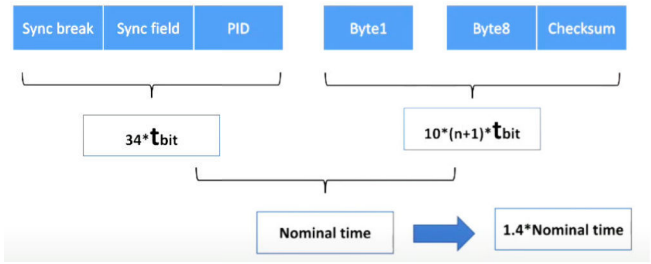
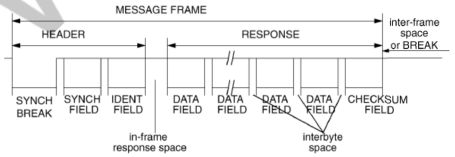
LIN Scheduling

Delay 40 %
Inter-frame sequence
Inter-Byte Sequence



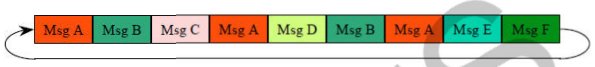
LIN protocol offers message timing predictability

- Time Triggered Approach
- Message Length is known
  - Number of transmitted data bytes is known
    - Minimum length can be calculated
  - Each Message has length budget of 140% of it's minimum length
    - maximum allowed length is known
    - distance between beginning of two messages

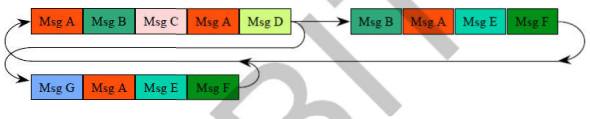


LIN protocol offers message timing predictability

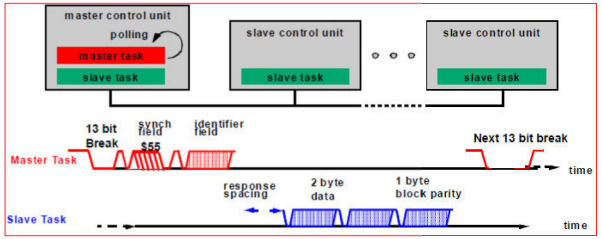
– Message sequence is known
  - Master uses scheduling table



– Use of different scheduling tables is possible
- Provides Flexibility



Data Transmission



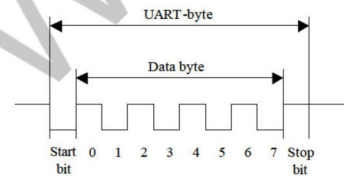
BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

LIN (Local Interconnect Network)

- Low speed serial network designed for body control in automotive applications
- Physical layer based on ISO 9141 (the K-line).
  - Single wire plus ground
  - Voltage level signaling
  - Dominant/recessive bit levels
  - Max 40 m wire length
  - 1-20 kbit/s
- Data Link
  - Media Access Control: Master/slave
  - Serial asynchronous byte oriented communication
- UART compatible
  - 64 uniquely identified messages
- Transport layer for diagnostic support
- Includes application layer development framework

Technical overview

- a single master and up to 16 slaves
- no arbitration
- deterministic traffic behavior and guarantees the latency times
- single wire
- Maximum data rate 20kbit
- byte-encoded according to the UART-protocol

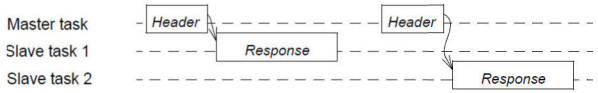


LIN frame

- Synchronization break,
- Synchronization byte,
- Identifier byte,
- Data bytes,
- Ckecksum byte.

LIN frame

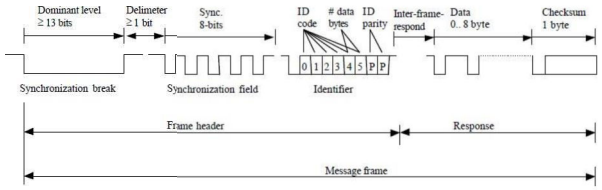
- A frame consists of a header (provided by the master task) and a response (provided by a slave task).



innovate achieve lead

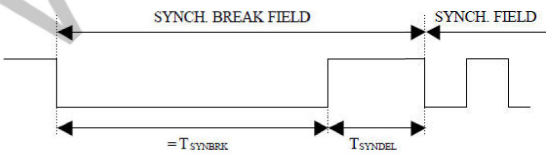
LIN message frame

- LIN message frame consists of a header and a response part.
- header: the SYNC-break, SYNC-field and the identifier- (ID) field



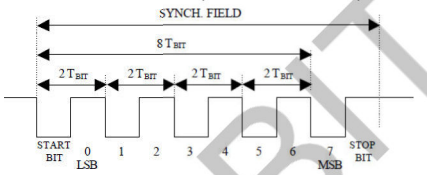
Synchronization break

- Marks the Beginning of a Message Frame
- consists of at least 13 bits of zeroes.
- the slaves are allowed to have a baud rate that differs with 15% to the masters
- Slave: to detect that a message is transmitted on the bus.
- For the master node, implemented in a microcontroller, the procedure of sending a SYNC-break involves some tampering with the UART-protocol.



Synchronization field

- Specific Pattern for Determination of Time Base
- master send a header, slaves synchronize their clocks every time a new message is received.
- Master: accurate resonator as a time reference.
- Slaves: synchronizes its clock from the falling edge of the start bit to the bit 7 of the SYNC-byte and divides it by 8.



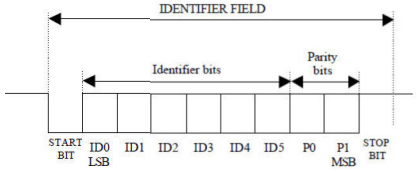
Identifier field

- The slave nodes on the network are addressed by the ID- field.
- Message Identifier: Incorporates Information about the sender, the receiver(s), the purpose, and the Data field length.
  - 2 Parity Bits protect the highly sensitive ID-Field.
  - Length 6 Bits, 64 Message Identifiers are possible.
  - 4 classes of 1/2/4/8 Data Bytes. The length coding is in the 2 LSB of the ID-Field. Each class has 16 Identifiers.

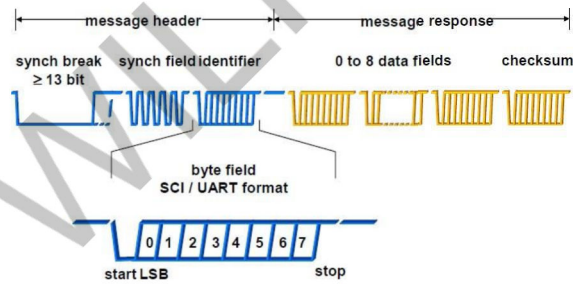
ID range	Frame Length
00-31 0x00-0x1F 000000-011111	2
32-47 0x20-0x2F 100000-201111	4
48-63 0x30-0x3F 110000-111111	8

Identifier

- The identifier field is sent by the master node to all LIN nodes
- This identifier normally contains one of 64 different values and includes 2 parity bits in the 8 bit data
- The identifier is normally associated with a collection of signals that are subsequently transmitted on the LIN bus
- In a specific case this can initiate SLEEP mode in the LIN slave nodes – in this case no further data is transmitted on the LIN bus

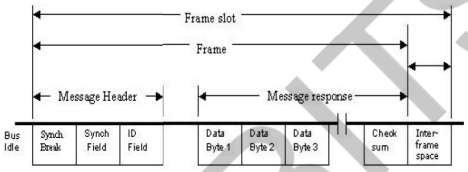


LIN Message Frame



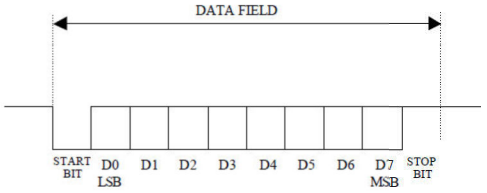
response

- The response contains one to eight data bytes and one checksum byte.
- The slave task is connected to the identifier and receives the response, verifies the checksum and uses the data transport.
- Messages are created when the master node sends a frame containing a header.
- The slave node(s) then fills the frame with data depending on the header sent from the master.



Data byte

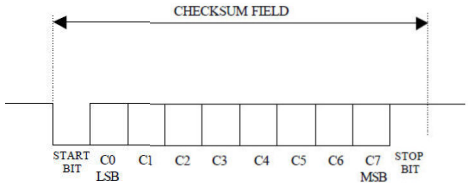
- The data length is defined by ID5 and ID6 from the identifier field.
- It can be 2, 4 or 8bytes.



BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

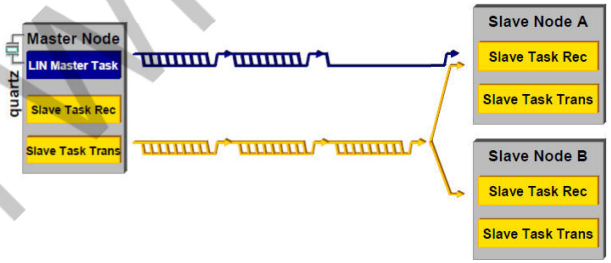
Cheksum byte

- The checksum (CRC) is computed only on the data field.
- All other fields are not included.



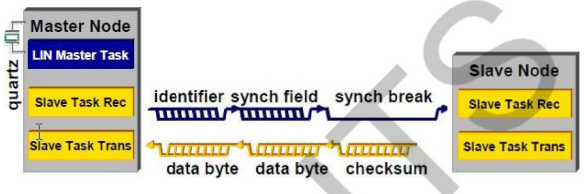
LIN Communication – Data from Master to Slave(s)

- a command frame (CMD frame).
- master sends a complete message frame for one ore more slaves



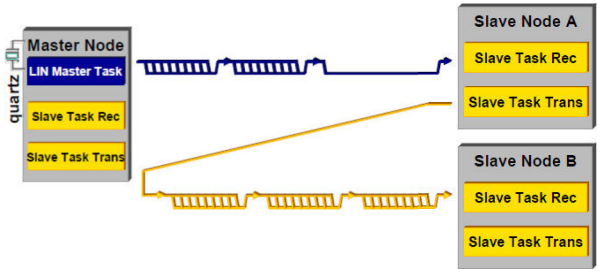
LIN Communication – Data from Slave to Master

- request frames (REQ frame).
- master requests a response from a specific slave (polling)



LIN Communication – Data from Slave to Slave

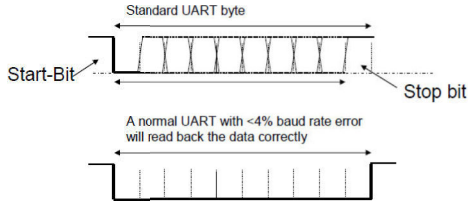
- information sent between sensor and actuator.
- one slave sends its response to one or more slaves



innovate achieve lead

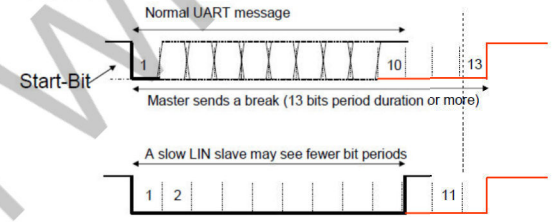
Frame Synchronisation - UART

- Initial conditions: +/- 4% baud rate accuracy relative the transmitting source
- A standard transmission of data will require matched send and receiver baud rates



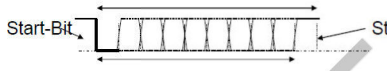
Frame Synchronisation - LIN

- Initial conditions: +/- 15% baud rate accuracy relative the the LIN master transmitting the synchronisation frame
- A synch break must be at least 13 bit periods in duration to allow for

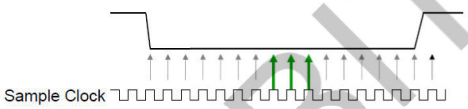


Bit-Synchronisation

- A start bit transition to a low logic level (dominant) indicates a start of a byte, least significiant first and completing with a logic high level (resessive) bit to indicate the STOP bit

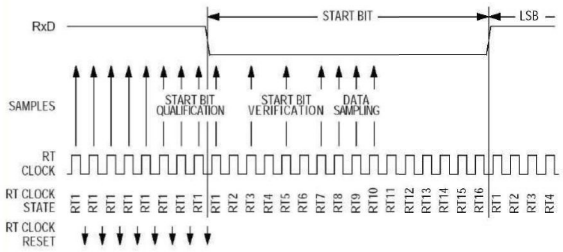


Data is sampled in the middle of the bit field:



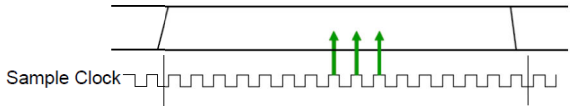
Bit Sampling

- Within a UART, clocked high speed sampling is used to aquire bit state



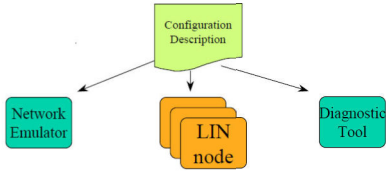
Bit-Synchronisation

- After recognition of a Low level in the start bit, the data is sampled at a rate 16 times the bit rate expected. The middle 3 samples must all agree for an error free reception of the data.
- A stop bit is expected after 1 start bit and 8 data bits in a typical message



Network Configuration

- LIN Concept includes configuration interface:
  - LIN description file describes complete LIN network and also contains all information necessary to monitor the network.
  - LIN Configuration Language Description is part of the LIN Specification and gives tools the possibility to configure the network and the nodes, diagnose the traffic, and/or simulate missing nodes.



The Work-flow

– **Data Input**
  - Definition of objects
  - Definition of relations between the objects
– **Data Processing**
  - Logical Signal Mapping
  - Signal Packing (Frame Editor/Frame Compiler)
  - Timing Analysis
– **Data Output**
  - Configuration file generation
  - Various optional customer-defined post-operations

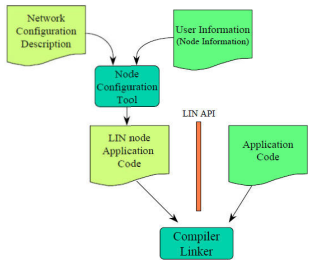
Application Programmers Interface

Standard API

•simplifies design of Application Code

•opens up the market for

competition.



LIN Tools by VCT

- **LIN Database Manager (LDM)**

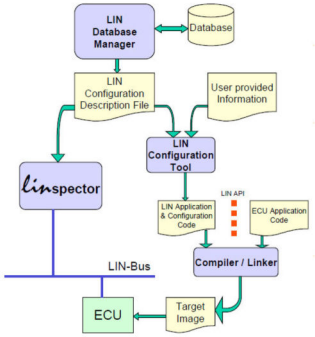
The LDM is a standalone offline tool, providing a user-friendly Windows interface for logically describing and configuring LIN systems at a high abstraction level.

- **LIN Configuration Tool (lcfg) and**

LIN Application Programmer's Interface (API) The LIN API provides the embedded SW developer an abstraction from details of information transfer. Together with the LIN Configuration Tool and an optimized embedded SW package the user gets correctness and quality together with efficiency and reconfiguration flexibility.

- **LINspector**

a highly flexible tool for testing and verifying communication for compliance with the LIN standard.



FAULT-TOLERANT

The role of fault tolerant communication is to address the higher communication safety needs. The fault tolerant communication increases the dependability of signal transmission between ECUs, sensors and actuators.

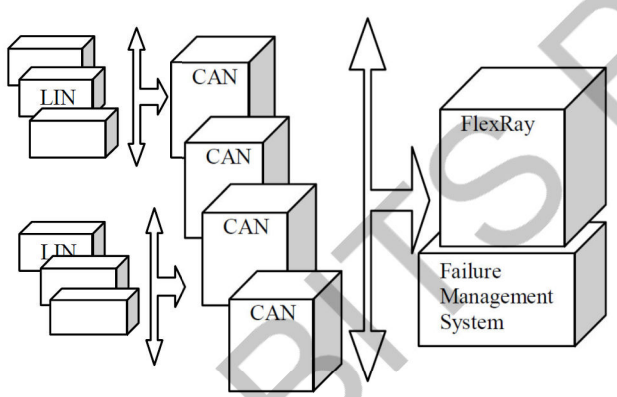
Faults:

- Timing errors,
- corruption of data during transmission,
- loss of frames
- loss of a communication channel

In order to provide fault tolerant communication, redundancy techniques are applied:

- information redundancy,
- Functional redundancy,
- time redundancy,
- structural redundancy.

Integrated Fault Tolerant Bus System



Integrated Fault Tolerant Bus System

- The integrated system supports fault tolerance using redundant networks.
- LIN bus systems are combined with extra redundant LINs that are used for a LIN master to send multiple messages to LIN clients. In case the clients do not receive any messages in a certain time, duplicated messages on a redundant LIN bus are reused for the waiting clients.
- Fault-tolerant CAN bus systems are implemented in the same way as the LIN bus does.
- FlexRay bus system which is used as the backbone for the integrated bus systems communicate with CAN buses.
- When LIN needs to communicate with other higher bus systems, LIN interconnects with CAN first and then communicate thru FlexRay bus.
- MOST messages are exchanged to other vehicle buses thru FlexRay bus, and vice versa.

BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

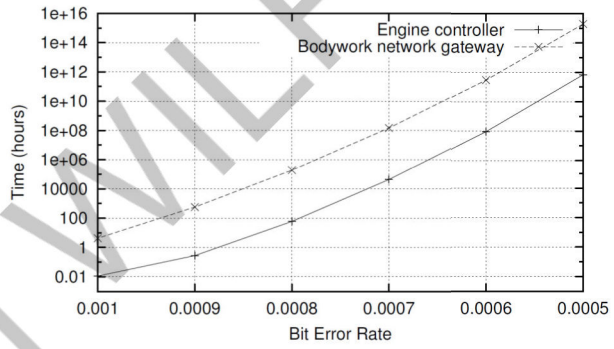
FAULT CONFINEMENT

The CAN protocol possesses fault confinement mechanisms aimed at differentiating between short disturbances caused by electromagnetic interferences (EMI) and permanent failures due to hardware dysfunctioning.

CAN fault confinement mechanisms are conceived to disconnect defective nodes from the network and prevent them from perturbing the whole network.

To estimate the probability of such events which can be achieved through the knowledge of the average hitting time of the bus-off state and of the variance of the bus-off hitting times.

FAULT CONFINEMENT



Average hitting times of the bus-off state for the engine controller and the bodywork network gateway with the Bit Error Rate (BER) varying from 0.0005 to 0.001 .

Event Triggered LIN Bus Latency

A LIN bus operates at 19.2 kbps. A single event-triggered LIN frame consists of a header (34 bits) and a payload (8 bytes). Calculate the transmission latency for one LIN frame in milliseconds (ms).

Solution:

- Total bits:

  34 bits + 8 × 8 bits = 34 + 64 = 98 bits

  Transmission latency:

  (98 bits / 19.2 × 10^3 bps) =0.005104 s = 5.104 ms

LIN Message Cycle Time

In a LIN network at 10.4 kbps, a periodic message consists of a header of 34 bits and payload data of 6 bytes. Calculate the transmission time (in milliseconds) for a single LIN message.

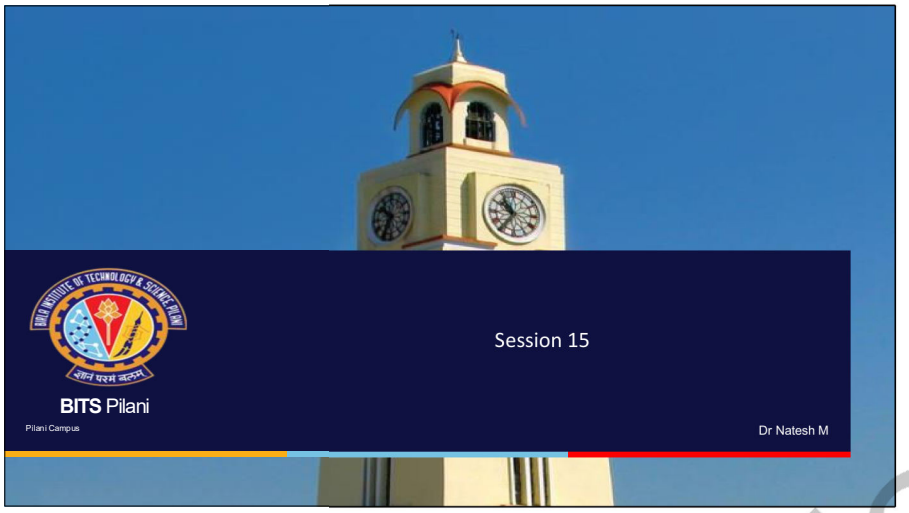
Solution:

- Total message bits:
  34 bits + (6 bytes×8) = 82 bits
- Transmission time:
  (82 bits / 10,400 bps) ≈ 7.8846 ms

Questions?

IMP Note to Self

BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES



IMP Note to Self

Start Recording

IMP Note to Students

- It is important to know that just login to the session does not guarantee the attendance.
- Once you join the session, continue till the end to consider you as present in the class.
- IMPORTANTLY, you need to make the class more interactive by responding to Professors queries in the session.
- **Whenever Professor calls your number / name ,you need to respond, otherwise it will be considered as ABSENT**

Geographical Routing

innovate achieve lead

BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

Contents:

Wireless sensor networks (**WSNs**) refer to networks of spatially dispersed and dedicated sensors that monitor and record the physical conditions of the environment and forward the collected data to a central location.

Geographic routing (also called **georouting** or **position-based routing**) is a routing principle that relies on geographic position information. It is mainly proposed for wireless networks and based on the idea that the source sends a message to the geographic location of the destination instead of using the network address.

Introduction

- Routing in wireless sensor networks is performed in a cooperative way, i.e. each node relies on its neighbors to forward its packets towards the network sink. All sensor nodes participate in the routing procedure acting as routers.
- This intricacy of the sensor networks implies that the routing functionality is distributed over all network nodes, which however have limited energy, memory and processing capabilities.
- Although a plethora of routing protocols for ad hoc and sensor networks has been proposed, geographical routing (GR) targets the efficient support of mobility and scalability while it turns out to be invulnerable against a set of security attacks when security comes into play.
- GR outperforms other solutions because it addresses at the same time scalability and mobility.
- The concept is to use geography for routing instead of measuring hops and flooding the current state of all network nodes to create a map.

The idea is that each node is characterized by its coordinates and each time a node needs to send a packet, it will forward it to the neighboring node that is closer to the destination than all the rest nodes.

To perform this decision, it is required to keep geographical information (the location) for each neighbor in its routing table.

This information is gathered through the so-called **"BEACON"** messages that each node periodically transmits announcing its location.

innovate achieve lead

Contd..

This localized operation has the following important advantages:

It facilitates the mobility support.

It is scalable.

It introduces minimal routing overhead.

GR is a proactive routing protocol since beacons are emitted periodically. The disadvantage of proactive routing protocols is that routing messages are exchanged even when no traffic to route exists.

Contd..

The realization of GR algorithms requires that each node is aware of its coordinates either due to the existence of a GPS (Global Positioning System) device in each sensor node or based on the implementation of some localization functionality.

This aspect has attracted much attention since it directly affects the cost of the sensor node and the achieved security

GEOGRAPHICAL ROUTING PROTOCOLS

- Greedy Perimeter Stateless Routing (GPSR)
- Geographical and Energy Aware Routing (GEAR)
- Energy Aware Greedy Routing (EAGR)
- Probabilistic Geographic Routing (PGR)
- Energy-aware Geographical Forwarding using Adaptive Sleeping (EnGFAS)
- Directional Location-based Randomized Routing (DLR)
- On Demand GPSR (OD-GPSR)
- Blind Geographic Routing (BGR)

Greedy Perimeter Stateless Routing (GPSR)

The most widely cited geographical routing protocol is the Greedy Perimeter Stateless Routing (GPSR) algorithm for wireless sensor networks initially presented by Karp and Kung (2000) in [1][1].

The algorithm consists of two methods for forwarding packets:

1. Greedy forwarding.
2. Perimeter forwarding.

innovate achieve lead

BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

Contd..

Greedy forwarding: the source node transmits the packet towards the neighbours that is closest to the destination (and furthest from the source node), which is the locally optimal choice of next hop.

Perimeter forwarding: When greedy forwarding is not possible and the packet has not reached its destination, i.e. there is a void between the node and the destination, perimeter forwarding is performed to travel around the void. This is achieved based on the proactive calculation of planar graphs.

Geographical and Energy Aware Routing (GEAR)

Energy consumption related improvements have been pursued by Yu, Estrin, and Govindan (2001) in [2] where "geographical and energy aware routing" (GEAR) is proposed.

In GEAR, the next hop is decided taking into account the value of the learned cost function for each neighbor apart from its location.

This "learned cost" function reflects the available (remaining) energy of each node, thus allowing for better load balancing among neighbors.

Energy Aware Greedy Routing (EAGR)

Haider, Javed, and Khattak (2007) in [14] have evaluated their Energy Aware Greedy Routing (EAGR) protocol which also relies on energy information to prolong the lifetime of the network.

In this work, nodes with remaining energy below a predefined threshold are considered as dead and are excluded from the list of candidate for forwarding neighbours.

Probabilistic Geographic Routing (PGR)

A combination of geographical information with energy information is also adopted by T. Roosta, M. Menzo, and S. Sastry (2005) in [3].

In order to forward a packet, the node selects a set of candidate nodes based on geographical information to guarantee that the packet will be forwarded and not travel backwards. These candidate nodes are then assigned a probability proportional to their residual energy and link reliability. Finally the next hop is decided taking into account these probabilities following a simple roulette wheel algorithm.

The performance of the algorithm is evaluated using the ns-2 simulation platform and is shown to offer higher throughput and longer system lifetime at the expense of slightly longer paths compared to GPSR.

innovate achieve lead

Energy-aware Geographical Forwarding using Adaptive Sleeping (EnGFAS)

Energy awareness is adopted by Shuhui Ma and Hong Ji (2006) in [18], where they present a cross-layer energy aware geographical Routing protocol.

In EnGFAS, nodes sleep not only to save energy but also to save collisions at the MAC layer.

Additionally, to avoid forwarding the fragments of a single application layer packet through different neighbours, the source node calculates whether a neighbour node has enough energy (more than Ethr) to forward the whole application layer packet. This energy threshold (Ethr) is defined based on the value in the 'duration' field of the RTS message of the MAC layer.

LOCATION INFORMATION

Although the straightforward solution is to equip each node with a GPS (Global Positioning System) device, it increases the cost and the size of the sensor node.

To overcome this drawback, localization algorithms have been designed and implemented. They allow for the calculation of the relative position of nodes using few nodes as anchors.

Although these algorithms require the exchange of messages and cooperation among the sensor nodes, which consumes bandwidth and transmission resources, the overall cost is significantly lower than that of the GPS device.

Localization techniques can be classified in two categories:

Triangulation localization

Free-range localization

Few nodes are assumed to know their location and broadcast it to all their one-hop neighbors. Based on three measurements, each node can calculate its own location comparing the signal strength from each neighbor or the time difference of the received messages.

Another option is to derive coarse grained location information applying heuristics based on the received beacon messages.

Contd..

Location inaccuracies in general lead to performance degradation while if location is intentionally falsified (by a malicious node), network collapse may be caused in geographically routed sensor networks.

In an attempt to deal with the localization issue, geographical routing protocols based on “Virtual coordinates” have been presented (e.g. [7], [8]).

innovate achieve lead

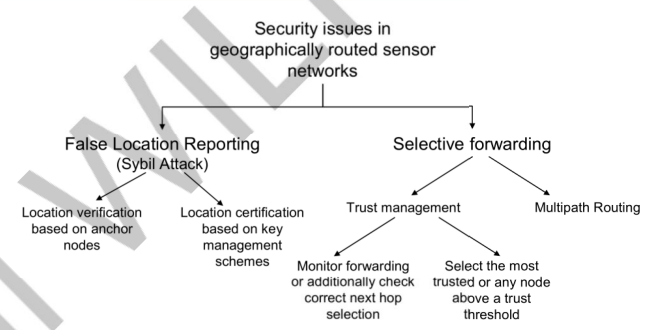
BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

SECURITY ISSUES IN GEOGRAPHICAL ROUTING

Wireless sensor networks are more prone to security attacks than legacy wired communication networks.

Geographical routing protocols are inherently less vulnerable to routing attacks mainly due to their local (almost stateless) operation.

Security Attacks



Defending Sybil Attack

To prevent Sybil attack, a key management scheme can be adopted. In [9], Hao et. al. report a number of key-based techniques that can be used, stressing however, that the limited processing capabilities of sensor node should be carefully taken into account.

Another way to prevent a sensor node from falsifying its location, is to verify its location challenging it. When the node receives the challenge, it should immediately reply to the verifier, through an ultrasonic channel, with a nonce that was included in the original challenge message. However, this solution requires extra hardware which may increase the cost and the size of the sensor node.

Defending Selective forwarding attack

A very easily implemented routing attack is selective forwarding or even blackhole attack. In the first, a node drops part of the packets it was assumed to forward while in the second it refuses to forward any received packet.

To defend against these attacks, the nodes should be capable of detecting malicious nodes and exclude them from their forwarding candidates list.

To detect this behavior it is necessary that each node checks whether the selected next hop neighbor has forwarded the packet either based on some type of acknowledgement or overhearing its transmissions. This way the trustworthiness of the neighbors can be defined and taken into account during routing decisions.

innovate achieve lead

Contd..

An enhancement is made to the GPSR where the next hop is decided taking account the trustworthiness of the neighbours apart from their location.

To evaluate the trustworthiness of each neighbour, each time node A selects node B as the next hop, it listens node's B transmission waiting (for a fixed time interval) to listen its own packet correctly forwarded.

$$T_i^{A,B} = \frac{S_i^{A,B} - F_i^{A,B}}{S_i^{A,B} + F_i^{A,B}}$$

This way, malicious node[illegible] and the packet delivery ratio of plain GPSR is improved by 30% for 50% of adversary nodes.

conclusion

In geographically routed wireless sensor networks, routing is performed on hop-by-hop basis where each hop is decided based on location information of all one-hop away neighbours.

This localized operation presents significant advantages which include the support of mobility, the support of high scalability while overhead and node requirements remain minimal.

To this end, geographical routing can be exploited to build scalable, dense, secure and energy-aware ad hoc wireless sensor networks supporting high mobility levels.

Numericals -GPSR

Design and implement a Geographic Routing - Gytar for a network of nodes with given coordinates, considering the impact of node mobility and varying transmission ranges on the routing efficiency. A list of node coordinates [(0,0), (3,4), (6,8), (1,1)] and a source node (0,0) and destination node is (6,8).

Solution

Source Node (0,0)

Destination node is (6,8).

Distance Calculation: The Euclidean distance formula will be used to calculate the distance between nodes: d= sqrt ((x2−x1)^2+(y2−y1)^2) =(*x*2−*x*1)2+(*y*2−*y*1)2.

Distance (destination node(6,8) to node(3,4)): sqrt((6-3)^2 + (8-4)^2) = sqrt (25) = 5

Distance (destination node(6,8) to node(1,1)): sqrt((6-1)^2 + (8-1)^2) = sqrt (74) = 8.60

So Node (3,4) is closest to the destination node so, the path selected for transmission will be as follows

[(0,0) -> (3,4) -> (6,8)]

innovate achieve lead

BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

Solution contd…

Gytar Routing Principle: Gytar routing is based on the concept of forwarding packets to the neighbor that is closest to the destination, which is a greedy approach to find the shortest path.

Transmission Range: The transmission range of each node will affect the connectivity and thus the routing decisions. Nodes within the transmission range of each other can directly communicate.

Numerical -GEAR

S (source) is at coordinates (0, 0) and D (destination) is at (10, 10).

Nodes A, B, C, and E are neighbors of S, with coordinates and remaining energy as follows:

- A: (2, 2), Energy: 80%
- B: (1, 1), Energy: 60%
- C: (3, 3), Energy: 90%
- E: (0, 1), Energy: 70%

We'll use a simple cost function: Cost = Distance / Energy.

How will the Geographical and Energy Aware Routing (GEAR) protocol be implemented in the above scenario?

Solution

S to Neighbors:

Cost(A) = sqrt((2-10)^2 + (2-10)^2) / 80 = 11.31 / 80 = 0.141

Cost(B) = sqrt((1-10)^2 + (1-10)^2) / 60 = 12.73 / 60 = 0.212

Cost(C) = sqrt((3-10)^2 + (3-10)^2) / 90 = 9.90 / 90 = 0.110

Cost(E) = sqrt((0-10)^2 + (1-10)^2) / 70 = 14.14 / 70 = 0.202

S forwards the packet to C (lowest cost).

Solution contd…

C to Neighbors (within recursive region):

Let us assume C's neighbors are:

F: (4, 4), Energy: 75%, Cost(F) = 0.15

G: (5, 5), Energy: 85%, Cost(G) = 0.12

H: (3.5, 3.5), Energy: 95%, Cost(H) = 0.08

C forwards to H (lowest cost).

- This process continues within the region until the data reaches D.

innovate achieve lead

BITS Pilani
Pilani | Dubai | Goa | Hyderabad | Mumbai
WORK INTEGRATED LEARNING PROGRAMMES

innovate achieve lead

Thank you

innovate achieve lead