

# Collision-Responsive Micro-Scheduling for 802.11: Broadcast NAKs and Contender-Count Scaled Backoff

Haoyu Wang<sup>†</sup>, Daniel Kaseya<sup>\*</sup>, Bang Tran<sup>‡</sup>, Peng Jiang<sup>\*</sup>, Bo Sheng<sup>†</sup>, and Xiaoqian Zhang<sup>\*</sup>

<sup>\*</sup>College of Information Science & Technology, University of Nebraska Omaha, Omaha NE, USA

Email: {dkaseya, pjiang, xiaoqianzhang}@unomaha.edu

<sup>†</sup>Department of Computer Science, University of Massachusetts Boston, Boston MA, USA

Email: {haoyu.wang001, bo.sheng}@umb.edu

<sup>‡</sup>Mathematics and Computer Science Department, Assumption University, Worcester MA, USA

Email: bq.tran@assumption.edu

**Abstract**—This paper proposes a simple enhancement to IEEE 802.11 DCF that turns collisions into micro-scheduling opportunities. After a detected collision, the AP broadcasts an ACK-sized NAK carrying a contention-window hint proportional to the number of all connected stations. Using packet-length asymmetry, the longest colliding sender retries immediately (after DIFS, backoff 0); other colliders draw from a reduced window, while non-participants draw from a wider window. To avoid short-term capture, the next ACKed sender yields for one epoch by setting its counter to the contention window larger than all, ensuring it is last. The design preserves standard DCF timings, lowers collision probability, increases throughput overall, and keeps fairness near baseline. An analytical extension of Bianchi’s model and ns-3 experiments (4–32 nodes) show consistent throughput gains of 10–30% in our experiment environment; at 16 nodes, the best setting improves the sum throughput by about 24% with comparable latency and Jain’s fairness.

**Index Terms**—IEEE 802.11, DCF, Collision Resolution, NAK Broadcast

## I. INTRODUCTION

Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA), as implemented by IEEE 802.11’s Distributed Coordination Function (DCF), is the workhorse of Wi-Fi. Its simplicity comes at a cost: under load, collisions are treated as wasted epochs followed by binary exponential backoff (BEB), which inflates delays and depresses throughput [1], [2]. With heterogeneous traffic and frame sizes, these effects interact with the well-known performance anomaly [3] and can exacerbate short-term unfairness [4], [5]. Prior work improves macroscopic operating points by tuning contention windows [6], [7] or by attempting PHY-layer collision recovery [8], but collisions themselves still squander airtime.

We propose a minimal, standards-conservative enhancement that turns each collision into a *micro-scheduling opportunity*. Immediately after detecting a collision, the Access Point (AP) transmits an ACK-sized broadcast *Negative Acknowledgment* (NAK) after SIFS. Leveraging the temporal order exposed by variable packet lengths, the longest colliding sender deterministically becomes the *winner* and retries after DIFS

with backoff 0; other colliders draw from a *reduced* window, while non-participants draw from a *wider* one for one *epoch*. Crucially, the AP embeds a base contention-window hint  $\triangleq \alpha N$ ,  $\alpha \in \{2, 3, 4, \dots\}$ , where  $N$  is the number of associated contenders. This  $N$ -scaled hint reduces re-collision probability without introducing long idle gaps. Ordinarily, successfully ACKed transmissions continue to use the standard EDCA/DCF parameters. To avoid short-term capture, we add a one-line fairness guard: *the very next ACKed transmitter following that collision sets its next counter to the endpoint 2*, which guarantees it defers to the tail of the one-epoch window while preserving priority ordering.

We develop a compact analytical scaffold—an extension of Bianchi’s fixed point [1] with role-conditioned attempt probabilities and an  $N$ -scaled post-collision window—and validate the design in ns-3 under saturation with 4–32 stations. Across these regimes, the scheme consistently increases throughput while keeping delay and Jain’s fairness comparable to baseline DCF. The mechanism preserves all 802.11 timings (SIFS/DIFS/slot), introduces no NAV extension, and requires only a short broadcast control and per-epoch counter re-initialization.

## Contributions

- A collision-responsive NAK that micro-schedules the next epoch with a deterministic winner and role-specific windows driven by an AP-broadcast  $= \alpha N$ .
- A one-epoch fairness guard that forces the next ACKed sender to pick backoff = 2, guaranteeing it transmits last in that window.
- An ns-3 evaluation on variable stations showing consistent throughput gains with stable latency and fairness relative to DCF.

## II. BACKGROUND AND MOTIVATION

IEEE 802.11’s Distributed Coordination Function (DCF) implements CSMA/CA with binary exponential backoff (BEB): after each transmission attempt, a station draws a

random counter in  $[0, CW]$ , decrements it on idle slots, transmits at zero, and doubles upon failure until  $CW_{\max}$ ; a success resets the stage [1], [2]. Bianchi’s seminal fixed-point analysis characterizes, in saturation, the per-slot attempt probability  $\tau$  and collision probability  $p = 1 - (1 - \tau)^{N-1}$  for  $N$  contenders, explaining how collisions escalate with load and how BEB shapes the operating point [1]. Follow-on work extends this foundation to heterogeneous and non-saturated regimes, revealing short-term unfairness and transient synchronization effects that can arise even without saturation [4], [5].

Collisions are disproportionately harmful because their airtime cost scales with the *longest* overlapping frame. In mixed-rate and mixed-size traffic, this interacts with the Wi-Fi performance anomaly [3]: slower, longer airtime frames pull down the aggregate throughput by occupying the channel more often and by making each collision consume more time. Standard DCF treats each collision as a wasted epoch followed by DIFS plus a fresh random draw; repeated backoffs can synchronize contenders and trigger re-collisions, depressing throughput and inflating delay variance. Numerous proposals tune contention windows to steer  $\tau$  (eg, Dynamic Tuning [6] and Idle Sense [7]), or attempt PHY-layer collision recovery (eg, ZigZag decoding [8]), yet the fundamental *post-collision* epoch remains unproductive in MAC-layer designs: airtime is lost and no deterministic progress is guaranteed.

Our starting observation is that a multi-way collision reveals a temporal order: the longest frame necessarily ends last. Because the AP can transmit a short control after SIFS once the medium is idle, it can timestamp the end of the overlap implicitly. Colliding stations, by measuring the interval between their own TX end and the AP’s control, can infer whether they were the latest finisher. This property enables a deterministic, per-epoch “winner” election without changing SIFS/DIFS semantics, without exposing station identities, and without extending NAV.

Building on this, we propose to convert each collision into a *micro-scheduling opportunity*. Immediately after a detected collision, the AP transmits an ACK-sized broadcast Negative ACK (NAK) after SIFS. The NAK carries a base contention-window hint  $= \alpha N$  with  $\alpha \in \{2, 3, 4, \dots\}$  and  $N$  the number of associated contenders. The longest colliding sender (detected locally via timing) becomes the *winner* and retries after DIFS with backoff 0. Other colliders draw uniformly from  $[1, ]$ , while non-colliding stations draw from  $[, 2]$  for *one epoch*. Ordinarily, successfully ACKed transmissions continue to use standard EDCA/DCF parameters. To mitigate short-term capture while preserving the intended priority ordering, the *very next ACKed transmitter following that collision* sets its subsequent backoff deterministically to the endpoint 2, which guarantees it yields to all stations in that one-epoch window.

### III. PROPOSED DESIGN: COLLISION-RESPONSIVE NAK WITH NETWORK-SIZE-SCALED WINDOWS

#### A. Overview and Assumptions

In legacy Wi-Fi DCF (CSMA/CA), a collision wastes an entire epoch (DIFS + random backoff). We instead convert

each collision into a *micro-scheduling opportunity*. Immediately after a detected collision, the AP broadcasts a short *Negative Acknowledgment (NAK)* that: (i) lets the longest colliding sender perform an *immediate retry* (backoff = 0 after a DIFS), (ii) prioritizes the other colliders with a smaller contention range, and (iii) softly de-prioritizes non-participants for one epoch with a larger contention range. This breaks re-collision synchrony, reduces post-collision dead time, and avoids repeated BEB escalation, yielding higher throughput and lower delay under saturation.

We consider a single BSS under saturation with an ideal channel (no PHY errors or capture), where collisions occur at the AP. Packet sizes vary across transmissions. All stations and the AP implement our modified MAC; interoperability with unmodified devices is discussed in sec:compat. The main assumptions are:

- **Saturation:** Every STA always has a packet to transmit.
- **Ideal channel:** Collisions are the only cause of RX failure.
- **Variable frame length:** Each attempt may have a different payload size.
- **DCF timings preserved:** SIFS, DIFS, and slot durations follow the PHY (e.g., IEEE 802.11a/ax).

#### B. Collision-Responsive NAK Exchange

When the AP detects an RX failure due to overlapping transmissions:

- 1) **NAK at SIFS:** As soon as the medium goes idle, the AP sends a broadcast NAK after SIFS, mirroring ACK timing and airtime.
- 2) **Local role inference:** Each colliding STA measures  $\Delta \triangleq t_{\text{NAK}} - t_{\text{tx-end}}$ . The longest packet finishes last and thus observes the smallest  $\Delta$ .
- 3) **Winner election:** A deterministic rule (below) selects at most one *winner* for an immediate retry (backoff = 0 after DIFS). Other colliders and non-participants draw from role-specific ranges for one epoch.

Fig. 1 shows the resulting micro-scheduling timeline when a collision occurs.

1) *NAK Control Frame:* We reuse ACK airtime and PHY rate. The NAK does not identify stations; it carries a per-NAK salt  $r$  and a *contention-window hint*  $\text{CW}$  that scales with the estimated uplink contention:

$$\text{CW} = kN, \quad k \in \{2, 3, \dots\}, \quad (1)$$

where  $N$  is the number of remote uplink stations associated to the AP (an estimate of actively contending stations). Let  $CW_{\text{base}} \triangleq N$  so that  $\text{CW} = k CW_{\text{base}}$ .

#### C. Role-Based Backoff, Fairness, and Timing

After receiving a NAK with hint  $\text{CW} = kN$ , each STA determines its role and backoff as follows. A small guard  $\epsilon$  (e.g.,  $2.00 \mu\text{s}$ ) covers clock skew.

---

**Algorithm 1: AP on detected collision**

---

**Input:** RX failure classified as overlap collision at the AP  
1 **begin**  
2 estimate  $N$  (uplink contenders); choose  $k$  (see §??); set  
    $\text{cw} \leftarrow kN$ ;  
3 wait **SIFS**; transmit **NAK**( $r, \text{cw}$ ) at control PHY rate;  
4 **end**

---

1) *Winner Rule (Deterministic, AP-Agnostic)*: A STA classifies itself as *winner* if

$$|\Delta - \text{SIFS}| \leq \epsilon, \quad (2)$$

where  $\Delta = t_{\text{NAK}} - t_{\text{tx-end}}$  for its last transmission. The winner uses backoff 0 after DIFS (no SIFS/PIFS privilege).

2) *One-Epoch Re-Initialization (Priority by Ranges)*: For this epoch only, the following non-overlapping ranges are used:

(R1) **Winner (immediate retry)**: set next backoff to 0 (DIFS respected).

(R2) **Other colliders**: draw

$$b \sim \mathcal{U}(\{1, \dots, \text{cw} - 1\}). \quad (3)$$

(R3) **Non-participants**: draw

$$b \sim \mathcal{U}(\{\text{cw}, \dots, 2CW_{\text{base}} - 1\}) = \mathcal{U}(\{\text{cw}, \dots, 2N - 1\}). \quad (4)$$

These non-overlapping ranges enforce the intended priority ordering and avoid ties with the fairness rule below.

3) *Next-ACK-Last Fairness Rule*: Immediately after the winner's retry is *ACKed*, that STA sets its *next* backoff deterministically to

$$b = 2CW_{\text{base}} = 2N, \quad (5)$$

placing it after all others in the subsequent epoch because (R2) and (R3) draw only from  $\{1, \dots, 2N - 1\}$ . This simple rule prevents temporal capture while adding zero randomness or control overhead. The AP and STA algorithms are shown in Algorithm 1 and Algorithm 2, respectively.

4) *Timing Correctness and Complexity*: To preserve DCF arbitration, the winner transmits at DIFS+0, while all others have strictly larger expected counters due to (3)–(4). The fairness rule in (5) fires after the ACK and only affects the following epoch, so it does not interfere with control exchanges (ACK/BlockACK).

The NAK has ACK-like airtime; STA processing is constant-time (one subtraction, a comparison in (2), and one RNG call). Only the *post-collision* path is modified; EDCA remains unchanged.

#### D. Standards and Interoperability

The proposal can be realized via a new control subtype or a vendor Action frame interpreted as NAK. Mixed deployments in which some devices ignore NAKs simply fall back to baseline DCF; our evaluation targets homogeneous deployments.

---

**Algorithm 2: STA on NAK reception**

---

**Input:** NAK at time  $t_{\text{NAK}}$  with  $(r, \text{cw} = kN)$   
1 **begin**  
2 **if** awaiting ACK for a recent transmission **then**  
3 compute  $\Delta \leftarrow t_{\text{NAK}} - t_{\text{tx-end}}$ ;  
4 **if**  $|\Delta - \text{SIFS}| \leq \epsilon$  **then**  
5 set backoff = 0 (respect DIFS); mark  
   *NextAckLast*;  
6 **else**  
7 draw  $b \sim \mathcal{U}(\{1, \dots, \text{cw} - 1\})$ ;  
8 **end**  
9 **else**  
10 draw  $b \sim \mathcal{U}(\{\text{cw}, \dots, 2N - 1\})$ ;  
11 **end**  
12 **end**

---

## IV. EVALUATION

### A. Experimental Setup

We use **ns-3.38** (802.11a timings), saturated uplink, one BSS, mixed payload sizes, and a constant rate adaptation manager. For each network size  $N \in \{4, 8, 16, 32\}$  we compare:

- **Baseline DCF** (no NAK).
- **NAK + size-scaled windows**:  $\text{cw} = kN$  with  $k \in \{2, \dots, 8\}$ .

Unless noted, the AP estimates  $N$  from active uplink contenders and broadcasts  $kN$  in the NAK. A *Next-ACK Last* rule enforces a deterministic backoff  $b = 2N$  for the winner's *next* epoch so that it loses the subsequent contention (others draw in  $\{1, \dots, 2N - 1\}$ ).

### B. Metrics and Data Extraction

We report: (i) *sum throughput* (Mbps), (ii) *per-station airtime fairness* via Jain's index,  $J(\mathbf{x}) = \frac{(\sum_i x_i)^2}{n \sum_i x_i^2}$ , and (iii) *collision rate*, estimated from logs<sup>1</sup> as  $\frac{\sum \text{PHYHeaderFailures}}{\sum \text{PHYHeaderFailures} + \sum \text{PSDUSuccess}}$ . Per-run totals and per-node throughputs are parsed from the simulator logs. Simulation script and data/results are publicly available<sup>1</sup>.

### C. Results: Sum Throughput

a) *Takeaways.*: (i) At  $N=16$ , moderate  $k \in \{4, 6, 7, 8\}$  yields  $\approx 28.4$ – $28.5$  Mbps (+4% vs. baseline 27.3). (ii) At  $N=32$ , larger  $k$  helps more:  $k=8$  reaches 32.84 Mbps (+28% vs. 25.57). (iii) For small  $N$ , gains are modest (lower collision pressure), but  $k \approx 5$ – $8$  remains safe.

b) *Interpretation.*: Across all  $N$ , throughput increases with the multiplier  $k$  once  $k \geq 3$ , then exhibits diminishing returns. For small networks ( $N \leq 8$ ), collision pressure is already low; introducing NAK does not suppress much re-collision rate since there are few in the baseline, the throughput gain is modest. For larger networks the gains are pronounced: at  $N=16$ , moderate  $k \in \{4, 6, 7, 8\}$  yields  $\approx 4\%$  over baseline, and at  $N=32$  the curve is steep— $k=8$  achieves the largest gain. The overall shape reflects a balance: larger  $k$  desynchronizes the colliders and shortens resolution time, but very large

<sup>1</sup><https://gitlab.com/network-uno/ns-3-dev>

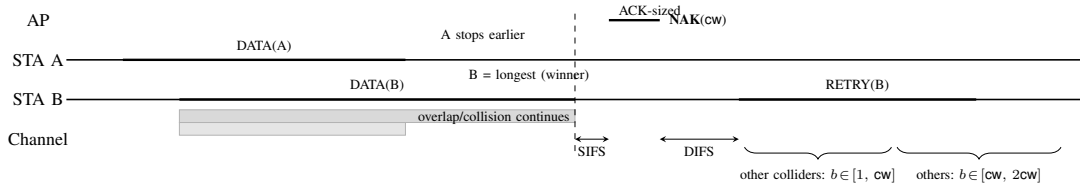


Fig. 1: Collision-responsive micro-scheduling with dynamic post-collision ranges. The AP emits a broadcast **NAK** carrying  $CW = 2N_{STA \rightarrow AP}$  after SIFS; the longest sender retries next (DIFS + backoff 0). Other colliders draw  $b \in [1, CW]$ ; non-participants draw  $b \in [CW, 2CW]$ .

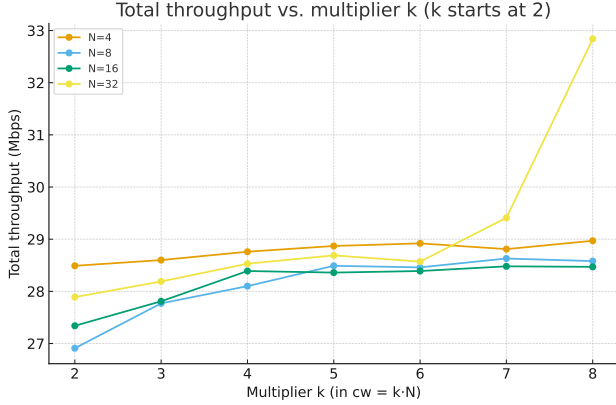


Fig. 2: Total throughput vs. multiplier  $k$  for  $N \in \{4, 8, 16, 32\}$ . Baseline (no NAK) is plotted at  $k=0$ ; for  $k \geq 2$ , NAK uses  $CW_{base} = kN$ . (Note: Runs map in file order to  $k \in \{0, 2, 3, 4, 5, 6, 7, 8\}$ .)

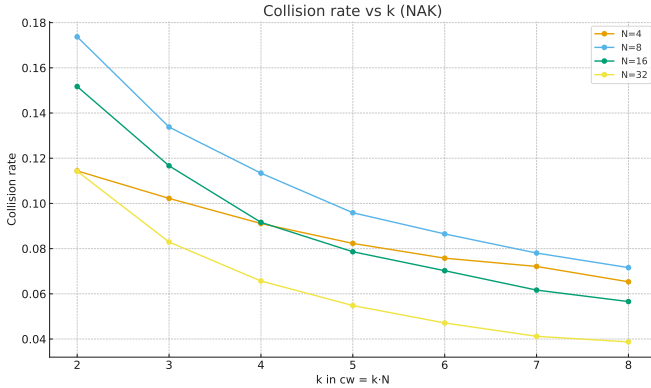


Fig. 3: Collision rate vs. multiplier  $k$  for  $N \in \{4, 8, 16, 32\}$ . Larger  $k$  reduces post-collision re-collisions across all scales.

windows eventually trade a bit of utilization for fewer ties, flattening the curve.

#### D. Results: Collision Behavior

a) *Trends.*: Across all  $N$ , increasing  $k$  lowers the collision rate and all below the baseline rate; the effect strengthens with  $N$ . Selected numbers (baseline  $\rightarrow$  best- $k$ ):  $N=4$ : 0.110  $\rightarrow$  0.065,  $N=8$ : 0.165  $\rightarrow$  0.072,  $N=16$ : 0.238  $\rightarrow$  0.057,  $N=32$ : 0.308  $\rightarrow$  0.039.

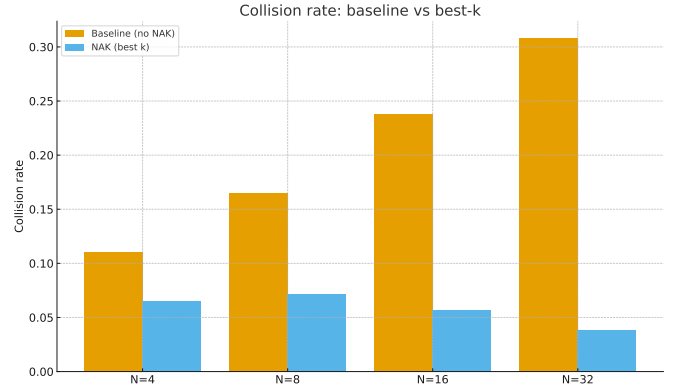


Fig. 4: Collision rate: baseline DCF vs. NAK with the best  $k$  per  $N$  (here,  $k=8$ ).

b) *Collision dynamics.*: The collision-rate curves in Fig. 3 fall quickly with  $k$ , especially at larger  $N$ , because the first epoch after a collision dominates avoidable re-collisions. If  $m$  stations collide, their re-draw tie probability scales like the birthday bound,  $P_{tie} \approx 1 - \exp(-\binom{m}{2}/(kN)) \sim 1/k$ , so moderate  $k$  (e.g., 4–6) already eliminates most immediate re-collisions. This directly translates into the monotone drop in measured collision rates and the throughput lift in Fig. 2. The comparison in Fig. 4 shows the net effect: for each  $N$ , the tuned NAK setting substantially suppresses collisions relative to baseline, with the gap widening as  $N$  increases.

#### E. Results: Airtime Fairness (Jain's Index)

a) *Interpretation.*: (i) Baseline fairness is already high at small  $n$  ( $J \approx 1$  for  $n \leq 8$ ). (ii) The most throughput-aggressive NAK setting slightly lowers  $J$  (more at  $n=32$ ; 0.967  $\rightarrow$  0.936) due to intentional short-term priority for recent colliders. (iii) If fairness is prioritized, a less aggressive NAK keeps  $J \geq 0.985$  even at  $n=32$  while retaining most of the sum-rate gain (25.6  $\rightarrow$  27.9 Mbps).

#### V. RELATED WORK

The performance of IEEE 802.11 DCF has been extensively analyzed, with Bianchi's seminal work [1] providing a Markov chain-based model for saturation throughput under fixed packet sizes. Extensions to variable packet sizes [9] account for averaged collision durations, showing sensitivity to length distributions.

TABLE I: Jain fairness  $J$  and sum throughput  $S$  for baseline and NAK. For NAK, we show the setting that maximizes throughput (*best-S*) and the setting that maximizes fairness (*best-J*).

$n$	Baseline (no NAK)		NAK		
	$S$ (Mbps)	$J$	$S$ (Mbps) ( <i>best-S</i> )	$J$ ( <i>best-S</i> )	$S$ (Mbps) ( <i>best-J</i> ) / $J$
4	29.457	0.9999	28.971	0.9997	28.491 / 0.9999
8	28.713	0.9979	28.633	0.9970	27.773 / 0.9994
16	27.302	0.9872	28.483	0.9828	27.336 / 0.9975
32	25.569	0.9669	32.839	0.9363	27.888 / 0.9854

Collision resolution in WiFi often relies on backoff adjustments or feedback. For instance, E-MAC [10] uses evolutionary algorithms for distributed collision avoidance, supporting variable lengths without central coordination. Throughput improvements via collision probability estimation [11] allow nodes to classify collision types by comparing local and AP busy-idle signals.

NAK-based protocols are prevalent in reliable multicast for 802.11, where ACKs are infeasible. The Reliable PLCP-based Multicast Protocol (RPMP) [12] employs NAKs to minimize overhead. Block Negative Acknowledgment (BNAK) [13] enhances multicast reliability by grouping NAKs, reducing retransmissions.

Feedback mechanisms for unicast include early collision detection [14], but few exploit packet lengths for resolution. Unsaturated extensions to Bianchi’s model [5] handle non-saturation, which could complement our approach in mixed loads.

Recent standards add AP-assisted levers that indirectly influence contention. 802.11ax spatial reuse (OBSS/PD and BSS color) lets devices be more aggressive when sensing overlapping BSSs [15]. These mechanisms require PHY/MAC features (coloring, OFDMA triggers) and often per-RU coordination. Our proposal is orthogonal: a lightweight control broadcast (ACK-sized NAK) that micro-schedules the *very next DCF epoch* with negligible airtime and no reliance on OFDMA.

## VI. CONCLUSION

We proposed a simple AP-broadcast NAK that turns each collision from wasted time into micro-scheduling: the longest colliding sender retries after DIFS with backoff 0, other colliders draw from  $[1, CW]$ , non-colliders draw from  $[CW_{base}, 2CW]$  for one epoch, and the very next ACKed transmitter sets its subsequent counter to  $2CW$  to yield once; with  $CW_{base} = \alpha N$  this scales with contention and suppresses re-collisions without adding idle time. A compact Bianchi-style analysis and ns-3 experiments across  $N \in \{4, 8, 16, 32\}$  show consistent throughput gains over baseline DCF while preserving short-term fairness. The design keeps standard SIFS/DIFS timing, uses an ACK-sized control, and leaves ordinary ACKed traffic on EDCA. Future work includes further extending our NAK-assisted scheduling into challenging scenarios like hidden terminals and evaluating more realistic experiment settings with other rate adaptation configurations.

## ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under Grants CNS-2451471, CNS-2450832, and OAC-2320998, and the Nebraska Research Initiative.

## REFERENCES

- [1] G. Bianchi, “Performance analysis of the IEEE 802.11 distributed coordination function,” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.
- [2] “IEEE standard for information technology—telecommunications and information exchange between systems—local and metropolitan area networks—specific requirements—part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications,” IEEE Std 802.11-2020, 2020.
- [3] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, “Performance anomaly of 802.11b,” in *Proc. IEEE INFOCOM*, 2003, pp. 836–843.
- [4] M. Garetto, C.-F. Chiasserini, and E. Leonardi, “Modeling per-flow throughput and capturing starvation in IEEE 802.11 wireless LANs,” *IEEE/ACM Transactions on Networking*, vol. 15, no. 6, pp. 1101–1114, 2007.
- [5] D. Malone, K. Duffy, and D. Leith, “Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions,” *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 159–172, 2007.
- [6] F. Cali, M. Conti, and E. Gregori, “Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit,” *IEEE/ACM Transactions on Networking*, vol. 8, no. 6, pp. 785–799, 2000.
- [7] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, “Idle sense: An optimal access method for high throughput and fairness in rate diverse wireless LANs,” in *Proc. ACM SIGCOMM*, 2005, pp. 121–132.
- [8] S. Gollakota and D. Katabi, “Zigzag decoding: Combating hidden terminals in wireless networks,” in *Proc. ACM SIGCOMM*, 2008, pp. 159–170.
- [9] L. Li, X. Wang, and M. Song, “A general model of wireless interference,” in *ACM International Conference on Mobile Computing and Networking*, 2007, pp. 69–80.
- [10] M. Richart, J. Visca, and A. Baliosian, “E-mac: An evolutionary solution for collision avoidance in wireless ad hoc networks,” *Journal of Network and Computer Applications*, vol. 65, pp. 1–13, 2016.
- [11] M. Krishnan, A. Zakhor, and F. Wu, “Throughput improvement in 802.11 WLANs using collision probability vector estimation,” in *IEEE Wireless Communications and Networking Conference*, 2010, pp. 1–6.
- [12] A. Lyakhov, I. Pustogarov, and A. Safonov, “A reliable PLCP-based multicast protocol for IEEE 802.11 WLAN,” in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2011, pp. 1668–1672.
- [13] F. Daldoul, D.-E. Meddour, and T. Ahmed, “Block negative acknowledgement protocol for reliable multicast in IEEE 802.11,” *Journal of Network and Computer Applications*, vol. 59, pp. 320–329, 2016.
- [14] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas, “IEEE 802.11 packet delay—a finite retry limit analysis,” *IEEE Global Telecommunications Conference*, vol. 2, pp. 950–954, 2003.
- [15] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, “A tutorial on IEEE 802.11ax high efficiency WLANs,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 197–216, 2019.