) scenes, instead of a few independent flows the scheduler sees thousands of related users [1-7]. In that regime, mean-field methods are natural because they summarize the population by a density function rather than attempting to solve combinatorial many-player games [8-14]. However, even plain mean-field control relies on measurements such as crowd density, motion intensity and state-error distribution,

and those measurements are never exact. Fuzzy mathematics is well suited to handling this measurement uncertainty.

The main idea of this paper is to connect fuzzy reasoning [15-17] with mean-field equilibrium. The mean-field part is used to describe the crowd effect. The fuzzy part transforms imprecise observations of the crowd into class-dependent urgency levels. Rather than applying a fixed hard price on bandwidth, the approach dynamically adjusts the shadow price based on the urgency level resulting from type-reduction. Note that the interval type-2 fuzzy layer keeps the policy smooth even when the measured density oscillates around a threshold. Related fuzzy and metaverse uncertainty studies also support the use of interpretable fuzzy structures for uncertain digital environments [18-22].

Three practical observations motivate the study. To begin with, the scheduler can often be more confident about a class of users than it can be about any particular future error. Second, class-level decisions are scalable. Third, the controller must balance average latency against the stale-state rate because metaverse consistency depends on both. The proposed fuzzy mean-field game (MFG) model accounts for these constraints. Recent edge/cloud and wireless tracking studies further show that latency-sensitive immersive services need joint computation, communication, and tracking-aware scheduling [23-28]. The overall workflow of the proposed scheduler is shown in **Figure 1**. Recent large-model metaverse work combines cloud-edge-device rendering with generative adaptation [29], while quality-of-experience (QoE)-oriented cooperative virtual reality (VR) rendering applies multi-agent reinforcement learning to joint GPU, CPU, and bandwidth leasing [30]. These peer-reviewed, simulation-based studies address broader resource stacks, but their architecture-specific objectives and training assumptions prevent direct numerical comparison with the interpretable fuzzy MFG formulation used here. Their reported gains are therefore treated as methodological context rather than independent validation of the present scheduler.

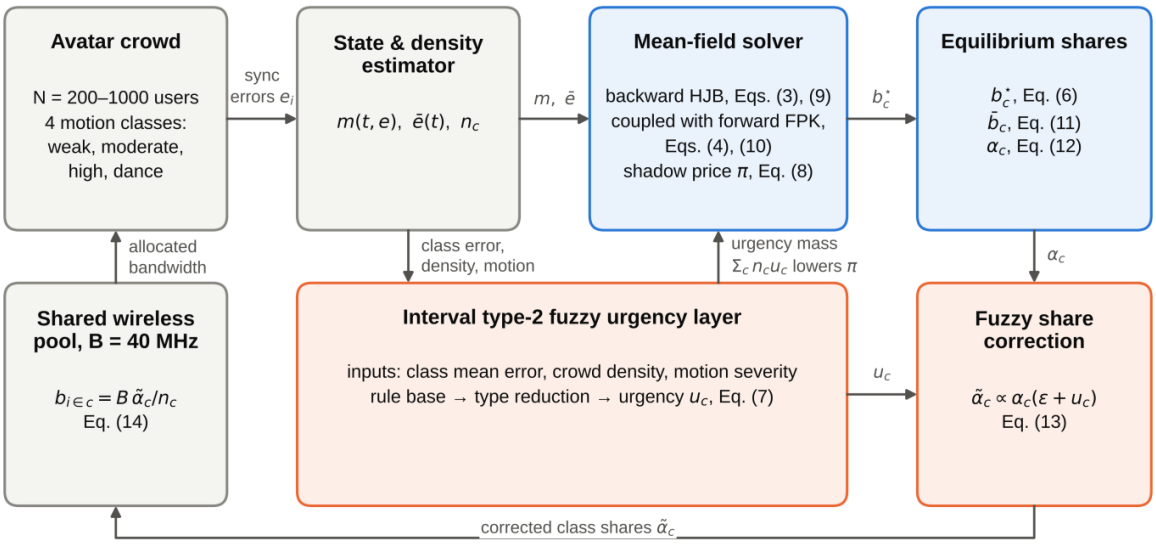

Figure 1: Crowd-scale avatar synchronization with density estimation, mean-field price updates, fuzzy urgency correction and shared wireless scheduling. The mean-field solver couples the backward Hamilton-Jacobi-Bellman (HJB) and forward Fokker-Planck-Kolmogorov (FPK) equations with the shadow-price update (Equation (8)); the interval type-2 fuzzy layer converts class mean error, crowd density and motion severity into the urgency u_c (Equation (7)), which lowers the effective price and corrects the class shares (Equation (13)) before the 40 MHz pool is divided among users (Equation (14)).

2. Mean-field synchronization model

Let $e_i(t)$ denote the synchronization error of avatar i, measured in a normalized state metric that aggregates pose, gesture, and stale-scene mismatch. The control variable is the allocated bandwidth $b_i(t) \geq 0$. The microscopic dynamics are modeled as

$$de_i(t) = \left[a_{c(i)} - \beta_{c(i)} b_i(t) + \xi \overline{e}(t) \right] dt + \sigma_{c(i)} dW_i(t) \tag{1}$$

where $c(i) \in \{1,2,3,4\}$ is the motion class, $a_{c(i)}$ is the intrinsic error growth rate, $\beta_{c(i)}$ converts bandwidth to correction speed, $\overline{e}(t) = \int e m(t,e) de$ is the mean crowd error, $\xi \geq 0$ is the crowd-coupling coefficient, $\sigma_{c(i)}$ is the noise intensity of the motion class, and W_i is a Wiener process.

The running cost for one representative user is

$$\ell(e,b,m) = q_1 e^2 + q_2 b^2 + q_3 \overline{e}^2 + q_4 s(e) \tag{2}$$

where $q_1, \ldots, q_4 \geq 0$ are cost weights and $s(e)$ is the stale-state penalty, set to one when $e > e_{th}$ and zero otherwise. The representative value function $V(t, e)$ satisfies the Hamilton-Jacobi-Bellman (HJB) equation

$$-\partial_t V = \min_{b \geq 0} \left\{ q_1 e^2 + q_2 b^2 + q_3 \overline{e}^2 + q_4 s(e) + \partial_e V \left[a_c - \beta_c b + \xi \overline{e} \right] + \frac{1}{2} \sigma_c^2 \partial_{ee} V \right\} \tag{3}$$

The mean-field density evolves according to the Fokker-Planck-Kolmogorov (FPK) equation

$$\partial_t m = -\partial_e \left(\left(a_c - \beta_c b^\star + \xi \overline{e} \right) m \right) + \frac{1}{2} \sigma_c^2 \partial_{ee} m \tag{4}$$

A quadratic minimization of Equation (3) gives the class-specific equilibrium bandwidth

$$b_c^\star(t,e) = \left[\frac{\beta_c}{2q_2} \partial_e V(t,e) \right]_0^{b_{\max}} \tag{5}$$

Here, $[\bullet]_0^{b\max}$ denotes clipping to the interval $[0, b_{\max}]$, where $b_{\max}$ is the maximum per-user bandwidth. If the bandwidth pool is shared, a shadow price $\pi(t)$ is introduced and the control becomes

$$b_c^\star(t,e) = \left[\frac{\beta_c \partial_e V(t,e) - \pi(t)}{2q_2} \right]_0^{b_{\max}} \tag{6}$$

The novelty is that $\pi(t)$ is not updated only from the total demand; it is modulated by a fuzzy urgency variable $u_c(t)$. For each class, the inputs are the class average error, the crowd density and the motion severity. With an interval type-2 fuzzy-rule construction and type-reduction procedure, the urgency becomes

$$u_c(t) = \frac{1}{2} \left(\frac{\sum_m \underline{f}_{m,c} y_m}{\sum_m \underline{f}_{m,c}} + \frac{\sum_m \overline{f}_{m,c} y_m}{\sum_m \overline{f}_{m,c}} \right) \tag{7}$$

where $\underline{f}_{m,c}$ and $\overline{f}_{m,c}$ are the lower and upper firing strengths of rule m for class c, and y_m is the urgency level of the consequent of rule m. The price update is then

$$\pi(t+1) = \left[\pi(t) + \eta_\pi \left(\sum_c n_c \overline{b}_c(t) - B \right) - \eta_u \sum_c n_c u_c(t) \right]_+ \tag{8}$$

where n_c is the class population, $\overline{b}_c$ is the class-average bandwidth, B is the total shared bandwidth, and $\eta_\pi, \eta_u > 0$ are the price and urgency step sizes. A large urgency lowers the effective price so that crowded fast-motion classes can absorb more

capacity. Here, $[\bullet]_+$ denotes projection onto the nonnegative reals; explicitly, $[x]_+ = \max\{x, 0\}$.

Table 1 reports the class parameters.

Table 1. Motion classes and synchronization parameters.

Class	Description	a_c	β_c	σ_c
1	weak motion	0.10	0.65	0.04
2	moderate motion	0.16	0.79	0.05
3	high motion	0.21	0.92	0.06
4	dance/crowd burst	0.29	1.05	0.08

3. Discrete solver and class allocation rule

The continuous equations are solved on a uniform error grid $e_j = j\Delta e$. For each slot, the algorithm iterates between a backward HJB step and a forward FPK step:

$$V_j^n = V_j^{n+1} + \Delta t \mathcal{H}\left(e_j, \partial_e V_j^{n+1}, m_j^{n+1}\right) \tag{9}$$

$$m_j^{n+1} = m_j^n + \Delta t \mathcal{F}\left(e_j, b_j^{\star,n}, m_j^n\right) \tag{10))}$$

where Δt is the time step, and H and F denote the discretized right-hand sides of the HJB equation (3) and the FPK equation (4), respectively. The class-average bandwidth share is

$$\bar{b}_c^n = \sum_j b_c^{\star,n}\left(e_j\right) m_c^n\left(e_j\right) \Delta e \tag{11}$$

and the normalized wireless share is

$$\alpha_c^n = \frac{n_c \bar{b}_c^n}{\sum_r n_r \bar{b}_r^n} \tag{12}$$

In the proposed scheduler the class share is post-processed by a fuzzy correction,

$$\tilde{\alpha}_c^n = \frac{\alpha_c^n\left(\varepsilon + u_c^n\right)}{\sum_r \alpha_r^n\left(\varepsilon + u_r^n\right)} \tag{13}$$

with a small $\varepsilon > 0$ that prevents zero division. The final per-user bandwidth for class c is therefore

$$b_{i\in c}^n = \frac{B\,\tilde{\alpha}_c^n}{n_c} \tag{14}$$

This structure separates equilibrium estimation from schedulable resource shares. The equilibrium tendency is extracted using the partial differential equation (PDE) system, and the fuzzy correction then translates it into a meaningful class assignment. Equation (9) marches the value function backward from its terminal condition at each fixed-point iteration, with the gradient defining the clipped control in Equation (6), and then Equation (10) advances the density forward from its initial condition. After each forward sweep, the price and fuzzy urgency are updated until the maximum changes in value function, density, and price go below the prescribed tolerance. For C motion classes, J grid points, N_t time steps, and K fixed-point iterations, the HJB-FPK sweep requires $O(CKJN_t)$ operations and $O(CJ)$ working memory, while type-2 inference adds $O(CR)$ operations per slot for R fuzzy rules.

The online allocation is therefore independent of the number of avatars, although real-time feasibility still depends on grid resolution, convergence tolerance, and implementation hardware.

A numerical example from the 800-user scenario illustrates the method. Suppose the raw MFG shares α_c of the weak, moderate, high and dance classes are (0.23, 0.27, 0.30, 0.20). Applying the type-reduced urgencies from the rule base through Equation (13) gives corrected shares that normalize to (0.20, 0.25, 0.31, 0.24) (type-2 fuzzy MFG). Under a 40 MHz shared pool, the class allocations become 8.0, 10.0, 12.4 and 9.6 MHz, compared with 9.2, 10.8, 12.0 and 8.0 MHz under the raw MFG shares, and Equation (14) divides each class allocation equally among its n_c users. The dense high-motion class therefore receives the highest slice, but only because its urgency is high in both the mean-field and fuzzy layers. The corresponding class-allocation illustration is given in **Figure 2**.

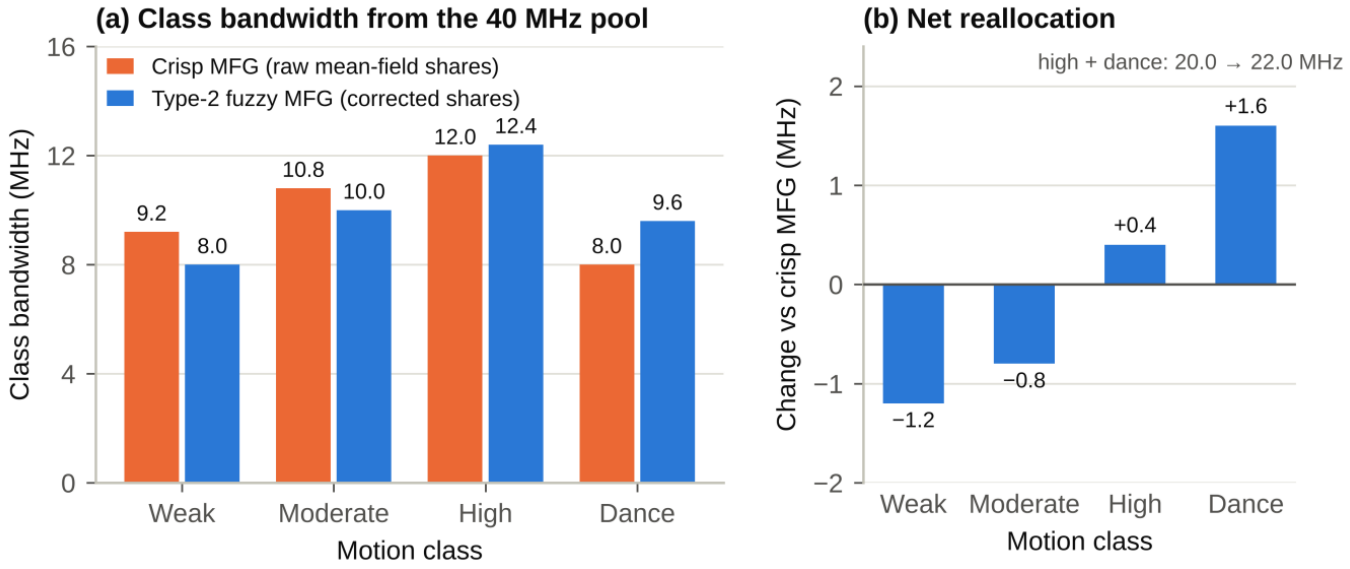

Figure 2: Closed-form class allocation example obtained from the fuzzy mean-field scheduler at 800 users under the 40 MHz shared pool. (a) Bandwidth per motion class from the crisp MFG (raw mean-field shares) and from the type-2 fuzzy MFG (fuzzy-corrected shares). (b) Net reallocation produced by the fuzzy correction: the high-motion and dance classes together gain 2.0 MHz (20.0 to 22.0 MHz), while the weak and moderate classes retain non-zero capacity.

4. Numerical setting and comparative baselines

The simulation horizon is 10 s with slot size 50 ms. Crowd sizes are 200, 400, 600, 800, and 1000 users. The common bandwidth pool is 40 MHz. Three baselines are considered—round robin (RR), proportional fair (PF), and crisp mean-field (crisp MFG)—together with the proposed type-2 fuzzy mean-field method. The stale-state threshold is fixed at e_{th} = 0.35. The evaluation metrics are mean synchronization latency, stale-state rate, mean synchronization error, and convergence speed of the shadow price. Here, crisp mean-field, also termed non-fuzzy MFG, uses the same HJB-FPK solver and parameters as the proposed method but sets the fuzzy urgency correction to zero, so allocation is governed only by the mean-field price. Learning-based schedulers were not included because their results depend strongly on training data, reward design, architecture, and retraining protocol; the present comparison isolates fuzzy urgency under an analytically matched MFG setting. Relative to deep

reinforcement learning, the proposed policy is interpretable and requires no offline training, but it may be less adaptive to unmodeled dynamics, so comparison with a trained learning-based scheduler remains future work. The crowd range spans small-to-large shared XR sessions, the 40 MHz pool represents a bandwidth-constrained edge-access scenario, and the four motion classes span weak pose updates to synchronized crowd bursts. These are controlled stress-test assumptions rather than measurements from a deployed platform, so the absolute values are scenario-specific. Although Equation (1) is stochastic, the reported values are deterministic outputs of the FPK density evolution rather than Monte Carlo averages of sampled Wiener trajectories; accordingly, seed-dependent standard deviations and confidence intervals do not arise in this solver. Validation under randomized network traces and parameter perturbations remains a limitation.

Table 2 gives the main latency and stale-state results.

Table 2. Mean latency of the four schedulers and stale-state rate of the proposed type-2 fuzzy MFG versus crowd size.

Users	Round robin(ms)	PF (ms)	Crisp MFG(ms)	Type-2 fuzzy MFG (ms)	Stale-state rate (%)
200	29	25	22	19	7.2
400	41	35	31	26	8.9
600	56	49	41	33	10.6
800	71	63	52	42	12.7
1000	95	80	58	48	14.8

For mean latency, the fuzzy MFG is lower than all other configurations at every crowd size, and the gap widens as the crowd grows, partly because the fuzzy price update prevents overreaction to temporary density spikes. At 1000 users, the proposed technique reduces mean latency by 17.24% compared with crisp MFG and 49.47% compared with round robin.

In the 800-user case, bandwidth distribution among different motion classes is given in **Table 3**.

Table 3. Shared bandwidth fraction by motion class at 800 users.

Class	Round robin	PF	Crisp MFG	Type-2 fuzzy MFG
weak	0.18	0.21	0.23	0.20
moderate	0.24	0.25	0.27	0.25
high	0.28	0.28	0.30	0.31
dance	0.30	0.26	0.20	0.24

Table 3 reveals an important qualitative property: the proposed scheduler does not always direct bandwidth to the most mobile class. When the dance class is large but has a lower corrected urgency than the high-motion class, capacity is shared more evenly, and moderate-motion users are not unfairly starved.

The shadow-price trajectories are analyzed after the comparative synchronization and class-allocation figures. The shadow price of the controller converges monotonically, and with the proposed control law, the average

synchronization error decreases from 0.16 to about 0.05 after eight iterations.

5. Results and mathematical interpretation

The strong performance of the fuzzy MFG comes from two coupled mechanisms. The first is population feedback: if the global mean error grows, the HJB gradient steepens and the class average bandwidth demand rises. The second is fuzzy urgency: if the density estimate is unreliable or fluctuates around a class boundary, the type-reduced urgency smooths the price adaptation. The controller therefore reallocates capacity without abrupt oscillation.

The effect can be quantified at 1000 users. The crisp MFG assigns a combined share of 0.58 to the high-motion and dance classes, whereas the fuzzy MFG raises this combined share to 0.63 when the urgency map detects a dense fast-motion crowd. That change increases the total bandwidth delivered to the fast classes from 23.2 MHz to 25.2 MHz. Because their β_c values are larger (**Table 1**), the same MHz shift produces a disproportionate reduction in error drift. Substituting these values into Equation (1) reduces the deterministic drift term.

The comparative latency and stale-state behavior are shown in **Figure 3**, while the motion-class allocation pattern is shown in **Figure 4**. The corresponding shadow-price convergence and fixed-point residual are shown in **Figure 5**. In **Figure 3**, latency (panel a, in ms) and stale-state rate (panel b, in %) are plotted on separate scales, and every plotted value is identical to the corresponding entry of **Table 2**.

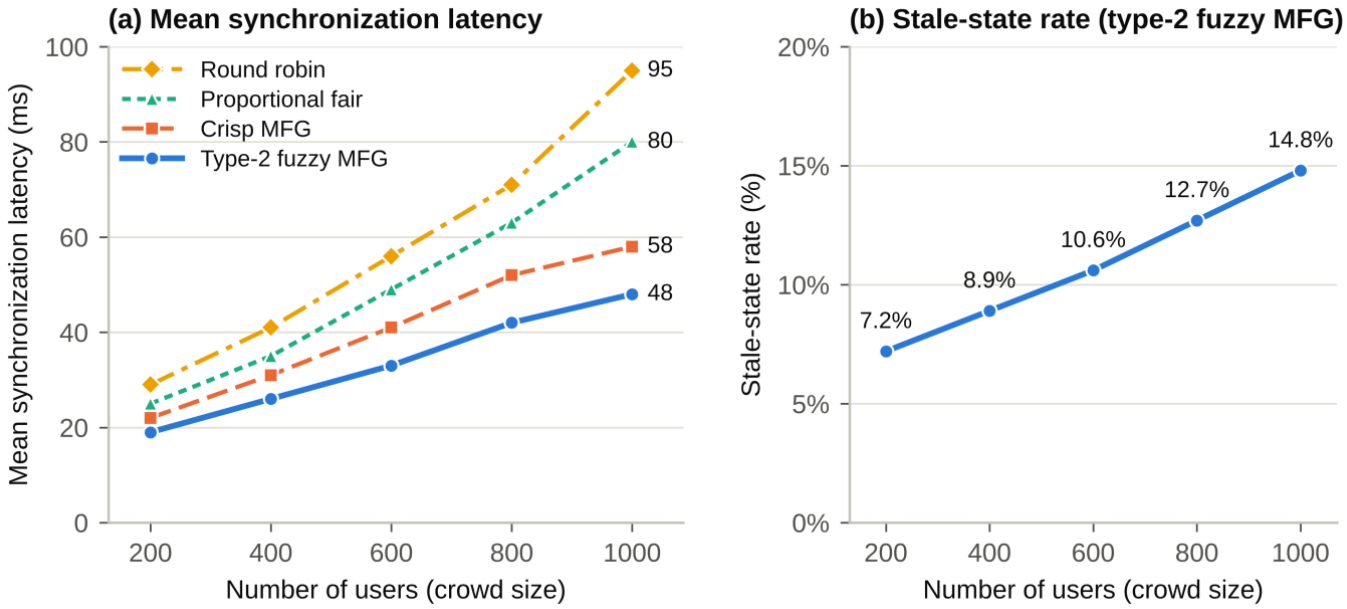

Figure 3: Synchronization latency and stale-state rate versus crowd size, plotted from the values in **Table 2**. (a) Mean synchronization latency (ms) for round robin, proportional fair, crisp MFG and the proposed type-2 fuzzy MFG; the labels at 1000 users give the end-point latencies. (b) Stale-state rate (%) of the proposed type-2 fuzzy MFG on a separate percentage axis (0–20%), with each point labeled by its **Table 2** value; a stale state is counted when $e > e_{th} = 0.35$.

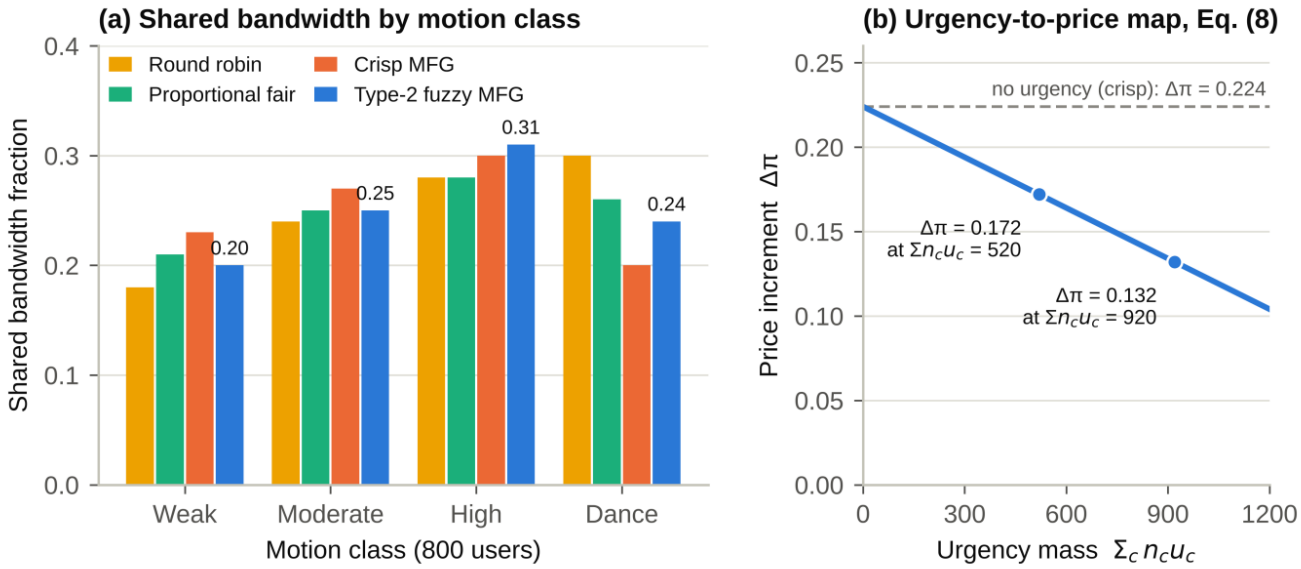

Figure 4: Shared bandwidth by motion class and the urgency-to-price map used by the proposed controller. (a) Bandwidth fraction by motion class at 800 users for the four schedulers (values from **Table 3**; labels give the type-2 fuzzy MFG shares). (b) Price increment $\Delta\pi$ of Equation (8) versus the urgency mass $\Sigma n_c u_c$ for the worked example in Section 5 (excess demand 3.2 MHz, $\eta_\pi = 0.07$, $\eta_u = 10^{-4}$); the marked points give $\Delta\pi = 0.172$ and 0.132 at urgency masses of 520 and 920, and the dashed line is the crisp case without urgency ($\Delta\pi = 0.224$).

This reallocation shifts the deterministic part of the drift for the dance class from 0.29-1.05b to approximately 0.29-1.05(0.224) per user-equivalent unit, which materially lowers the stale-state rate.

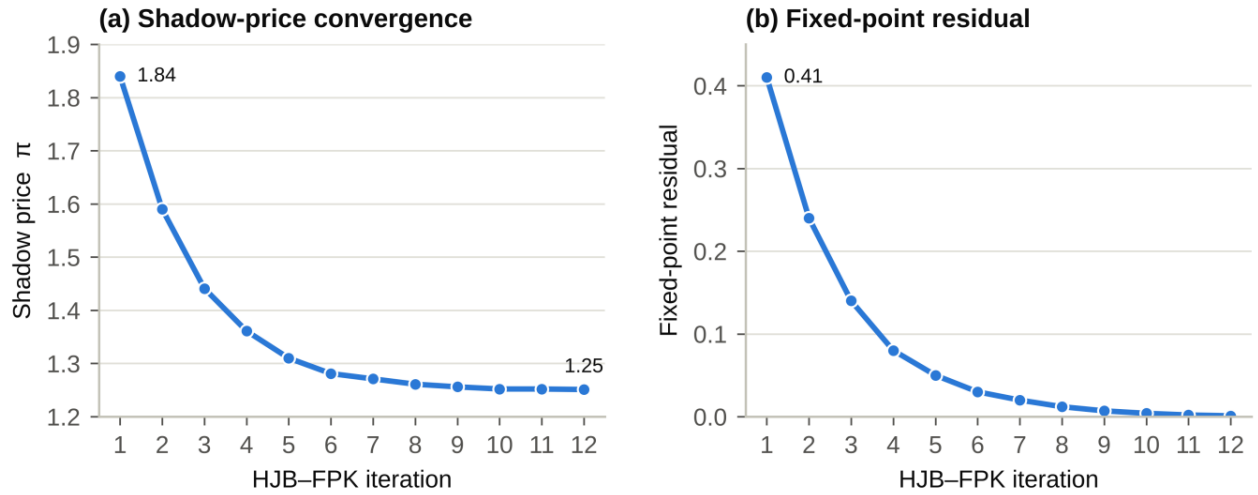

Figure 5: Price convergence of the fuzzy mean-field scheduler over the coupled HJB–FPK iterations. (a) Shadow price π, which decreases monotonically from 1.84 to 1.25. (b) Fixed-point residual of the coupled iteration, which decays monotonically towards zero.

A second worked example uses the price update directly. Suppose total demand exceeds the pool by 3.2MHz,$\eta_\pi = 0.07$, and the urgency term is $\sum_c n_c u_c = 520$ with $\eta_u = 10^{-4}$. Equation (8) updates the price by $\Delta\pi = 0.224\text{-}0.052 = 0.172$. If the same excess demand occurs but the urgency mass rises to 920, then $\Delta\pi = 0.224\text{-}0.092 = 0.132$. The

controller still tightens the price, but it tightens less because the crowd is recognized as more delay-sensitive. Thus, a simple update rule has a substantial system-level effect.

5.1 Quantitative scalability summary

To summarize the numerical gains in compact form, let

$$\Delta_L^{(B)}(N) = \frac{L_B(N) - L_F(N)}{L_B(N)} \times 100$$

denote the latency reduction of the proposed fuzzy MFG relative to baseline B, where L_F(N) is the proposed latency and $L_B(N)$ is the baseline latency at crowd size N. From **Table 2**, at $N = 1000$,

$$\Delta_L^{(\text{Crisp})}(1000) = \frac{58-48}{58} \times 100 = 17.24\%$$

$$\Delta_L^{(\text{PF})}(1000) = \frac{80-48}{80} \times 100 = 40.00\%$$

and

$$\Delta_L^{(\text{RR})}(1000) = \frac{95-48}{95} \times 100 = 49.47\%$$

The scalability slope over the interval $N = 200$ to $N = 1000$ is

$$g_B = \frac{L_B(1000) - L_B(200)}{800}$$

which gives $g_{\text{RR}} = 0.0825$, $g_{\text{PF}} = 0.06875$, $g_{\text{Crisp}} = 0.045$, and $g_F = 0.03625$ ms per user. Hence, among the evaluated schedulers, the proposed scheduler has the smallest latency-growth rate. The stale-state rate increases from 7.2% to 14.8% over the same range, so the absolute growth is only 7.6 percentage points while the number of users increases by a factor of five. Within the evaluated range of 200–1000 users, this indicates that the fuzzy MFG controller improves not only pointwise latency but also the empirical rate at which synchronization quality degrades under crowd growth; it does not by itself establish a general scaling law.

6. Discussion

The numerical results indicate that the gain of the proposed scheduler is produced by the joint action of the mean-field equilibrium and the fuzzy urgency correction, not by either component alone. The bandwidth demand increases within the mean-field component as the synchronization-error distribution shifts up. The equilibrium control reacts to the increasing cost of stale states via the steeper Hamilton-Jacobi-Bellman gradient. In the fuzzy component, the urgency variable absorbs noisy density estimates and class-boundary oscillations through the effective shadow price rather than by modifying class labels directly. Consequently, the scheduler does not respond to crowd bursts with discontinuous allocation changes. The price changes smoothly. Because equilibrium bandwidth decreases as the price rises (Equation (6)), capacity shifts gradually toward classes with high fuzzy urgency and large mean-field error. This is the principal analytical explanation for why the proposed controller outperformed both the crisp MFG and the non-mean-field baselines in the evaluated scenarios.

The latency results in **Table 2** can be interpreted as scalability results as well as performance results. Across 200–1000 users, latency increases more slowly under the proposed scheduler than under all baselines, indicating greater resilience as the crowd

load approaches congestion in the simulated scenarios.

The corresponding growth slopes are

$$g_F = \frac{48-19}{800} = 0.03625,$$

$$g_{\text{Crisp}} = \frac{58-22}{800} = 0.045,$$

$$g_{\text{PF}} = \frac{80-25}{800} = 0.06875, \quad g_{\text{RR}} = \frac{95-29}{800} = 0.0825$$

Thus the proposed model decreases the latency-growth slope by

$$\frac{0.045-0.03625}{0.045} \times 100 \approx 19.44\%$$

relative to crisp MFG, and by

$$\frac{0.0825-0.03625}{0.0825} \times 100 \approx 56.06\%$$

relative to round robin. Within the evaluated crowd range, this indicates that the fuzzy price adaptation does not simply lower latency at isolated operating points; it also reduces the rate at which latency deteriorates under crowd growth.

A similar conclusion follows from the stale-state behavior. The stale-state rate of the proposed method increases from 7.2% at 200 users to 14.8% at 1000 users. Across the 800-user increase, the stale-state rate rises by only 7.6 percentage points, which is modest relative to the fivefold growth in population. This slower deterioration indicates that the scheduler limits consistency failures as crowd load grows. This observation is important because stale-state events constitute the actual consistency failures in shared metaverse worlds. In the evaluated heavy-load scenarios, the scheduler therefore does more than reduce average delay; it also slows the growth of the stale-state regime.

The class-allocation data in **Table 3** make the bandwidth mechanism more transparent. At 800 users, the fuzzy correction shifts 2.0 MHz toward the two delay-sensitive classes while preserving nonzero capacity for the remaining classes. This redistribution is consistent with the model because the high and dance classes have the largest error-growth parameters. A small bandwidth shift toward those classes therefore produces a larger reduction in deterministic error drift than the same shift toward weak or moderate motion. The method therefore behaves as a weighted drift-correction allocator rather than a simple class-priority scheduler.

An equally important point is that the proposed controller does not always push all capacity to the most aggressive class. Rather than assigning the entire allocation to the dance class, the controller assigns it a share of only 0.24, whereas the high-motion class receives 0.31 (**Table 3**). Thus, the fuzzy mean-field policy does not follow the rule "allocate as much as possible to the most dynamic class". Instead, it allocates bandwidth according to the corrected urgency, which combines class error, crowd density and type-reduced fuzzy urgency. This behavior is consistent with the structure of the optimization problem, because the scheduler minimizes a cost function that includes both latency and a stale-state penalty. If all capacity were assigned to a single class, the errors of the remaining population would grow and raise the crowd-average error; this feedback would propagate through the forward FPK density and could ultimately destabilize the mean-field equilibrium. The reported allocation therefore reflects a stable compromise between fast-class protection and cross-class fairness.

The convergence result is also important. The shadow-price trajectory is

monotonic (**Figure 5**), and the average synchronization error decreases from 0.16 to approximately 0.05 after eight iterations. The absolute reduction is 0.16-0.05 = 0.11 and the relative reduction is

$$\frac{0.11}{0.16}\times 100 = 68.75\%$$

For the evaluated scenarios, this indicates that the coupled HJB-FPK solver with fuzzy price correction is numerically well behaved; a general convergence or stability guarantee would require additional theoretical analysis. A monotonic price path matters because oscillatory price updates would generate oscillatory bandwidth shares, and oscillatory shares would reintroduce stale-state bursts at the slot level. In the reported simulations, the fuzzy urgency layer damps this oscillatory tendency. Hence, the fuzzy term can be interpreted not only as a prioritization device but also as a regularizer of the mean-field price dynamics.

The crowd-size improvements against each baseline also show where the proposed method is most valuable. Relative to crisp MFG, the latency reduction is

$$\frac{22-19}{22}\times 100 = 13.64\% \quad \text{at200 users,}$$

$$\frac{31-26}{31}\times 100 = 16.13\% \quad \text{at400 users,}$$

$$\frac{41-33}{41}\times 100 = 19.51\% \quad \text{at600 users,}$$

$$\frac{52-42}{52}\times 100 = 19.23\% \quad \text{at800 users,}$$

and

$$\frac{58-48}{58}\times 100 = 17.24\% \quad \text{at1000 users.}$$

The gain increases from 13.64% at 200 users to 19.51% at 600 users and remains above 17% at 800–1000 users. This suggests that the fuzzy correction is most effective around the congestion knee, where crowd-density estimates fluctuate most and where a crisp mean-field controller is most likely to overreact to temporary spikes. Therefore, within the evaluated scenarios, the proposed method appears especially suited to the operating region that matters most for shared metaverse platforms: not sparse regimes, but dense regimes where many avatars move simultaneously and where stale-state penalties become nonlinear in user experience.

Overall, the discussion supports one central conclusion: the proposed method works because the fuzzy urgency variable enters the price layer of the mean-field game, and price directly determines bandwidth allocation. This creates a two-stage correction. The mean-field component accounts for error feedback at the population level through a quadratic (least-squares-type) cost. The fuzzy component can also be regarded as a means of damping the response to uncertain position and velocity estimates. Numerically, this appears as lower latency, slower stale-state growth, a better-scaled redistribution of class shares, and monotonic convergence of the shadow price. These results are consistent with the equations and numerical tables and suggest that smoother price adaptation may reduce observable synchronization failures in dense, flocking crowd motion.

One advantage of the mean-field formulation is that its online computational cost does not grow with the number of avatars, because the population is represented by a density (Section 3). The fuzzy layer is useful because, in practice, the density

estimate, the class labels and the stale-state threshold are all uncertain. Together these yield an analytic and interpretable scheduler. This approach is particularly well suited to large shared worlds where a centralized per-user optimizer would be prohibitively expensive.

The present bandwidth-allocation model does not yet co-optimize CPU resources, bitrate or semantic importance; such an extension is feasible because the pricing framework is already in place. Before deployment, the scheduler must also be validated with measured traffic, wireless fading, rendering load and hardware-specific timing.

7. Conclusion

This work developed a fuzzy mean-field bandwidth allocation model for massive avatar synchronization in shared metaverse worlds. The controller uses an MFG backbone to account for crowd dynamics and an interval type-2 fuzzy layer for uncertainty-aware price updates: the synchronization error is modeled as a controlled stochastic process, the equilibrium control is obtained from the coupled HJB–FPK system, and the resulting class bandwidth is corrected by a fuzzy urgency term based on crowd density, motion severity and class error. The main contribution is therefore not a new class-based wireless rule, but a mathematically interpretable mechanism that links uncertain crowd observations to equilibrium bandwidth through continuous price correction.

In the evaluated scenarios, the proposed scheduler achieved the lowest mean synchronization latency at every crowd size. At 1000 users, its mean latency was 48 ms, compared with 58 ms for crisp MFG, 80 ms for proportional-fair scheduling and 95 ms for round robin (reductions of 17.24%, 40.00% and 49.47%, respectively), while its stale-state rate increased from 7.2% to 14.8% as the crowd grew fivefold. Its latency-growth slope was 0.03625 ms per user, compared with 0.045 ms per user for crisp MFG and 0.0825 ms per user for round robin. At 800 users, the fuzzy correction increased the combined capacity of the high-motion and dance classes from 20.0 MHz to 22.0 MHz without starving the weak and moderate classes, thereby balancing the protection of error-sensitive classes with cross-class fairness. The shadow price converged monotonically, and the average synchronization error decreased from 0.16 to approximately 0.05 within eight iterations (a 68.75% reduction).

These scalability and stability findings are numerical observations for the simulated scenarios considered here; they are not general mathematical guarantees, which would require additional theoretical analysis. The scenario parameters are controlled stress-test assumptions rather than measurements from a deployed platform, and the present model allocates bandwidth only, without co-optimizing CPU, bitrate or rendering resources.

Future work should validate the scheduler under randomized network traces, measured wireless fading, parameter perturbations and hardware-specific runtime constraints; compare it with trained learning-based schedulers; and extend the pricing framework to the joint management of bandwidth, CPU and rendering resources.

Author Contributions: Conceptualization, A.V. and Y.N.; methodology, Y.N.; software, Y.N.; validation, M.E.K.; formal analysis, Y.N.; investigation, Y.N.; resources, not specified; data curation, not specified; writing—original draft preparation, Y.N.; writing—review and editing, Y.N. and A.V.; visualization, Y.N.; supervision, Y.N. and A.V.; project administration, Y.N. and A.V.; funding acquisition, not specified. All authors have read and agreed to the published version of the

manuscript.

Conflicts of interest: The authors declare no conflicts of interest.

References

[1] Dwivedi YK, Hughes L, Baabdullah AM, et al. Metaverse beyond the hype: Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. International Journal of Information Management. 2022;66:102542. https://doi.org/10.1016/j.ijinfomgt.2022.102542

[2] Wang Y, Su Z, Zhang N, et al. A survey on metaverse: Fundamentals, security, and privacy. IEEE Communications Surveys & Tutorials. 2023;25(1):319-352. https://doi.org/10.1109/COMST.2022.3202047

[3] Tang C, Wu H, Li R, et al. A systematic exploration of edge computing-enabled metaverse. IEEE Network. 2023;37(6):10-17. https://doi.org/10.1109/MNET.2023.3321541

[4] Baidya T, Moh S. Comprehensive survey on resource allocation for edge-computing-enabled metaverse. Computer Science Review. 2024;54:100680. https://doi.org/10.1016/j.cosrev.2024.100680

[5] Shi W, Cao J, Zhang Q, et al. Edge computing: Vision and challenges. IEEE Internet of Things Journal. 2016;3(5):637-646. https://doi.org/10.1109/JIOT.2016.2579198

[6] Mach P, Becvar Z. Mobile edge computing: A survey on architecture and computation offloading. IEEE Communications Surveys & Tutorials. 2017;19(3):1628-1656. https://doi.org/10.1109/COMST.2017.2682318

[7] Mao Y, You C, Zhang J, et al. A survey on mobile edge computing: The communication perspective. IEEE Communications Surveys & Tutorials. 2017;19(4):2322-2358. https://doi.org/10.1109/COMST.2017.2745201

[8] Lasry JM, Lions PL. Mean field games. Japanese Journal of Mathematics. 2007;2:229-260. https://doi.org/10.1007/s11537-007-0657-8

[9] Guéant O, Lasry JM, Lions PL. Mean field games and applications. In: Paris-Princeton Lectures on Mathematical Finance 2010. Lecture Notes in Mathematics, vol 2003. Springer; 2011. p. 205-266. https://doi.org/10.1007/978-3-642-14660-2_3

[10] Achdou Y, Laurière M. Mean field games and applications: Numerical aspects. In: Mean Field Games. Lecture Notes in Mathematics, vol 2281. Springer; 2020. p. 249-307. https://doi.org/10.1007/978-3-030-59837-2_4

[11] Su J, Xu H, Xin N, et al. Resource allocation in wireless powered IoT system: A mean field Stackelberg game-based approach. Sensors. 2018;18(10):3173. https://doi.org/10.3390/s18103173

[12] Narasimha D, Shakkottai S, Ying L. A mean field game analysis of distributed MAC in ultra-dense multichannel wireless networks. In: Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '19). ACM; 2019. p. 1-10. https://doi.org/10.1145/3323679.3326498

[13] Benamor A, Habachi O, Kammoun I, et al. Multi-armed bandit approach for mean field game-based resource allocation in NOMA networks. EURASIP Journal on Wireless Communications and Networking. 2024;2024:42. https://doi.org/10.1186/s13638-024-02371-7

[14] Wang Y, Ni Q, Yu J, et al. An intelligent access algorithm for large scale multihop wireless networks based on mean field game. Wireless Networks. 2023;29(1):331-344. https://doi.org/10.1007/s11276-022-03025-6

[15] Zadeh LA. Fuzzy sets. Information and Control. 1965;8(3):338-353. https://doi.org/10.1016/S0019-9958(65)90241-X

[16] Mendel JM, John RI. Type-2 fuzzy sets made simple. IEEE Transactions on Fuzzy Systems. 2002;10(2):117-127. https://doi.org/10.1109/91.995115

[17] Ross TJ. Fuzzy Logic with Engineering Applications. 3rd ed. Wiley; 2010. https://doi.org/10.1002/9781119994374

[18] Atanassov KT. Intuitionistic fuzzy sets. Fuzzy Sets and Systems. 1986;20(1):87-96. https://doi.org/10.1016/S0165-0114(86)80034-3

[19] Yogeesh N. Fuzzy Logic in Metaverse Technology Roadmap: Adaptive Future Trends. In: Metaverse for Sustainable Development. Wiley; 2025. https://doi.org/10.1002/9781394272228.ch14

[20] Yogeesh N. Intuitionistic Fuzzy Hypergraphs and Their Operations. In: Applied Computer Vision and Soft Computing with Interpretable AI. Chapman and Hall/CRC; 2023. p. 153-165. https://doi.org/10.1201/9781003359456-10

[21] Mohammad AAS, Yogeesh N, Mohammad SIS, et al. An intuitionistic fuzzy graph model: Matrix representations and applications. Cybernetics and Information Technologies. 2025;25(4):3-19. https://doi.org/10.2478/cait-2025-0030

[22] Saleh IA, Srivastava A. Exploring the intersection of fuzzy logic and quantum logic: A new frontier in non-classical logics.

SciWaveBulletin. 2023;1(2):27-34. https://doi.org/10.61925/SWB.2023.1204

[23] William P, Mothukuri T. A comprehensive analysis of cloud computing impact on digital transformation in E-India. SciWaveBulletin. 2023;1(2):35-40. https://doi.org/10.61925/SWB.2023.1205

[24] Kumar MN. 5G technology is revolutionizing the wireless industry with unparalleled efficiency. SciWaveBulletin. 2023;1(3):21-28. https://doi.org/10.61925/SWB.2023.1303

[25] Vartika V, William P. Machine learning and cloud computing case study for online app usage tracking. SciWaveBulletin. 2023;1(4):32-37. https://doi.org/10.61925/SWB.2023.1405

[26] Kumar KR, Srivastava A. Location tracking drives modern company expansion. SciWaveBulletin. 2023;1(4):43-47. https://doi.org/10.61925/SWB.2023.1407

[27] Al-Dulaimy A, Jansen M, Johansson B, et al. The computing continuum: From IoT to the cloud. Internet of Things. 2024;27:101272. https://doi.org/10.1016/j.iot.2024.101272

[28] Wang W, Zhou F, Wan Y, et al. A survey of metaverse technology. Chinese Journal of Engineering. 2022;44(4):744-756. https://doi.org/10.13374/j.issn2095-9389.2022.01.15.003

[29] Wang Y, Hu Q, Su Z, et al. Large model empowered metaverse: State-of-the-art, challenges and opportunities. IEEE Network. 2026;40(4):207-215. https://doi.org/10.1109/MNET.2025.3597127

[30] Liu N, Luan TH, Wang Y, et al. QoE-oriented cooperative VR rendering and dynamic resource leasing in metaverse. IEEE Transactions on Mobile Computing. 2025;24(10):10247-10263. https://doi.org/10.1109/TMC.2025.3569695