Wireless Sensor-Actuator Network for Cell-Level Treatment Based on Protocol of Collision Segregation via Learning

MO ZHAO\textsuperscript{1,2}, (Member, IEEE), AND ROBERT H. BLICK\textsuperscript{2,3}, (Member, IEEE)

\textsuperscript{1}The Mathworks, Inc., 1 Apple Hill Dr, Natick, MA 01760, USA
\textsuperscript{2}Department of Electrical and Computer Engineering, University of Wisconsin-Madison, 1415 Engineering Dr. Madison, WI 53706, USA
\textsuperscript{3}Center for Hybrid Nanostructures, University of Hamburg, Jungiusstr. 11c, Hamburg 20355, Germany

Corresponding author: Mo Zhao (e-mail: mzhao8@wisc.edu).

This work was supported in part by the Wisconsin Alumni Research Foundation (WARF) through an Accelerator grant and the Defense Advanced Research Projects Agency (DARPA) MOLDICE program.

ABSTRACT

State-of-the-art technology of nano-sensors and actuators enables diagnostics and treatment on the scale of biological cells. However, an advanced medical application should allow the direct feedback of a sensed signal to an actuation, e.g., an action potential propagation through an axon or a special cell activity might be sensed and suppressed by an actuator through voltage stimulation or chemical agent delivery. Such a complex procedure calls for building a communication network between sensors and actuators, but is challenged with issues such as intensive signal collisions, protocol simplification, and communication efficiency. We propose a collision segregation method and the protocol of Collision Segregation via Learning (CSL). Simulations of stress tests show its advantages as compared to the performance of the traditional methods for wireless body area networks (WBANs). The potential applications are discussed in detail, such as photothermal cancer therapy and anti-cancer drug delivery.

INDEX TERMS

biosensor, cancer treatment, nanonetworks, wireless body area networks, wireless communication protocol.

I. INTRODUCTION

WITH the rapid development of bio- and nanotechnology, the state-of-the-art sensors can extract information from biological cells, such as the action potential, pH-value, pharmacokinetic parameters, proteinase activity, and even molecular events, by converting chemical information or conformational change into electrical or optical signals [1]–[13]. In turn, intra-body actuators can act on cells, such as exerting specific stimulations and adjusting the therapeutic dose on targeted cells [14]–[18]. A more advanced system of sensors-actuators in the future should be able to perform appropriate actuation based on collecting feedback signals from all related sensors. For example, an action potential propagation through an axon can be sensed and then suppressed by an actuator downstream via electric stimulation; chemicals can be delivered to a cell on specific activity or status detected. Such a complex system can enable the cell-level automatic therapy, which would be advantageous, but calls for communication between those sensors and actuators, restraint by the demanding intra-body environment and the device size.

In the past decade, there has been a significant progress on using nanotechnology for cell-level cancer therapy: The localized surface plasmon resonance effects of gold nanoparticles, which strongly enhance the absorption from near-infrared (NIR) radiation, have been applied to photothermal cancer ablation, pointing towards a very perspective therapy path [19]–[21]. Similarly, single-walled carbon nanotubes (SWCNTs), with their low cost, also show a strong optical absorbance in NIR spectral windows [22]. This intrinsic optical properties, with SWCNTs’ functionalization and transport capabilities, can be used for cancer cell positioning [23], photothermal ablation [24]–[26] and drug delivery [27], [28]. In these novel therapy modes, the irradiation dose to destroy tumor cells relies on the absorption potency of nanoantennas, but the quantitative photothermal model for the intra-body absorption of plasmonic nanomaterials has widely remained absent from testing [19]. This forms an obstacle for effective clinical use.

The wireless body area networks (WBANs), which inter-
connect tiny nodes with sensing or actuating capabilities in, on, or around a human body for short range communications, have much attracting attention in recent years [29]–[32]. To adapt the human body environments and reduce power consumption, in February 2012, IEEE published the first international standard for WBANs, IEEE Std 802.15.6 [31]–[34]. WBANs may consist of wearable or implantable sensor nodes that sense biological information from the human body and transmit to a control device worn on the body or placed at an accessible location for ambulatory health/therapy status monitoring [29], [30]. WBANs’ intra-body communication (IBC) is based mainly on capacitive or galvanic coupling. Technical challenges were magnified for the IBC because of the lack of a corresponding mathematical model and the impossibility of conducting intra-body measurements [31], [32], [34]. These make it difficult to leverage WBANs for the application of the cell-level automatic therapy.

Furthermore, the emerging research on nanonetworks provides another opportunity for cell-level sensing-actuating networks [35]–[37]. Nanonetworks can interconnect nanomachines, which are tiny components consisting of an arranged set of molecules and can only perform very simple computation, sensing or actuation tasks [35]. Due to their size and energy constraints, classical communication paradigms based on continuous signal modulation is not feasible. Currently, introducing an innovative protocol stack model that captures the specific characteristics of nanonetworks is still in its early stages and become an active area of research [38].

As a solution, we suggest a new localized body area network solution, which can achieve a precise and personalized dose control of drug or irradiation through close-loop feedback. This will be a powerful promotion of the current personalized medicine system, with optimal doses for each patient to maximize efficacy and minimize side effects [9], [11]. Another challenge for the application is reducing the circuity size and the communication protocol complexity. Unlike other comparable protocols including WBAN [33], radio-frequency identification (RFID) [39], and the wireless sensor network (WSN) [40], which consider more complex communication situations, the application has a strong restraint on size and power, and needs only simple messages to communicate, including the sensing information, the command towards an actuator, the acknowledgement (ACK) signal, and the extensions, such as relay route.

We propose a protocol of Collision Segregation via Learning (CSL) to form a layered ad-hoc network of biomedical sensors and actuators that can be deployed with or even inside biological cells. In comparison with other protocols such as IEEE 802.15.6 for WBAN, this solution is well adapted for cell-level sensing-actuating tasks considering the requirements on size/complexity, power consumption, communication range, actuation delay, etc. For example, the conventional CSMA/CA method, widely used in the wireless communication [40], [41], no longer works well for such applications, not only due to expensive overhead and circuit complexity, but also because that burst signal transmission and intensive collisions commonly happen in bio-sensing/actuating and may cause serious issues of loss or delay.

In the following, we specify the model of the sensor-actuator network with double-layer nodes in Sec. II. The collision segregation method and the CSL protocol are introduced in Sec. III and Sec. IV, respectively. Simulation results in Sec. V show the advantage of the CSL protocol by comparing with conventional standards. We give specific application examples in Sec. VI for better understanding the solution.

II. MODEL OF SENSOR-ACTUATOR NETWORK

The sensor-actuator network is composed of two layers of nodes: the primary sensors/actuators, as the primary nodes (or simply note as nodes), and their nearby secondary sensors/actuators, as the secondary nodes, as shown in Fig. 1. Ideally, the whole coverage of the network can be divided into cellular regions; each region is covered by a primary node, including the secondary nodes within the region. For communication of primary nodes, surrounding primary nodes usually has overlap of signal coverage, so a challenge is to overcome signal collisions. The communication scenario can be summarized as following:

a) Primary nodes are micron-scale devices that can not only perform sensing/actuation assisted by the secondary nodes (based on optical antennas for example), but also communicate with other primary nodes or the central controller if in range based on simple wireless communication protocols, for delivery of environmental information, actuation instruction, etc. All primary nodes are synchronized by an external
clock, which could be realized by a periodically pulsed laser (also providing power to the individual node circuits). Nevertheless, not all nodes can send signals to the central controller and communicate with a specific node, and relays may be needed if such communication is necessary. For device simplicity, this sensor-actuator network uses only one frequency channel, thus no frequency division for multiple access. It adopts the shared-frequency ad-hoc mode.

b) Secondary nodes are composed of simple particles with specific biochemical, mechanical, optical, thermal, or electromagnetic properties, but without computation capability and thus unable to perform complex communication. They are usually used for sensing signals of variations, and concentrating/converting energy upon excitation. For example, after sensing chemical signals has reached a certain threshold, it can generate a signal indicating an abnormal event, and trigger the nearby primary sensor/actuator to initiate a communication to notify others about this abnormal event. For another example, it can release medicine quantitatively or implement optothermal conversion upon a trigger signal, for medical treatment on the micrometer level. Such signal transmission involving the secondary sensor/actuator is outside the frequency band of the primary nodes, thus not affecting the communication channel of the primary nodes.

In reality, the network presents a much more complex structure than Fig. 1, as the example indicated in Fig. 2, where a primary node may be topologically connected to either more than or less than its surrounding nodes. Additionally, due to the uncertain overlap of signal coverage of different primary nodes, the number of communication channels for each node is also uncertain. Thus, with limited resources, the channels cannot be allocated to every primary node prior to network setup, but must go through a learning process after the network connection is determined.

The nodes can communicate with each other via optical or radiofrequency carrier signals. For the optical option, signals can be carried by ultrashort optical pulses from an external pulsed laser, which also provides power and the synchronous clock for all nodes.

III. COLLISION SEGREGATION METHOD

In biomedical applications, an event, such as an axon action potential, may trigger a series of other events. The signal timing of nearby sensors/actuators (S/As) are usually strongly related. Thus, there is a high probability for dense collision of signals while the primary nodes communicate to transmit information or issue commands. If we adopt the conventional CSMA/CA protocol, the loss of messages due to frequency collision can prevent the system from working reliably. In contrast, adopting regular TDMA (time division multiple access) method would waste the resource and delay the overall response. In order to solve the problem, we propose the method of collision segregation and a corresponding protocol, according to the feature of cell-level sensing and actuation. It leverages the basic methods of time division, collision detection, and collision avoidance, and adapts the setting to the application for optimization. This method can meet the requirements of high reliability and efficiency of network communication with lowest cost.

The collision segregation method has the following features:

(1) Adaptive 2D TDMA: Using the general TDMA, the communication time of every node is divided into \( \tilde{T} \) time slots, each with a slot ID from 1 to \( \tilde{T} \) and running in loop; every node is assigned a slot ID and sending is only allowed in that slot (while receiving can be performed in any slot). Without a preset boundary for \( \tilde{T} \), we want to adaptively allocate a slot ID to each node according to the network topology, in order to minimize \( \tilde{T} \) and eliminate possible collisions. This would require the nodes with the same topological neighbor to have different slot IDs, which is actually a type of graph coloring problem [42]. However, considering the slot utilization and the challenge to compute the optimum allocation, we prefer to set an upper limit of \( \tilde{T} \), denoted by \( T \), and combine the slots. A slot ID (denoted by \( \tau \)) above \( T \) would be replaced by a slot ID smaller than \( T \) (e.g., using \( \tau \bmod T \)). Thus, there would be multiple slots combined in the same slot, and each of the slots is assigned a standing ID \( s = [\tau/T] \), denoting the sending priority in case of collision, where messages from the lower standing nodes would be resent after a scheduled delay. The pair of slot ID and standing ID assigned to every node actually establishes a two-dimensional TDMA, which is half implemented by the time slot allocation and half implemented by the collision detection in the following. (see Fig. 2) Furthermore, we expect the standing ID to change periodically in order to balance the transmitting priority of all nodes and avoid blocking of low-standing nodes. After every sending slot, a node increases the standing ID by 1 with a preset mod, which brings the number back to 0 when reaching maximum. As a result, this adaptive
2D TDMA provides a good way to balance efficiency and collisions, and is particularly suitable for intensive sending scenario due to improved time utilization.

(2) Collision detection: We introduce a collision detection code (CDC) at the beginning of each message (after the preamble), for each node to distinguish from other sending nodes in the same slot, with the value $C = 2^s$. Assuming the CDC to be a series of binary bits, it should have only a single ‘1’ on the position corresponding to the standing ID. Thus, with the 2D-TDMA, the sending nodes should transmit different CDCs in a slot, such that collisions causing overlaps can be easily detect. A sending node should listen to others’ CDC when it is not transmitting (or transmitting ‘0’). If it hears ‘1’ before its ‘1’ to transmit, it knows that another node with a higher standing is transmitting within the slot, and will stop transmitting its ‘1’, hold for $s + 1$ cycles and resend the message (however, hearing ‘1’ after transmitting, indicating a collision with a lower standing node, should not block transmitting). In addition, for a receiving node that has no transmitting task in the slot, hearing multiple ‘1’ bits indicates a collision, which would block its further receiving and ACK reply. Besides, with the change of standing ID, the CDC of a node is circularly left shifted by a bit every time slot. Note that while collision detection is a concise and efficient method for communication, such as CSMA/CD of Ethernet with bus topology, the implementation in common wireless scenario is difficult due to near-field effect and synchronization, but utilizing collision detection in wireless networks is still desirable and under investigation [43].

(3) Collision avoidance for hidden nodes: The receiving node replies an ACK signal within the same frame, provided that the received message is valid; if the sender does not receive this ACK, the message is considered as lost, so that the node holds for $s + 1$ cycles and resends the message (however, hearing ‘1’ after transmitting, indicating a collision with a lower standing node, should not block transmitting). This approach helps identify the collision after sending the data, and avoids the loss of data due to collisions. This is particularly useful for hidden nodes, which use the same slot but cannot see each other while transmitting, causing the failure of collision detection. Because the message in the system has short and fixed length, using ACK should have better performance than another collision avoidance method CTS/RTS (Request to Send / Clear to Send), as pointed by IEEE Std 802.11 [41].

It is worth remarking that the method optimizes the energy cost, compared to the common collision detection/avoidance methods. For the exposed collisions, a sending node listens to the CDC before transmitting its own CDC bit, and if hearing a ‘1’, it stops massage sending and entering the energy-saving state for a few cycles. For the hidden collisions, a receiving node uses CDC to detect collisions: if finding more than one ‘1’ bit in CDC, the receiving node early terminates receiving operations and enters the energy-saving state for this slot.

### IV. COMMUNICATION PROTOCOL

We propose a communication protocol based on the collision segregation method and an adaptive learning process for routing and slot allocation. It provides reliability and performance with sufficient simplicity adapted for the tiny S/A. A learning mode that initiates the slot ID, standing ID and routing is designed before the working process. We name this protocol Collision Segregation via Learning (CSL). In order to physically simplify the signals for size and power optimization and also to shorten the communication delay time, we perform a cross-layer design [44], [45], which considers implementation complexity of physical layer and functional requirements of logic layers.

### A. WORKING MODE

In the physical layer of the protocol, the signal is carried by optical or RF pulses. Every bit in communication being 1 or 0 can be represented by the existence of a pulse. This simple digital encoding favors circuit simplification and device miniaturization. The continuous time of communication is split into slots, which individually accommodates a frame of message, the fundamental unit of data transmission. In our prototype each frame has a length of 4 words or 64 bits, as shown in Fig. 3. There is a short inter-frame space (SIFS) between contiguous slots, which is used for post-processing of the last frame, switching between transmitting and receiving, and preparation for the next frame. Additionally, as stated in section II, every node is only allowed to send messages in a specific slot, which is associated with a slot ID (from 1 to $T$). Due to limited time division, we set the maximum slot ID at 7 (i.e., $T = 7$), and other slot IDs are folded in this range.

The composition of a frame is illustrated in Fig. 3, containing the following parts:

1. Type bits utilize 2-bit 00 to distinguish a message in the working mode from the learning-mode messages. Note that sending 00 means transmitting no signal during the period, which favors the collision detection for saving transmitting-receiving switching.

2. Collision detection code (CDC) takes up to 4 bits and only the bit corresponding to the node’s standing ID is set to 1. The standing ID ranges from 1 to 4. Given the 7 slots, our 2-dimensional time division hence supports up to $4 \times 7 = 28$ communication channels (in comparison, the need is generally less than 20 for a network according to simulations shown in the following section).
(3) Destination node ID indicates the destination node of a message and takes up 10 bits, which is assigned uniquely to each node before deployment, allowing 1024 nodes in a network. If a receiver detects that the received destination node ID is different from the its node ID, it would terminate further receiving actions in the slot.

(4) Message serial number utilizes 6 bit, ranging from 0 to 64, is used to identify each message, for possible use of sorting messages and checking for duplicates.

(5) Source node ID indicates the source node of a message and takes up 10 bits.

(6) Data in the message frame takes 24 bits, containing the sensed information or the action command.

(7) Cyclic redundancy check (CRC) code utilizes 6 bits, computed based on the prior 56 bits. A wrong CRC code received by the destination node would not trigger the reply of ACK signal.

(8) Acknowledgement (ACK) signal utilizes 2 bits. The first bit is reserved as a buffer space, for receive-transmission switching and data processing. Failure in collision detection and faulty CDC or destination ID would also block replying ACK. From the sender’s perspective, it will hold any sending for a number (equal to the standing ID) of sending cycles before resend, if the designated ACK is not received at the end of the frame.

This protocol is fairly simple to implement for every node. From the memory aspect, a node needs to maintain a sending queue of messages and a resending timer, besides the memory for each part of the frame being received, and the constant memory of node ID (\(i\)), and time slot ID (\(\bar{r}\)). Note that the upper bar in the context denotes the variables scoped to a specific node and cached in its memory.

\[\text{ROUTE}\text{hop}\text{upstream ID}\text{source ID}\text{max node ID}\]

\[\text{BACK}\text{max hop}\text{upstream ID}\text{source ID}\text{downstream num}\]

**FIGURE 4.** Frame structure of route learning protocol denoted simply as \(\text{ROUTE}(h, i_u, i_s, N)\) and \(\text{BACK}(h_m, i_u, i_s, n_m)\), respectively.

B. LEARNING MODE FOR ROUTING

In this mode, every node tries to learn what node is its upstream node and what nodes are its downstream nodes, in terms of the central controller (CC). For the purpose of routing, we expect the shortest path as the principle for learning the upstream, where the distance can be measured by the number of hops a message needs to run from the CC to the node. Accordingly, each node can send a hop number to its neighbors, namely 1 plus the hop count from CC to itself. The receiver would cache this hop number, accept the sender as its upstream, and broadcast a new hop number message to its downstream, provided that the receiver’s current hop number is undetermined or greater than the received hop number.

This is the first learning stage prior to the network communication, where a classic TDMA can be applied: Every node transmits in a distinct time slot that is associated with their node ID before network deployment. In other slots, a node can only receive messages from others. This simply provides each node with a separate uplink channel.

The message frame structure in a time slot is described by Fig. 4, where the message \(\text{ROUTE}(h, i_u, i_s, N)\) provides the downstream node with the information to learn, including the hop number from CC (\(h\)), the upstream node ID (\(i_u\)), and the maximum node ID (\(N\)), and the message \(\text{BACK}(h_m, i_u, i_s, n_m)\) notifies the upstream node that the branch has completed learning. Additionally, every node has a cache of its hop number (\(\bar{h}\)) and the maximum hop for a node in the network (\(h_m\)), it also remembers its unique upstream node (\(i_u\)) and a list of downstream nodes (\(I_d\)).

Initially, the CC sends \(\text{ROUTE}(1, 0, i_{CC}, N)\) to start the process, where \(i_{CC}\) represents the node ID of CC. The algorithm for any node after receiving a message is described in the following:

\[
\text{if received a message } \text{ROUTE}(h, i_u, i_s, N) \text{ then}
\]

- \(\text{if its slot counter was not activated then}
  \]
- \(\text{start the counter with initial value } i_s;\)
- \(\text{set the loop bound as } N;\)
- \(\text{if node } i_s \text{ was not in its neighbor list then}
  \]
- \(\text{add node } i_s \text{ in};\)
- \(\text{if } i_u = \bar{i} \text{ then}
  \]
- \(\text{add node } i_s \text{ as downstream in } I_d;\)
- \(\text{stop the timer; }\)
- \(\text{else if } h < \bar{h} \text{ or } \bar{h} \text{ is unset then}
  \]
- \(\text{set } \bar{h} = h;\)
- \(\text{set node } i_s \text{ as upstream (} \bar{i}_u = i_s;\)
- \(\text{transmit } \text{ROUTE}(h + 1, \bar{i}_u, \bar{i}, N) \text{ in the next available slot; }\)
- \(\text{else if node } i_s \text{ is in } I_d \text{ then}
  \]
- \(\text{remove node } i_s \text{ from } I_d;\)
- \(\text{if traversed } I_d \text{ and no ROUTE to send then}
  \]
- \(\text{transmit } \text{BACK}(h_m, \bar{i}_u, \bar{i}, \text{size}(I_d)) \text{ in the next available slot; }\)
- \(\text{else if received a message } \text{BACK}(h_m, i_u, i_s, n_m) \text{ then}
  \]
- \(\text{if } i_u = \bar{i} \text{ and no ROUTE to send then}
  \]
- \(\text{set } h_m = \text{max}(h_m, h_m);\)
- \(\text{sort } i_s \text{ in } I_d \text{ in a descending order of } n_m;\)
- \(\text{if traversed all downstream nodes in } I_d \text{ then}
  \]
- \(\text{transmit } \text{BACK}(h_m, i_u, i_s, \text{size}(I_d)) \text{ in the next available slot; }\)

The process finishes when the CC receives the BACK messages from all of its downstream nodes. Then, it is the time to start the next learning mode for the slot allocation.

C. LEARNING MODE FOR SLOT

As pointed in the above discussion, a perfect adaptive slot allocation requires the nodes with the same topological
neighbor to have different slot IDs, which is equivalent to the graph coloring problem known as a NP-complete problem [42]. It is impractical to find the optimum allocation, given the limitation of time and memory. Especially, in the scenario of the communication network, a node cannot easily share all the known information with other nodes in the process of the graph searching, making the algorithm much more demanding than the common case. Thus, we propose a learning algorithm to find a suboptimum allocation for this challenging problem, where each node, instead of looking at all possible permutations of available slot IDs and downstream nodes, always tries to assign the smallest available slot ID to the downstream node with the most neighbors. This should give the downstream nodes more flexibility to use smaller slot IDs, thus optimizing the maximum number used. The algorithm utilizes three types of messages, START, FINISH, and SLOT, shown in Fig. 5. For any of the nodes in the process, the algorithm works as the following.

```
if received a message START(i_d, i_s, \tau) then
    mark sender's slot ID (\tau) in the slot pool as occupied;
    if i_d = i then
        pick smallest available slot ID in the pool as \bar{\tau};
        transmit SLOT(0, i, \bar{\tau}) in the next slot;
    else if received a message SLOT(i_d, i_s, \tau) then
        if i_d = i (indicating a conflict) then
            pick smallest available slot ID in the pool as \bar{\tau};
            transmit SLOT(0, i, \bar{\tau}) in the next slot;
        else if i_d = 0 and \tau is occupied in the slot pool then
            reply SLOT(i_s, 0, \bar{\tau}) to sender in the next slot;
        else if received a message FINISH(i_d, \bar{\tau_m}, \bar{\tau}) then
            mark sender's slot ID (\tau) in the slot pool as occupied;
            if i_d = i then
                update cached \bar{\tau_m} = max(\bar{\tau_m}, \bar{\tau_m});
                execute SendNext();
            else if transmitted a message SLOT(0, i, \bar{\tau}) then
                if no SLOT(\bar{\tau}, 0, 0) is received in the next slot then
                    execute SendNext();
```

The central controller acts as a special node in this mode. It sends a START message to start the slot learning mode, and

```
FUNCTION SendNext() move the downstream pointer to the next node;
if the downstream pointer points to node i_p then
    send START(i_p, \bar{\tau}, \bar{\tau}) to node i_p;
else
    send FINISH(i_u, \bar{\tau_m}, \bar{\tau}) message to upstream;
```

ends the process after receiving the FINISH messages from all of its downstream nodes. Subsequently, it broadcasts the maximum slot ID \bar{\tau_m} used in the network to all other nodes by relay (note that no collision needs to consider for identical messages). As the learning result, every node will memorize the crude slot ID \bar{\tau_m} and use \bar{s} = \lfloor \bar{\tau_m}/T \rfloor to initialize the CDC with \bar{s} = \lfloor \bar{\tau_m}/T \rfloor as its length for circular shifting.

This slot learning process is fast, easy to implementation, and with good accuracy. It only costs \mathcal{O}(N \times n) time and \mathcal{O}(n) memory, where \textit{N} denotes the node number and \textit{n} denotes the neighbor number of a node. In Fig. 6, we compare the learning result of the slot number (before combination) with the optimum number obtained by searching all possible allocations, based on a large scale of random networks with 50 nodes. The horizontal axis is the maximum number of neighbors for a node (n_m), which characterizes the network bottleneck and determines the theoretical lower limit of the slot number (n_m + 1).

V. SIMULATION RESULTS

In the following, we use simulation results to compare the performances of the proposed CSL method with the conventional methods including TDMA and CSMA/CA (Carrier Sense Multi-Access/Collision Avoidance) adopted by IEEE 802.11, IEEE 802.15.4, IEEE 802.15.6, etc. In the simulation, we consider two types of CSMA/CA methods, one employing ACK message (hereinafter referred to as CA-ACK) and another with the RTS/CTS mechanism (here-
In order to enhance performance, both of them also utilize the virtual carrier-sensing mechanism, with NAV (Network Allocation Vector) included in the 64-bit frame. For the random backoff mechanism, the contention window is with width 8–1024, which were adopted by 802.11. Furthermore, we adopt the method of adaptive time division multiple access (ATDMA) for comparison, which uses the result of our learning process for slots but without slot combination. This method is obviously better than the simple TDMA with a presumed number of time divisions (from the topologically most complex node if known, or using the number of nodes).

In the simulation, we randomly generate a network of 50 nodes as Fig. 3 with arbitrary positions and connections. Each node follows a Poisson process to generate random messages for transmitting with a given average rate, where the destination and the message data is randomly chosen from available values. Thus, we implement Monte-Carlo simulations for the network, where a scale of 50 displays a good accuracy. Because the correlated biochemical changes and the corresponding sensing/actuating signals usually happen intensively, we must adapt the network to the extreme scenarios. Two testing conditions are hence interesting: (1) Continuous stress test, with high volume of sending tasks approaching the full load, where we test various generation rates in simulations and depict the result in Fig. 7. (2) Impulse test, with a high generation rate much higher than full-load but only lasting for a short time (using the first 14 slots in our test). Fig. 8 shows the response versus the impulse magnitude (measured by the average number of generated messages per node).

The average delivery rate of message transmission, indicating the ratio of received messages in generated messages, and the average delay, indicating the time for a message from being generated to being received, are two figures of merit of communication performance. For different protocols under the same load stress, the higher the delivery rate, the better communication efficiency.

The simulation results show that the conventional CSMA/CA maintains an acceptable delivery rate in case of light load of message transmission, but has an even worse performance under heavier load. The method CA-ACK performs better than CA-RTS, because the short frame in use favors smaller overhead. The ATDMA method shows an even better performance, as the time slots are well utilized in the context of heavy load, and this also proves the power of our slot learning process. Our CSL protocol has the best performance, which further reduces the number of slots to avoid wasting resources and integrates the methods of collision detection and collision avoidance for resolving collisions. A similar result of average delay for impulse response is shown in Fig. 8, where the average delay of the CSL protocol is also the best among the four, no matter if considering heavy or light load, which proves it the most suitable network communication protocol for the heavy-load application.

**VI. PROSPECTIVE APPLICATIONS**

This solution has good application prospects, such as in cancer treatment. In the following we discuss two prospective application examples.
A. PHOTOTHERMAL CANCER THERAPY

In this application example, secondary nodes adopt single-walled carbon nanotube (SWCNT) antennas [25], [26] or gold nanorod antennas [19], [20], which function as both sensors and actuators, and the ad-hoc network of primary nodes consists of only sensors, which, as the design in Fig. 9, can realize communications based on optical antennas. The energy needed for photothermal therapy is provided by an external near-infrared continuous-wave (CW) laser [20], [24].

As the secondary nodes, the nanoparticles are conjugated to (or coated with) anti-EGFR antibodies [20] or folate [24], [26], so to automatically locate tumor cells. Applying the nanoparticles via an intravenous injection, they can flow along blood vessels and reach, for example, the liver, and cluster on target tumor cells, as depicted in Fig. 10a. Under irradiation of the laser pulses, they generate fluorescent signals, informing the nearby primary sensor of presence and location of cancer cells. Due to the strong clustering effect on cancer cells, nanoparticles straying in the normal cells generate very weak fluorescent signals, which are much lower than the receiving threshold of the primary sensors and are ignored.

On receiving a signal from secondary sensors, the primary sensor will send a request signal to the central controller, to notice the area of cancer cells and request for increasing localized energy for photothermal therapy. After the request signal is received, the controller will scan and irradiate the designated locations on a certain strategy, such as increasing the irradiation dose and duration. The SWCNT antennas or gold nanorod antennas, as secondary actuators, have a strong absorption for the near-infrared irradiation [24], whose energy is converted into heat to kill cancer cells. After cancer cells are destroyed at the designated locations, their sorption to the nanoparticles will vanish, so that these nanoparticles will dissipate quickly and finally be eliminated from the body. If signals of a primary sensor cannot reach the controller, it can notice the controller via relay from other nodes.

B. ANTI-CANCER DRUG DELIVERY

Another example is the anti-cancer drug delivery, which adopts ultra pH-sensitive (UPS) nanoparticles as the secondary sensors for cancer cell tracking and positioning [46], and magneto-electric nanoparticles (MENs) as secondary actuators for anti-cancer drug delivery and controlled release [47], as depicted in Fig. 10a and 10b.

Like the above example, the biomarker-specific antibodies are attached on the drug-loaded MENs and steer these MENs to the tumor cell membrane [47], after they are injected intravenously into the human body. Through a similar targeting process, the UPS sensors bind to tumor cell surface receptors or even enter tumor cells by receptor-mediated endocytosis. The UPS sensors have an ultra pH-sensitive fluorescence emission function with a near-infrared emission range, which can be turned on/off within less than 0.25 pH unit, with a large fluorescence activation ratio (higher than 300-fold) [46]. Because the tumor microenvironment has a stronger acidity than normal cells, the UPS sensors stay silent in normal cells (pH 7.4), but become activated in tumor extracellular milieu (pH 6.5-6.8) and inside tumor cells (pH 5.0-6.0) through the sharp pH transition [46], [48]. After arriving at the tumor cells via blood circulation, the UPS sensors will emit fluorescence signals to the nearby primary sensor, then the primary sensor will send stimulation to MEN actuators. Under the stimulation from the primary nodes, the drug-loaded MENs can generate localized electric fields to electroporate the membrane and enter the cancer cells. Next, under a greater stimulation, the MENs will release anticancer drug inside tumor cells [47].

C. NON-MEDICAL APPLICATIONS

Our CSL protocol also has potential applications outside the medical area. As it be initiated for adapting the network with burst transmission with the simplest design, the applicable scope can actually extended. The protocol is a promising candidate when the network satisfies the following conditions: (1) restricted or benefited in ad-hoc mode, such as for vehicular and military use; (2) challenged by burst transmission or heavy load; (3) a single communication message is short; (4) frequency division is unavailable or expensive. Examples include vehicle-to-everything (V2X) communication and wireless communication of robots.

VII. CONCLUSION

We focus on a body-area wireless sensor-actuator network that has promising applications in cell-level diagnosis and treatment. The network is composed by two layers of nodes: the primary nodes, which can communicate with others in range for delivery of environmental information, action instruction, etc., and the secondary nodes, which are simple particles with specific biochemical, mechanical, optical, thermal, or electromagnetic properties. System-level challenges for realizing such a network includes burst message transmission for correlated biochemical changes, reducing protocol complexity for device miniaturization, communication efficiency, etc. For the communication between primary nodes, we have proposed the collision segregation method and the corresponding protocol Collision Segregation from Learning (CSL), featured by adaptive 2D TDMA, collision detection, and in-frame collision-avoidance ACK. A learning mode is designed to acquire the routing and time slot for communication in the working mode. Simulations of continuous stress tests and impulse tests show its advantages compared to the performance of existing standards for wireless body area networks (WBANs). Additionally, we discussed the perspective applications in photothermal cancer therapy and anti-cancer drug delivery.

REFERENCES

FIGURE 10. Distribution of the cell-level sensors/actuators for cancer treatment. (a) Secondary sensors/actuators cluster on target tumor cells. (b) Secondary sensors (UPS nanoparticles) and actuators (MENs) can further enter tumor cells by receptor-mediated endocytosis and stimulation from primary nodes, respectively.


[40] IEEE Standard for Local and Metropolitan Area Networks – Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), IEEE Std. 802.15.4-2011, 2011.


***