The University of Iowa
Department of Computer Science

Real-Time Scheduling in Low-Power Mobile Wireless Networks

Behnam Dezfouli | Marjan Radi | Octav Chipara

The 13th Annual IEEE Consumer Communications & Networking Conference (CCNC)
9-12 January 2016, Las Vegas, USA

Real-Time Networks

- Correctness depends on both ***functionality*** and ***timeliness***

- Used in various applications such as industrial automation
  - Several organizations: HART, ISA, WINA, ZigBee

WirelessHART

- Uses IEEE 802.15.4 standard (250 Kbps)
- Centralized network management (centralized TDMA-based scheduling)
- Time-synchronized communication
- No intra-network interference: Only one device can send in a given time slot and channel
- Compatibility with existing HART devices
- ***Does not support mobility***

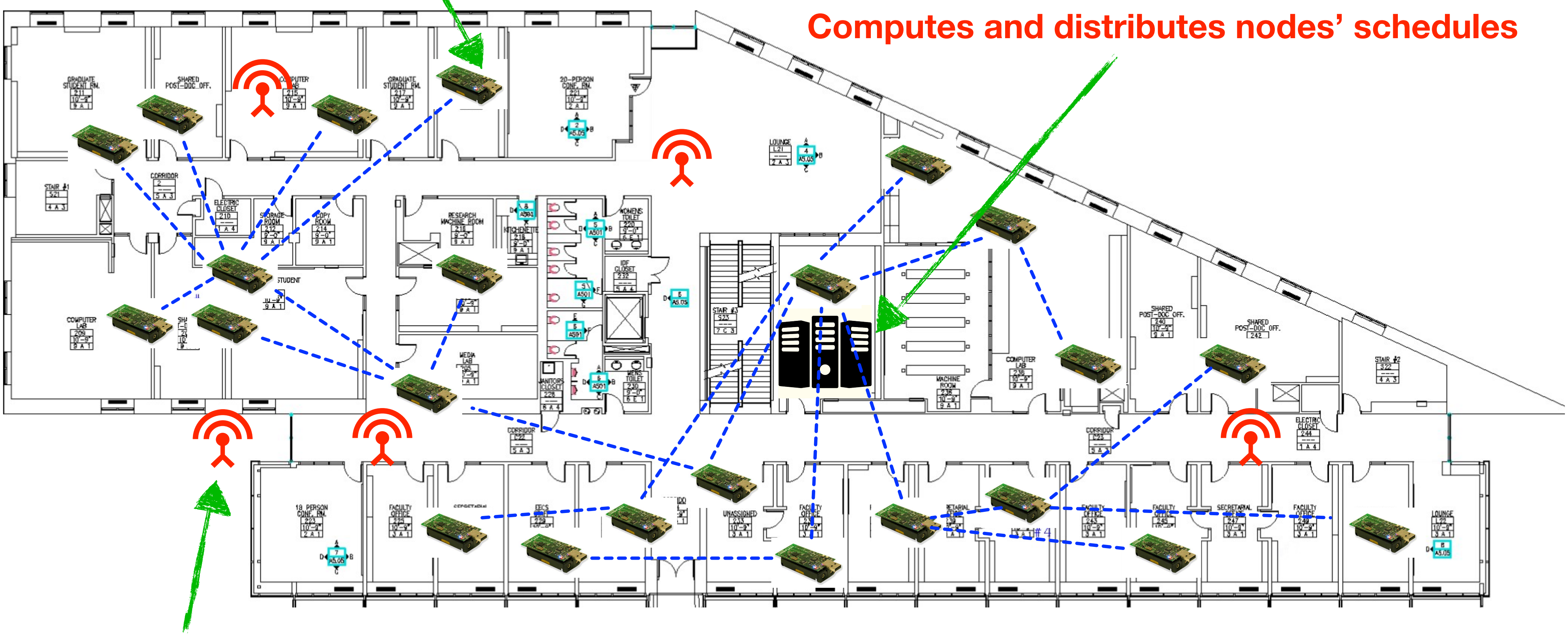
Real-Time Networks

- Existing solutions for real-time networks do not support mobility.
- Limits the applicability of these solutions to applications with mobile entities such as patients, robots, firefighters, etc.
- **How to support real-time communication with mobile nodes?**

Architecture

A Low-Power Wireless Infrastructure Node
Communicates in a real-time manner with the Gateway

Gateway
Communicates with the nodes
Computes and distributes nodes' schedules



A Low-Power Wireless Mobile Node
Communicates in a real-time manner with the Gateway

Impact of Mobility on Data Forwarding Paths

Low energy consumption

↓

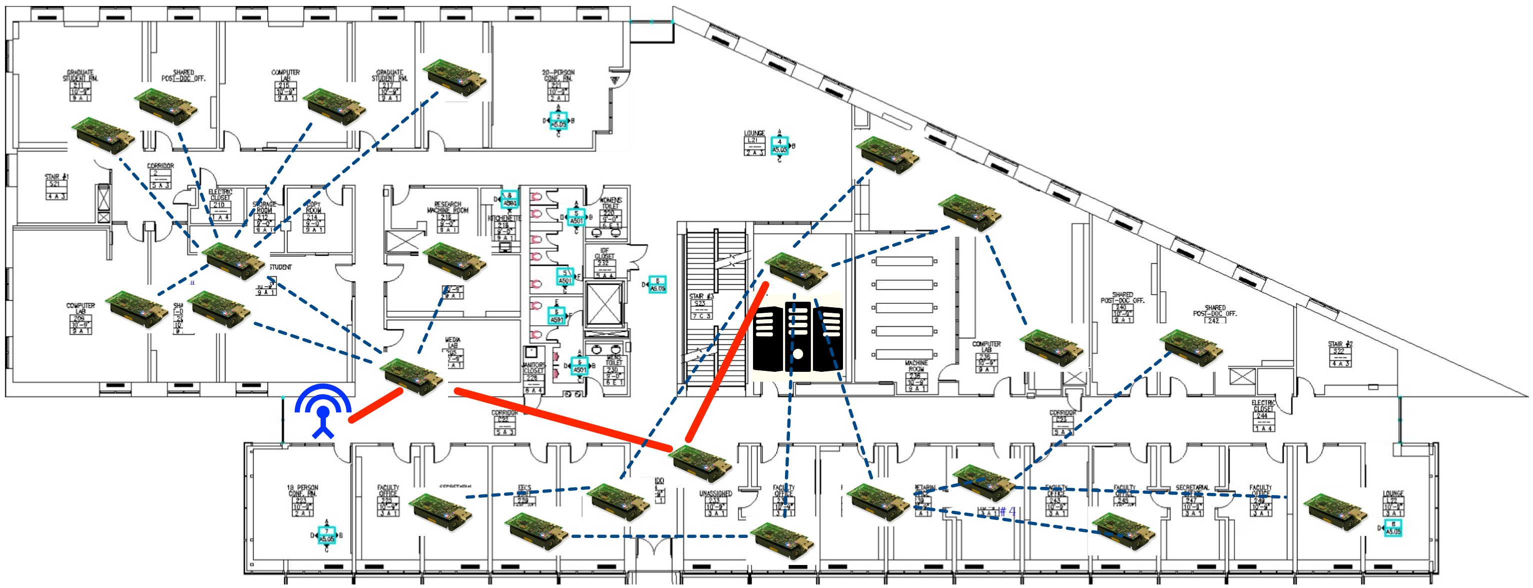
Short communication range

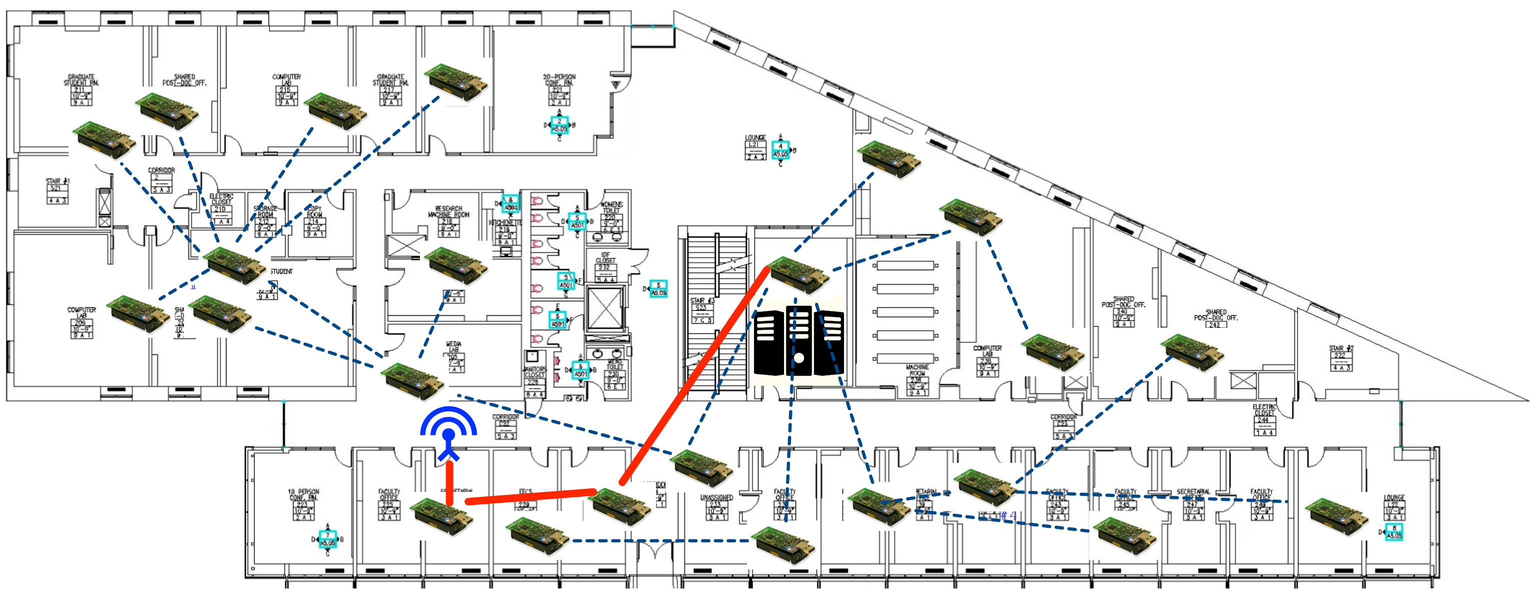
↓

Frequent association with infrastructure nodes

↓

Frequent changes in data forwarding paths





Two Bandwidth Reservation Strategies

1. On-Demand Bandwidth Reservation

- Gateway performs scheduling for bandwidth reservation *whenever the communication path changes*
- **Shortcomings:**
  - Connection loss: Gateway may not be able to reserve bandwidth
  - Mobile nodes should frequently request for bandwidth reservation: A huge bandwidth is used for exchanging control data

2. On-Join Bandwidth Reservation

- Bandwidth is reserved *over all the communication paths upon node join*
- Gateway admits a mobile node if the new scheduling is successful
- **Shortcomings:**
  - If performed naively, the number of admitted mobile nodes would be very small
  - We propose techniques to address this shortcoming

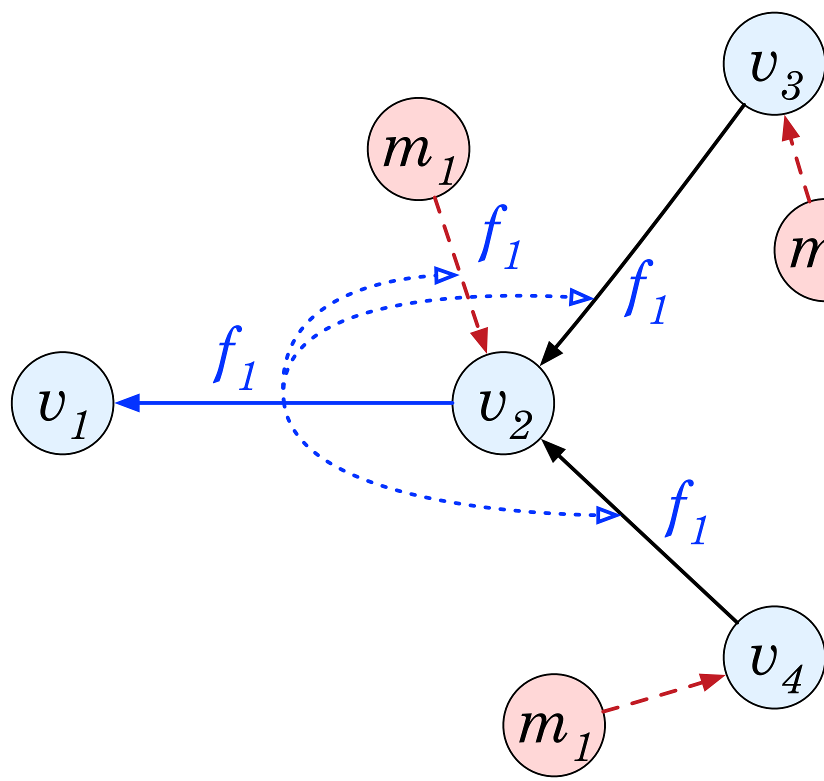
Efficient Bandwidth Reservation Techniques

We propose techniques for improving the bandwidth reservation efficiency of mobile nodes' data flows.

Technique 1

A transmission (v_i, v_j, f_q) should be released after transmissions $\{(m_r, v_i, f_q)\} \cup \{(v_l, v_i, f_q) | v_l \in \Upsilon(v_i)\}$ have been scheduled. $\Upsilon(v_i)$ is the set of the children of node v_i.

Transmission (v_2, v_1) can be considered for scheduling after transmissions (v_3, v_2), (v_4, v_2), and (m_1, v_2) have been scheduled.

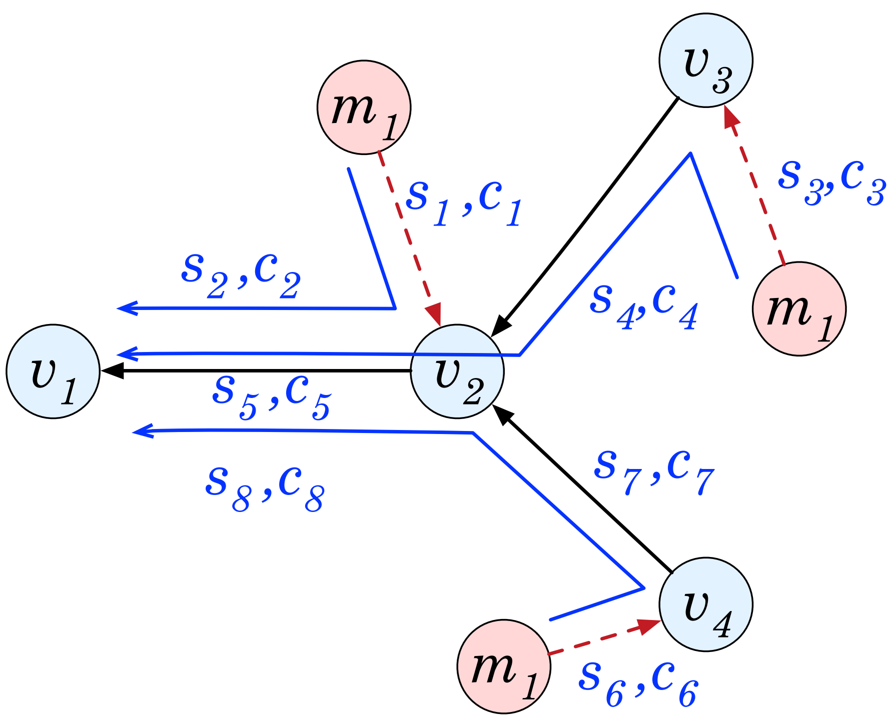


Efficient Bandwidth Reservation Techniques

Without Technique 1

<u>8</u> **different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1**

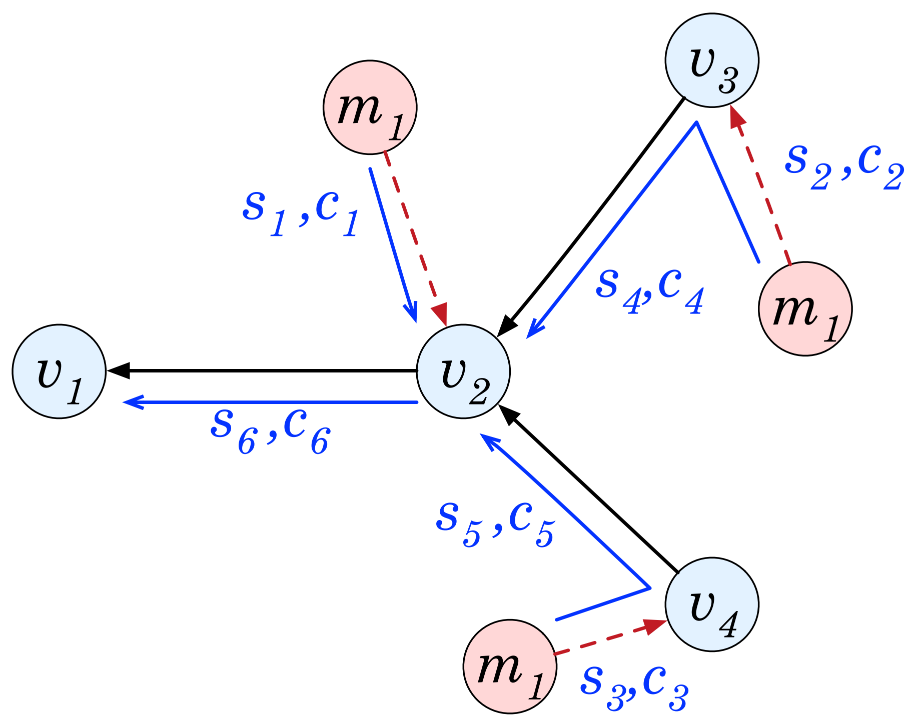
We refer to this approach as the
Basic Scheduling Algorithm (BSA)



With Technique 1

<u>6</u> **different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1**

We refer to this approach as the
Enhanced Scheduling Algorithm (ESA)



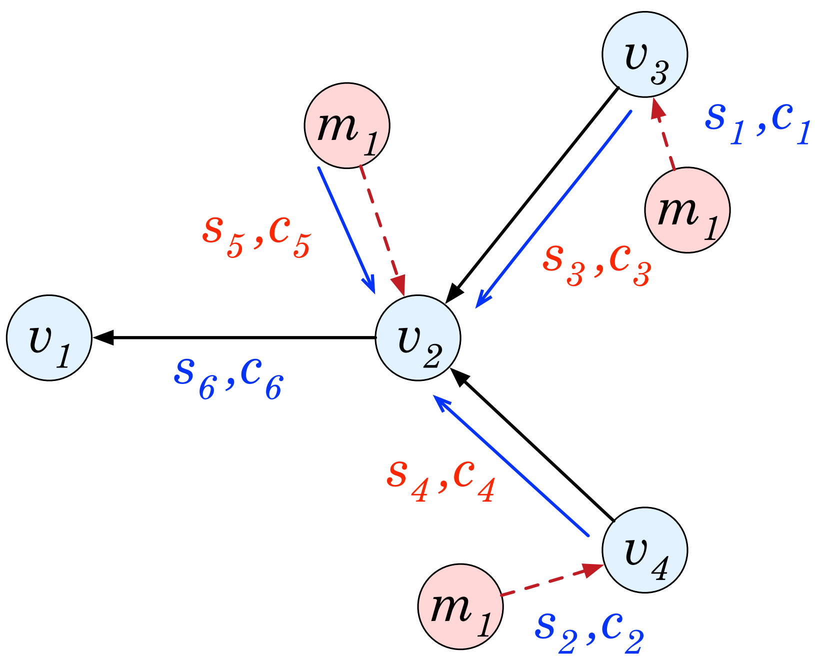
Efficient Bandwidth Reservation Techniques

Technique 2

Any subset of $\{(m_r, v_i, f_q)\} \cup \{(v_l, v_i, f_q) | v_l \in \Upsilon(v_i)\}$ can be combined.

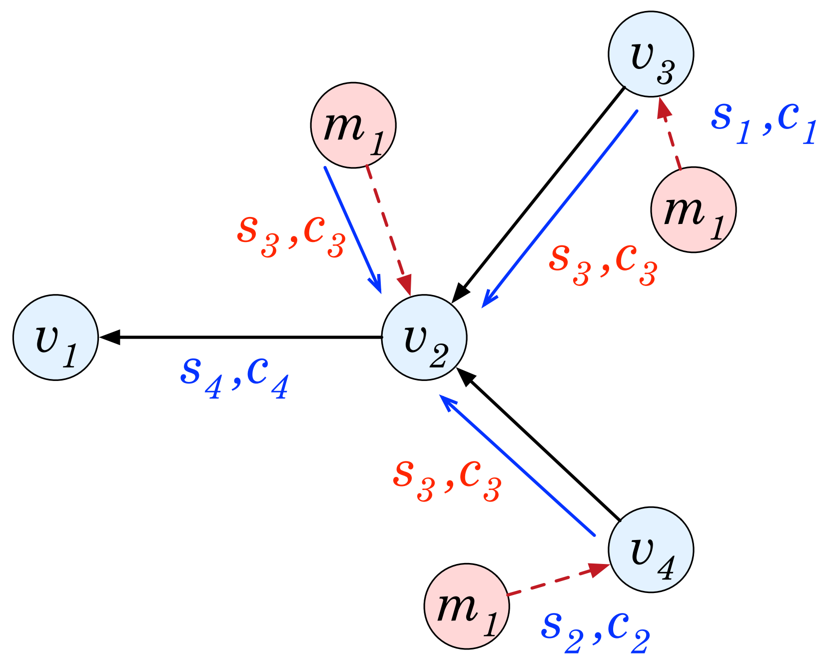
Without Technique 2

6 different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1



With Technique 2

4 different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1



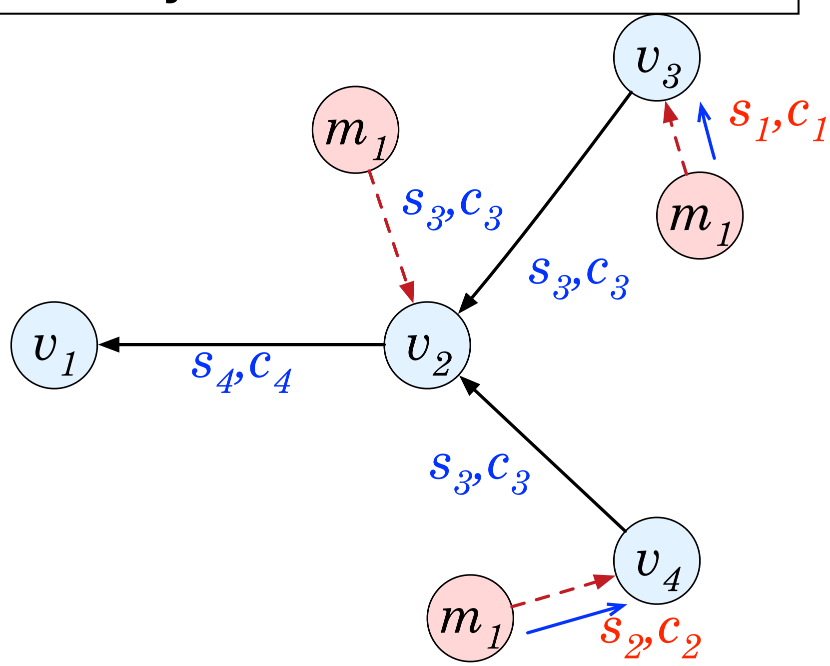
Efficient Bandwidth Reservation Techniques

Technique 3

For a set $\{(m_r, v_i, f_q) | v_i \in \hat{\mathbf{m}}_r\}$, which is the set of transmissions for flow f_q from a mobile node m_r to the potentially associable infrastructure nodes, any subset of this set can be combined.

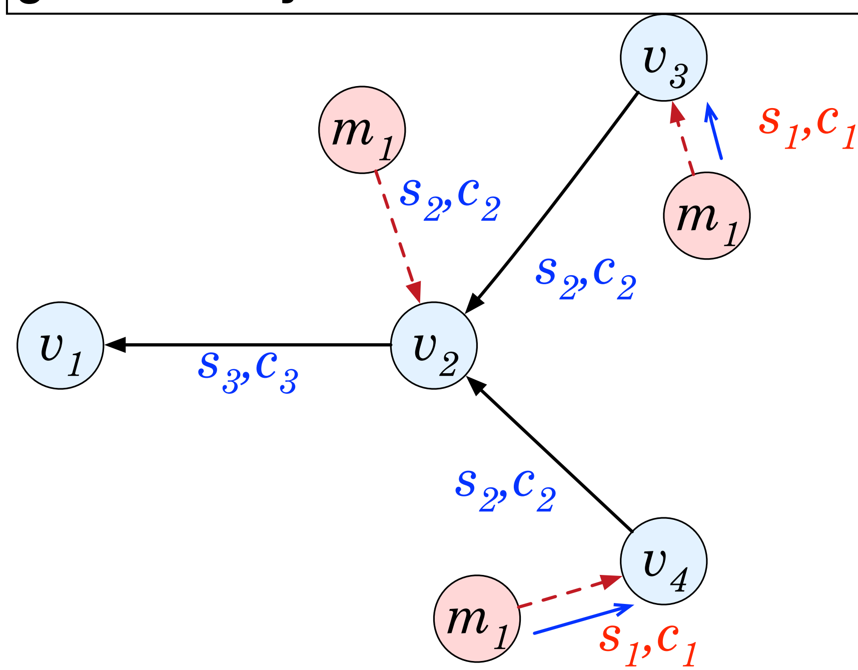
Without Technique 3

4 different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1



With Technique 3

3 different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1



Efficient Bandwidth Reservation Techniques

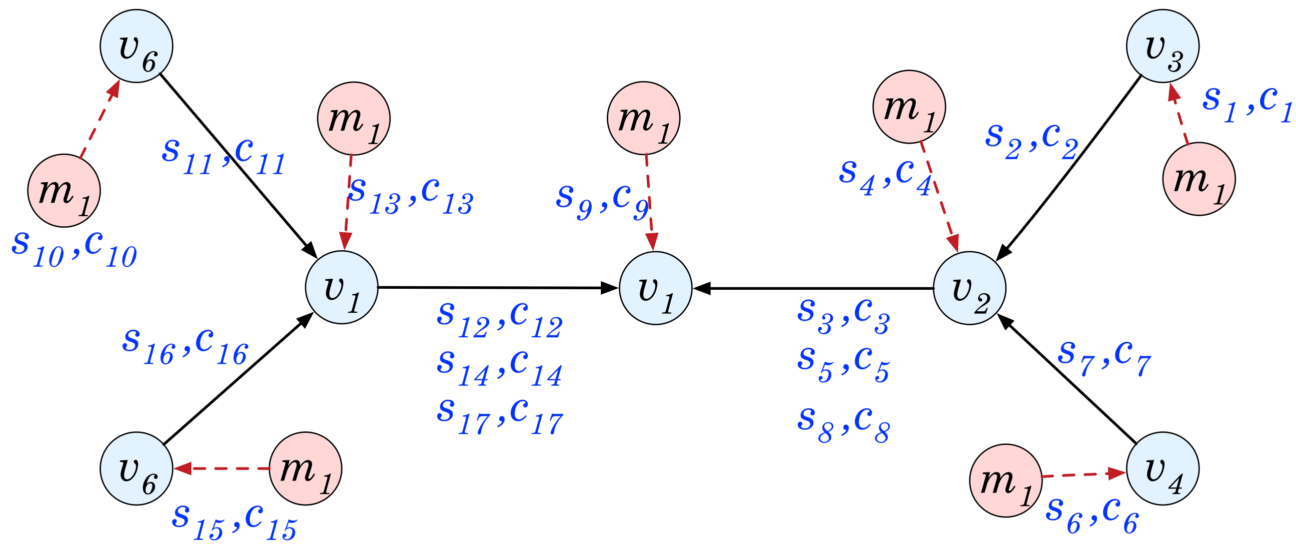
Technique 4

When Technique 1 is applied and the upstream graph is a spanning tree, a released transmission (w, z, f_q) can be combined with any scheduled transmission (x, y, f_q).

Efficient Bandwidth Reservation Techniques

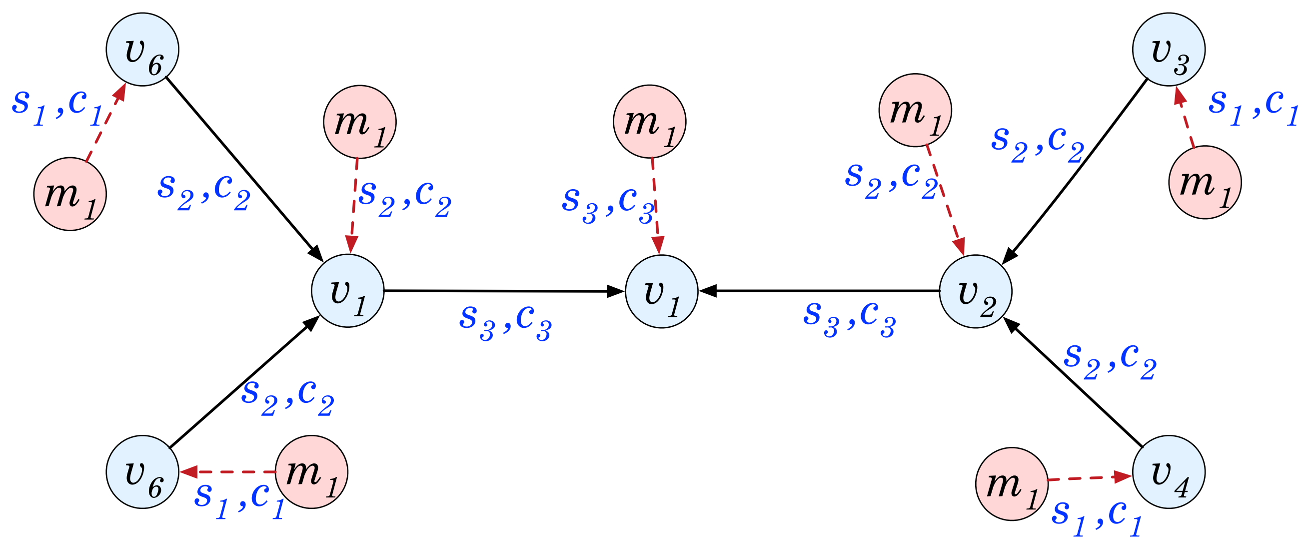
No Technique Employed

17 different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1



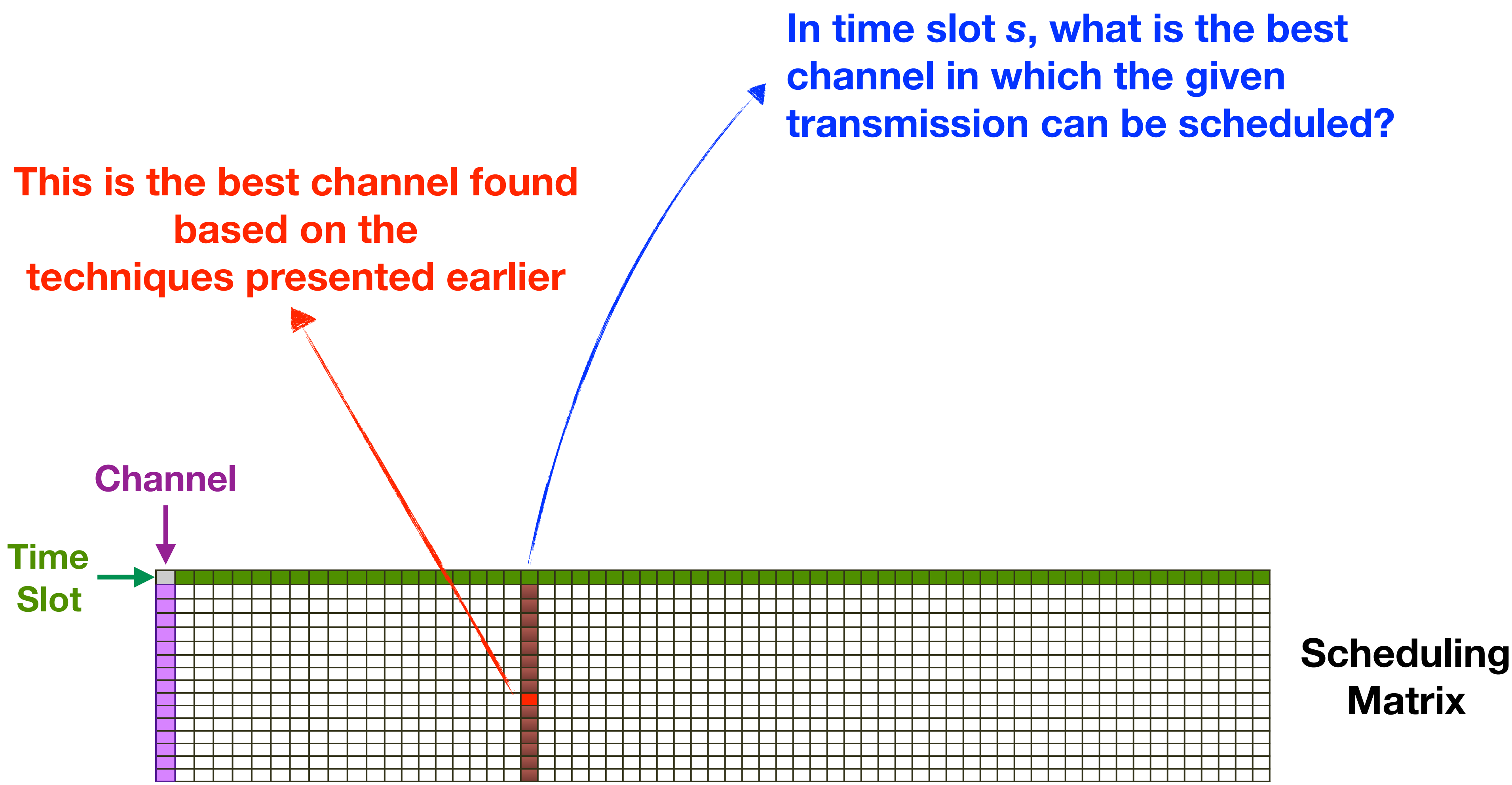
All Techniques Employed

3 different slot-channel combinations are required to forward a flow f_1 generated by mobile node m_1

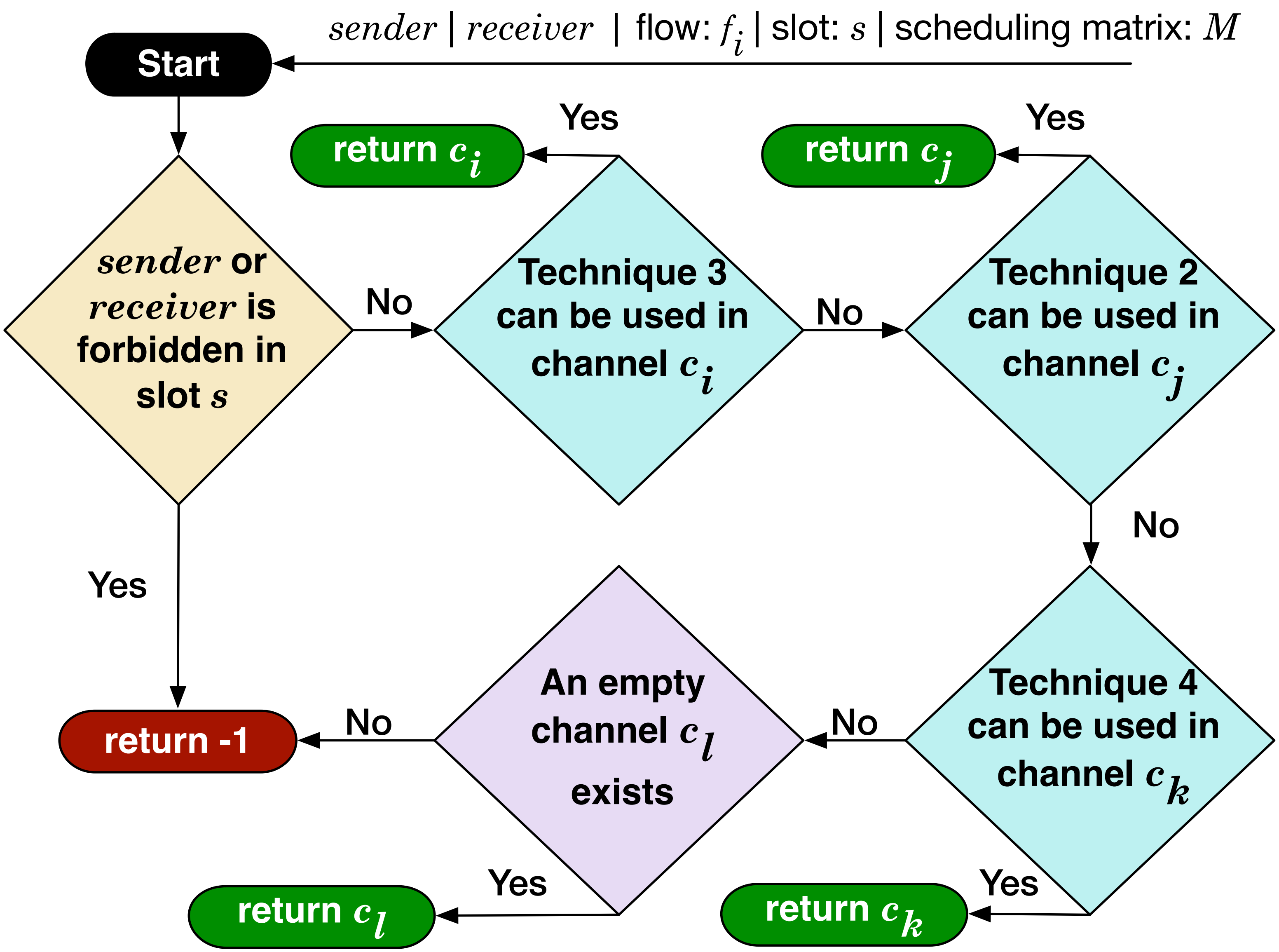


Channel Search Algorithm (CSA)

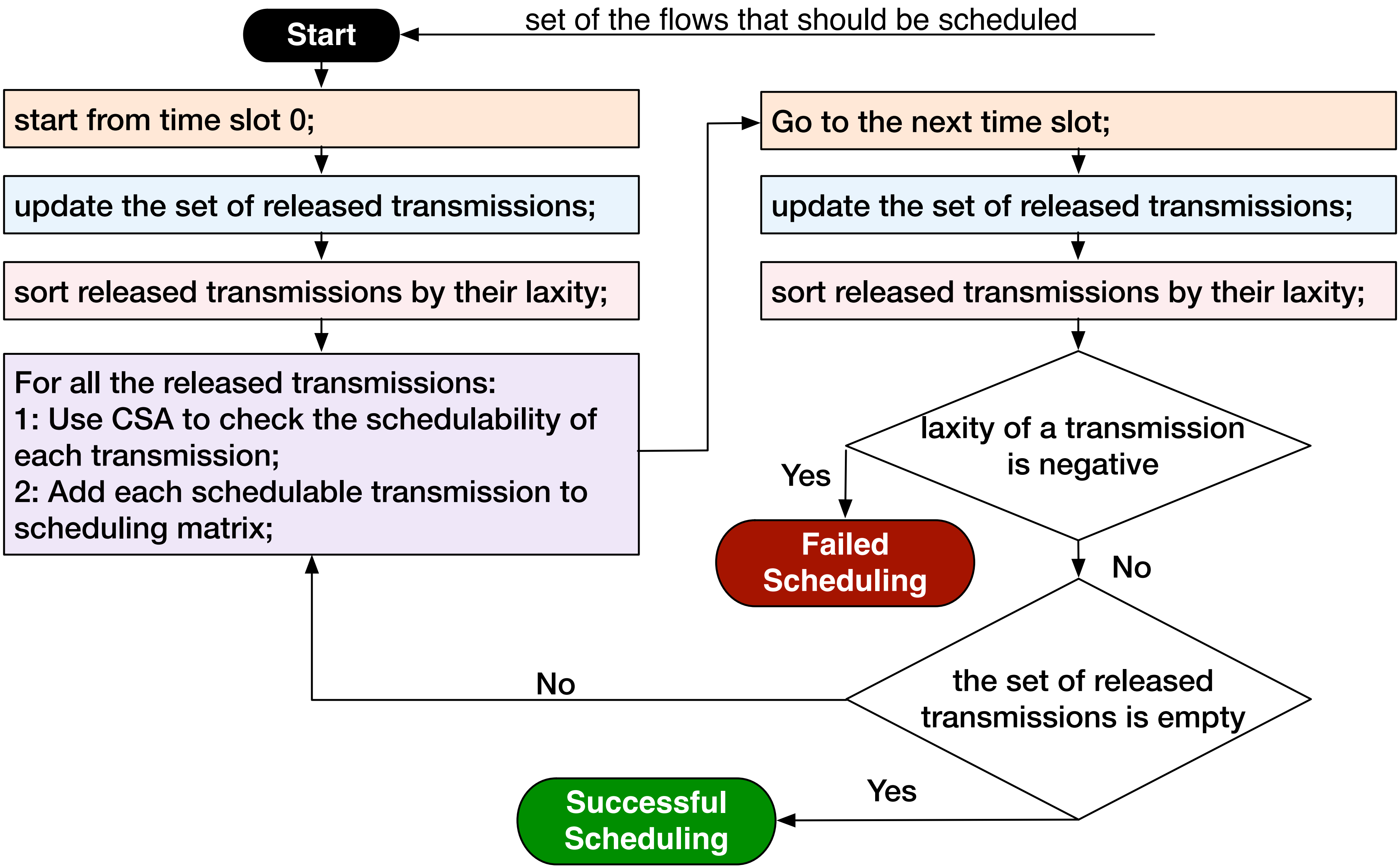
Having a transmission (w, z, f_i), the ***channel search algorithm (CSA)*** finds the best cell (channel) in a given time slot to schedule this transmission.



Channel Search Algorithm (CSA)



Mobility-Aware Scheduling Algorithm (MASA)



Mobility-Aware Scheduling Algorithm (MASA)

Algorithm 2: Mobility-Aware Scheduling Algorithm (MASA)

Input: **F**: set of the flows that should be scheduled
Output: generates scheduling matrix $\mathcal{M}[C][T]$ if the scheduling was successful, otherwise returns "unsuccessful"

begin
  $T \leftarrow$ least common multiplier of flows' periods;
  $\Theta \leftarrow \emptyset$; $\Theta_{rel} \leftarrow \emptyset$; $\Theta_{new\_sch} \leftarrow \emptyset$;
  $s \leftarrow 0$;
  `updRelTrans`$(s, \mathbf{F}, \Theta, \Theta_{rel}, \Theta_{new\_sch})$;
  Sort Θ_{rel} in ascending order of laxities;
  while $\Theta_{rel} \neq \emptyset$ **do**
    for $index \leftarrow 1$ *to* $|\Theta_{rel}|$ **do**
      $(w, z, f_i) \leftarrow$ the first transmission in set Θ_{rel};
      $c_j =$ `CSA`$(w, z, f_i, s, \mathcal{M}[C][T])$;
      if $c_j \neq -1$ **then**
        `addSchedule`$(w, z, f_i, s, c_j, \Theta_{rel})$;
    $s \leftarrow (s + 1) \bmod T$;
    `updRelTrans`$(s, \mathbf{F}, \Theta, \Theta_{rel}, \Theta_{new\_sch})$;
    Sort Θ_{rel} in ascending order of laxities;
    for *every transmission* (x, y, f_i) *in* Θ_{rel} **do**
      if `laxity`$(x, y, f_i, s) < 0$ **then** **return** unsuccessful ;
  return $\mathcal{M}[C][T]$;

Evaluate the schedulability of released transmissions in the order of their laxity.

Call CSA to find the best channel through applying the presented rules

Add schedule to the scheduling matrix

Update the set of released transmissions when a new time slot is considered.

Scheduling fails if the the deadline of a flow cannot be satisfied

Mobility-Aware Scheduling Algorithm (MASA)

Procedure `updRelTrans` $(s, \mathbf{F}, \Theta, \Theta_{rel}, \Theta_{new\_sch})$
  for *every flow* f_q *in* $\mathbf{F}$ **do**
    if $s \bmod p_q = \phi_q$ **then**
      $m_o \leftarrow$ the mobile node generating flow f_q;
      for *every* v_j *in* $\hat{\mathbf{m}}_o$ **do**
        $\Theta_{rel} \leftarrow (m_o, v_j, f_q)$;
        $\Theta \leftarrow$ links on the path from v_j to v_{root} ;
      for *evey transmission* (v_l, v_n, f_q) *in* Θ **do**
        remove duplicates of the transmission;
  for *every link* (w, z, f_i) *in* Θ_{new_sch} **do**
    if (*transmission* (z, y, f_i) *exists in* Θ) and
    (*no transmission* $(*, z, f_i)$ *exists in* $\Theta \cup \Theta_{rel}$) **then**
      $\Theta_{rel} \leftarrow (z, y, f_i)$;
      remove (z, y, f_i) from Θ;
    remove (w, z, f_i) from Θ_{new_sch};
  return;

Procedure `addSchedule` $(w, z, f_i, s, c_j, \Theta_{rel})$
  $\mathcal{M}[c_j][s] \leftarrow (w, z, f_i)$;
  remove (w, z, f_i) from Θ_{rel};
  return;

Procedure `laxity` (x, y, f_i, s)
  $s' = s \bmod p_i$;
  if $s' \geq \phi_i$ **then** **return** $d_i + \phi_i - s' - h_x$;
  else **return** $d_i + \phi_i - (s' + p_i) - h_x$;

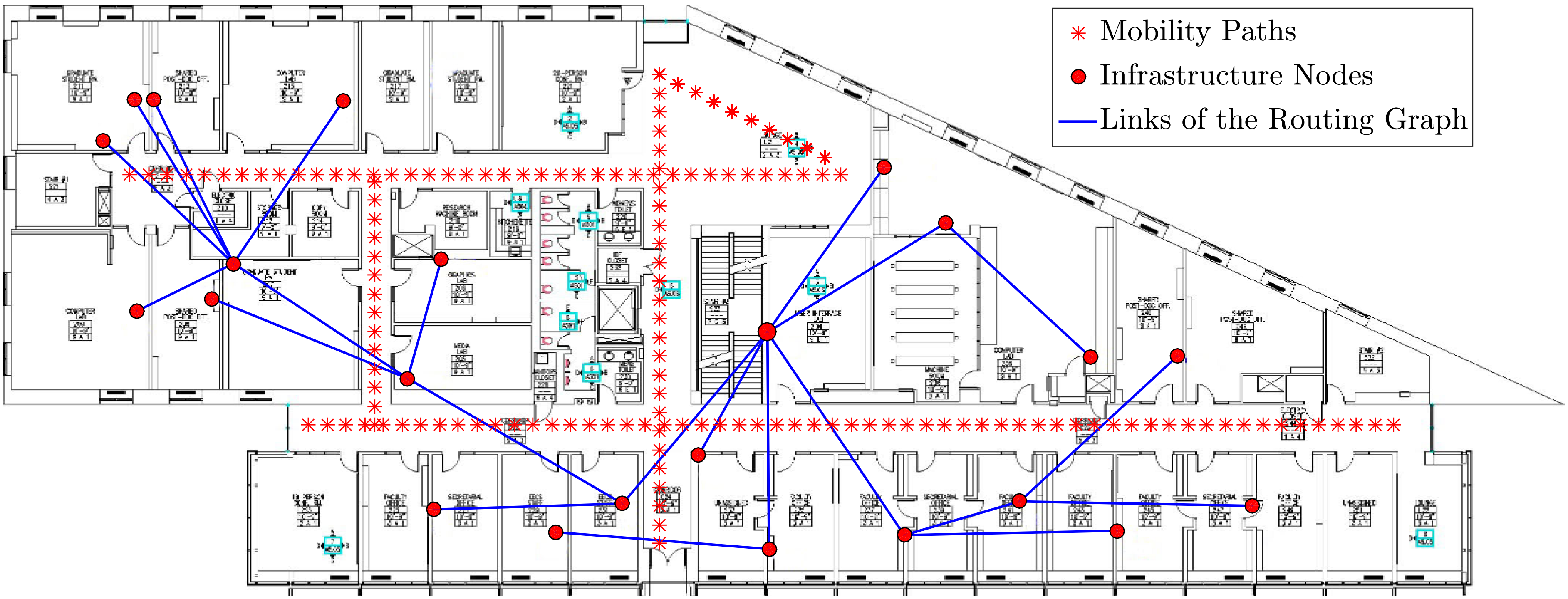
If a flow generates a new packet in this time slot, then add all the required transmissions.

Update the set of released transmissions considering the newly scheduled transmissions.

Add a transmission to the scheduling matrix.

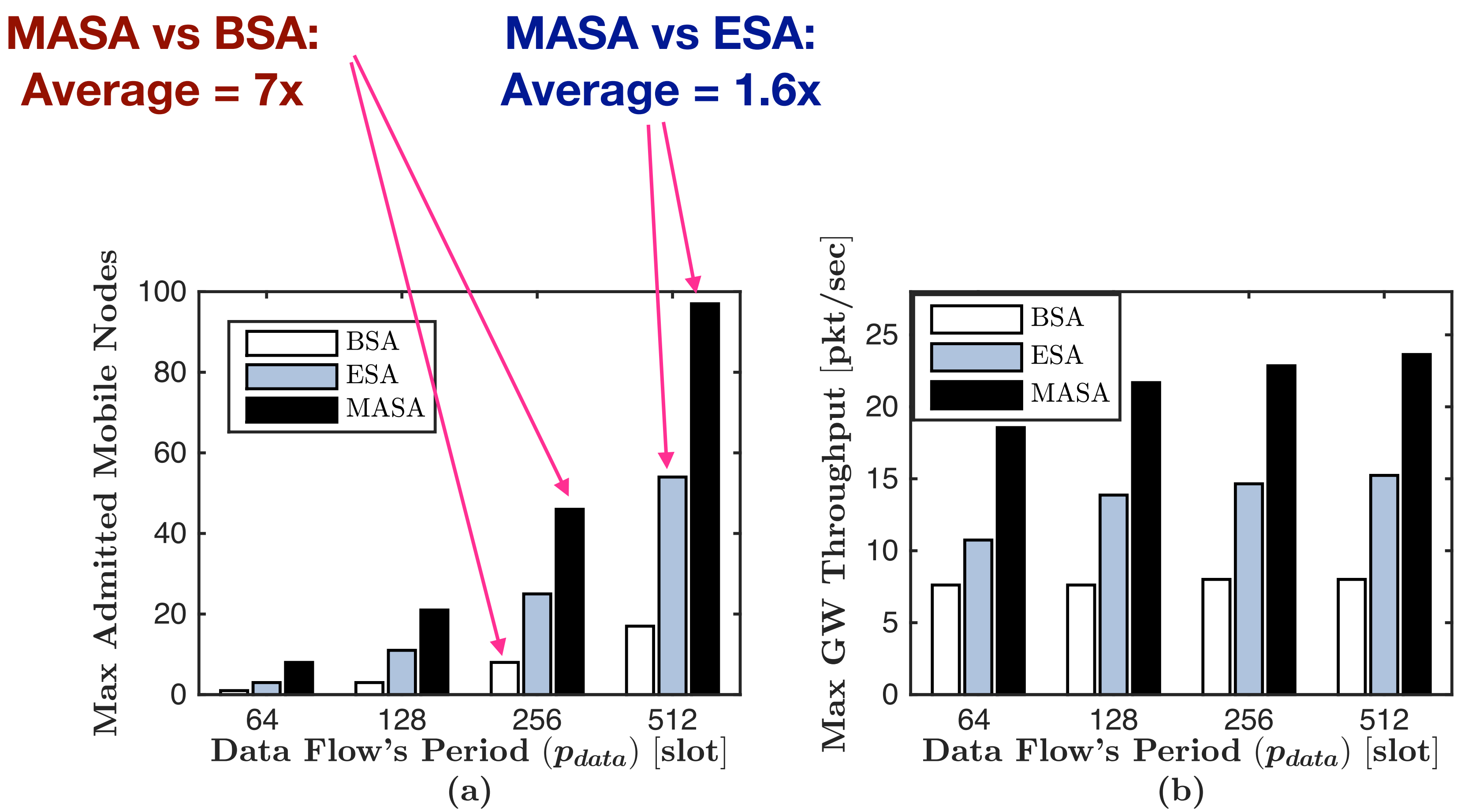
Compute the laxity of a transmission in time slot *s*.

Configuration



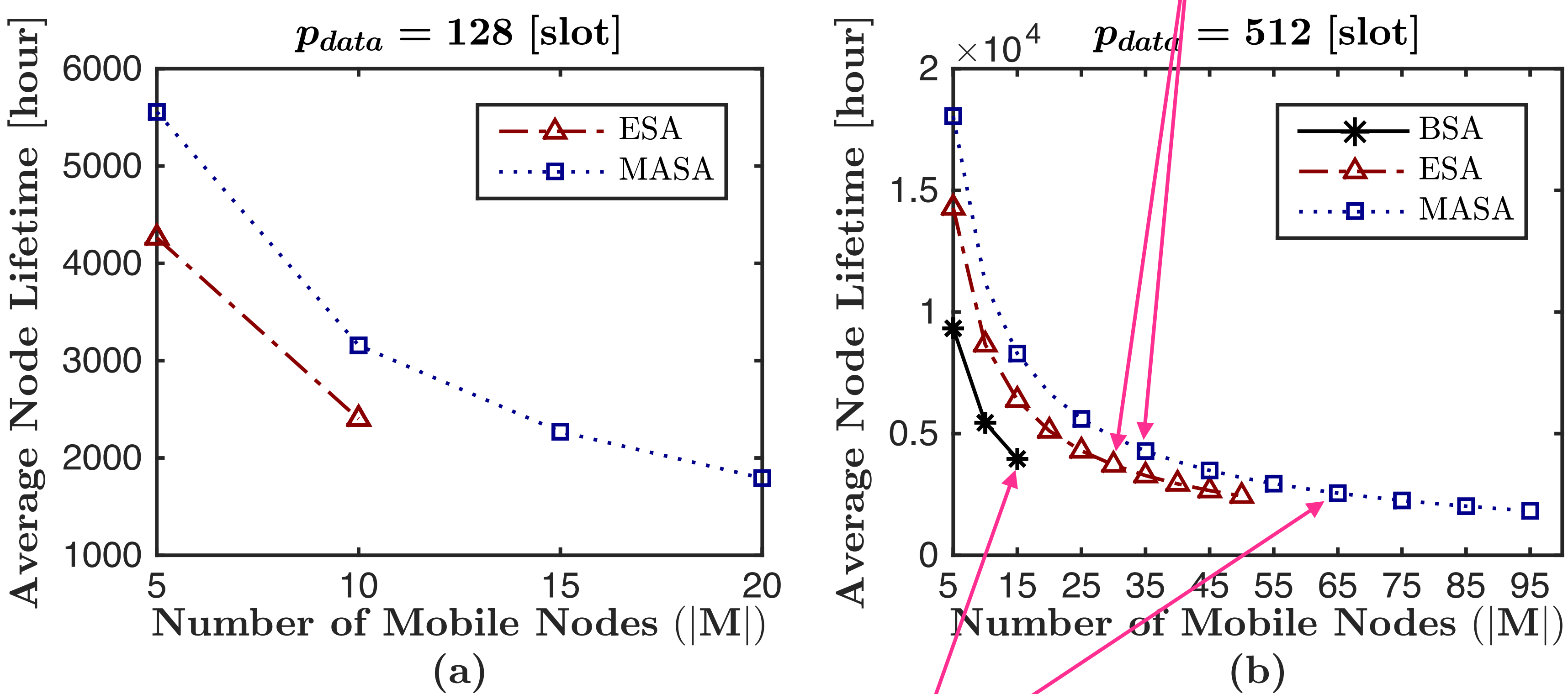
$p_{beac} = d_{beac} = p_{rpt} = d_{rpt} = p_{ctr} = d_{ctr} = p_{req} = d_{req} = 512$
$\phi_{beac} = \phi_{rpt} = \phi_{ctr} = \phi_{req} = 0$
$p_{data} \in \{64, 128, 256, 512\}$ | $\phi_{data} \in [0, p_{data} - 1]$ | $d_{data} = p_{data}$
Time Slot Duration = 10ms (as defined by the WirleessHART standard)
Packets: 802.15.4 compatible - maximum 127 bytes
Battery: 2500mAh 3V | Radio Transmission Power = 0dBm

Scalability



Network Lifetime

MASA vs ESA: Average = 110%



MASA vs BSA: Average = 30%

Acknowledgement

Behnam Dezfouli
University of Iowa

Marjan Radi
University of Iowa

Octav Chipara
University of Iowa

This work was supported by
the National Science Foundation and
the Roy J. Carver Charitable Trust