

Article

A Novel Self-Forming Virtual Sub-Nets Based Cross-Layer MAC Protocol for Multihop Tactical Network

Rukaiya Rukaiya * , Shoab Ahmed Khan, Muhammad Umar Farooq  and Farhan Hussain

Department of Computer and Software Engineering, College of Electrical and Mechanical Engineering (CEME), National University of Sciences and Technology (NUST), Islamabad 44000, Pakistan; shoabak@ceme.nust.edu.pk (S.A.K.); ufarooq@ceme.nust.edu.pk (M.U.F.); farhan.hussain@ceme.nust.edu.pk (F.H.)

* Correspondence: rukaiya@ceme.nust.edu.pk

Abstract: A tactical network mainly consists of software-defined radios (SDRs) integrated with programmable and reconfigurable features that provide the addition and customization of different waveforms for different scenarios, e.g., situational awareness, video, or voice transmission. The network, which is mission-critical, congested, and delay-sensitive, operates in infrastructure-less terrains with self-forming and self-healing capabilities. It demands reliability and the need to survive by seamlessly maintaining continuous network connectivity during mobility and link failures. SDR platforms transfer large amounts of data that must be processed with low latency transmissions. The state-of-the-art solutions lack the capability to provide high data throughput and incorporate overhead in route discovery and resource distribution that is not appropriate for resource-constrained mission-critical networks. A cross-layer design exploits existing resources to react to environment changes efficiently, enable reliability, and escalate network throughput. A solution that integrates SDR benefits and cross-layer optimization can perform all the mentioned operations efficiently. In tactical networks, SDR's maximum usable bandwidth can be utilized by exploiting radios' autonomous behavior. This paper presents a novel virtual sub-nets based cross-layer medium access control (VSCL-MAC) protocol for self-forming multihop tactical radio networks. It is a MAC-centric design with cross-layer optimization that enables dynamic routing and autonomous time-slot scheduling in a multichannel network environment among SDRs. The cross-layer coupling uses link-layer information from the hybrid of time division multiple access and frequency division multiple access (TDMA/FDMA) MAC to proactively enable distributed intelligent routing at the network layer. The virtual sub-nets based distributed algorithm exploits spectrum resources and provides call setup with persistently available k -hop route information and simultaneous collision-free transmission of voice and data. The experimental results over extensive simulations show significant performance improvements in terms of minimum control overhead, processing time in relay nodes, a substantial increase in network throughput, and lower data latency (up to 76.98%) compared to conventional time-slotted MAC protocols. The design is useful for mission-critical, time-sensitive networks and exploits multihop simultaneous communication in a distributed manner.

Keywords: cross-layer design; mission-critical networks; multihop networks; software-defined radios; tactical communication; virtual sub-nets



Citation: Rukaiya, R.; Ahmed Khan, S.; Farooq, M.U.; Hussain, F. A Novel Self-Forming Virtual Sub-Nets Based Cross-Layer MAC Protocol for Multihop Tactical Network. *Appl. Sci.* **2021**, *11*, 2470. <https://doi.org/10.3390/app11062470>

Academic Editor: Christos Bouras

Received: 12 February 2021

Accepted: 5 March 2021

Published: 10 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Tactical networks are mobile ad hoc networks (MANETs) of self-organized, self-configured software-defined radios (SDRs) to guarantee mission-critical information exchange. Their autonomous behavior (including mobility, node failures, and link failures) strains connectivity, which requires decentralized solutions to assure network survivability and communications reliability. At the same time, there is a critical demand for communication capacity with services diversity. Examples include the need for wideband communications to transfer large volumes of data, narrowband communications for extended

communication ranges, and bounded channel access latency to share mission-critical information [1]. Other important perilous issues with regards to networked communication include communication-on-the-move and interoperability. A distributed system should address these issues with self-forming, self-healing, and self-organizing capabilities. The developed system should be intelligent enough to cater to tactical communication rudiments, be technically radical, and not rely on a central controlling hub.

In tactical networks, SDRs are provisional in mission-critical areas such as battlefields and catastrophic environments to meet immediate tactical needs. These SDRs autonomously communicate and keep track of each other to enable seamless single-hop and multihop communication by configuring and reconfiguring themselves through coordination with their neighbors. SDRs ensure flexibility for every particular scenario, and the communication among radios can be optimized using a specific waveform that can switch dynamically. It is capable of operating various wireless communication protocols without changing or updating the hardware. Since the radios keep changing their positions, neighbors must know the routes or at least the next hop toward a distant destination. In many cases, there is also a need for alternate paths to compensate for failed communications [2]. Therefore, it is evident that the reliability of routes and the opportunity to make quick routing decisions are critical features.

In ad hoc networks, a message from a source node to a destination node may traverse many hops, multiplying network capacity consumption and usually in a larger network, resulting in a long chain of links. However, it is not just the hopping of data packets that contributes to capacity utilization but also the overhead associated with routing the data. For instance, many mainstream ad hoc routing protocols depend on frequent network-wide broadcasts to gain a picture of the network as it continues to change [3–5]. These protocols may not be appropriate for tactical radio networks, as network-wide broadcasts can noticeably increase response latency in emergencies.

In tactical radio networks, quality-of-service (QoS) guarantees (confined transmission latency, minimum guaranteed data rate, bounded call setup delay, etc.) are rated higher than spectral efficiency and increased bandwidth. The contention-based (carrier sense multiple access (CSMA)) techniques for emergencies result in collisions that trigger retransmissions, further magnifying call setup delays and power loss [6–8]. However, the noncontention-based approaches (such as time division multiple access (TDMA)) provide reliable communication but induce delays in the transmission where each node has its fixed slot that reduces spectral efficiency. Whereas frequency-division-multiple-access (FDMA) based approaches encounter cross-talk and signal interferences that are not adequate in military applications and should be handled precisely. However, a hybrid of TDMA/FDMA approaches can combine their strengths to overcome each other's inadequacies.

Due to the dynamic nature of MANETs, higher layer functionalities are added depending on the complexity of the application or service. Since higher layers increase overhead and latency, the cross-layer approach is used to mitigate layering's adverse effect in the mission-critical environment. It enables reliability and escalates network performance, i.e., better throughput, reduced latency, and less bit error rate through parameter controlling from a layer to another [9].

To mark improvements in a tactical radio network, we propose a novel virtual sub-nets based cross-layer medium access control (VSCL-MAC) design that provides a self-forming and self-organizing intelligent approach to radio communications. It involves neighborhood discovery, which keeps nodes informed about their k -hop neighbors and helps nodes discover routes to these neighbors. Each SDR exchanges control frames to communicate intended transmission, establish routes for data transmission, and perform virtual confinement of radios, called sub-nets or subnetworks. This virtual division of the network enables simultaneous collision-free communication using a hybrid of TDMA/FDMA approaches. The distributed intelligent approach eliminates the neighbor discovery phase's periodical runs and enables efficient route discovery and simultaneous data transmission using FDMA for non-intermittent high network throughput in a time-distributed manner.

The VSCL-MAC design supports unicasting and enables multihop communication within a uniform framework. It finds a route between source and destination using a common table, unlike other ad hoc routing protocols [10]. The design also provides dynamic slot allocation to simultaneous data transmission and assigns multiple slots to SDRs according to the data size, such as text or multimedia data. The adapted time slot allocation for radio transmission over multiple FDMA carriers ensures throughput maximization and QoS in tactical communication.

1.1. Research Contributions

In designing efficient routing protocol for multihop networks, MAC protocols have a large influence in finding optimal routes to make it efficient and self-operating [11,12]. The proposed VSCL-MAC is designed to be the first to consider tactical network as a group of subnetworks and embeds route discovery as part of the MAC layer rather than the network layer. The design utilizes major contributions to the research field and provides

1. Intelligent distributed dynamic routing and data slot scheduling for multihop simultaneous data transmissions over non-overlapping frequency sub-bands.
2. Close interaction with the MAC layer to empower efficient multihop route discovery, allowing radios to have updated neighbor information without incorporating route discovery overheads.
3. Self-forming and self-regulatory virtual subnetworks for efficient utilization of resources without having any centralized control.
4. Reliable and interference-free channel access for control and data transmissions with maximization of network throughput, minimum call setup, and access delays.

2. Related Work

Few research studies based on cross-layer schemes have been carried out to support distributed communication in multihop ad hoc and tactical networks. A cross-layer framework CL-TDMA for tactical SDRs that provides self-forming and self-healing features was proposed in [13]. The design uses TDMA as the MAC layer and ad-hoc on-demand distance vector (AODV) on network layer. control messages for multihop network routing and finds slots for collision-free data transmission. It performs better in throughput, call setup delay, and latency than traditional TDMA. A cross-layer scheme that offers decision-based routing is proposed in [14], designed for unmanned flying ad hoc networks (UFANETS). The method uses cross-layer metrics such as signal-to-noise ratio (SNR), received signal strength indication (RSSI), and AODV routing packets to establish data transmission routes. It overcomes issues of latency, unsuccessful data transmission ratio, and high usage of AODV protocol messages. A cross-layer cognitive system designed by the authors of [15] operates with distributed coordination and local information exchange among SDRs. The system integrates dynamic spectrum sensing (DSA), routing, and network coding for a multihop cognitive radio network. It also optimizes spectrum utility by joining the backpressure algorithm with network coding to support unicast and multicast traffic. A selective cross-layer-based routing scheme based on probabilistic technique combined with AODV to resolve route selection issues such as relay node selection and optimal route establishment based on energy and distance was proposed in [16]. The scheme uses a population-based stochastic optimization technique to schedule slot assignment during data transmission.

A cross-layer routing protocol for visible light ad hoc networks (LANETS) was proposed in [17]; it interacts closely with the MAC layer to maximize network throughput with the focus on the reliability of routes. The design uses route reliability scores (RSS), a utility-based three-way handshaking process computed on immediate neighbors' information to reach the sink node and mitigate the effects of link and transmission failures. A link-adaptive multihop routing protocol for tactical combat radios was presented in [18]. It adapts link state changes and selects a stable route to perform video streaming over TDMA. The design is compared with link adaptation protocols, AODV and optimized link

state routing (OLSR) with carrier-sense multiple access/collision avoidance (CSMA/CA) structures. An application-centric network architecture for tactical communication systems that can applicably respond to the dynamic network topology, link quality at the link layer, and provide a solution to the limitations of conventional single-cost function-based routing protocols at the network layer was proposed in [19]. The designed architecture offers better QoS and dynamic traffic distribution to all available paths. A robust cross-layer routing and resource allocation algorithm for orthogonal frequency-division multiplexing (OFDM) based wireless ad hoc networks was proposed in [20]. The algorithm uses limited knowledge such as maximum transmit power and total bandwidth at the transmitting node to perform joint resource allocation and routing.

Another cross-layer autonomous route discovery mechanism was proposed in [21] for vehicular networks (VANETs). The mechanism incorporates a cross-layer cluster-based routing protocol to set stable clusters up that improve routing path lifetime and decrease link breaks for multimedia data transmission. The design uses mobility metrics and the relative distance among nodes to find reliable relay nodes to the destination and improves reliability, throughput, and energy consumption depending on nodes velocity and density. A cross-layer neighbor discovery algorithm based on hexagonal clustering and GPS is proposed by [22]. The algorithm uses hello messages at the MAC layer, sent in TDMA method combined with hexagonal clustering algorithm and GPS. A random back-off mechanism is used to improve the efficiency of sending hello messages. The design effectively finds the ratio of neighbors per cycle and reduces packet collision probability and throughput in the network.

An energy-efficient and position-aware routing protocol for wireless sensor networks (WSN) is proposed by [23], which uses energy and hop-distance-based formulation for routing mechanism. The protocol selects a neighbor with maximum energy level and minimum hop distance for data forwarding, impacting network lifetime, packet delay, and throughput. An efficient route selection routing information protocol (ERS-RIP) is proposed by [24] for multihop WSNs under a jamming environment. The design uses energy cost for route selection and performs routing on optimal paths using RIP that occasionally exchanges network updates. The system achieves better results in terms of end-to-end delay, throughput, packet delivery rate, and network lifetime compared to DSR and other protocols.

The designs mentioned above rely on other layer parameters to yield routing and slot assignment information incorporated in control frames [25,26]. This increases the control overhead in the execution of route discovery, channel, and slot assignment. In contrast, our design excels in radios' autonomous behavior and performs all the mentioned operations in a distributed manner. The design fairly schedules collision-free simultaneous data transmission to all network nodes using virtual sub-nets. It employs a distributed neighbor and routes discovery algorithm for multihop routing using cross-layer design capabilities. The proposed VSCL-MAC design is unique in network organization and route establishment.

3. Proposed VSCL-MAC Architecture

The design predominantly comprises control and data units operating on TDMA frame cycles and enables multihop communication by maintaining k -hop neighbor information at every network node. Therefore, the design includes the neighborhood discovery phase at the start of the network and follows with control frames and data transmissions, as shown in Figure 1.

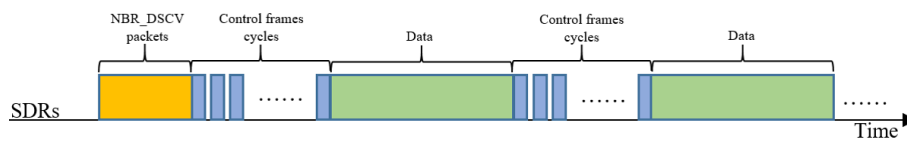


Figure 1. Time division multiple access (TDMA) frame cycles in proposed virtual sub-nets based cross-layer medium access control (VSCL-MAC) design.

Given the limitations and importance of expeditious tactical communication, each unit of the protocol operates over TDMA. The neighbor discovery unit runs only once at the start of the network; hence, it is called the “one-time neighbor discovery” process. In the first TDMA frame, the initial $2N$ slots are used to transmit neighbor discovery (NBR_DSCV) packets. Each node transmits the packets in its two-hop vicinity on fixed slots for k -hop neighbor discovery. After neighborhood discovery, control frames C_i are transmitted for the node’s coordination on a non-contention based time-slotted common control channel. These frames are used to send REQ (request), REQ_ACK (request acknowledgment), or BRDSCT (broadcast) packets and perform route establishment for multihop data transmission. In the control unit, the number of control slots is fixed; each corresponds to the node MAC ID. These control slots’ duration is specified in bits that depend on the number of radios in the network. In the data unit, DATA packets are transmitted over dynamic time slot vectors to send actual messages, i.e., text, images, or voice. The duration of data slots for a node depends on the type of data the sender wants to transmit.

The following fundamental elements are used to descend a cross-layer MAC design for tactical SDR-based multihop communication network.

3.1. One-Time Neighborhood Discovery

As soon as the network starts functioning in tactical communication, each SDR sends NBR_DSCV packet to all its one-hop neighbor(s) in its slot. The format of NBR_DSCV packet is shown in Figure 2, where transmitter ID is the MAC ID of sending node, i.e., for a network of N nodes, $\log_2(N)$ bits are required for the node’s MAC ID. The four-bit message type field identifies message format, i.e., NBR_DSCV, REQ, REQ_ACK, BRDSCT, combined_REQ or combined_ACK. The bits value 0000 indicates the NBR_DSCV packet, and NBR_MAP field gives hop count information of each neighbor, i.e., two-bit code for each node representing $2^2 = 4$ possible combinations, with 01, 10, and 11 defining 1, 2, and 3-hop, respectively. The bit code 00 represents that the node is not in a 1-hop or 2-hop neighborhood; usually, it represents itself.

Transmitter MAC ID	Message Type	NBR_MAP (bit code for hop counts)
-----------------------	-----------------	--------------------------------------

Figure 2. Format of neighbor discovery packet.

The transmission of NBR_DSCV packets begins with the lowest node ID, and in the first N slots, each SDR broadcasts the NBR_DSCV packet. Each shares its NBR_MAP with minimum hop information of all its neighbor(s), including itself. Upon receiving the NBR_DSCV packet, each node extracts the bit code (hop count) of other nodes and inserts the entry (hop count + 1) in its neighbor table (NBR_table). In NBR_table, each node N_i saves the node ID of its 1-hop neighbors in the first column of the table, each in a separate row, and keeps 2, 3, and 4-hop neighbor(s) ID in the corresponding column and in the row of 1-hop neighbor from which it gets the entry, as depicted in Table 1.

Table 1. Neighbor table of node N_i .

1-Hop NBR	2-Hop NBR	3-Hop NBR	4-Hop NBR
MAC ID of direct neighbors	MAC ID of neighbor(s), received from 1-hop NBR(s)		

The process of neighbor discovery and the built-up NBR_table are described in Figure 3 it repeats for the next N slots, and radios that cannot get information in the first receive it in this round.

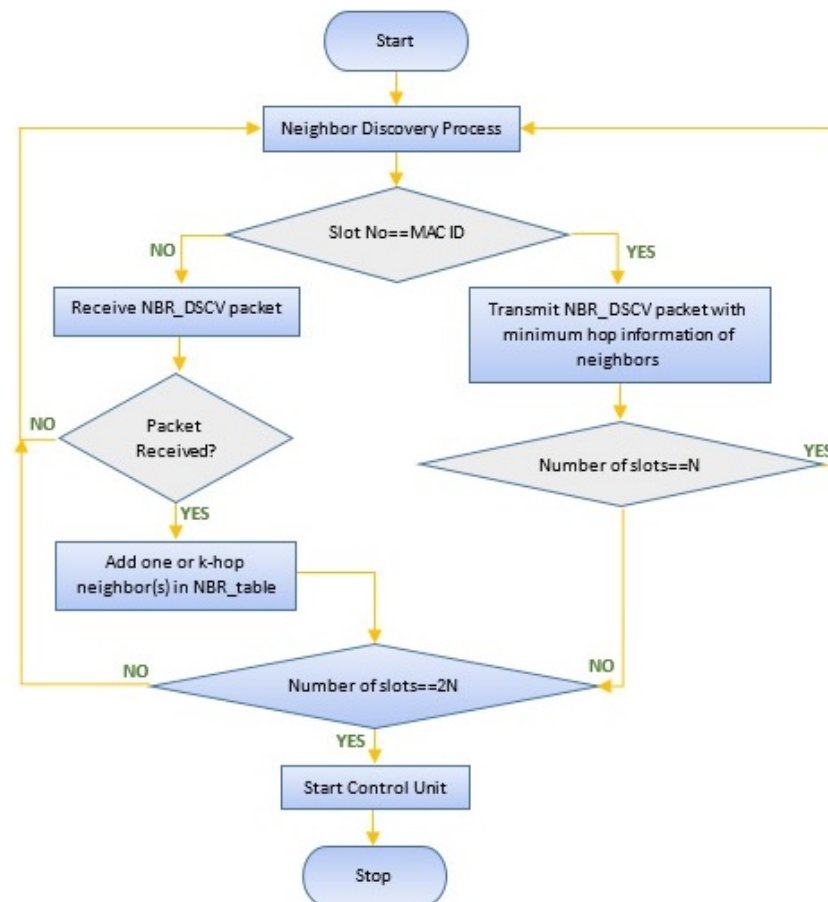


Figure 3. One-time neighbor discovery process to build the NBR_table.

The proposed algorithm can maintain more than a 3-hop neighbor’s information, depending on the number of bits used to send hop information of neighbors. Tactical networks are congested; more than 4-hop is considered to be out of bounds for radios.

3.2. Route Discovery Using NBR_Table

Many of the ad hoc routing protocols rely on advertising HELLO packets for route discovery. Most of them use physical layer parameters such as a line of sight (LOS), energy levels, directional communication links, and signal-strength-based distance measurements [27,28]. However, these approaches consume significant time to operate and use extensive bandwidth resources to find neighbors rather than send data.

The proposed VSCL-MAC design uses the NBR_table to perform efficient route discovery and makes no assumption on prior knowledge, such as the number or relative position of neighbors. A source node searches its NBR_table for the 1-hop neighbor to reach the destination node involving minimum hops. Each node has confirmed a 2-hop destination path in its NBR_table, and further, it consents the next node that shows the minimum hop route to the destination to handle it.

In the proposed design, all nodes update the NBR_table using NBR_MAP received in NBR_DSCV packets and later in control messages that keep nodes aware of its active neighbors for route discovery. It eliminates the need for any other routing protocol, considerably reduces required neighbor discovery time, and eradicates route discovery packet transmissions. The proposed VSCL-MAC design uses a hybrid routing mechanism and enables continuous route maintenance without using extra signaling packets. It decreases the excessive overhead of hello packets; adaptive link connectivity improves connectivity and coverage when the node is far away from the destination.

3.3. Control Frames Transmission Unit

It is the foremost part of VSCL-MAC protocol and encompasses major design utilities. Therefore, before sending data, SDRs go through a control unit. In this unit, SDRs exchange intended transmissions using REQ and REQ_ACK packets, send in their respective time slots, and also establish a route for data transmissions. It enables single as well as multihop data transmission for which packets are sent to the next-hop node selected from the NBR_table, further transmitted to the next node until they are received by the destination. The format of REQ and REQ_ACK packets is shown in Figure 4; they contain source, destination, and the next-hop node’s MAC ID. The NBR_MAP is the same as that of the NBR_DSCV packet, and the transmission table (TRANS_table) contains intended transmissions of the TX node as well as other nodes that help in forming sub-nets and time and frequency allocation in distributed manner.

Source MAC ID	Destination MAC ID	Next hop MAC ID	Message Type	NBR_MAP	Transmission table (TRANS_table)
------------------	-----------------------	--------------------	-----------------	---------	-------------------------------------

Figure 4. Format of request and acknowledgment control packets.

The addition of NBR_MAP in control packets benefits when a node does not receive information on any of its k -hop neighbors during the neighbor discovery process and also when the neighborhood changes due to mobility or link failures. The concerned nodes can update and maintain their NBR_table by extracting the hop information of its k -hop neighbor(s) from received control packets.

When sending the REQ packet, the source node adds the intended transmission in TRANS_table, which contains all the route paths of the transmission perceived through NBR_table. The format of TRANS_table is shown in Table 2 in which the first and second columns contain the transmitter’s and receiver’s MAC ID of the transmission, respectively.

Table 2. Transmission table of node N_i .

Transmitter	Receiver(s)
Source/Relay node	Relay/Destination node
-	-
Transmission of all 1- and k -hop neighbors	
-	-

Upon receipt of the REQ packet, all overhearing neighbor(s) add the shared transmissions in their TRANS_table. The receiving next-hop node of the REQ message can be the destination itself or a relay node. In the case of destination, the next-hop ID is the same as of destination ID, and the node responds with the REQ_ACK packet in its time slot, whereas, in another case, the next-hop relays REQ in its corresponding slot.

The proposed design can send combined REQ and REQ_ACK packets, or nodes can have multiple requests from different transmitters [29]. This reduces the number of control packets and puts a marginal effect on control overhead, call setup delays, and network

convergence. Furthermore, in SDRs communication, not all radios have messages all the time; therefore, if a node does not have any REQ or REQ_ACK message to send, it sends a broadcast (BRDCST) message in its time slot. The BRDCST packet contains NBR_MAP and TRANS_table to maintain similar neighbor and transmission information on every node when not received from REQ or REQ_ACK packets.

The TRANS_table alone helps form autonomous sub-nets, frequency selection, and time slot vectors for collision-free simultaneous data transmissions. For k -hop communication, the protocol runs multiple control frame cycles; it is essentially used to establish a route and maintain similar information on all nodes, as each frame can be heard within its two-hop vicinity.

3.4. Sub-Nets Formation for Simultaneous Data Transmission

The proposed protocol allows SDRs of the tactical network to be confined into multiple virtual sub-nets. These sub-nets are formed autonomously using entries of TRANS_table, which is similar at every radio; thus, it results in the same sub-nets structure and number. The transmission pattern of nodes shapes sub-nets' structures and has a maximum diameter equal to the network's total number of nodes. Each SDR looks its TRANS_table up and forms virtual sub-nets based on the following rules:

1. The transmitter MAC ID is equal to the receiver MAC ID of the other entry.
2. The receiver MAC ID is equal to the transmitter MAC ID of the other entry.

In the case of either rule, SDR belongs to the same sub-net and selects the data channel accordingly. After sub-nets formation, the proposed design uses the FDMA technique that assigns different frequencies to different nodes. The selection of frequency for data transmission is adhesive with the pattern of sub-nets formation. Each node selects a frequency f_h autonomously from a frequency band of range 0 to f_{h-1} and matches with the lowest ID of the sub-net members. There is no collision in frequency selection among sub-nets because each sub-net has a different lowest node ID, e.g., in Figure 5, sub-net 1 has node4, sub-net 2 has node2, and sub-net 3 has node1 as the lowest member ID of the sub-nets; therefore, use frequency $f_4, f_2,$ and f_1 for data transmission, respectively. The control packets use one frequency from the same range, which is common and time-slotted.

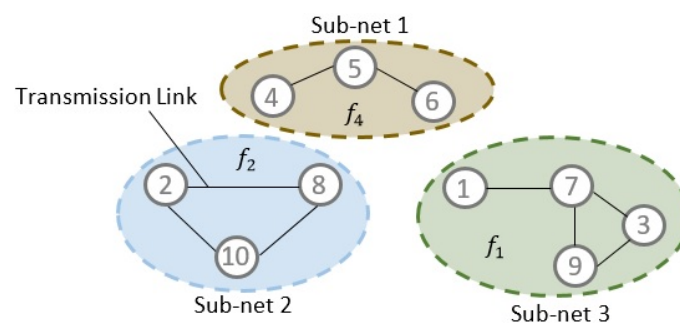


Figure 5. Sub-nets formation and frequency selection for data transmission.

This virtual confinement of nodes makes the network efficient in resource distribution in a dynamic multichannel network environment. In tactical communication, SDRs use adaptive frequency hopping techniques in which radios continuously hop over multiple frequencies; those techniques are out of this paper's scope. It provides transmission security in jamming and other transmission attacks [30,31].

3.5. Time Slots Allocation for Data Transmission

The proposed VSCL-MAC design integrates an intelligent dynamic slot allocation algorithm that allows multihop data transmission. Each SDR can allocate multiple time slots on the channel depending on the type of data SDR wants to transmit, i.e., text or multimedia messages. The algorithm runs at each SDR and calculates the time slot vector (TSV) for all frequency sub-bands selected by each of the sub-nets using the same

TRANS_table. The algorithm allocates time slots starting from the lowest node ID of the sub-net and ensures the source node of multihop transmissions gets the slot before the relay node; this is discussed in the next section. Rather than allocate multiple slots to a transmitter, there is a possibility of making the size of data slots variable for different data requirements, but this will create a time synchronization problem among radios.

3.6. Data Unit

In the data transmission unit, SDRs of each sub-net send data in time slots allocated on corresponding data channels. It enables all SDRs to execute collision-free and interference-free data transmission with efficient use of bandwidth. After completion of data transfer, all reserved time slots and frequencies are released and reserved again accordingly.

4. Working of Algorithm

We consider an ad hoc wireless tactical network comprised of N nodes, connected through wireless links of same-channel properties, and has a unique identifier denoted as ID. Each network node is connected to a set of neighbors, moves with consistent speed, and has low mobility during protocol functioning. In our network model, the NBR_table corresponds to discover 1- to 4-hop neighbors and routing paths. The algorithm can work for any number of SDRs, usually ranging from 10 to 80 radios. We assume that there are only Q nodes (half of the radios, $Q < N$) which act as traffic sources, and the TRANS_table is maintained to list all intended transmissions involved.

Now, assume a topology of 10 SDRs with IDs 1, 2, 3, 4, 5, 6, 7, 8, 9, and 10. In the first TDMA cycle, each SDR sends the NBR_DSCV packet in its time slot with the information they have in their NBR_table as depicted in Figure 6. Suppose that N_1 sends NBR_DSCV packet with no information in its NBR_table and that it made its entry in the NBR_table of SDRs $N_3, N_4, N_6,$ and N_9 as shown in Table 3 (the colors of each transmission show entries, added by matching colored arrows in Figure 6). Then, N_2 sends the packet to its neighbors $N_3, N_4, N_7, N_8,$ and N_{10} . Similarly, N_3 transmits the discovery packet to not only announce itself to its 1-hop neighbors but also shares hop information of N_1 to N_2 and N_2 to N_1 , as they are not the neighbors of each other but of N_3 . The NBR_DSCV packet of N_3 is shown in Figure 7, where the two-bit code is the hop information of each node which is at 1- or 3-hop away from N_3 i.e., N_1 and N_2 are represented at 1-hop with bit code 01. All other remaining SDRs similarly send NBR_DSCV packets. After one-time neighbor discovery, each SDR knows about its 1- to 4-hop neighbors.

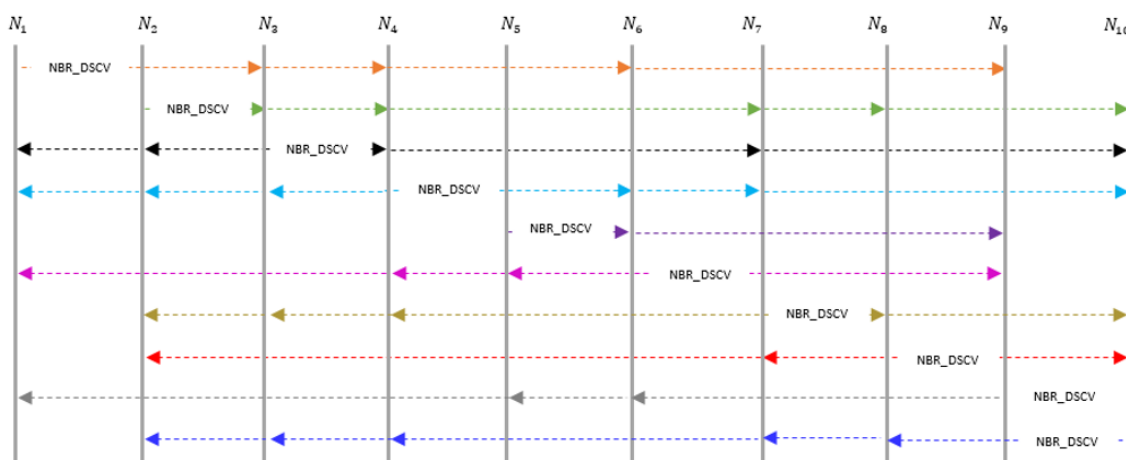


Figure 6. Flow diagram of one-time neighbor discovery process for N nodes, where $N = \{1, 2, 3, \dots, 10\}$.

TX_ID	Msg_Type	Hop Count (Bit code)
3	0000	01 01 00 - - - - -

Figure 7. Neighbor discovery packet of node N_3 .

For the control unit description, we consider five data messages sent by SDRs $N_1, N_2, N_3, N_4,$ and N_5 to SDRs $N_7, N_9, N_2, N_5,$ and $N_6,$ respectively. Each SDR has the next-hop information for the transmission perceived from its NBR_table, as shown in Figure 8. In the control unit, all SDRs send control messages to exchange intended transmissions in their respective time slots, as shown in Figure 9. Suppose N_1 has data for N_7 , sends REQ to its one-hop neighbor N_3 , and is overheard by all its neighbors. N_1 knows the path ($N_1 \rightarrow N_3 \rightarrow N_7$) to reach N_7 from its NBR_table and adds this route in its TRANS_table. Upon receipt, N_3 relays the REQ to destination N_7 . The REQ packet of N_1 , shown in Figure 10, lets its neighbors $N_3, N_4, N_6,$ and N_9 update their TRANS_table, as shown in Table 4 (Text color in Table 4 is indicating the entries received from the packet flow shown in Figure 9).

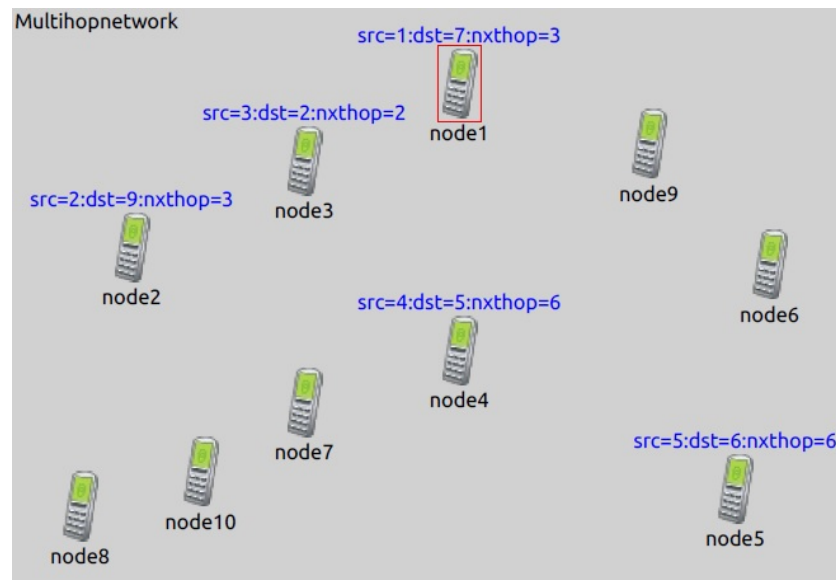


Figure 8. Network topology of ten nodes with $M = 5$.

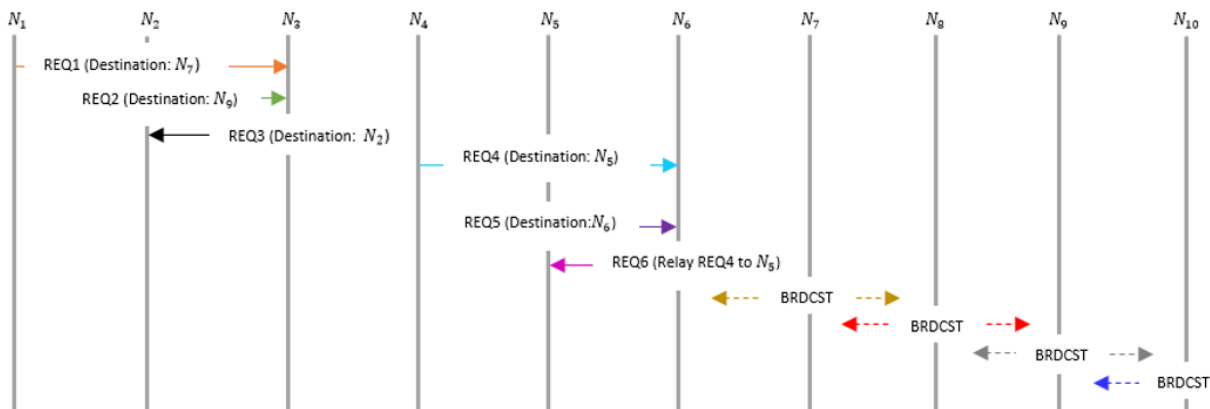


Figure 9. Flow graph of transmission of first control frame messages for N nodes, where $N = \{1, 2, 3, \dots, 10\}$.

REQ Packet 01					
Source	Destination	Next_hop	Msg_type	NBR_MAP	TRANS_table
1	7	3	0001		

Figure 10. Request packet of node N_1 .

Table 4. Transmission table of N_1 and its neighbors after first REQ message.

Transmitter		Receiver	
1		3	
3		7	

Corollary 01: REQ packets contain the information of the source, destination, and next-hop nodes. The neighbors of SDR can extract the 1-hop transmission directly from fields (source→next_hop). Hence, it is not shared in the TRANS_table field of the control message, i.e., N_1 only shares 3→7 in its REQ packet. The REQ packet receiver can add further path information itself, if available in its NBR_table, e.g., N_3 adds 1→3 from the REQ packet and 3→7 from its NBR_table. This enables SDRs to send more intended transmissions as maximum transmissions shared in a control packet is equal to M .

In the next time slot, N_2 sends the REQ message to N_3 for N_9 that is at 3-hop away. N_2 has confirmed 2-hop path information (2→3, 3→1) in its NBR_table but does not know whether N_1 is directly connected to N_9 or not. Therefore, it only adds a 2-hop path in its TRANS_table and sends the REQ message to N_3 . All the neighbors of N_2 update their TRANS_table as shown in Table 5. N_3 receives the REQ, knows the path to reach at N_9 , and adds 1→9 in its TRANS_table later shares it with its neighbors in the next control slot. The other neighbors add the same information as N_2 as they do not have the transmission route in their NBR_table.

Table 5. Transmission table of N_2 and its neighbors after the REQ message transmission of N_2 .

TRANS_Table (N_2, N_7, N_8, N_{10})		TRANS_Table (N_3)		TRANS_Table (N_4)	
Transmitter	Receiver	Transmitter	Receiver	Transmitter	Receiver
2	3	1	3	1	3
3	1	3	7	3	7
-	-	2	3	2	3
-	-	3	1	3	1
-	-	1	9	-	-

In the third time slot, N_3 has data for N_2 and, at the same time, has two more transmissions to relay in its queue (3→7 and 3→1). In this condition, N_3 sends a combined_REQ message with destinations and next-hops, as shown in Figure 11. This reduces the number of control packets; otherwise, every time, SDR consumes one slot for each REQ message.

Combined_REQ Packet 03					
Source	Destination	Next_hop	Msg_type	NBR_MAP	TRANS_table
3	7,1,2	7,9,2	0110		

Figure 11. Combined request packet of N_3 .

Corollary 02: The size of the combined_REQ packet is same as that of a REQ packet because the TRANS_table field can now hold $M/2$ transmission entries. Therefore, it can send more than one REQ message in a single packet.

Corollary 03: If an SDR does not have REQ or REQ to relay, then it can use the slot to send REQ_ACK if it has already received the request for it. In another case, SDR has a relay REQ and REQ_ACK in its queue; then, it uses the first-come, first-serve approach

and transmits other packets in the next control frame slot. In neither case, SDR sends a BRDCST packet to synchronize both tables with updated information.

Similarly, N_4 sends REQ to N_5 , which should be traversed from N_6 . N_5 sends REQ to N_6 and N_6 relays the REQ of N_4 to N_5 in their time slots, whereas N_7 has no REQ and sends REQ_ACK to N_3 , according to Corollary 03. The first control frame contains control transmissions of all SDRs, as shown in Table 6.

Table 6. First control frame contains transmissions of all software-defined radios (SDRs).

Control Frame 01					
Slot no	1	2	3	4	5
Packet	REQ for 7 via 3	REQ for 9 via 3	combined_REQ for 7,9 and 2	REQ for 5 via 6	REQ for 6
Slot no	6	7	8 to 10		
Packet	Relay REQ of 4 for 5	ACK to REQ of 3	BRDCST		

The second control frame contains REQ (direct and relayed), REQ_ACK, combined_ACK, and BRDCST packets transmission. In the sixth slot of the frame, N_6 sends combined_ACK for both REQ messages received from N_4 and N_5 previously, as shown in Table 7.

Table 7. Second control frame contains transmissions of all SDR.

Control Frame 02					
Slot no	1	2	3	4	5
Packet	Relay REQ of 2 for 9	ACK to REQ of 3	ACK to REQ of 1	BRDCST	ACK to REQ of 6
Slot no	6	7	8	9	10
Packet	combined_ACK to 4 and 5	BRDCST		ACK to REQ of 1	BRDCST

The third and last control frame of discussed topology contains REQ_ACK of N_1 and N_3 . In contrast, others transmit a BRDCST message to share those transmissions which were not sent previously, necessary to synchronize TRANS_table of all nodes, as shown in Table 8.

Table 8. Third control frame contains transmissions of all SDRs.

Control Frame 03				
Slot no	1	2	3	4 to 10
Packet	ACK to REQ of 3	BRDCST	ACK to REQ of 2	BRDCST

After the transmission of all control frames, each SDR has all the intended transmissions in its TRANS_table, as shown in Table 9 (due to the size of TRANS_table, transmission with same receivers are combined and written in ascending order).

On completion of the control frames transmission, all SDRs built the same TRANS_table using Algorithm 1, which runs on each network node and forms virtual sub-nets based on the transmission information in the table. For sub-nets formation, Algorithm 1 (line no. 17 to 23) matches each transmission with every other transmission of the table. If a TX of any transmission matches with the TX or RX of any other transmission, then the TXs and RXs of both transmissions belong to the same sub-net. If no match is found, then the TX and RX of transmission i are the only members of the sub-net S_v . For the above topology, two sub-nets are formed, $S_1 = \{1, 2, 3, 7, 9\}$ and $S_2 = \{4, 5, 6\}$, as shown in Figure 12, which also shows the roles of each node played for the transmissions.

Table 9. Transmission table of all SDRs after completion of the control phase.

TRANS_Table (N ₁)		TRANS_Table (N ₂)		TRANS_Table (N ₃)	
Transmitter	Receiver(s)	Transmitter	Receiver(s)	Transmitter	Receiver(s)
1	3,9	1	3,9	1	3,9
2	3	2	3	2	3
3	1,2,7	3	1,2,7	3	1,2,7
4	6	4	6	4	6
5	6	5	6	5	6
6	5	6	5	6	5

TRANS_table(N ₄)		TRANS_table(N ₅)		TRANS_table(N ₆)	
Transmitter	Receiver(s)	Transmitter	Receiver(s)	Transmitter	Receiver(s)
1	3,9	1	3,9	1	3,9
2	3	2	3	2	3
3	1,2,7	3	1,2,7	3	1,2,7
4	6	4	6	4	6
5	6	5	6	5	6
6	5	6	5	6	5

TRANS_table(N ₇)		TRANS_table(N ₈)		TRANS_table(N ₉)	
Transmitter	Receiver(s)	Transmitter	Receiver(s)	Transmitter	Receiver(s)
1	3,9	1	3,9	1	3,9
2	3	2	3	2	3
3	1,2,7	3	1,2,7	3	1,2,7
4	6	4	6	4	6
5	6	5	6	5	6
6	5	6	5	6	5

TRANS_table(N ₁₀)	
Transmitter	Receiver(s)
1	3,9
2	3
3	1,2,7
4	6
5	6
6	5

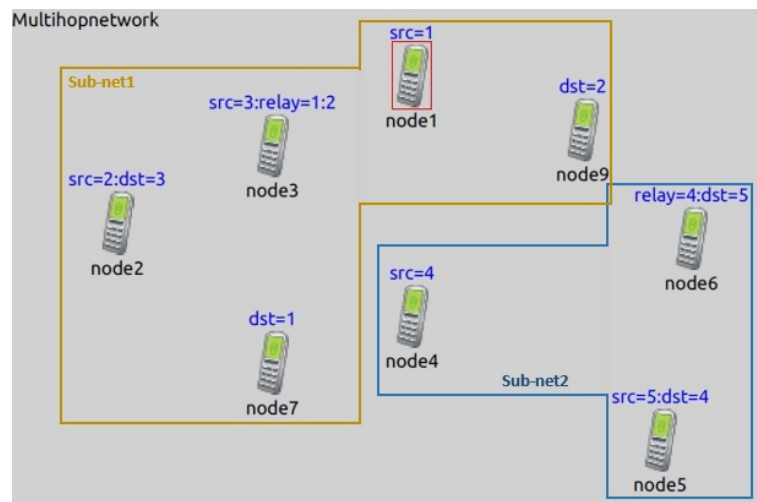


Figure 12. Sub-nets formation after control frames transmission.

Algorithm 1: Route establishment and sub-nets formation in VSCL-MAC protocol.

Notation:

1. The finite set of symbols δ_s denoted by $\Sigma^* = \{N, \Gamma(N), S\}$, where N = number of nodes, $\Gamma(N)$ = set of one-hop neighbors, S = number of virtual sub-nets.
2. N_i : elements of N with $1 \leq i \leq N$, M = number of data messages.
3. $ctrl_slots$ = control time slots, U = number of transmissions.

Input: N nodes topology, NBR_table , $ctrl_slots$.

Output: $TRANS_table$ of N_i

Initialization: $TRANS_table \leftarrow \phi$, $Slot_no \leftarrow 1$

```

1 while ( $ctrl\_slots == true$ ) do
2   if ( $slot\_no == MAC\_ID \ \&\& \ N_i == source\_node$ ) then
3     Add transmission in  $TRANS\_table$  of  $N_i$ 
4      $N_i$  sends REQ packet with  $M$  unshared transmissions of its  $TRANS\_table$ 
5   end
6   if ( $slot\_no == MAC\_ID \ \&\& \ N_j == TXnode$ ) then
7     // It is a relay node
8     if ( $REQ$  of the transmission has already received) then
9        $N_j$  sends REQ packet to  $N_k$  with  $M$  unshared transmissions of
10       $TRANS\_table$ 
11    else
12       $N_j$  sends BRDCST packet with  $M$  unshared transmissions of
13       $TRANS\_table$ 
14    else if ( $N_j == destination\ node \ || \ RX\ node$ ) then
15      Receives REQ packet of  $N_i$  and sends REQ_ACK to  $N_i$ 
16      Insert TX and RX in  $TRANS\_table$  of  $N_j$ 
17      Update  $NBR\_table$  of  $N_j$ 
18    end
19  end
20 end
21 end
22 // Virtual sub-nets formation
23 Input:  $TRANS\_table$  of nodes.
24 Output: Virtual sub-nets  $S$ .
25 while ( $i == U$ ) do
26   //  $i$  and  $j$  refer transmissions of the table
27   Add TX( $i$ ) and RX( $i$ ) in sub-net  $S_v$ 
28   Search  $TRANS\_table$  for TX( $i$ )
29   if ( $TX(i) == TX(j)$ ) then
30     Add RX( $j$ ) in sub-net  $S_v$ 
31     if ( $TX(i) == RX(j)$ ) then
32       Add TX( $j$ ) in sub-net  $S_v$ 
33     end
34   end
35 end
36 end

```

These virtual sub-nets perform an autonomous selection of the frequency channel. Each member of the sub-net uses the same frequency channel to communicate with each other. This virtual confinement of radios does not make any physical change in network topology; rather, it separates the communication paths between radios. For instance, the sub-nets in Figure 13 represent a valued graph in which SDRs connected with links labeled by the same frequencies are members of the same sub-net, i.e., SDRs N_1, N_2, N_3, N_7 , and N_9 use frequency channel f_1 , whereas N_4, N_5 , and N_6 use f_4 for data transmission in a data unit.

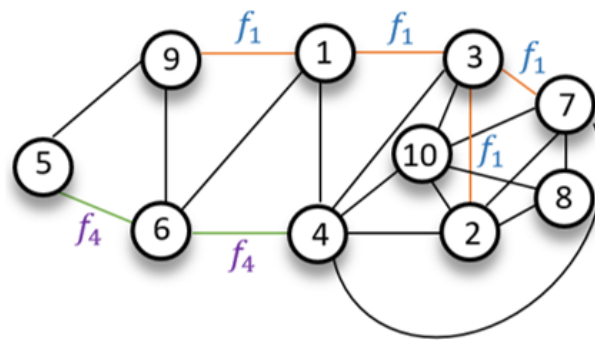


Figure 13. Virtual confinement of radios using different frequency channels.

Algorithm 1 describes the working of the control unit for route establishment, maintaining TRANS_table and sub-nets formation that privileges simultaneous collision-free data transmission.

After channel selection, all SDRs calculate time slot allocation vectors (TSVs) for each frequency channel. Algorithm 2 steps the time slot scheduling for TX(s) and relay SDRs down; they are identified in the TRANS_table. It first identifies relay SDRs and the TX(s) for their number of occurrences as the transmitter. The allocation starts from the lowest node ID to the highest. If an SDR is the only transmitter and not a relay SDR, then it gets the current slot in the vector, i.e., the sub-net2 members are N_4, N_5 , and N_6 and the allocation starts with N_4 , which gets the first slot because it has data for N_5 and needs to be relayed by N_6 , as shown in Figure 14.



Figure 14. Time slot vectors for sub-nets.

Algorithm 2: Data slot scheduling.

Notation:

1. t_r = time slot, n_i = TX node of sub-net S_v .
2. i, j , count are finite non-negative integers.

Input: Transmissions belong to sub-net S_v .

Output: Time slot vectors (TSVs).

Initialization: int $i, j, m \leftarrow 1, count \leftarrow 0$

```

1 for ( $m \leftarrow 1 \dots n$ ) do
2   Find TX and Relay nodes in  $U$  transmissions
3   while ( $j == U$ ) do
4     if ( $n_i == RX$  of transmission  $j$ ) then
5        $n_i$  is the Relay node
6       count  $\leftarrow$  count+1 // count occurrence of  $n_i$ 
7       // Schedule transmission of all TX(s) of  $n_i$ 
8       Find transmission in which  $n_i$  is the RX
9       Assign  $t_{r-1}$  to TX of that transmission
10    end
11    else if ( $n_i != RX$  of transmission  $j$ ) then
12     Assign  $t_r$  to  $n_i$ 
13    end
14  end
15  Repeat steps 4 to 11 for all TX members of sub-net  $S_v$ 
16 end
  
```

The allocation policy for relay SDR is different. The algorithm ensures that if the RX of transmission i and the TX of the transmission j are the same, then the TX of transmission i should get the prior slot of the relay SDR, i.e., in sub-net1, N_2 has data for N_3 that should be sent via N_3 . However, there is also a transmission in which N_3 has data for N_2 ; then, the algorithm schedules the transmission $3 \rightarrow 2$ before the transmission $2 \rightarrow 3$. These transmissions can be scheduled interchangeably, as both transmissions have no relation and N_3 has its data to transmit to N_2 , whereas N_2 first needs to send it to N_3 and then N_3 sends it to N_1 . The resultant TSV1 has the third slot scheduled for transmission $3 \rightarrow 1$, which carries data of N_2 .

The data slot allocation on each frequency channel is illustrated in Figure 15. The transmission can hold multiple consecutive slots for data transmission, as discussed in the next section. Within allocated time slots of a slot vector, SDRs perform simultaneous data transmissions without disturbing each other's transmissions.

Time slot Frequency	t_1	t_2	t_3	t_4	t_5	t_6	t_7
f_1	3→2	2→3	3→1	1→3	1→9	3→7	-
f_4	4→6	6→5	5→6	-	-	-	-

Figure 15. Time slot and frequency distribution of sub-net's transmissions.

The designed methodology makes every node aware of all the transmissions in the network. There are cases in which many nodes are not involved in any transmission; they may run the control phase at frequency f_0 parallel to the data transmissions.

The proposed VSCL-MAC cross-layer design embraces distributed intelligent algorithms, where nodes can discover communication routes without any upper-layer protocol support and perform simultaneous multihop data transmissions in a multichannel environment. In tactical communication, radios have a limited amount of data; therefore, they consume fewer time slots in data transmission.

5. Finite State Machine for Control and Data Transmissions

We use a finite state machine (FSM) to implement and analyze the working of VSCL-MAC design. The proposed cross-layer MAC design describes the same set of states, events, conditions, and actions required to operate FSM. The control and data packet transmissions, transition details, and required actions are shown in Figure 16.

The FSM defines the interaction between all conceivable states, events, and actions for the transmit and receive path. The states of FSM have the following denotations.

- IDLE: there is no control or data message to transmit.
- LISTEN: neighbor SDR(s) overhear control messages.
- WAIT REQ_ACK: TX SDR is waiting for the ACK of the REQ message it sent.
- WAIT STATE: SDR is waiting for the control period to be over and starts data transmission upon timeout.
- TRANSMIT DATA: the control period is over, and SDR sends data on the selected frequency and scheduled data slots.

The events that take place in FSM have the following denotations:

- DATA_available: SDR has data to transmit, available in its queue.
- REQ_received: SDR received a REQ message.
- REQ_received (to relay): SDR received a REQ message that needs to be relayed.
- REQ_ACK_recieved: the ACK of the sent REQ is received at SDR.
- REQ_ACK_timeout: no ACK is received till the timeout.
- Overheard_REQ, REQ_ACK, BRDSC of neighbor(s): the neighbor(s) of SDR overheard control packets transmission.
- CTRL_frames_completed: the transmission of all control frames is completed.
- DATA_transmitted: SDR has transmitted the data message.

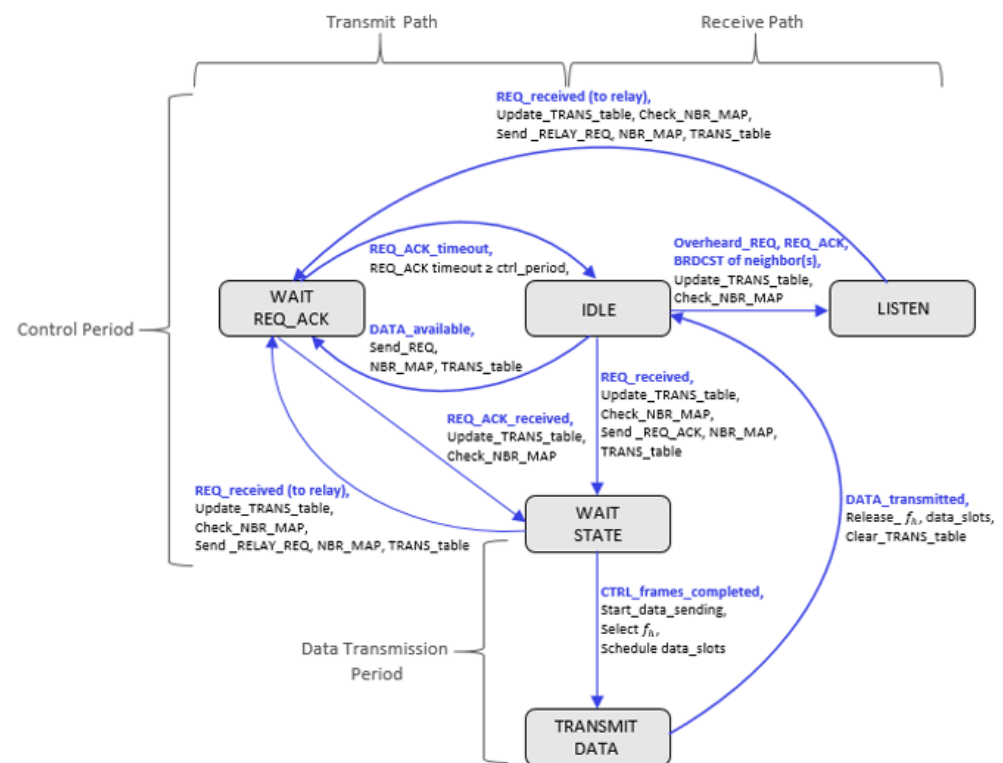


Figure 16. Finite state machine for control and data transmissions.

The FSM remains in an idle state until an event is raised to start the state transition process. Figure 16 illustrates that when an event *DATA_available* is set, *Send_REQ* action is performed that combines with the actions of sending *NBR_MAP* and *TRANS_table*. These actions are taken when the FSM goes from the *IDLE* state to *WAIT REQ_ACK* state. Next, FSM examines for either *REQ_ACK_received* or *REQ_ACK_timeout* event, and the transition depends on which event occurred. If an event *REQ_ACK_timeout* is perceived, then the FSM goes to the *IDLE* state, where it waits for data to be available and again sends the *REQ* message. After the *REQ_ACK_received* event, the actions *Update_TRANS_table* and *Check_NBR_MAP* are executed, and FSM goes into the *WAIT STATE*. If an event *CTRL_frames_completed* that identifies the completion of control frames transmission has taken place, then *Start_data_sending* action is enabled and it triggers two more events *Select f_h* and *Schedule_data_slots* to go into the *TRANSMIT DATA* state. When FSM executes the event *DATA_transmitted*, it transits to the *IDLE* state by performing actions *Release f_h* , *data_slots* and *Clear_TRANS_table*. It also deals with the event *Overheard_REQ/REQ_ACK* or *BRDCST* for those SDRs that are neighbors of the TX SDR and transit from *IDLE* to *LISTEN* state. At this state, the actions of *Update_TRANS_table* and *Check_NBR_MAP* are performed. When an event of *REQ_received* occurs while being in the *IDLE* state, then the action of *Send_REQ_ACK* is executed along with *Update_TRANS_table* and *Check_NBR_MAP* actions. At this event, the FSM goes from an *IDLE* state to *WAIT STATE*. The rest of the state transition diagram can be inferred similarly.

It is noted that sub-nets formation, frequency selection, and data slot scheduling do not depend on the FSM model and run on the information available in *TRANS_table*, built during the control frames transmissions.

6. Experiments, Results, and Performance Evaluation

The proposed VSCL-MAC design is evaluated via extensive simulations on a different set of topologies in OMNET++5.4.1 with INET framework 4.6.0. The protocol is simulated for single and multihop communication scenarios. The number of frequencies f_h is assumed to be more than the number of nodes, or almost twice the number of nodes. A random

mobility pattern is considered for nodes, and during protocol execution, speed of nodes is slow. The parameters and their notation are listed in Table 10.

Table 10. Parameters and their notation.

Notation	Parameters
R_t	Network data rate
D_slots	No. of data slots
L	No. of control frames
f_h	No. of frequencies
k	No. of hops
M	No. of messages
S_c	Size of control slot
S_d	Size of data slot
T	TDMA frame duration

6.1. Simulation Results and Analysis

We derived the proposed VSCL-MAC design performance in terms of call setup delay, control overhead, network throughput, and latency.

6.1.1. Call Setup Delay

For call setup delay, we consider the last bit of REQ message sent from the source node until it receives the REQ_ACK message for the REQ. We evaluated call setup delay for 5 to 50 nodes that have data to transmit on different hops. In VSCL-MAC design, the average call setup delay is low as compared to conventional TDMA-based MAC because participating nodes can send combined REQ and REQ_ACK packets, which significantly decreases the call setup time and total number of control frames transmission, as shown in Figure 17, whereas in conventional TDMA-based design, each REQ or ACK packet consumes an individual time slot to be transmitted and increases call set up time linearly with the increase in number of nodes N .

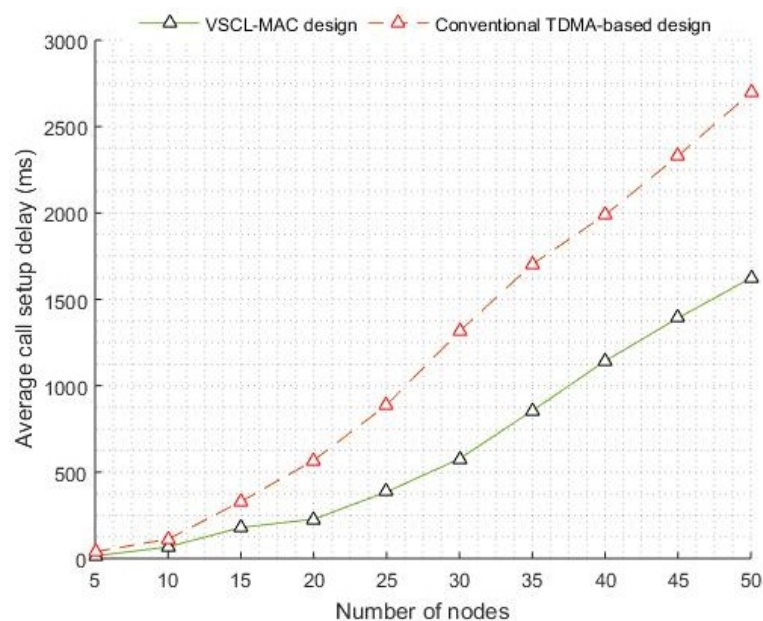


Figure 17. Average call setup delay for VSCL-MAC and TDMA-based MAC design.

In some cases, the call setup delay in the proposed design abbreviates more when the sender of REQ or ACK is the same for most of the transmissions and has the packet available in its queue concurrently. This has a significant impact on average call setup time and also reduces the number of control frames L . In these scenarios, traditional TDMA-based

design consumes many control frames compared to VSCL-MAC, as shown in Figure 18. This substantial increase in L is not favorable for time-critical tactical applications.

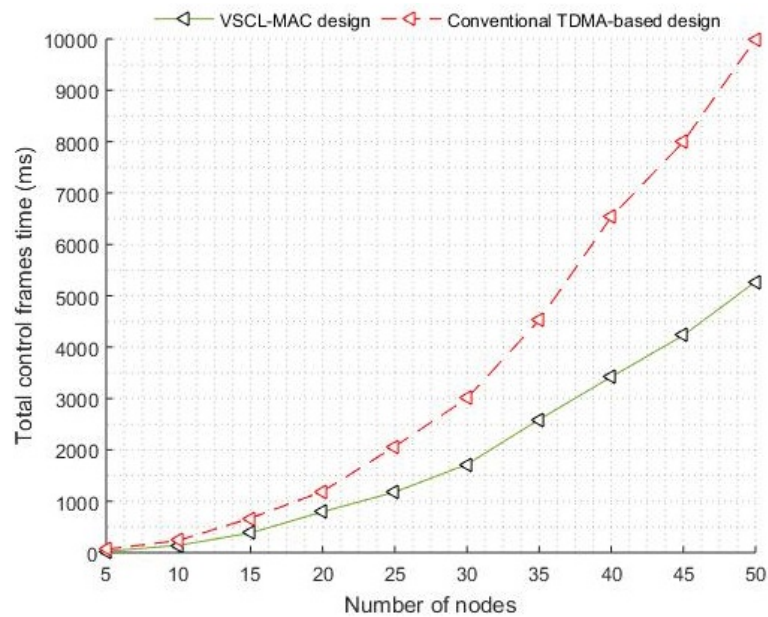


Figure 18. Total control frames transmission time for VSCL-MAC and TDMA-based design.

We evaluated the call setup time of network topologies when all nodes transmit on the same hop and analyzed the average call setup delay of VSCL-MAC, as shown in Figure 19. For 1-hop transmissions, a total of 2.5 or 3 control frames are consumed. The design is simulated for maximum 4-hop transmission, which merely has 2200 ms on average of call setup delay with a network of 50 nodes and consumes only five control frames.

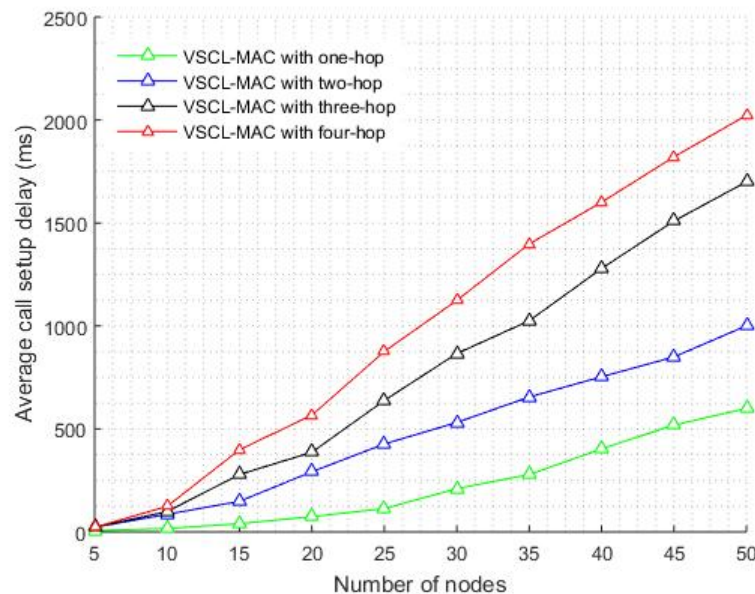


Figure 19. Average call setup time with respect to number of hops in VSCL-MAC design.

We observed that with k -hop transmission, the VSCL-MAC protocol consumes k control frames to set up a call specially when the source node uses its preceding nodes to relay the packet and has a greater MAC ID than the source and relay node(s). This limits slots wastage and reduces call setup delay. Usually, in the last control frame, most source nodes receive ACKs of the REQ messages and the design consumes $k + 1$ rounds of control

frames. Even in the worst-case scenario with 4-hop transmissions, L does not exceed more than six control frames and does not decrease bandwidth utilization for data transmission to the lowest range.

Therefore, it can be analyzed that the total number of control frames L depends on maximum hop transmissions, the complexity of the network, and the set of control parameters.

6.1.2. Control Overhead

In VSCL-MAC, the control overhead is much less than conventional TDMA-based MAC design because control packets contain only transmission information, and nodes take an autonomous decision on frequency selection and time slots scheduling by using TRANS_table. When L is small, the control overhead is low. For example, in a tactical network of 30 radios, the control overhead adds up to 38.57% in VSCL-MAC and 83.08% in TDMA-based design. This control overhead, including call setup time, is low when the network is small, and most of the radios are intending to communicate on lower hops.

6.1.3. Network Throughput of VSCL-MAC Protocol

The network throughput $TH_{network}$ of the proposed design is calculated using Equation (1), which describes the number of bits b_m of message m transmitted by a node n_i on data channel h of data rate R_t kbps. The data is transmitted in a data slot d of TDMA frame of duration T .

$$TH_{network} = \sum_{h=1}^F \left(\frac{\sum_{i=1}^N (n_{ih} \times b_m / R_t)}{T_d} \right) \quad (1)$$

For network throughput analysis, we consider the channels of capacity 200 kbps to 1000 kbps based on the M-ary modulation scheme selected by the authors of [32]. Different data sizes are considered according to the requirement of application with the adjustment of conflicting design parameters such as data rate, bandwidth, delay concerning the number of hops k , and radio transmit power by the model presented in [32]. The size of data slot S_d is double the size of control slot S_c , e.g., for 30 nodes, S_d is 28.6 ms. We assume three different messages according to the data type and its size, specified in Table 11 and the number of data slots is equal to number of messages M .

Table 11. Particulars of data messages with reference to slot and data size for $N = 30$.

Message Type	Bit Identification	No. of Slots Allocated	Data Slot Time (ms)	Data Size
text	00	1	28.6	10 Kb
Image/visuals	01	2	57.2	100 Kb
voice/video	10	3	85.8	1 Mb

6.1.4. Network Throughput Concerning R_t

In consideration of above settings, the slot size S_d of 28.6 ms is not enough to send 1250 bytes (10 Kb) of data over R_t of 200 kbps and 300 kbps. However, with R_t of 400 kbps to 1000 kbps, each TX node of sub-nets can transmit all 10 Kb of text message, keeping the data slot size S_d unchanged, as shown in Figure 20. For visuals and voice messages, $TH_{network}$ increases with R_t , as more data bits can be transmitted. It is also due to the allocation of two and three data slots to each TX for respective message types and has a diverse impact on channel utilization and network throughput.

Nevertheless, the increase in R_t only favors the prosperous transmission of all bits under a slot; it does not ensure utilization of all data slots. Henceforth, the channel capacity goes subverting and critically affects $TH_{network}$. Under low R_t such as 200 kbps and 300 kbps, 57.2% and 86.67% bits are transmitted from 10 Kb, respectively. For visual messages, nodes can send 11.4% to 57.2% bits of the message over R_t of 200 kbps to 1000 kbps, and for voice messages, the bit transmission percentage goes from 1.72% to 8.58% for respective data rates.

6.1.5. Impact of Increase in Slot Size and Number of Slots on $TH_{network}$

The proposed VSCL-MAC design improves results for large data sizes if we increase the data slot size, i.e., $S_d = 57.2$ ms for text and other types accordingly. The nodes can send more bits in a single slot and have a marginal impact on $TH_{network}$. However, this will not increase channel utilization if $N < D_slots$ in the TDMA frame. When we increase S_d , the overall size of the TDMA frame T likewise increases.

We observed that with large data sizes, M number of slots are not enough, as each node is assigned multiple data slots. Therefore, to analyze the impact on $TH_{network}$, we increase the number of data slots and set it to N , i.e., $D_slots = N$, where $N = 30$. With an increase in D_slots , the TDMA frame's size increases to 858 ms, and $TH_{network}$ decreases to almost half for small data sizes, as depicted in Figure 21. This happens because, for a small data size, most slots are wasted, whereas with large data sizes, more nodes get the slots to transmit data but merely impact $TH_{network}$ as the number of transmitting nodes on each frequency is not equal, and channel capacity goes undermining on few frequency channels.

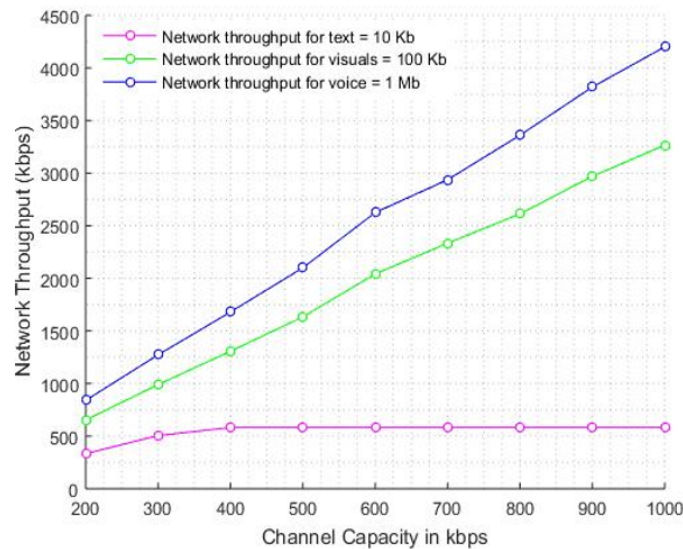


Figure 20. Network throughput of proposed design with different data types for $N = 30$.

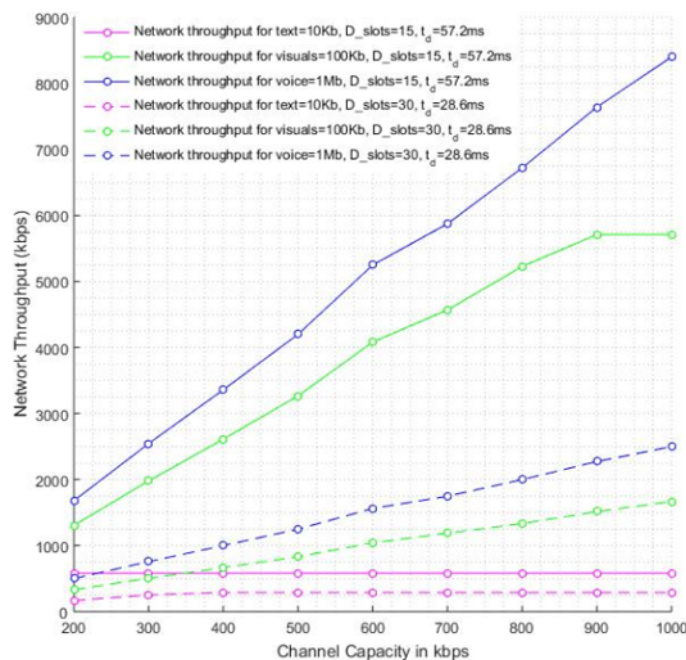


Figure 21. Network throughput with different number of sizes of data slots for $N = 30$.

To deal with the issue, we can repeat calculated time slot vectors TSVs of each sub-net on the selected frequency channel. It will increase $TH_{network}$ and channel utilization for small values of R_t .

7. Comparison of VSCL-MAC with Other Cross-Layer Protocols

In this section, we present the qualitative and quantitative comparison of proposed VSCL-MAC protocol with other cross-layer and MAC protocols.

7.1. Qualitative Comparison

All previously proposed cross-layer designs discussed in Section 2 are missing the realistic behavior of the tactical environment and specify our proposed design's novelty. The investigation shows that none of the techniques provide complete autonomous and distributed design, which works in multichannel tactical network settings. Consequently, it is impossible to mark fair quantitative comparison with other cross-layer designs, as all utilize PHY or network layer support to enable collision-free multihop communication. However, we performed a qualitative comparison with deference to the expedient of undertaking ad hoc tactical networks in terms of control overhead for route discovery and set collision-free simultaneous data transmissions up, described in Table 12.

Table 12. Qualitative comparison of VSCL-MAC protocol with other cross-layer protocols.

Application Category	Protocols and References	Comments
A cross-layer framework for SDR based multihop tactical network	CL-TDMA [13]	A TDMA-based cross-layer MAC for multihop tactical networks exploits empty slots and provides route calculation using AODV control messages, whereas our proposed VSCL-MAC eliminates the use of control packets to perform route discovery and uses the same NBR_table. All the requests for intended transmissions are forwarded directly to the selected next-hop neighbor instead of sending the broadcast message to all neighbors. It also schedules dynamic slot allocation for collision-free simultaneous data transmission, which increases the network throughput.
Cross-layer design for routing and route discovery in VANETs	ACO-AODV [16]	The protocol is designed for vehicular networks and uses stochastic approaches for global coordination and slot allocation at MAC layer, such as ant colony and particle swarm optimizations, which take uncertain convergence time. The probability distribution can change for each iteration of the algorithm. These techniques are generally not suitable for real-time, mission-critical networks. Our proposed VSCL-MAC design enables dynamic slot allocation and frequency selection autonomously without using any global coordination among nodes of the network. It provides guaranteed communication with performance gain in throughput.
Cluster-based routing protocols for ad hoc networks	CCBR [21], ND_HC [22]	The designed protocols use clustering concepts to find reliable relay nodes as cluster heads and optimal paths. constant bit rate (CCBR) uses average relative velocity to elect cluster head. The cross-layer neighbor discovery ND_HC algorithm exchanges hello messages using hexagonal clustering and GPS over TDMA. These approaches inhibit the autonomous behavior of network and induce control overhead not only in routing but also to select cluster heads. The VSCL-MAC involves no central points and exchanges minimum control messages to enable multihop routing.
Energy-efficient and position-aware routing mechanisms for WSNs	[23], ERS-RIP [24]	The designed protocols are suitable for event-based applications and use energy levels and hop distance for data forwarding. ERS-RIP uses routing information protocol to measure optimal paths between source and destination, whereas VSCL-MAC is fully decentralized and does not incorporate any routing protocol for route discovery, which reduces control overhead induced by routing protocols. The continuous and updated neighbor information keeps the nodes updated for multihop routes without sending any extra control messages.

7.2. Quantitative Comparison with Conventional TDMA-Based MAC Design

The quantitative comparison for control and data latency is performed between proposed VSCL-MAC and conventional TDMA design for the different number of messages M . We considered multiple hops to evaluate the latency for both packet types, i.e., control and data. The proposed scheme has low latency values for every hop network compared with TDMA, as shown in Figure 22. In the proposed design, with an increase in the number of hops k , the number of control frames L also increases with the count of one, e.g., $L + 1$, whereas in conventional TDMA design, L is large, as each transmission consumes an individual slot for every transmission. The increase in latency lowers the maximum throughput for the channel and makes it intelligible that the network performance lessens with an increase in the number of nodes.

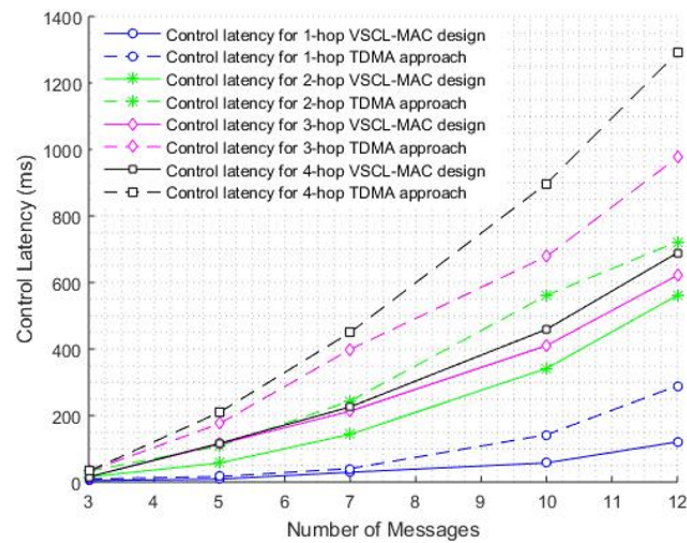


Figure 22. Control latency comparison between proposed design and conventional TDMA.

In the case of data latency, TDMA uses more control frame cycles and takes a relatively great amount of time to start data transmission, as shown in Figure 23. This transpires because each SDR will have to wait for its data slots for its next data transmission, whereas in VSCL-MAC, the simultaneous data transmission in the same time slot makes the latency lower, and dynamic allocation of slots increases $TH_{network}$ due to sub-nets formation.

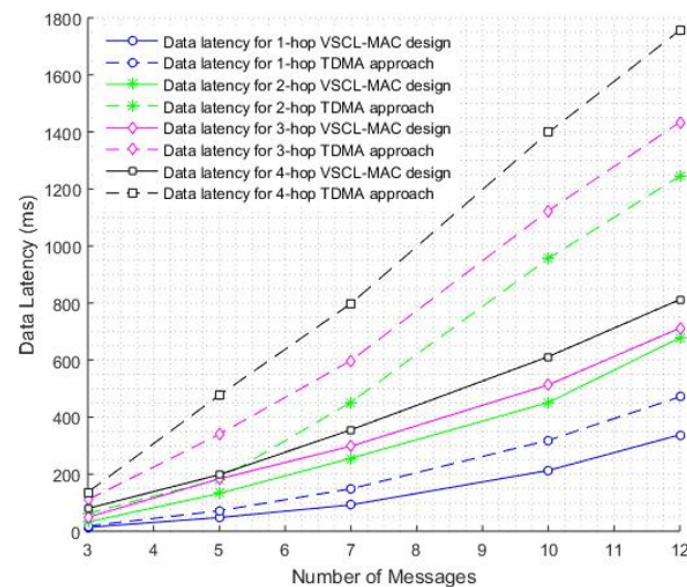


Figure 23. Data latency comparison between proposed design and conventional TDMA.

8. Observations and VSCL-MAC Design Constraints

Several important observations were made over an analysis of performance measures.

1. The protocol is valid for both single and multihop communications. Moreover, each radio can be engaged in only one call at a time.
2. Increase in R_t and combined REQ/REQ_ACK packets significantly reduces call setup time. An increase in S_d has a significant impact on $TH_{network}$.
3. Latency increases if different relay nodes are selected for control transmissions and protocol starts functioning as conventional TDMA.
4. The control packet size increments with the expansion in N and subsequently impacts call setup delay.

The proposed design can deal with conventional MAC issues, i.e., transmission on different frequency sub-bands mitigates collisions and interference among radios. It does not involve retransmissions as radios transmit control messages in their slot, whereas virtual confinement of SDRs avoids collisions in data messages. The distant radios may go for retransmissions that deal with the respective TCP and ARQ mechanism provided by the network layer protocol.

The VSCL-MAC protocol favors concurrent transmissions over multiple channels without disturbing each other's transmissions. It gives numerous advantages with less control signaling and can schedule bidirectional correspondence among peer nodes. Utilizing the method, a network can uphold diverse transmission strategies, i.e., broadcast and multicast, by just referring to sub-nets address.

9. Conclusions

In accordance with the multihop tactical communication, we proposed a novel self-forming virtual sub-nets based MAC-centric cross-layer design. Given the impact of network connectivity in MANETs, it is a difficult and essential task to manage time-sensitive communication with the dynamic behavior of the network. Therefore, we presented a distributed intelligent design that allows parallelization of computation and leads to a significant reduction in routing overhead. The proposed VSCL-MAC provides an efficient neighbor discovery process (embedded in control frames) that keeps the nodes updated about their neighbors. The design does not employ any route discovery mechanism and purges system-wide broadcast, reducing call setup time. The idea of virtual sub-nets formation enables simultaneous collision-free multihop data transmission. The self-forming and self-operating capabilities make the design stand out in terms of network throughput and lower the data latency up-to 76.98% compared to conventional TDMA. Furthermore, the qualitative comparison of the proposed design with other state-of-the-art cross-layer designs proves its validity for tactical applications.

Author Contributions: Conceptualization, S.A.K. and R.R.; methodology, R.R. and S.A.K.; software, R.R. and M.U.F.; validation, R.R., S.A.K., and M.U.F.; formal analysis, R.R.; investigation, R.R.; resources, F.H.; writing—original draft preparation, R.R.; writing—review and editing, R.R., M.U.F., and F.H.; visualization, R.R.; supervision, S.A.K. and R.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the data supporting the results is provided in the article. The data scenarios not mentioned in article can be provided whenever needed.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cheung, L.Y.; Yin, C.W. Designing Tactical Networks-Perspectives from a Practitioner, dSTA HORIZONS. Technical Reports. 2013, 122–133. Available online: <https://www.dsta.gov.sg/docs/default-source/dsta-about/designing-tactical-networks-perspectives-from-a-practitioner.pdf?sfvrsn=2> (accessed on 10 March 2021).
2. Younes, O.; Thomas, N. Analysis of the expected number of hops in mobile ad hoc networks with random waypoint mobility. *Electron. Notes Theor. Sci.* **2011**, *275*, 143–158. [[CrossRef](#)]
3. Nazhad, S.H.H.; Shojafar, M.; Shamshirband, S.; Conti, M. An efficient routing protocol for the qos support of large-scale manets. *Int. J. Commun. Syst.* **2018**, *31*. [[CrossRef](#)]
4. Jahir, Y.; Atiquzzaman, M.; Refai, H.; Paranjothi, A.; LoPresti, P.G. Routing protocols and architecture for disaster area network: A survey. *Ad Hoc Netw.* **2019**, *82*, 1–14. [[CrossRef](#)]
5. Gawas, M.A.; Gudino, L.J.; Anupama, K. Congestion-adaptive and delay-sensitive multirate routing protocol in manets: A cross-layer approach, *Int. J. Commun. Syst.* **2019**. [[CrossRef](#)]
6. Moraes, R.; Vasques, F.; Portugal, P. Survey of real-time communication in csma-based networks. *Netw. Protoc. Algorithms* **2010**, *2*, 158–183. [[CrossRef](#)]
7. Stukova, I.; Crawford, B.A. *Short-Term Self-Moving Tactical Networks in Austere Environments*; Technical Report; Naval Postgraduate School: Monterey, CA, USA, 2019.
8. Lee, J.S.; Yoo, Y.-S.; Choi, H.S.; Kim, T.; Choi, J.K. Energy-efficient tdma scheduling for uvs tactical manet. *IEEE Commun. Lett.* **2019**, *23*, 2126–2129. [[CrossRef](#)]
9. Fu, B.; Xiao, Y.; Deng, H.; Zeng, H. A survey of cross-layer designs in wireless networks. *IEEE Commun. Sur. Tutor.* **2013**, *16*, 110–126. [[CrossRef](#)]
10. Ghribi, M.; Meddeb, A. Survey and taxonomy of mac, routing and cross layer protocols using wake-up radio. *J. Netw. Comput. Appl.* **2020**, *149*. [[CrossRef](#)]
11. Xu, Y.; Wang, Q.; Xu, Y.; Liu, J.; He, C. MPDMAC-SIC: Priority-based distributed low delay MAC with successive interference cancellation for multihop industrial wireless networks. *Comput. Commun.* **2020**, *154*, 48–57. [[CrossRef](#)]
12. Ajmini, N.; Abdelhamid, L.; Pascal, M.R. SPEECH-MAC: Special purpose energy-efficient contention-based hybrid MAC protocol for WSN and Zigbee network. *Int. J. Commun. Syst.* **2021**, *34*. [[CrossRef](#)]
13. Nosheen, I.; Khan, S.A.; Ali, U. A cross-layer design for a multihop, self-healing, and self-forming tactical network. *Wirel. Commun. Mob. Comput.* **2019**. [[CrossRef](#)]
14. Alkış, O. Improvement of Routing Protocols for Unmanned Flying ad hoc Networks (ufanets) by Using Cross Layer Metrics. Ph.D. Thesis, Middle East Technical University, Ankara, Turkey, 2019.
15. Soltani, S.; Sagduyu, Y.E.; Scanlon, S.; Shi, Y.; Li, J.H.; Feldman, J.; Matyjas, J.D. Cross-layer design and sdr implementation of dsa, backpressure routing and network coding. *arXiv* **2019**, arXiv:1912.05658.
16. Gawas, M.A.; Govekar, S.S. A novel selective cross layer based routing scheme using aco method for vehicular networks. *J. Netw. Comput.* **2019**, *143*, 34–46. [[CrossRef](#)]
17. Jagannath, J.; Melodia, T. VI-route: A cross-layer routing protocol for visible light ad hoc network. In Proceedings of the 2019 IEEE 20th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Washington, DC, USA, 10–12 June 2019; pp. 1–9.
18. Seo, M.; Kim, K.; Ko, Y.-S.; Kim, K.; Kim, D.; Choi, J.W.; A study of mobile ad-hoc network protocols for ultra narrowband video streaming over tactical combat radio networks. *J. Korea Inst. Military Sci. Technol.* **2020**, *23*, 371–380. [[CrossRef](#)]
19. Wang, M.-C.; Chuang, Y.S.; Davidson, S.A.; Liu, B. In the design of tactical communication systems. In Proceedings of the 2018 Annual IEEE International Systems Conference (SysCon), Vancouver, BC, Canada, 23–26 April 2018; pp. 1–6.
20. Kordbacheh, H.; Oskouei, H.D.; Mokari, N. Robust cross-layer routing and radio resource allocation in massive multiple antenna and ofdma-based wireless ad-hoc networks. *IEEE Access* **2019**, *7*, 36527–36539. [[CrossRef](#)]
21. Shafi, S.; Ratnam, D.V. A cross layer cluster based routing approach for efficient multimedia data dissemination with improved reliability in vanets. *Wirel. Personal Commun.* **2019**, *107*, 2173–2190. [[CrossRef](#)]
22. Wang, Q.; He, X.; Chen, N. A cross-layer neighbor discovery algorithm in ad hoc networks based on hexagonal clustering and gps. In Proceedings of the 6th Annual 2018 International Conference on Geo-Spatial Knowledge and Intelligence, Hubei, China, 14–16 December 2019; Volume 234, p. 012050.
23. Shivalingagowda, C.; Jayasree, P.; Sah, D.K. Efficient energy and position aware routing protocol for wireless sensor networks. *KSII Transact. Int. Info Syst. (TIIS)* **2020**, *14*, 1929–1950. [[CrossRef](#)]
24. Mageswari, R.U.; Baulkani, S. Routing technique for data transmission under jammer in multihop wireless network. *J. Ambient Int. Hum. Comput.* **2020**, 1–9. [[CrossRef](#)]
25. Sah, D.K.; Amgoth, T. Parametric survey on cross-layer designs for wireless sensor networks. *Comput. Sci. Rev.* **2018**, *27*, 112–134. [[CrossRef](#)]
26. Gupta, N.; Dhurandher, S.K. Cross-layer perspective for channel assignment in cognitive radio networks: A survey. *Int. J. Commun. Syst.* **2020**, *33*. [[CrossRef](#)]
27. Wei, Z.; Han, C.; Qiu, C.; Feng, Z.; Wu, H. Radar assisted fast neighbor discovery for wireless ad hoc networks. *IEEE Access* **2019**, *7*, 176514–176524. [[CrossRef](#)]

28. Robinson, Y.H.; Krishnan, R.S.; Julie, E.G.; Kumar, R.; Thong, P.H. Neighbor knowledge-based rebroadcast algorithm for minimizing the routing overhead in mobile ad-hoc networks. *Ad Hoc Netw.* **2019**, *93*. [[CrossRef](#)]
29. Rukaiya, S.A. Self-forming multiple sub-nets based protocol for tactical networks consisting of sdrs. *IEEE Access* **2020**, *8*, 88042–88059; doi:10.1109/ACCESS.2020.2992599. [[CrossRef](#)]
30. Elmasry, G.F. *Tactical Wireless Communications and Networks: Design Concepts and Challenges*; John Wiley & Sons: Hoboken, NJ, USA, 2012.
31. Shih, C.-F.; Xhafa, A.E.; Jianwei, Z. Techniques for Generation of a Frequency Hopping Sequence. U.S. Patent 9,614,579, 4 April 2017.
32. Nosheen, I.; Khan, S.A.; Khaliq, F. A mathematical model for cross layer protocol optimizing performance of software-defined radios in tactical networks. *IEEE Access* **2019**, *7*, 20520–20530. [[CrossRef](#)]