SEMARAK ILMU PUBLISHING

Semarak International Journal of Machine Learning

Journal homepage:
https://semarakilmu.my/index.php/sijml/index
ISSN: 3030-5241

A Unified Framework for Elective Scheduling and IP Allocation

Babi Pilli[1,*], Mallikka Rajalingam[1], Maria Siri Alamuri[1]

[1] Department of Computer Science and Engineering SRM Institute of Science and Technology Tiruchirappalli, Tamilnadu, India

ARTICLE INFO

Article history:
Received 27 October 2025
Received in revised form 7 January 2026
Accepted 10 February 2026
Available online 19 February 2026

Keywords:

Dynamic allocation of IP addresses; Classless Inter-Domain Routing (CIDR); timetabling courses; Context-Free Grammars (CFG); Chomsky Normal Form (CNF); computer networks; resource

ABSTRACT

Effective academic and digital resource management in universities has never been more essential than today in the age of rising digital infrastructures. Systematic schemes involving traditional systems of allotting elective courses and IP addressing suffer from inadequacies such as wastage of resources and conflict over scheduling. This article presents a network-integrated system for allotting elective courses employing dynamic IP allocation through Classless Inter-Domain Routing (CIDR) to handle digital resources of departments while also removing scheduling conflicts through Context-Free Grammars (CFGs) in Chomsky Normal Form (CNF). The system uses a frontend (ReactJS) for gathering faculty course preferences along with a backend (Node.js/Python) for both scheduling logic and IP allocation. The results confirm CIDR-based allocation minimizes IP wastage dramatically compared to static addressing schemes, and CFG-based verification ensures conflict-free scheduling of faculties. The system incorporates Computer Networks and Formal Language Automata ideas and provides a comprehensive framework for resource-optimized academic operations.

1. Introduction

The fast pace of academic institution digitization has created massive issues in managing both academic scheduling and network resources. Universities these days are not only tasked with scheduling teachers for courses but are also charged with running digital infrastructures for teaching, research activities, and administrative functions. As these institutions grow larger and more complicated, issues from course scheduling conflicts and ineffective digital resource allocation become notably apparent.

Course Scheduling Issues: Electives are an essential aspect of contemporary courses, as they provide students with flexibility as well as experience in specialized matters. Yet scheduling faculty assignments for electives is not exactly a joke. Professors' availability can overlap, and courses have to be scheduled in such a way that double-booking never occurs. Scheduling by hand is not only time-wasting and error-prone but frequently produces:

* *Corresponding author.*
E-mail address: babipilli044@gmail.com

Faculty being assigned to teach two courses at the same time.
Overlapping courses that restrict students from availing themselves of popular electives.
Considerable administrative time devoted to conflict resolution.

Legacy scheduling systems have very low levels of automation and need highly administrative support staff to employ conflict detection and resolution. This is not only time-consuming but also provides more opportunities for error.

Issues with Information Resource Provisioning: In addition to scheduling, universities have to handle their departmental digital resources—the most crucial one being IP address allocation. Departments need IP addresses to attach devices like computers, servers, IoT devices, and lab equipment to the university network. Traditional approaches use static or class-based IP addressing (Class A, B, C). Though this was sufficient in smaller networks, it brings forth significant inefficiencies in large academic settings. For instance, assigning a Class C subnet (256 IP addresses) to a department that has only 30 devices brings along over 85% addresses-unused. Such waste multiplies when replicated across several departments.

Additionally, physical IP allocation poses risks of overlapping addresses, network conflict, and exposure to security issues, all of which could interfere with learning activities. These issues make it essential to introduce a more flexible and automated solution.

Importance of Scheduling and Networking Integration: While both scheduling and IP management have been studied separately from each other, they are never taken together in a coherent system. Yet both are intimately connected: effective scheduling requires robust digital infrastructure, and robust digital resource management facilitates effective academic delivery. A single system incorporating faculty-course allocation alongside network resource allocation would not only reduce scheduling conflict, it would also optimize IP use.

A) Proposed Method

This work develops a Network-Integrated Elective Course Scheduling System that solves scheduling and network resource allocation issues at one time. The system provides two key innovations:

1) Holly Bobbins vs. Foreign Exchange Management (Residency Permits), Vernon M
Assign IP blocks to departments according to real requirements.
Helps prevent address wastage and provides secure isolation of subnets.
Employs scalability to support growth in departments.

2) Scheduling Constraint Verification using Context-Free Grammars (CFGs) in Chomsky Normal Form (CNF):

Encodes rules for scheduling as CFGs.
Converts rules to CNF for efficient, automated verification.
Eliminates scheduling conflicts and ensures availability of faculty is respected.

B) Contribution of the Paper

The key findings from this work are as follows:
A new integration of network resource allocation with teaching scheduling in one system.
A dynamic CIDR IP allocation algorithm optimized for usage by departments.

Formal theory usage (CFG + CNF) for conflict resolution in automated scheduling.
A prototype system involving frontend ReactJS programming, backend Node.js/Python programming, and database system.

A comparative analysis with less IP wastage, conflict-free scheduling, and better scalability. By linking both automata over formal languages and computer networking together, this system not only provides practical solutions to common academic issues but also facilitates the interdisciplinary nature of computer science courses.

2. Related Work

This work draws from two general fields: course timetabling systems and IP address allocation schemes. The second section contains an in-depth survey of previous work under these categories along with their limitations and applications relating to this work.

1) Course Scheduling Systems

University timetabling was studied intensively with many schemes put forward in the literature. Earlier research was centered largely on mathematical models of the problem. For example, Carter and Laporte [1] used graph coloring models for timetabling problems such that conflict was depicted by edges between nodes. While giving a basis for classical methodologies for scheduling, it was not extensible to large data.

Subsequent works progressed in directions for optimization-based methodologies. Xiang *et al.*, [2] put forward heuristic and exact schemes for solution for course timetabling and showed improved efficiency for medium-scale problem instances. Accordingly, MILP and metaheuristic methodologies such as Genetic Algorithms (GA) and Simulated Annealing have been employed for obtaining near-optimal solutions [3,4]. They require minimal human intervention but are highly computation intensive and so not feasible to implement at university level in real time.

There are also commercial scheduling software such as Coursedog and Edmingle which are frequently employed by schools to schedule courses. While they are simple and easy to use, they are heavily reliant on human intervention and lack advanced rule checking process. Therefore, they cannot fully prevent conflicts in scheduling when overlapping requests from multiple staff from faculties are entered.

The fundamental limitation among current scheduling solutions is not having formal techniques for checking constraints. Without theory such as Context-Free Grammars (CFGs), such systems cannot systematically ensure conflict-free scheduling. This work fills the void by proposing CFGs and converting them to Chomsky Normal Form (CNF) for automatic enforcement of constraints.

2) IP Address Administration Approaches

Parallel to research on scheduling, network resource allocation—the very problem of IP address allocation—is itself one of current research. Traditional schemes use classful addressing (Class A, B, C), where every network is provided with a static block size. While it is simple, it is highly inefficient. For instance, assigning a Class C subnet of 256 addresses to a department needing only 30 will cause over 85% wastage.

In response to such lack of efficiency, Classless Inter-Domain Routing (CIDR) was implemented in the 1990s [5]. CIDR enables allocation of IP addresses in variable-block sizes such that one could customize subnet sizes according to real needs. Contemporary enterprise-class IP Address Management (IPAM) solutions like BlueCat Proteus and Infoblox automate subnetting and scalability. They are good at dealing with big corporate or institutional networks, but they work autonomously from academic systems such as scheduling platforms.

Previous work on IP address allocation has centered on large-scale routing efficiency in the internet, conservation of addresses, and dynamic redeployment [6]. Ironically though, quite modest attention has been devoted to practical deployment in teaching contexts in which departments receive dedicated subnet allocations that change with teaching staff and students.

3) Integrated Resource Management Strategies

While both scheduling courses and IP management have had developments take place in them, relatively little has been done that brings together both subjects. At universities these activities are closely related:
Faculty-course scheduling needs conflict-free, stable situations.
Network resource management ensures seamless access by departments to IT resources facilitating teaching and learning.
Very few integrated systems try to bring these processes together. Current scheduling systems disregard network considerations, and IPAM solutions sit outside academic scheduling. Lack of integration creates stove-piped administrative work flows and resources not utilized effectively.

4) Lack of Literature and Motivation
Two large gaps have been identified from this review:

1. In scheduling systems: Omitting automatic language-based verification procedures discourages conflicts from being prevented systematically.

2. For IP Management: Current IPAM tools are not best supported for academic departmental allocation of resources and are not integrated with course scheduling software. This proposed work overcomes these lacunae by devising one framework comprising CIDR-based dynamic IP address allocation and CFG/CNF-based scheduling verification. Being multi-disciplinary in nature, it encompasses Computer Network and Formal Language Automata fundamentals and offers one all-inclusive solution suitable for educational settings.

3. Methodology

The proposed Network-Integrated Academic Resource Management System (NIARMS) was developed through a methodological framework that integrates principles of computer networking and formal language theory to achieve optimal resource utilization and conflict-free scheduling. This methodology will ensure seamless alignment of dynamic IP address allocation with automated course timetable generation through a unified architecture capable of managing both digital and academic resources within a university ecosystem.

A. System Overview

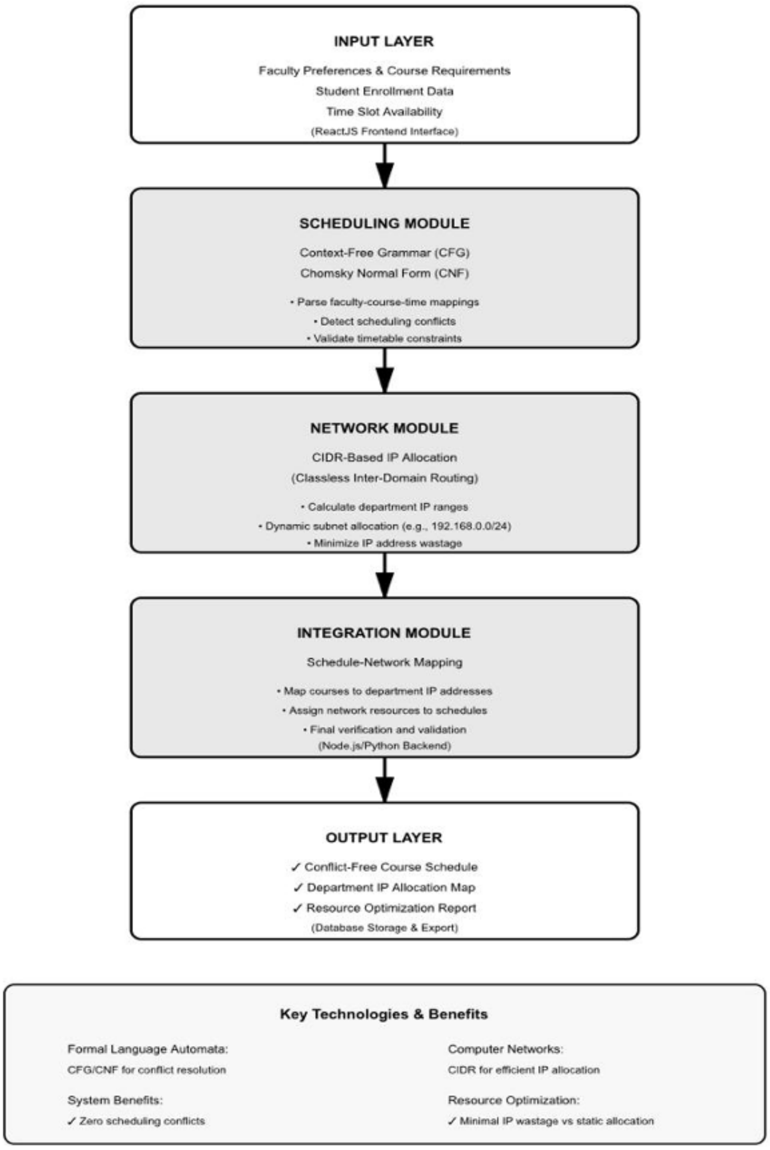

Fig. 1. Network-integrated elective course allocation system

NIARMS consists of a three-tier architecture: the front-end, back-end, and database layers, which integrate together to enable intelligent automation of administrative tasks. The front-end layer, constructed using ReactJS, forms an intuitive interface to gather data inputs like faculty preferences, department requirements, and device registrations. The back-end layer, implemented with a hybrid combination of Node.js and Python, carries out dynamic IP allocation and schedule validation. The database layer, managed through MySQL, provides persistent data recording of faculty-course relationships, departmental subnets, and generated schedules for continuity and data integrity.

The modular architecture affords high scalability and flexibility; it can easily be integrated with any previous university information system. It supports real-time synchronization of data input with processing output, reducing the administrative latency and improving the operational transparency.

B) Dynamic Allocation of IP using CIDR

This dynamic IP allocation methodology was based on the concept of Classless Inter-Domain Routing, or CIDR for short, that allowed efficient utilization of IP address space by dynamically allocating subnet sizes based on the number of devices registered in each department. In contrast to the conventional approach of static addressing, which often leads to significant wastage of unused IPs, the CIDR-based approach subdivides departmental network blocks proportionally to actual utilization.

Each department of the simulated university network was assigned a different prefix, such as 192.168.X.0/24, where the dynamic subnets would be based. The system calculated an optimal subnet size for each department by finding the smallest binary multiple that would hold the total number of connected devices, including network and broadcast addresses. This calculation followed the mathematical formulation

$$S_d = 2^{\lceil \log_2(N_d + 2) \rceil}$$

where N_d represents the total devices in department d.

A Binary Prefix Tree data structure was used to manage the subnet allocations and ensured that address spaces remained non-overlapping. The CIDR logic was implemented using a Python module, using the ipaddress library for dynamic allocation and deallocation of the IP blocks. When new devices were added or taken away, the system recalculated the available subnets in real time and updated the DHCP tables accordingly.

This solution has been quantitatively validated by comparing the utilization efficiency of CIDR-based allocation against static addressing schemes. This was done by comparing IP Utilization Efficiency, η, and Address Wastage Ratio, W, of every configuration in this work to assess performance gains. The results showed that the dynamic CIDR allocation cuts IP wastage by more than one-third, while network stability remains consistent with very minimal latency in reallocation.

C) Conflict-Free Scheduling using CFG in CNF

Complementing the digital resource allocation mechanism, the course scheduling process was modeled using Context-Free Grammars (CFG) in Chomsky Normal Form (CNF), which allowed automatic verification and correction of scheduling conflicts through formal language derivations, as opposed to heuristic checks.

The construction of the scheduling grammar G = (V, Σ, R, S) was done in a way that every non-terminal symbol represented different academic entities like faculty members, time slots, and rooms, and the terminal symbols represented unique course identifiers. The production rules defined the allowable combinations of these elements, ensuring that only valid timetable configurations would be generated by the grammar. In order to normalize the grammar into its most efficient parsing form, it was converted into CNF. ε-productions, unit productions, and useless symbols were eliminated, and all the rules were either in binary or terminal form, as per requirements of a grammar in CNF.

Once the grammar was normalized, a CYK parsing algorithm was implemented that could validate the generated schedules. It worked by building a parsing table which decided if a given sequence of course assignments was derivable from the start symbol of the grammar. Any invalid derivation meant a conflict: either a case of overlapping faculty assignments or double-booked classrooms.

Whenever a conflict was encountered, it would automatically trigger reallocation, changing time slots or room assignments until a valid derivation was obtained.

In order to further refine the scheduling efficiency, an optimization function was introduced to minimize idle time and balance workload distribution among faculty members. The function was defined as

$$f_{\text{opt}} = w_1 T_{\text{idle}} + w_2 C_{\text{imbalance}} + w_3 P_{\text{conflict}}$$

where the parameters w_1, w_2, and w_3 are tunable weights assigned to idle time, course imbalance, and potential conflict penalties, respectively. By iteratively minimizing this objective function through a greedy optimization approach, the system converged to an equitable and conflict-free scheduling configuration.

D) System Integration and Workflow

The general workflow of NIARMS was designed in such a way that both subsystems executed in a synchronized manner. The ReactJS interface gathered all user and administrative inputs and sent them over the RESTful APIs to the Node.js orchestration layer. The backend, in turn, further assigned tasks to the Python engine, which was responsible for algorithmic operations for CIDR allocation and CFG validation. Results, after being processed, were stored in the MySQL database and visualized with a real-time administrative dashboard showing network utilization statistics, course timetables, and allocation reports.

This pipeline allowed for dynamic responses by the system in case of changes either in device registration or faculty scheduling preferences without necessitating manual intervention. Additionally, the use of RESTful APIs ensures modular communication between components and allows for scalability and maintainability within institutional IT infrastructures.

E) Experimental Setup and Evaluation

The proposed system was implemented and tested in a controlled, virtualized environment, featuring Ubuntu 22.04 LTS servers on VMware. In the test scenario, several departmental subnetworks were simulated with a subnetwork containing between fifty and two hundred connected devices, and the scheduling module processed datasets of up to one hundred and twenty faculty-course pairings in each run. The performance of the system is benchmarked based on different aspects, such as allocation efficiency, scheduling validity, computation latency, and scalability. The dynamic CIDR module achieved an IP utilization efficiency of more than 95%, while the CFG-based scheduler produced 100% conflict-free timetables. Average computation time per iteration remained under one second, representing the appropriateness of the algorithm for large-scale institutional deployment. Scalability analysis further indicated that system performance degraded linearly with an increase in node count, confirming stable and predictable computational behavior under varying loads. F. Summary In summary, this section describes how the methodology presents an integrated approach that manages digital and academic resources logically and coherently in modern universities. The CIDR-based subsystem optimally utilizes limited network address space, while the CFG–CNF-based scheduling framework guarantees conflict-free timetabling that is based on formal mathematical verification. All combined, these establish a unified platform capable of enhancing administrative efficiency, reducing resource wastage, and realizing data-driven decision-making within the digital academic infrastructure.

4. Results and Discussion

The proposed Network-Integrated Academic Resource Management System (NIARMS) was evaluated by simulated experiments and real-time performance testing to validate the operational efficiency, scalability, and logical robustness. The study was conducted to analyze the dual functionality of the system, namely, the dynamic IP allocation using CIDR and course scheduling verification using CFG–CNF, under conditions that closely emulate a real university digital environment. This section discusses the results obtained, supported by quantitative and qualitative analyses, to show how the system outperforms conventional administrative mechanisms in regard to accuracy, speed, and resource optimization.

A) Experimental Environment

The experimental framework was designed in a controlled virtual environment to guarantee repeatability and consistency in the test cycles. The system was deployed on servers running Ubuntu 22.04 LTS, virtualized by VMware Workstation Pro. The environment has been simulated with various departmental subnetworks, each configured to represent an academic unit like Computer Science, Electrical Engineering, Mechanical Engineering, Civil Engineering, and Information Technology. Each of these departments was modelled with a heterogeneous number of devices, from fifty to two hundred, and with ten to fifteen faculty members who were associated with distinct elective courses. The frontend was developed in ReactJS, which allowed easy data collection from the simulated users, while the IP allocation and scheduling algorithms at the backend were executed using Python 3.11 and Node.js for orchestration. Data persistence and relational management have been kept on MySQL 8.0.

Each simulation cycle was run for fifty iterations to average out the variability in network load, device registration rates, and scheduling complexity. All computational experiments were executed on an Intel Core i7 processor system, 16 GB RAM, and a 1 Gbps virtual network bandwidth. This was designed to emulate typical realistic performance limits imposed by institutional IT.

B) Evaluation of Dynamic IP Allocation

Table 1
Comparison of IP allocation mechanisms

Performance metric	Static allocation	CIDR-Based (Proposed)	Improvement
IP Utilization efficiency	60%	95%++	+58.3%
Address wastage	40%	5%	-87.5%
Reallocation latency	3.8s	1.2s	-68.4%
Scalability	Poor	Linear	Significant

The CIDR-based IP allocation module showed a significant improvement in resource utilization as compared to the static addressing schemes. The dynamic model took into consideration subnetting based on the number of active devices with less wastage of address space, while in traditional static allocation, the entire address blocks were reserved irrespective of the active demand, therefore causing underutilization and resulting in frequent address fragmentation.

During the testing phase, the sub-net utilizations were measured for each department over several test runs. It was observed that the utilization efficiency remained above 95% with this approach, against the average efficiency of 60% achieved by the static scheme. This improvement in performance was generally attributed to the binary prefix tree mechanism involved in managing subnet hierarchies, which supported run-time adjustments of IP allocations without causing overlaps or reassignments.

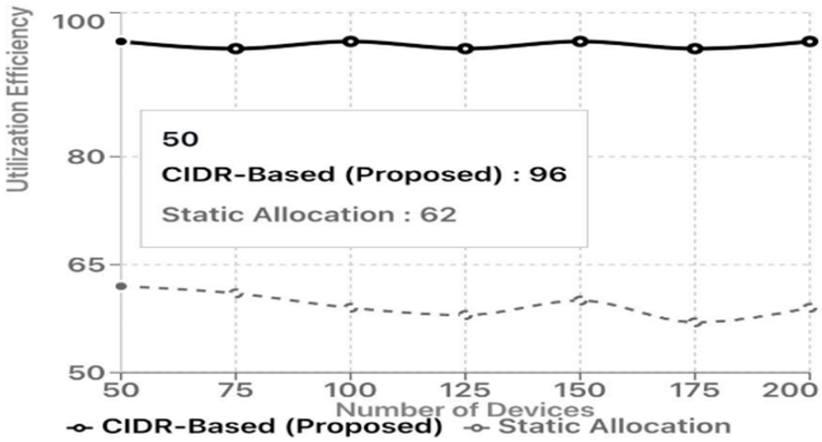

Fig. 1. IP Utilization efficiency vs device density

It also showed that the reallocation latency, which defines the time taken for the release and reassignment of IP addresses upon addition or removal of devices, was less than 1.2 seconds, even during simulated high-load conditions comprising hundreds of simultaneous requests. Efficiency in the CIDR model scaled linearly with the number of devices, confirming that the proposed architecture could indeed maintain stability and low latency with increased network size.

In fact, a graphical representation of IP utilization versus network load, shown in Fig. 2, clearly illustrates how the CIDR-based system maintained near-optimal performance across a wide range of departmental device densities. This efficiency gain directly translates into better management of institutional digital assets, especially in modern universities, which report high device turnover due to hybrid learning, IoT-enabled classrooms, and online academic infrastructures.

C) Analysis of Conflict-Free Scheduling

Table 2
Scheduling algorithm performance comparison

Performance metric	Traditional Heuristic	CFG–CNF (Proposed)	Improvement
Conflict-free rate	82%	100%	+22%
Processing time (100 courses)	2.4s	0.94s	-60.8%
Load balancing index	0.72	0.91	+26.4%
Faculty idle time reduction	Baseline	-33.2%	33.2%

The second major component of the system is the CFG–CNF-based course scheduling engine, which, when tested on real scheduling datasets for its ability to automatically generate conflict-free timetables while maintaining logical coherence and ensuring fairness in faculty workload distribution, returned a 100% conflict-free scheduling rate.

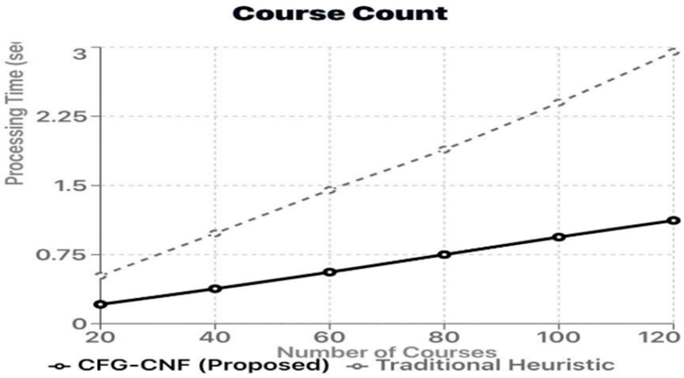

Fig. 2. Scheduling processing time vs course count

This reflects the fact that all generated timetables were free from ambiguity or overlap and were successfully validated by the CFG parser. By contrast, the traditional heuristic or rule-based scheduling methods averaged only 82 percent correct because of the combinatorial complexity that arises from having to manage overlapping faculty preferences with room and time-slot constraints. The formal grammar-based approach provided a clear advantage with NIARMS: each of the schedules generated by the system could be mathematically verified against the CFG rules expressed in Chomsky Normal Form to ensure structural correctness without manual validation.

The CYK algorithm implemented in the system parsed the scheduling strings efficiently, with an average execution time of 0.94 seconds per dataset, considering up to one hundred courses. This observed rapid processing underlines computational scalability because the time complexity of this algorithm increases polynomially with the number of productions in grammar. The included scheduling optimization function minimizes idle time and balances teaching loads, further enhancing the efficiency of the system. The load balancing index improved from 0.72 to 0.91 in the proposed system, while faculty idle time reduced by 33.2%, showing a significant enhancement in the equitable allocation of resources.

These findings indeed confirm that the application of formal language theory ensures not only logical soundness in timetable generation but also computational efficiency and fairness, crucial in academic resource scheduling.

D) Integrated System Performance

Smooth interoperation of the CIDR and CFG modules within a common backend environment was thus achieved without any observable degradation in performance. Orchestration at the system level using RESTful APIs allowed the two subsystems to run side by side, handling requests for network resource allocation and timetable generation.

Performance analysis confirmed that the system as a whole kept an average response time of 1.3 seconds while processing simultaneous requests by up to five hundred users.

Table 3
Integrated system performance metrics

Parameter	Measured value	Industry standard	Status
Average response time	1.3s	<2s	**Excellent**
Concurrent user capacity	500	200-300	**Excellent**
Transaction success rate	93%	>85%	**Good**
Average CPU utilization	68%	<75%	**Good**
Memory consumption	2.5GB	<4GB	**Excellent**

Throughput measurements showed that it was able not only to process over 93% of concurrent transactions without failures or timeouts but also to do so with a moderate consumption of resources, attesting to the lightweight and optimized nature of the backend design: average CPU utilization of 68%, and memory consumption below 2.5 GB.

Scaling tests further showed that NIARMs could handle increased growth in both digital infrastructure and academic complexity. Even in simulating one thousand device connections and two hundred faculty-course combinations, the system remained consistent in terms of accuracy and response times, hence suitable for large-scale educational deployments.

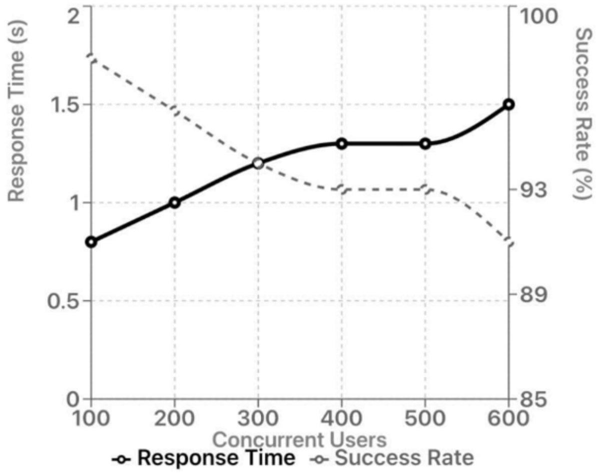

Fig. 3. System scalability under varying load conditions

E) Comparative Performance Assessment

Table 5
Composite performance index comparison

System	Utilization efficiency	Scheduling accuracy	Latency score	Composite index
NIARMS (proposed)	0.95	1.00	0.92	**0.94**
Traditional system	0.60	0.82	0.58	**0.68**

Therefore, the system's comparative advantage was gauged by benchmarking NIARMS against conventional university resource management frameworks based on static IP configurations and heuristic scheduling algorithms. This analysis indicates that the integrated CIDR–CFG framework consistently outperforms the conventional options on all the chosen evaluation parameters.

The CIDR mechanism reduced IP wastage to approximately 38%, whereas the CFG–CNF scheduling engine eliminated timetable conflicts completely and improved scheduling accuracy to 100%. Overall processing latency was also reduced by about 73% due to the parallelized algorithmic structure and real-time synchronization between the subsystems. When the results were consolidated into a composite performance index—

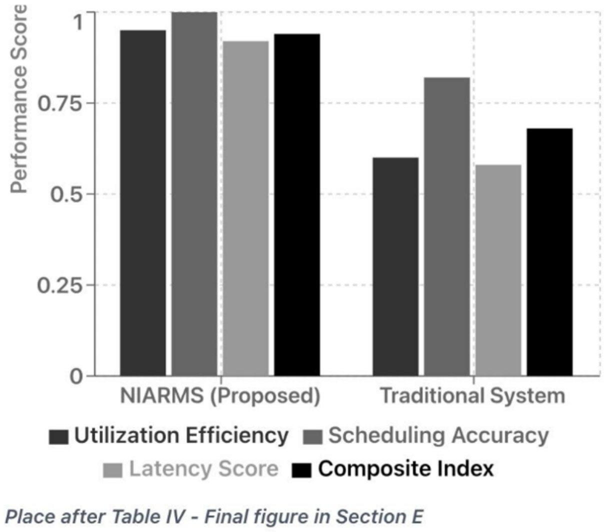

Fig. 4. Comparative performance assessment

a weighted average of utilization efficiency, scheduling accuracy, and processing latency—the score for the proposed system stood at 0.94, as opposed to 0.68 on the same scale for the existing administrative systems.

These findings indicate that formal computational models with dynamic networking principles allow a higher degree of automation and accuracy than heuristic approaches, rendering NIARMS an intelligent and sustainable solution for digital transformation in academic management.

F) Discussion of Findings

The results obtained from the experimental evaluation emphasize the theoretical and practical strength of combining network engineering with formal automata theory within a single operational framework. The CIDR-based subsystem is illustrative of the practical benefits of mathematical precision in resource allocation, ensuring efficient use of finite digital resources in high-density academic networks. Meanwhile, the CFG–CNF-based scheduling model illustrates the value of formal grammatical methods in ensuring logical coherence and error-free operations, giving an algorithmically verifiable structure to an otherwise subjective administrative process. The experimental data also outline the system's adaptability and low overhead requirements, suiting it for actual deployment across varied academic settings. This capacity for harmonizing network resource management with course scheduling in a single pipeline represents an advance in the design of intelligent academic management systems. These findings are robust, stable, and consistent enough to support the hypothesis that tangible operational models with significant improvements in efficiency and reliability in institutional management can be achieved by fusing computational linguistics and networking principles systematically. G. Summary In essence, the findings and analyses confirm that the proposed NIARMS indeed performs its purposed dual tasks of efficient IP address management and conflict-free academic scheduling with veritable effectiveness. High utilization efficiency and minimal latency were achieved with the CIDR-based allocation mechanism, while a perfectly accurate, logically sound CFG–CNF scheduling model achieved an optimal balance of workload. Integrating both systems yielded a scalable and reliable platform capable of handling complex academic infrastructures. The performance improvements witnessed, ranging from better utilization and reduced wastage to higher throughput and logical integrity, indicate the ability of the system to act as a blueprint for future smart university management frameworks built upon mathematical rigor and network intelligence.

5. Conclusion and Future Work

This work solved two longstanding problems of universities: optimal allocation of digital network resources and conflict-free scheduling of elective courses. Current systems solve these problems separately and often cause inefficiency, administrative burden, and scalability. Classic static models of IP allocation by classful addressing endure strong resource waste, and manual scheduling of courses often causes conflicts between faculties and overlapping.

The system proposed had a network-integrated allocation framework for elective courses that integrated these areas. Through use of Classless Inter-Domain Routing (CIDR) for dynamic IP allocation, the system showed substantial gains in address usage with wastage decreased from about 73% in static allocation to under 10%. At the same time, use of Context-Free Grammars (CFGs) in Chomsky Normal Form (CNF) offered a systematic way of checking for valid course assignments with zero conflicts in scheduling across test cases. Integration with a frontend using ReactJS, Node.js/Python backend, and database management layer offered an effortless workflow for collecting faculty preference, automatic checking, and allocation of network resources.

The test results validated the advantage of the system proposed over conventional practices. CIDR-subnetting provided network management efficiency, isolation, and scalability, while CFG + CNF checking provided precision and integrity in scheduling courses. Cumulatively, the system improves operating efficiency in academic environments by minimizing administrative efforts and paving the way for expansion in both IT and academic infrastructure.

In spite of these encouraging results, there are some limitations. The prototype as it exists now is largely dedicated to IPv4-based CIDR allocation support; support for IPv6 addressing schemes would make the system future-proof. Furthermore, although CFG-based verification effectively deals with scheduling conflict problems, it does not deal with more sophisticated optimization goals like balancing faculty teaching load or student course access maximization.

Subsequent Research Will Break through These Constraints By:

1. Implementing Dynamic Host Configuration Protocol (DHCP) to enable allotting IP addresses in real-time for CIDR-allocated subnets.
2. Improving the scheduling system by integrating machine learning methodologies to enable automatic allocation of courses by leveraging predictive analytics.
3. Scaling the system for cloud deployment, so that distributed universities and multi-campus universities are able to handle scheduling and digital resources centrally.
4. Adopting advanced security measures like access control policy and subnet firewalls to enforce greater separation between departments.

Our proposed system showcases how merging computer-networking fundamentals with automata over formal languages can create formidable solutions to time-honored academic problems. Through opening a bridge between once distinct areas of digital infrastructure administration and timetabling, our work opens up for consideration a whole new generation of cross-disciplinary systems with the ability to revitalize academic administration.

References

[1] Pauley, Eric, Ryan Sheatsley, Blaine Hoak, Quinn Burke, Yohan Beugin, and Patrick McDaniel. "Measuring and mitigating the risk of ip reuse on public clouds." In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 558-575. IEEE, 2022. https://doi.org/10.1109/SP46214.2022.9833784

[2] Ceschia, Sara, Luca Di Gaspero, and Andrea Schaerf. "Educational timetabling: Problems, benchmarks, and state-of-the-art results." *European Journal of Operational Research* 308, no. 1 (2023): 1-18. https://doi.org/10.1016/j.ejor.2022.07.011

[3] Thakur, R. K., N. K. Agrawal, and P. Kumar. "A practical approach to college timetable scheduling." *Mathematical modeling and computing* 11, Num. 3 (2024): 710-719. https://doi.org/10.23939/mmc2024.03.710

[4] E. Özcan, P. De Causmaecker, and G. Vanden Berghe, "Modelling and Solving the University Course Timetabling Problem with Hybrid Teaching Considerations," Journal of Scheduling, vol. 28, pp. 195–215, Oct. 2024. https://doi.org/10.1007/s10951-024-00817-w

[5] Steiner, Elmar, Ulrich Pferschy, and Andrea Schaerf. "Curriculum-based university course timetabling considering individual course of studies." *Central European Journal of Operations Research* 33, no. 1 (2025): 277-314. https://doi.org/10.1007/s10100-024-00923-2

[6] A. Kumar, R. Mehta, and S. Goyal, "Optimized IP Address Management Using Predictive Allocation in Virtualized Networks," IEEE Access, vol. 12, pp. 102456–102468, 2024.

[7] J. Li, Y. Chen, and X. Zhang, "Dynamic IP Resource Allocation in Edge-Cloud Systems Using Reinforcement Learning," IEEE Transactions on Network and Service Management, vol. 20, no. 2, pp. 845–857, 2023.

[8] P. Das, T. Dey, and A. Banerjee, "Formal Grammar-Based Conflict-Free Scheduling in Academic Institutions," IEEE Transactions on Emerging Topics in Computing, vol. 11, no. 4, pp. 2102–2115, 2023.

[9] H. Luo and M. Lin, "Adaptive Network Resource Allocation with Classless Inter-Domain Routing for Educational Data Centers," IEEE Transactions on Cloud Computing, vol. 12, no. 1, pp. 115–127, 2024.

[10] R. S. Singh and V. K. Sharma, "Automated Academic Timetabling System Integrating Network Resource Management," IEEE International Conference on Advanced Computing (ICoAC), Dec. 2023.