Моделиране, Симулиране и Верификация на Безжични Мобилни Ad-Hoc Мрежи

Доц. Д-р Анна Лекова

ALEKOVA@ISER.BAS.BG

Институт по Системно Инженерство и Роботика

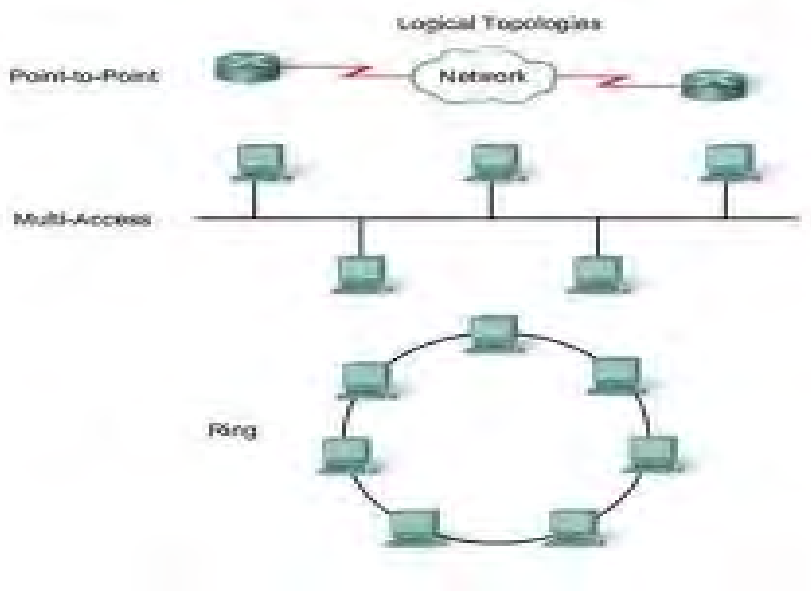
БАН

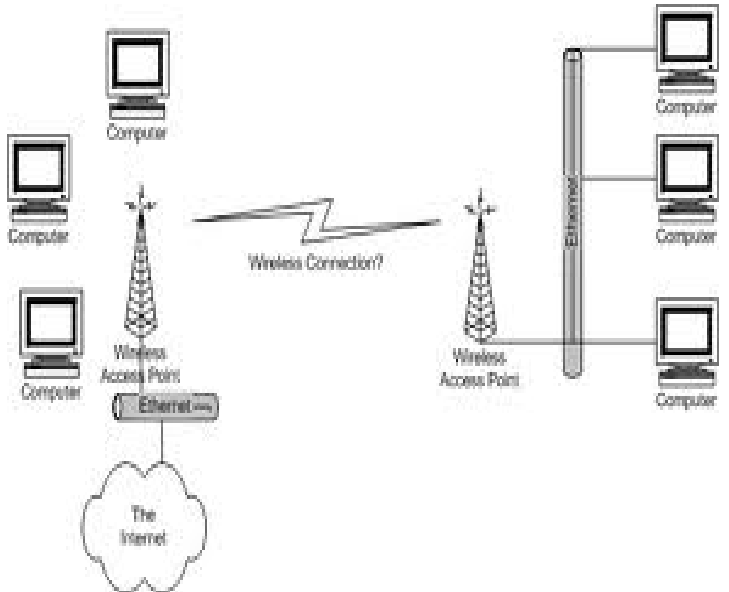
План на Лекциите

- ✓ Компютърни Мрежи. Въведение, Характеристики и Изисквания
- ✓ Мрежова Архитектура на Безжични Мрежи. IEEE 802.11 стандарти
- ✓ Безжични комуникации с ad-hoc инфракструктура. Мобилни Ad Hoc Мрежи (MANETs).
- ✓ Видове протоколи за маршрутизация в MANETs , традиционни и нови проблеми
- ✓ OLSR Протокол , алгоритъм за маршрутизация
- ✓ AODV Протокол , алгоритъм за маршрутизация
- ✓ Запознаване с UPPAAL
- ✓ Моделиране, симулиране и верифициране на система transmitter/receiver в UPPAAL
- ✓ Формален модел на R-OLSR в UPPAAL
- ✓ Инсталиране на UPPAAL . Курсова работа.
- ✓ Заключение

Компютърни Мрежи (КМ)
Въведение, Характеристики и Изисквания

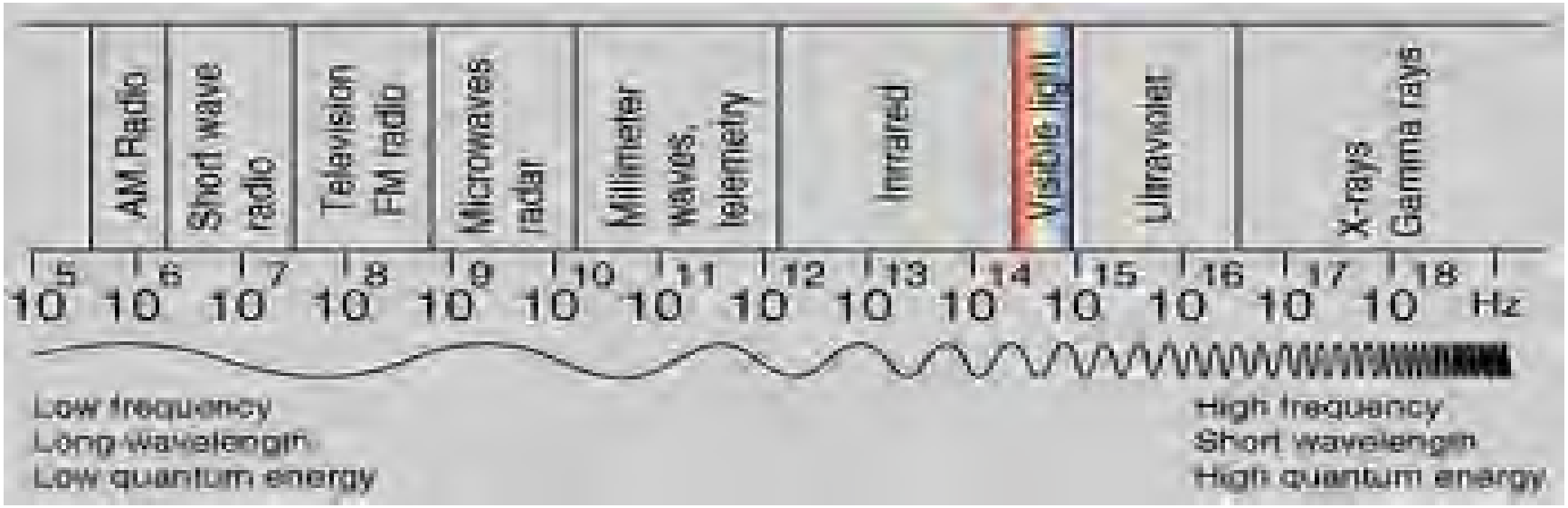
- КМ да осигури свързаност (connectivity) между компютърни устройства – възли (nodes)
- Свързаност на различни нива
- най-ниско ниво - КМ свързва възлите чрез физическа среда (physical medium) – връзка (link)
- Point-to-point, multiple-access, ring





Мрежови Връзки – реализирани на различни физически носители

- Телефонни линии, коаксиални кабели, оптични връзки и сигнали, разпространяващи се в пространството (space links) - AM, FM радио вълни, микровълни, инфрачервени
- Разпространяват Сигнали - електромагнитни вълни, честота и дължина на вълната
- Електромагнитен Спектър



Модулация

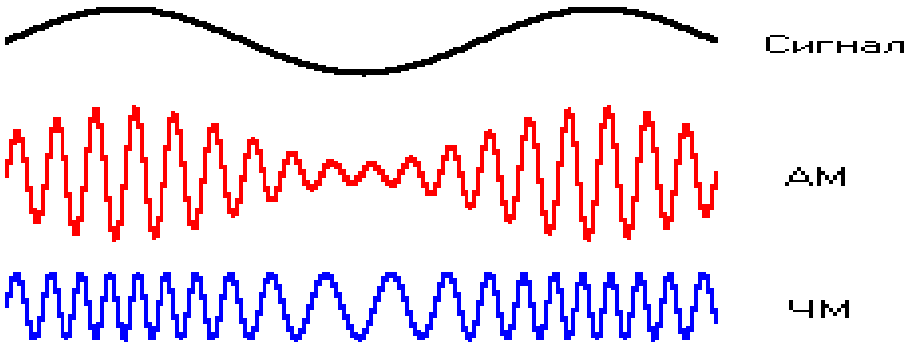
- Пренасяне на спектъра на сигнала, носител на информация във високочестотната област - изменение на параметрите на високочестотен сигнал, наречен носещ, под въздействието на сигнала, носител на информация, наречен модулиращ
- Различни видове модулации - изменя се някой от параметрите на носещия сигнал

Амплитудна модулация (АМ)

Честотна модулация (FM)

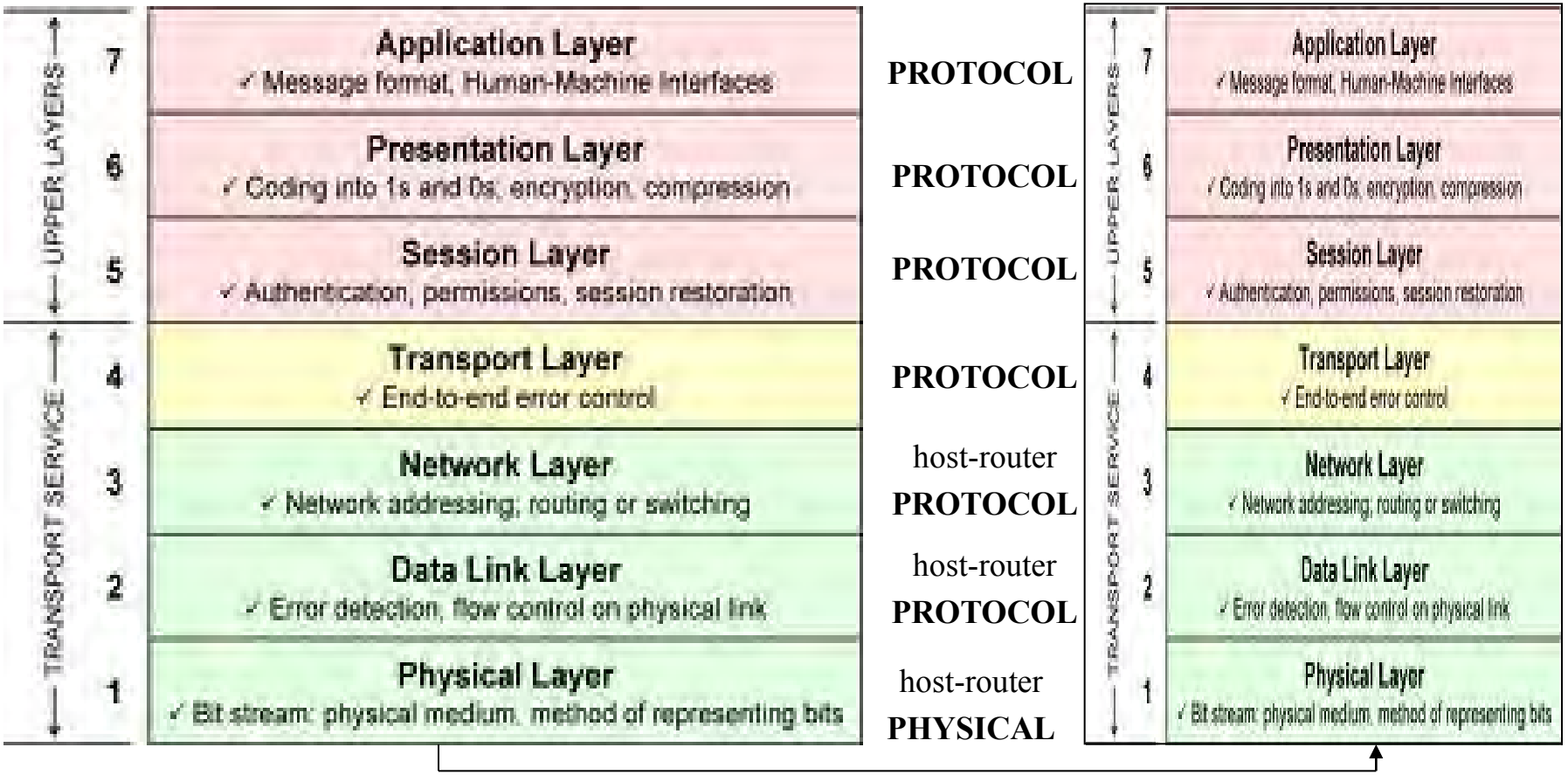
Фазова модулация

Импулсна модулация



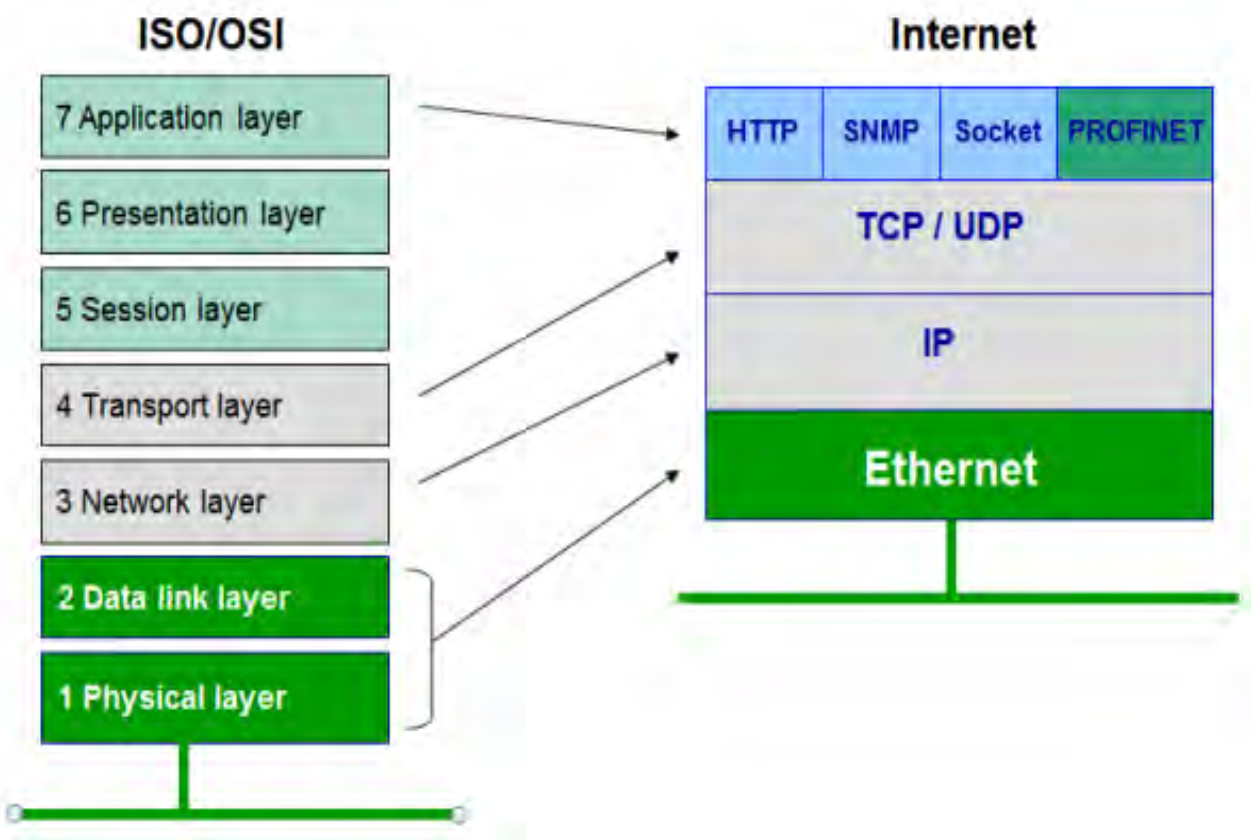
Мрежова Архитектура

- Абстракция при големи и сложни мрежи – да се дефинира модел на КМ – OSI (Open Systems Interconnection) architecture



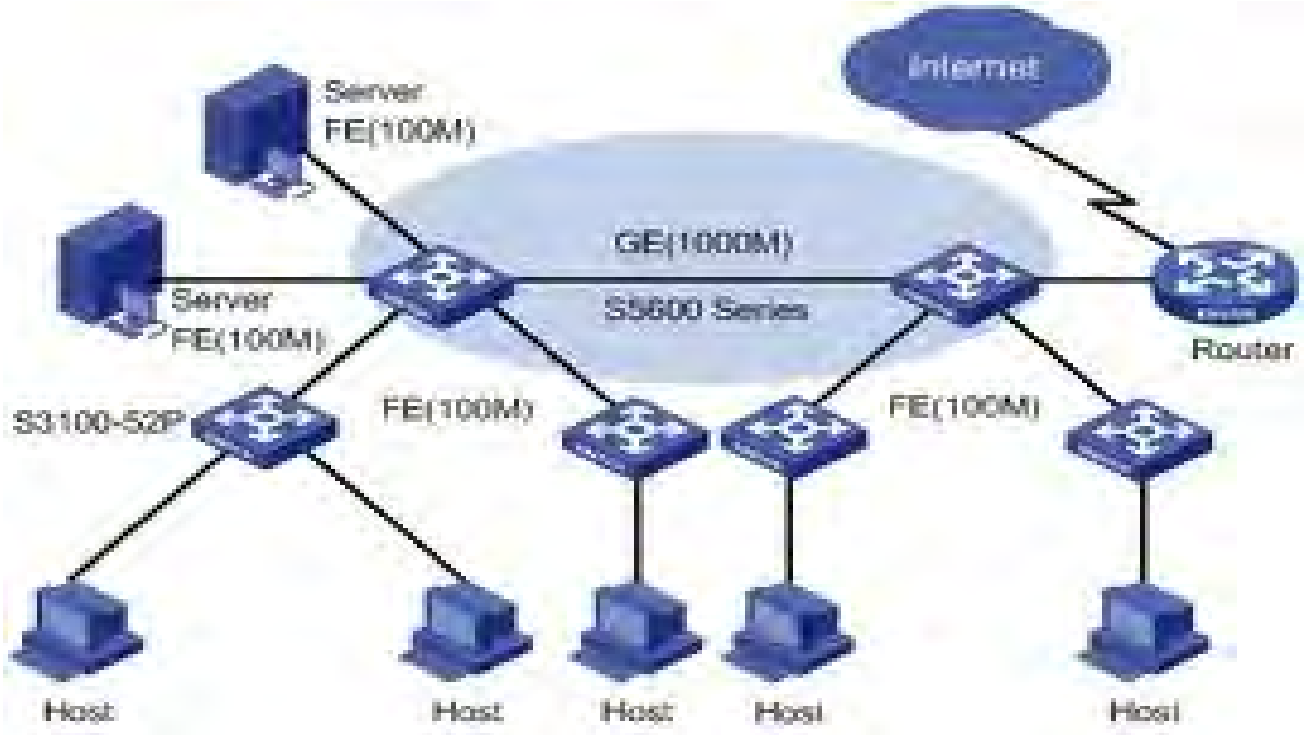
Internet Архитектура

- TCP/IP архитектура – 2-та основни протокола



Компютърни Мрежи (КМ)
Въведение, Характеристики и Изисквания (3)

- Директна връзка – сложно за управление и скъпо решение
- Индиректно свързване на група от възли – Switched networks
- Forwarding nodes – възли, свързани към повече от една връзка, специален софтуер за препредаване на данни (пакети) – store-and – forward – Packet-Switched Networks
- Router (Gateway)



Компютърни Мрежи (КМ)
Въведение, Характеристики и Изисквания (4)

- всеки възел има адрес –стринг от байтове
- възел- сорс (source) и възел-дестинация (destination)
- маршрутизация - когато сорсът иска КМ да изпрати данни към дестинация - възелът -switch или -router решават как да препратят пакетите до дестинацията на базата на адресите им

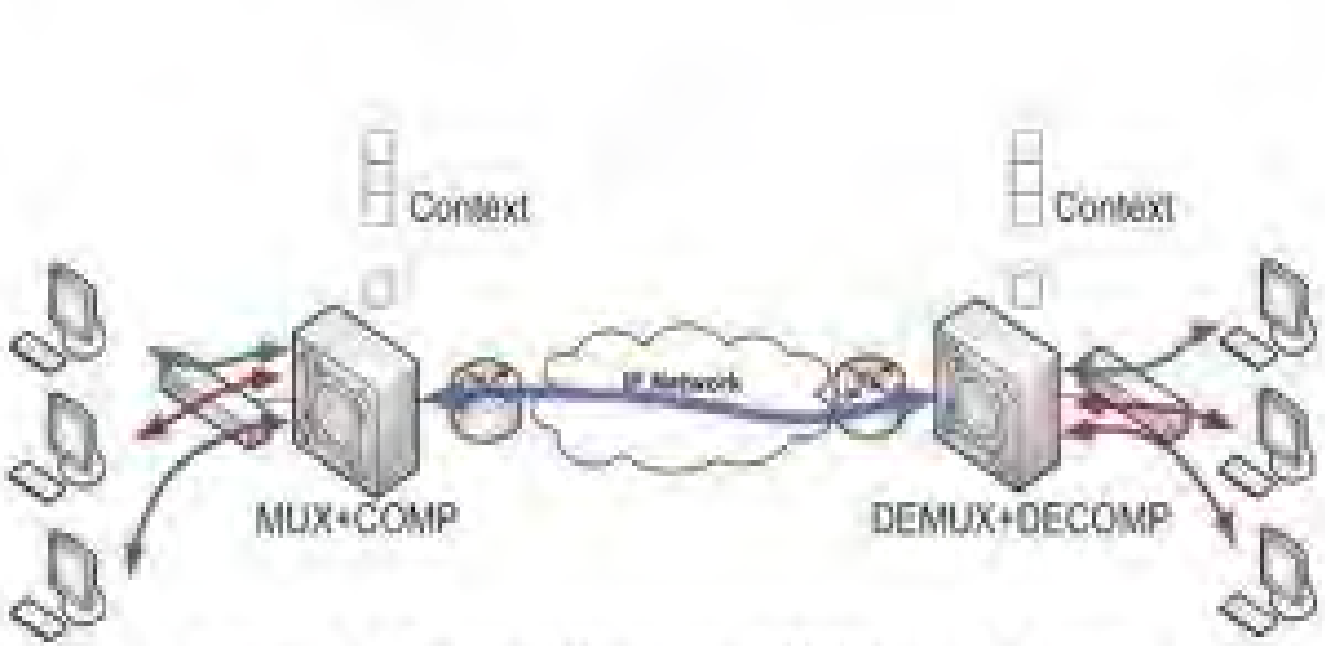
  - когато сорсът иска да изпрати данни само на една дестинация – **unicast**

  -когато сорсът иска да изпрати данни на всички дестинации – **broadcast**

  -когато сорсът иска да изпрати данни на определена група от дестинации – **multicast**
- КМ да поддържа маршрутизиране за multicast и broadcast адреси

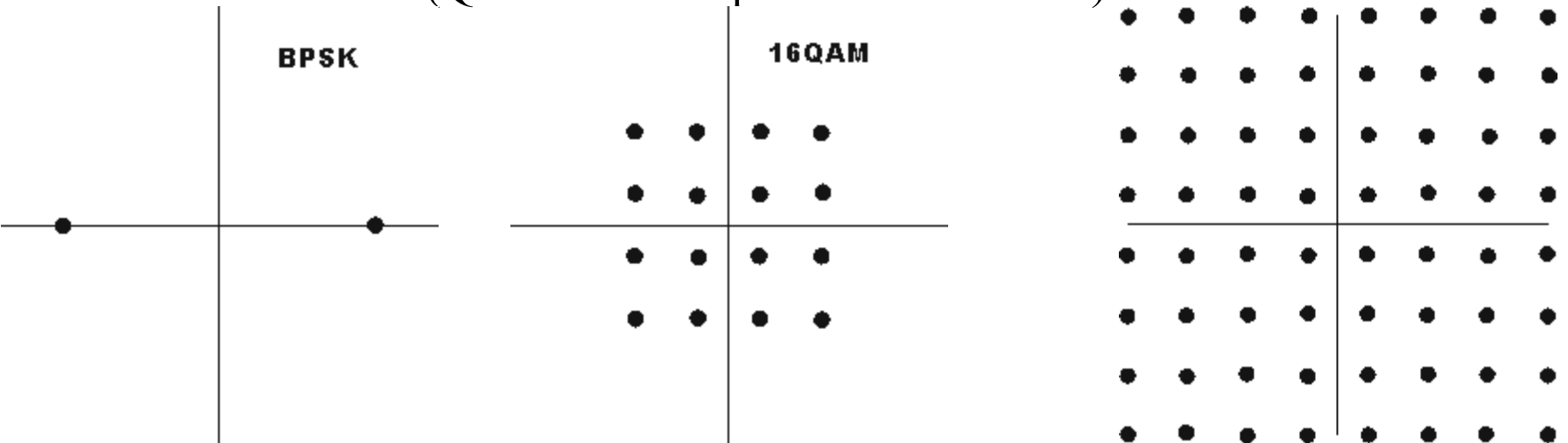
Споделяне на Физическата Среда

- Индиректна Packet-Switched КМ – как възлите споделят едни и същи връзки, които те искат по едно и също време
- Multiplexing – системните ресурси се споделят от много потребители, данните се мултиплексират по физическите връзки в КМ



Методи за Мултиплексиране и Модулация

- TDM - Time-Division Multiplexing
  - STDM – Synchronous Time-Division Multiplexing
- FDM – Frequency Division Multiplexing
  - OFDM - Orthogonal Frequency Division Multiplexing
- CDM - Code Division Multiplexing
  - FHSS - Frequency-hopping spread spectrum
  - DSSS - direct-sequence spread spectrum (фазова модулация)
  - HR-DSSS (High rate DSSS)
- BPSK (бинарна фазова манипулация)
- 16-QAM или 64-QAM - Квадратурна амплитудна модулация за 16 и 64 състояния (Quadrature Amplitude Modulation)



Класификация на КМ

- Класификация на КМ по размер
  - LANs – Local Area Network (до 1км)
  - WANs- Wide Area Network
- Класификация на КМ по физически носители на връзките (тип на електромагнитната вълна) – жични, безжични, коаксиални кабели, оптични влакна
- Класификация на КМ по типа на архитектурата – point-to-point, switch, circuit or packet, cloud, peer-to-peer, etc
- Класификация на КМ по методите за мултиплексиране, модулация и кодиране

Безжични Мрежи

- Информационни и Комуникационни услуги on the move базирани на 802.11/Wi-Fi wireless networking
- Безжична комуникация – Infrastructure mode

  Традиционните клетъчни мрежи (base station infrastructure)
- Безжична комуникация – Infrastructureless mode (Ad-hoc mode)– мултихоп маршрутизация
- Мобилни Ad-hoc Безжични Мрежи (MANETs)

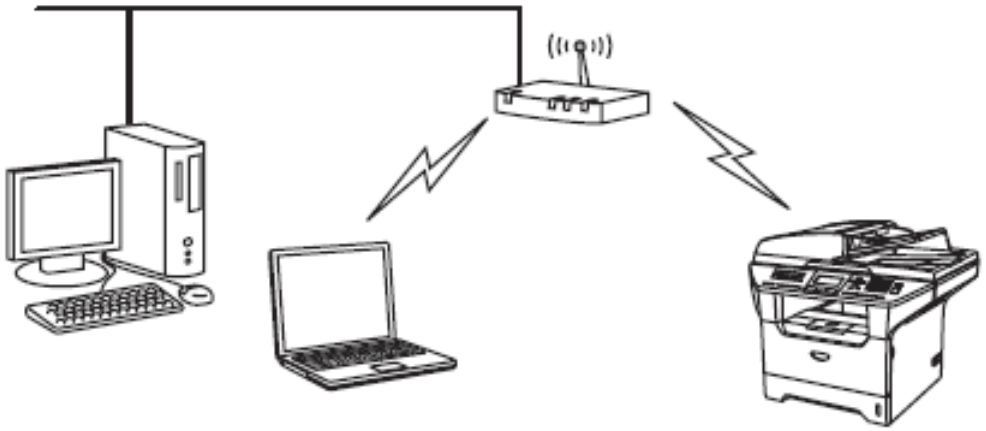
  Всяко устройство в MANETs е свободно да се движи независимо и във всяка посока => често променя връзките си с другите у-ва
  - Mobile Ad-hoc Networks (dense with mobile nodes)
  - WSNs (gateway, sinks, sensor nodes)
  - VANETs (vehicles and roadside equipment)
  - MESH (reliable by redundancy)
  - DT-MANETs (sparse)
  - iMANET (mobile nodes and fixed Internet-gateway nodes)

Изисквания към Безжичните Мрежи за Подобряване Качеството на Предаване

- **Throughput** (да се максимизира капацитета)
- **Number of nodes** (100 възела в различни клетки)
- **Connection to backbone LAN**, (LAN extension)
- **Service area**, (100- 300m)
- **Battery power consumption** (stand by режим, специален MAC протокол за контрол върху изразходваната мощност)
- **Transmission robustness and security** (уязвими към хакери и намеса, изисква специален дизайн на протоколи за сигурност)
- **License-free operation** (не изисква лиценз за честотна лента)
- **Handoff/roaming** (MAC протокол да улеснява преминаването от една клетка в друга)
- **Dynamic configuration**, автоматично добавяне и премахване на терминални у-ва

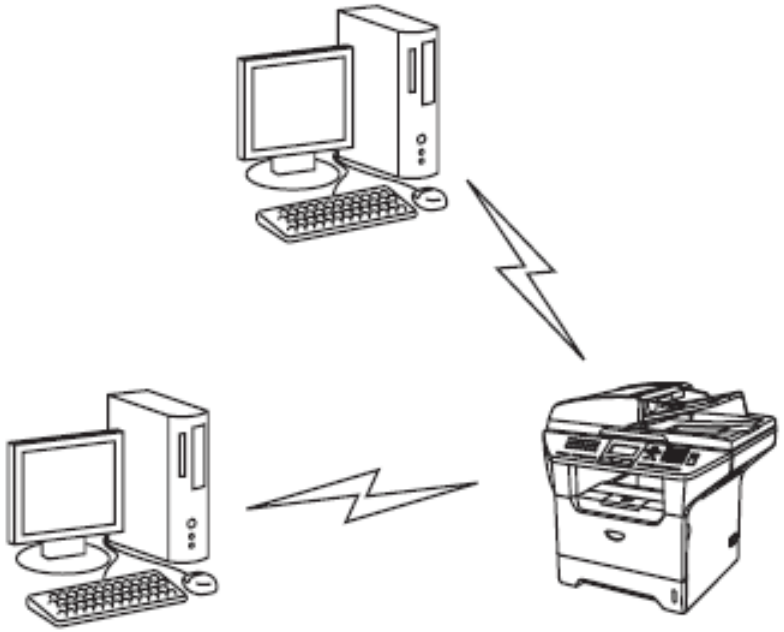
Безжична комуникация – Infrastructure mode

- централна точка за достъп в центъра на мрежата
- безжичните устройства комуникират помежду си чрез точка за достъп



Безжична комуникация – Ad-hoc mode

- не е необходима централна точка за достъп
- всеки клиент на безжичната жрежа комуникира директно с другите



Приложения на Wireless LAN

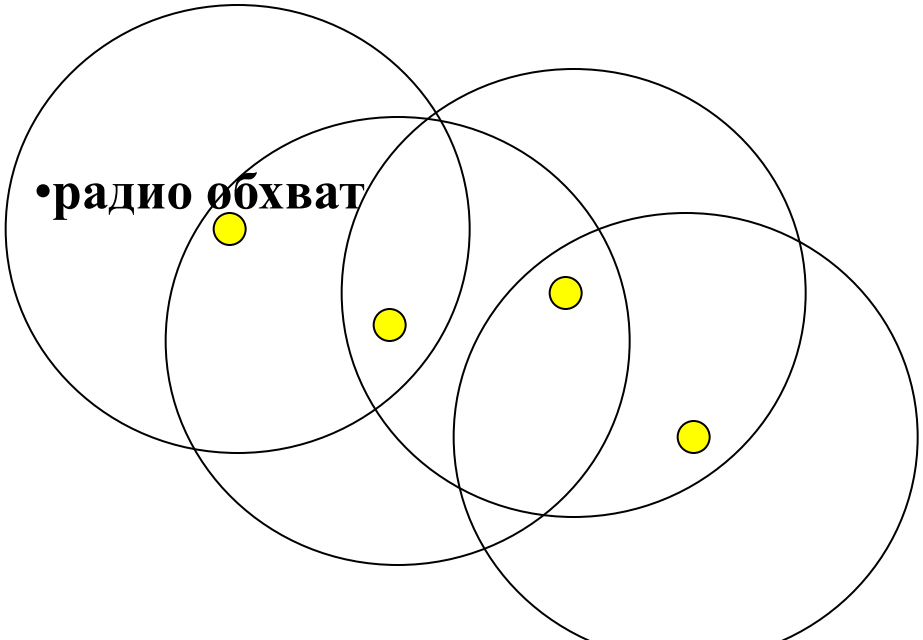
- **LAN extension**
  - Когато поддържането и инсталирането на жична инфраструктура е скъпо решение или не е разрешено - wireless LAN
- **Cross Building interconnection**
  - Много LANs могат да бъдат свързани с point-to-point wireless LAN (routers и point-to-point LAN bridges)
- **Nomadic Access (Mobile IP)**
  - Осигурява безжична връзка между LAN hubs и мобилни терминали/лаптопи/smart телефони (сгради, кампуси)
- **AdHoc networking**
  - Временна peer-to-peer мрежа, която се установява за конкретно приложение и време. Напр. група служители на среща свързват компютрите си за определен период от време. Wireless hop-by-hop трансмисия.
- **Sensor networks**

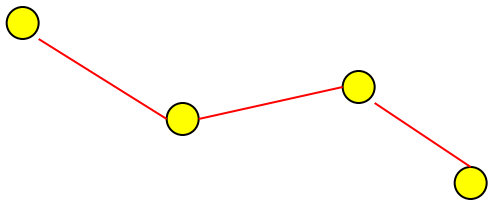
  - много и малки сензори са свързани в безжична мрежа и предават данни до специално определени възли, наречени или gateway или sink или Base Station, които са енергийно мощни. Междинни възли за агрегиране на данните.

Мобилни Ad Hoc Мрежи (MANETs)

Мобилните устройства изграждат спонтанно безжична мрежа за предаване на информацията

- Самоорганизираща се - когато няма инфраструктура
- Сравнително евтина мрежа
- Multi hop mode – за достигане дестинацията
- По-сложни технологични решения на всички слоеве

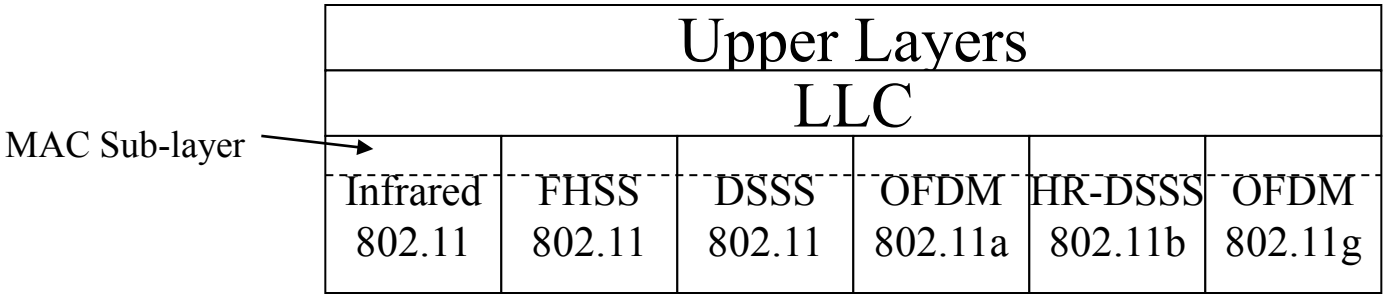




IEEE 802.11

- **IEEE 802.11** - известен още като Wi-Fi,
- дефинира набор от стандарти за Wireless LAN/WLAN, в основата на Wi-Fi технологията, която позволява електронните устройства да предават и получават данни чрез **радио вълни**
- 802.11x - обозначава набор от стандарти (a, b, g, n, y, c–f, h, j) с различна **оперативна честота (**2.4, 3.6, 5 и 60 GHz) , Data Rate-Type (0,9 Mbit/s -23 Mbit/s), Data Rate – Max (2 Mbit/s- 54 Mbit/s), обхват в сграда(20 метра- 50 метра), обхват на открито (100 метра- 5000 метра).
- Включва шест техники за модулация във въздушна среда

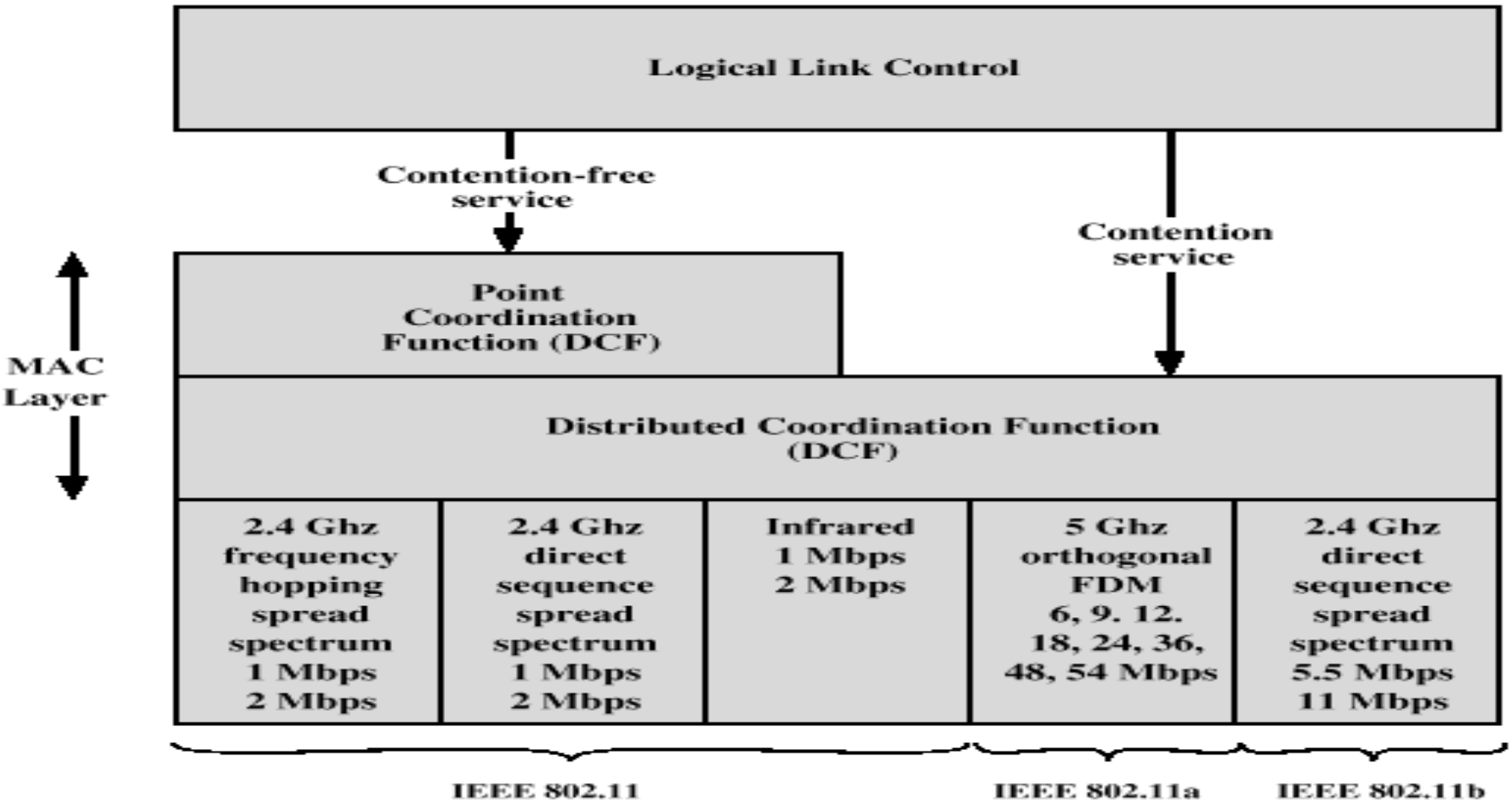
Физическа Среда в 802.11 Стандарта



- **Direct-sequence spread spectrum**
  - 2.4 GHz ISM band
  - Осигурява пренос от 1 and 2 Mbps
- **Frequency-hopping spread spectrum**
  - 2.4 GHz ISM band
  - Осигурява пренос от 1 and 2 Mbps
- **Infrared**
  - 1 и 2 Mbps
  - Дължина на вълната между 850 и 950 nm
- **IEEE 802.11a**
  - Използване на 5-GHz band
  - Осигурява пренос от 6, 9 , 12, 18, 24, 36, 48, 54 Mbps
  - Използва orthogonal frequency division multiplexing (OFDM)
  - Модулиране на носителя BPSK, QPSK, 16-QAM or 64-QAM
- **IEEE 802.11b**
  - Осигурява пренос от 5.5 and 11 Mbps
  - Complementary code keying (CCK)
  - Модулиране на носителя BPSK, QPSK

802.11 MAC SubLayer

- DCF – нова MAC технология на 802.11



802.11 DCF

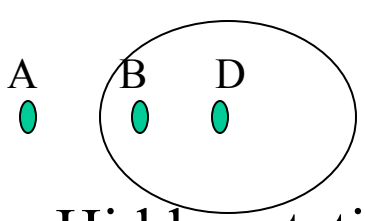
- Distributed coordination function (DCF) - MAC технология на IEEE 802.11
- DCF имплементира CSMA/CA , двоично експоненциален backoff алгоритъм, избягва колизии и осигурява различни нива Inter-Frame spacing (IFS),

$$BackoffTime = random() \times aSlotTime$$

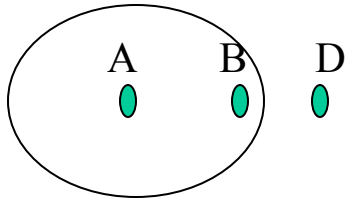
IFS	802.11b	802.11g	802.11a	802.11n 2.4GHz	802.11n 5GHz
SIFS	10µs	10µs	16µs	10µs	16µs
Slot Time	20µs	Long = 20µs Short = 9µs	9µs	Long = 20µs Short = 9µs	9µs
DIFS	50µs	Long = 50µs Short = 28µs	34µs	Long = 50µs Short = 28µs	34µs

Контрол на Достъп

- CSMA/CD (**Carrier Sense Multiple Access / Collision Detection**) не работи при wireless LANs
  - Устройствата не различават слаб сигнал от грешки и колизии
  - **Hidden stations**: **D** изпраща на **B** (в обхват са). **A** слуша радио комуникацията (carrier sense), че е свободна и погрешно решава, че може да изпрати на B. Колизия в устройство B.
  - **Exposed stations**: **A** изпраща на **B** (в обхват са), **B** иска да изпрати на **D**, но не може тъй като слуша радио комуникацията с **A**, че не е свободна и погрешно решава че не може да изпрати на **D**.



Hidden stations

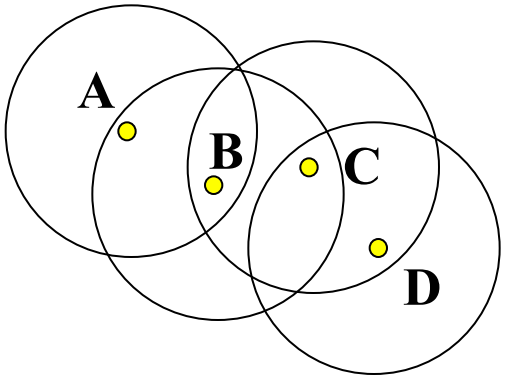


Exposed stations

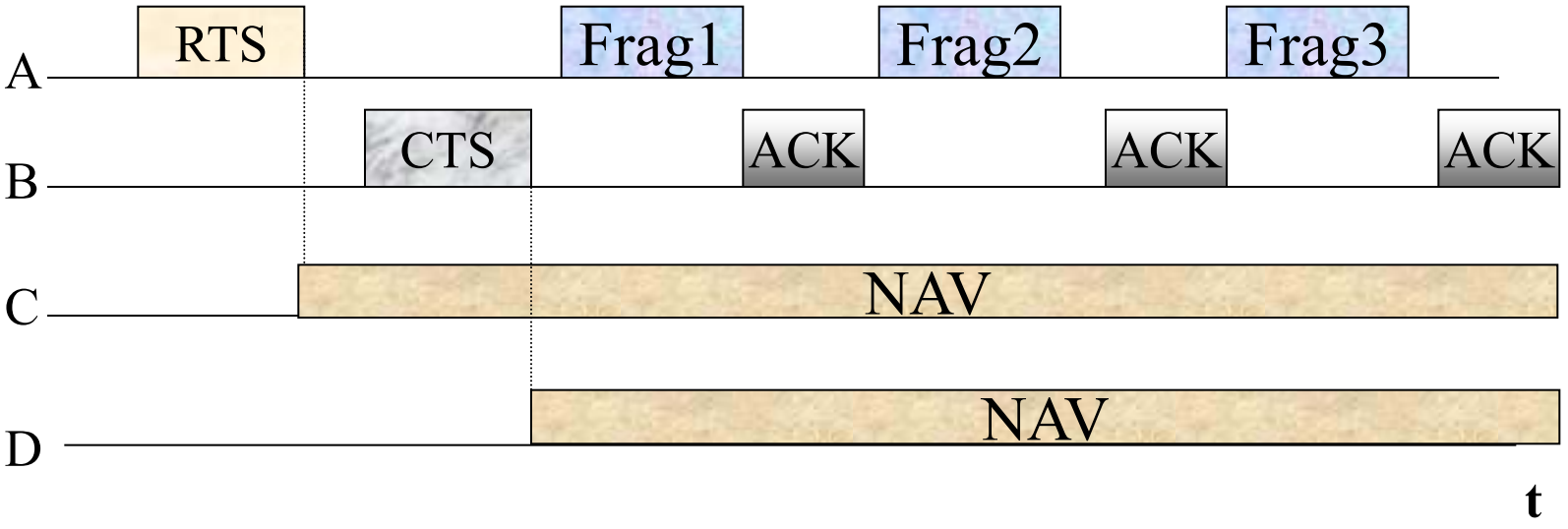
DCF Frame exchange protocol

- Wireless MAC решава проблемите, използвайки DFWMAC- Distributed Foundation Wireless MAC protocol
- Осигурява:
  - a distributed access control mechanism, наречен **distributed coordinate function** (DCF)
  - незадължителен централизиран контрол, наречен **Point Coordination function (PCF)** осигуряващ свободни услуги (напр. base stations to backbone)

- Протокол за предаване на фрейми
  - Възелът сорс изпраща данни
  - Възелът дестинация потвърждава (ACK)
  - Ако сорсът не получава ACK, препраща фрейма
- Предаване на контролни съобщения преди предване 1ви фрейм
  - Възелът сорс иска request to send (RTS) + Network Allocation Vector (NAV)
  - Възелът дестинация отговаря с clear to send (CTS) + NAV
  - Възелът сорс изпраща данните
  - Възелът дестинация потвърждава с ACK



DCF + MACAW

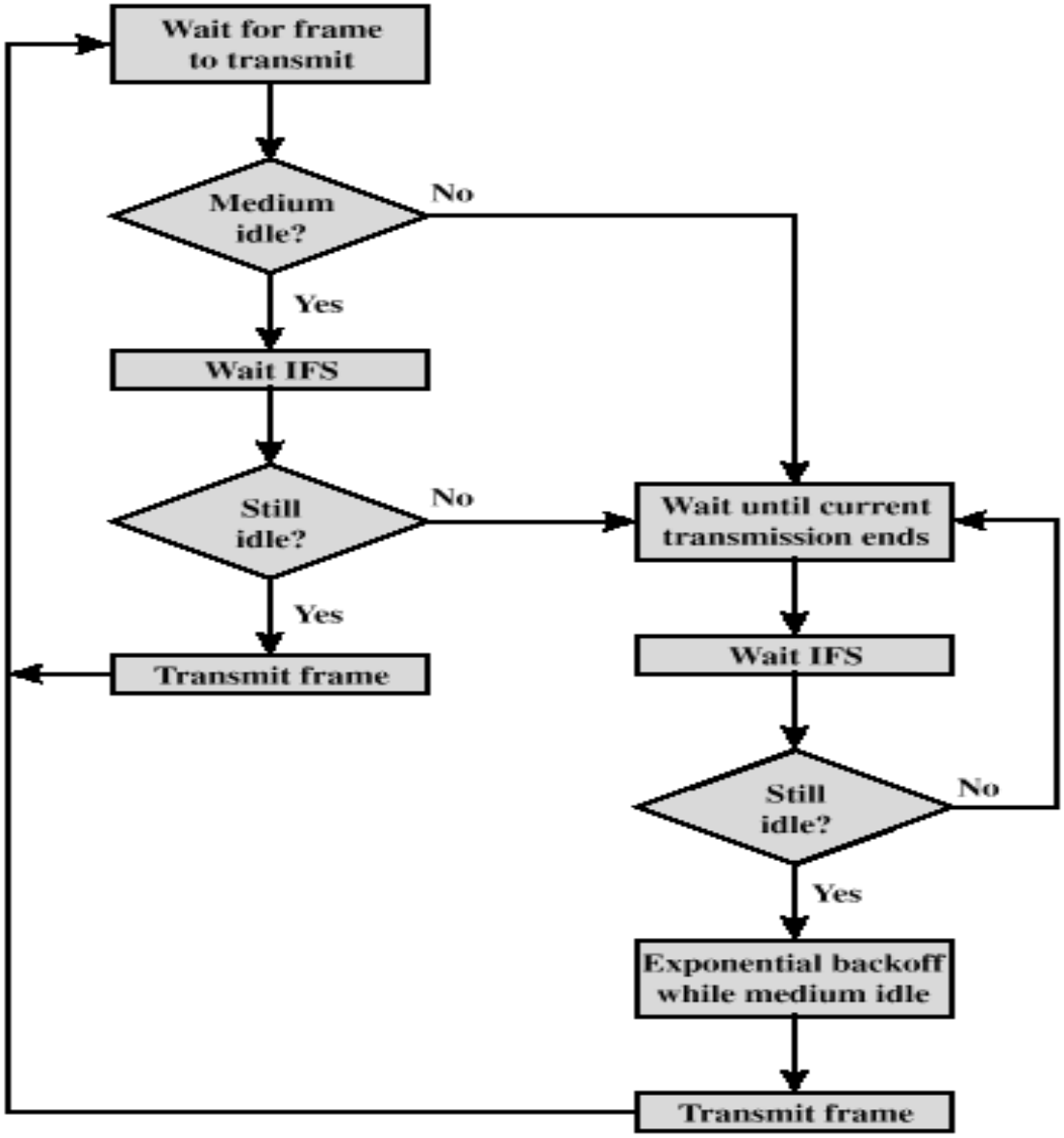


802.11b MAC има вградени няколко InterFrame Spacing, за да поддържа DCF и PCF едновременно

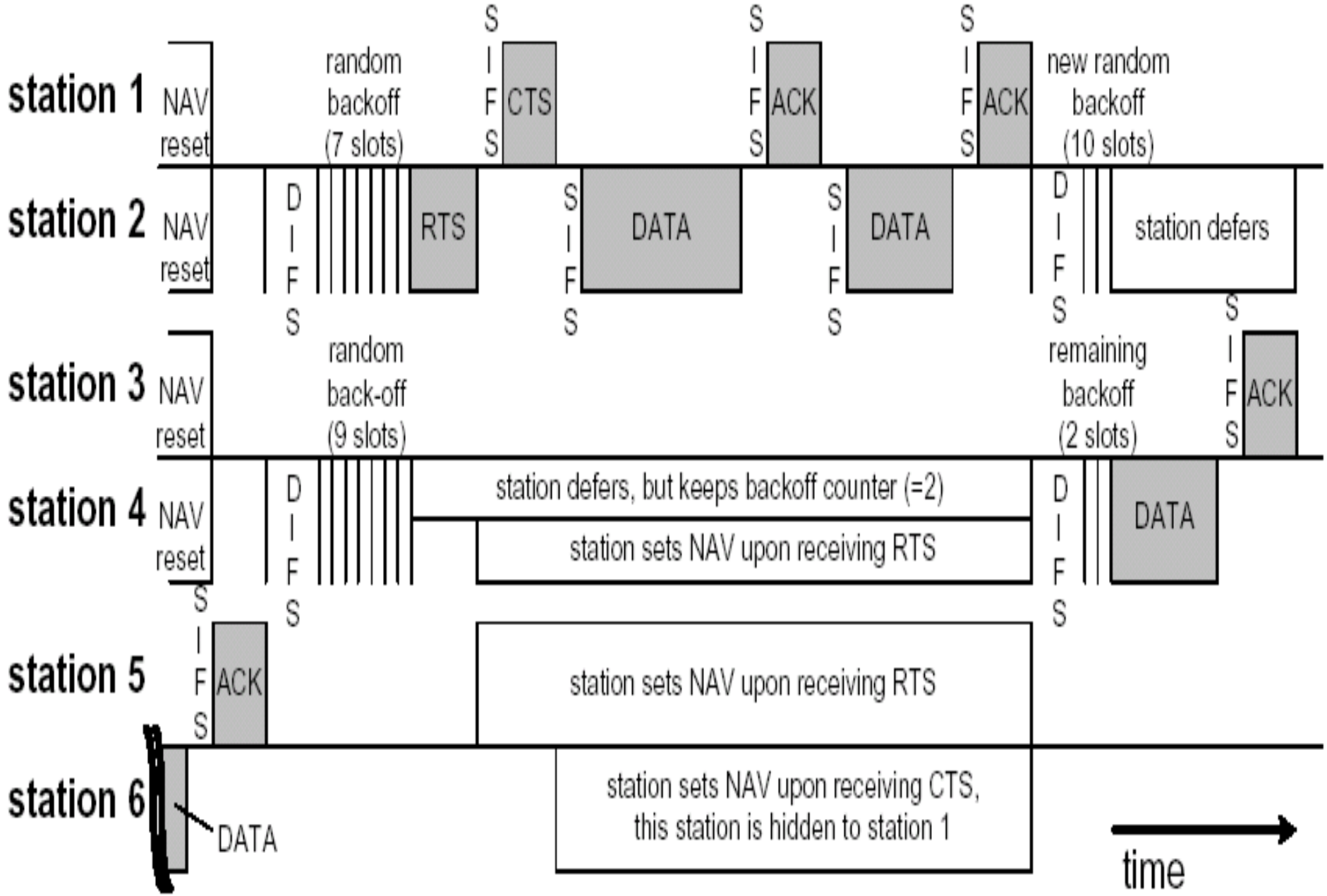
- След изпращане на фрейма се изчаква определен интервал от време
  - **SIFS (**Short InterFrame Spacing) – 10μs. Идеята е различните възли да не изпращат едновременно фреймите. Фрейми като ACK изчакват SIFS преди предаване. Няма Random Backoff.
  - **PIFS** (PCF InterFrame spacing) – Механизъм за приоритет. Polling механизъм, с който се контролира кой възел да изпраща. 30μs. Няма Random Backoff.
  - **DIFS** – 50μs. Използва Random Backoff.

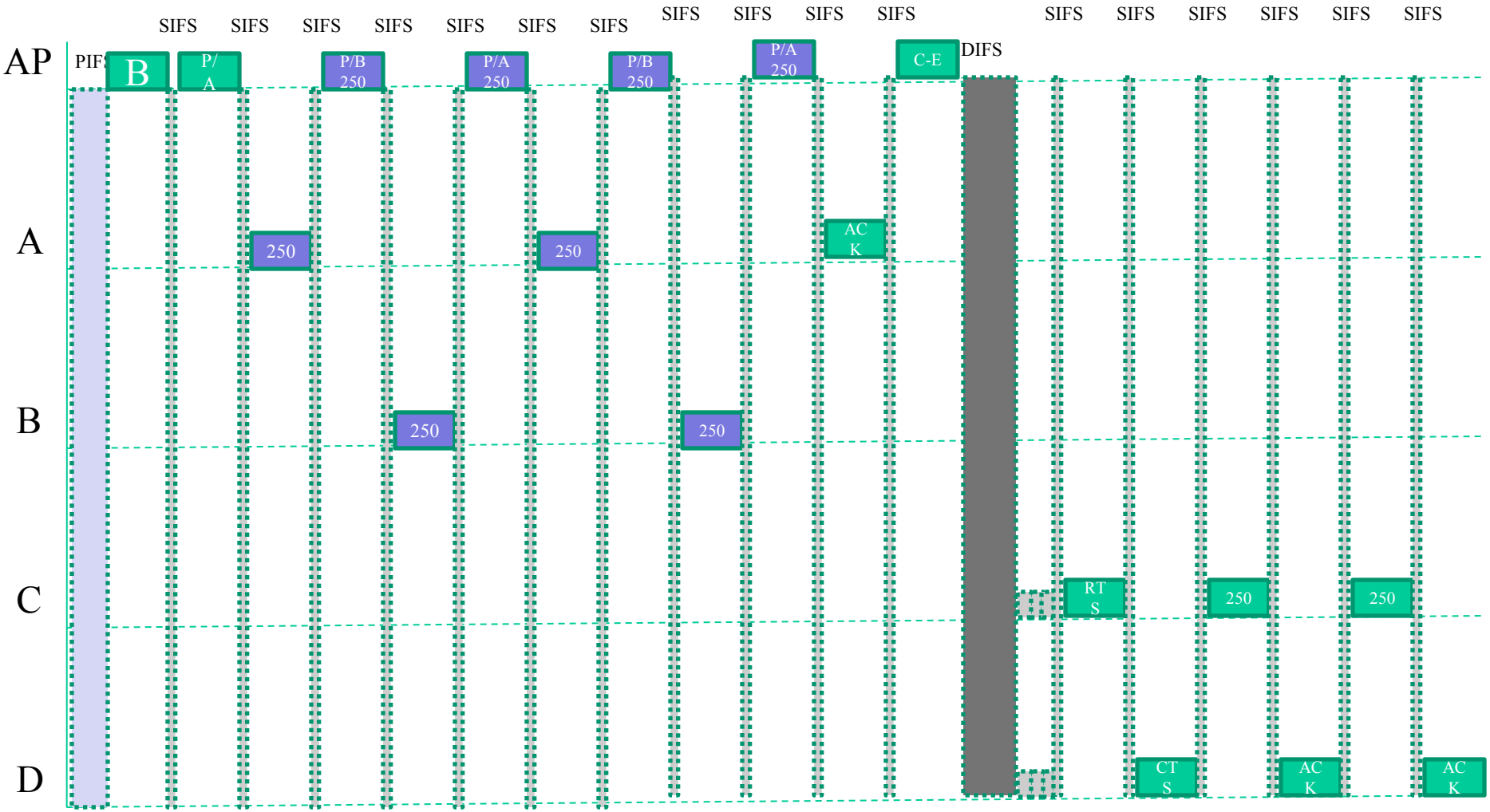
- Random Backoff – 1) генерира се число; 2) Изчаква се времето на DIFS докато каналът е свободен; 3)Намалява се Random Backoff на всеки 20μs докато каналът е свободен; 4) Ако каналът бъде зает (т.е. друг възел достига 0 преди нас) – stop() и повтаряне на стъпки 2-4.

MAC Logic -> DCF + MACAW



DCF c Hidden Nodes (NAV)




AP
A
B
C
D
PIFS
SIFS
B
P/A
P/B 250
P/A 250
P/B 250
250
250
250
P/A 250
AC K
C-E
DIFS
RT S
250
250
CT S
AC K
AC K

Кога е необходима MANET?

- Няма изградена инфраструктура или е частично разрушена
- Не изисква поддържане на базова мрежова инфраструктура

-скъпо решение при радио мрежи за късо разстояние (ZigBee range ~ 10m)

-скъпо решение да се пусне широколентов достъп поради отдалеченост на самото място или малкия брой абонати

Приложения

- Персонализирани Мрежи

-мониторинг на здравето

- Служби за сигурност, отбрана,

- военна екипировка, битка

- доставяне на мобилен достъп до Internet и обществени база данни

- Цивилна среда

-конферентни срещи

-спортни и културни събития

- Борба с бедствия и аварии

-спасяване на хора

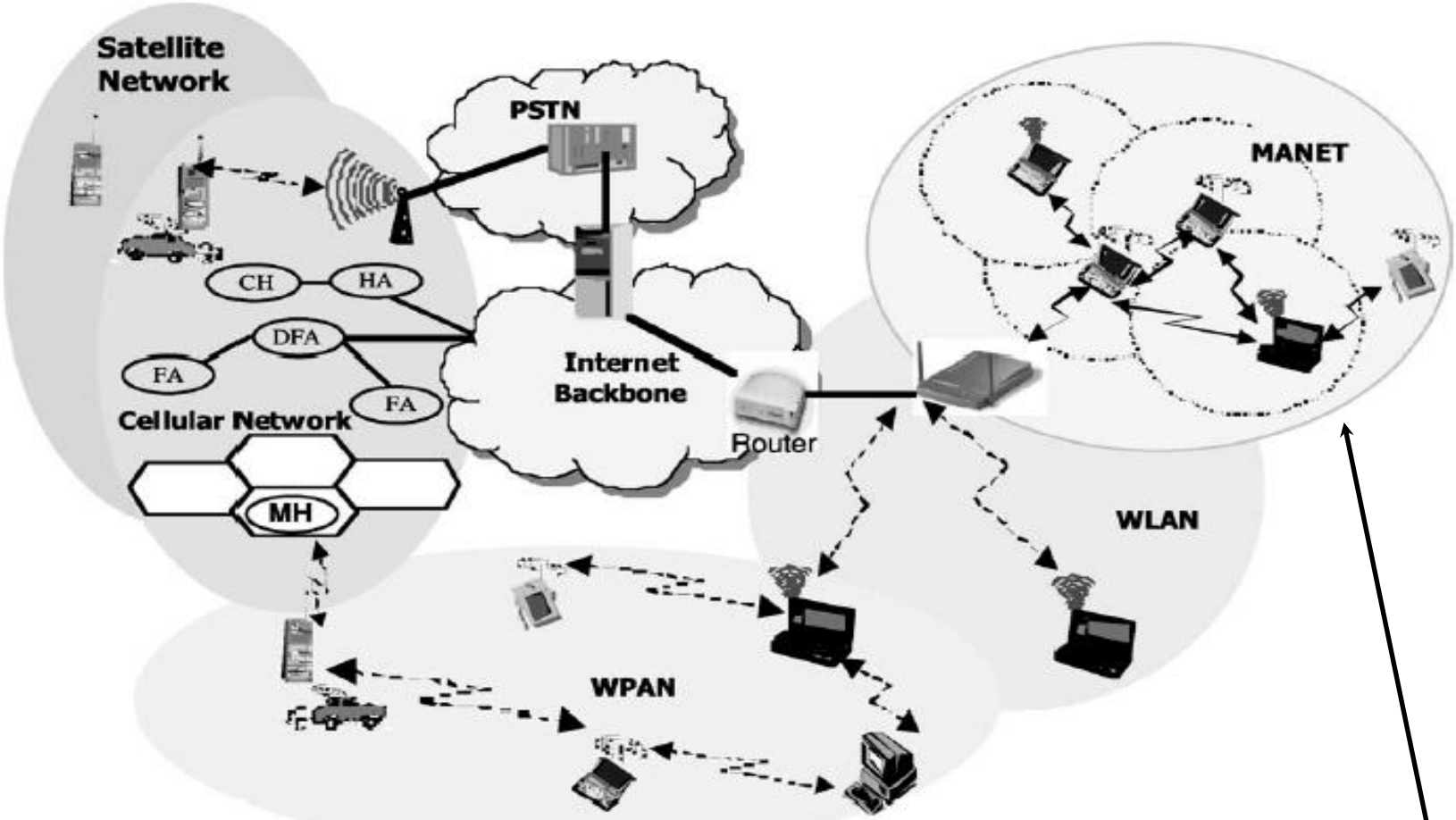
-премахване на препятствия и пожароспасителни операции

- Мониторинг и Data Mining с MANETs

-събиране на данни и data mining при мониторинг на различни замърсявания на въздуха

- Роуминг в 4G мрежи

4G-Глобална Интегрирана Мрежа - MANETs може да е част от 4G



Мобилни Безжични Мрежи (MANET) - динамична топология и неопределеност

Изследователска Дейност в Областта

- IEEE 802.11 for Wireless LANs
  - MAC
  - PHY
- IETF (Internet Engineering Task Force) MANET group
  - Верифицира и утвърждава възможните стандарти в областта
- Протоколи за Маршрутизация:
  - unicast – AODV, DSR, ZRP, TORA, CBRP, CEDAR
  - multicast – AMRoute, ODMRP, AMRIS

Сигурност в MANETs

- PGP (Pretty Good Privacy) - encrypted digital signature
- Компромис между сигурност и performance
  Подобрения на протоколи или нови протоколи
- Ограничена сигурност, базирана на хардуер
- Типове атаки - активни и пасивни
  - Най-често използвани са пасивните атаки – подслушване и разкриване на информация
  - Активни атаки: Denial of service (непредаване на пакети), Data modification by viruses, Trojans и worms

Класифициране на атаките в MANETs

- Application Layer: Malicious code
- Transport Layer: Session hijacking, Flooding
- Network Layer: Sybil, Flooding, Black Hole, Grey Hole. Worm Hole, Link Spoofing, Link Withholding, Location disclosure etc.
- Data Link/MAC: Malicious Behavior, Selfish Behavior, Active, Passive, Internal External
- Physical: Interference, Traffic Jamming, Eavesdropping (подслушване)

Изисквания към Протоколите за маршрутизация

- Достатъчно прости и лесни за конфигуриране
- Осигуряване надеждна и стабилна работа на мрежата

  - Алтернативни пътища, „Релета - Буфери” - препращат трафика към другите у-ва
- Реактивни, Проактивни и Хибридни протоколи - зависи от гъстота и степен на мобилност на възлите
  - по–голяма гъстотата -> по– малко “релета” са нужни
  - по-голяма мобилност – реактивни протоколи
- **Сигурност** при маршрутизирането

Сигурност на Маршрутизацията в Безжичните Мобилни Ad-Hoc Мрежи

Denial of Service Attacks (DoS) – Cisco

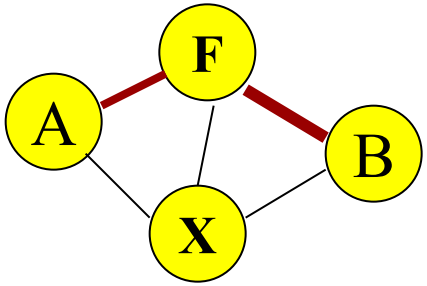
- DoS атака – недобронамерен опит от човек или група от хора да накарат устройствата да отказват или променят мрежови услуги за клиентите

- DoS атака при MANETs –анализирани относно отказ за маршрутизиране на пакетите, които се игнорират вместо да се препредават

- Selfishness (or blackholing) – Пречи на данните, генерирани от източника да достигнат до дестинацията

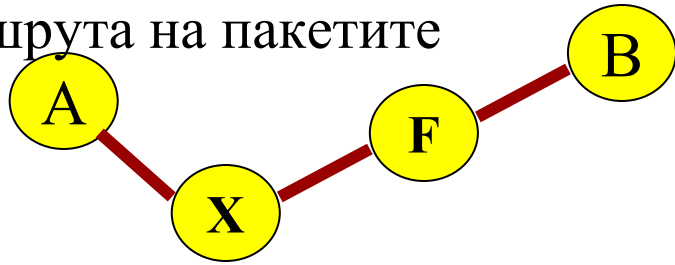
MANETs - Традиционни и Нови Проблеми

- Самоконфигуриране и децентрализиране
- Мрежовата топология се сменя често



t1 Междинният възел F трябва да препредава данните от A до B

Мобилността предизвиква смяна на маршрута на пакетите



t2 Два междинни възела (X и F) трябва да препредават данните от A до B

Маршрутизиране в MANETs

- Маршрутизиране – процес на обмяна на информация между един и друг възел в мрежата (в infrastructure КМ - възелът - switch или -router решават как да препратят пакетите до дестинацията на базата на адресите им)
- Механизъм за маршрутизиране – препредаване на пакетите до достигане на дестинацията като се използва най-ефикасния път
- Ефективността на пътя се измерва с различни метрики – брой hops, трафик, сигурност и др.
- В Ad-hoc мрежите всяко устройство действа и като рутер и може да бъде конфигурирано за различни типове протоколи

Типове Протоколите за Маршрутизация при MANETs

- Проактивни маршрутизиращи протоколи

  Optimized Link State Routing (OLSR),Destination-Sequenced Distance-Vector (DSDV),

  Global State Routing (GSR), Fisheye State Routing (FSR),

  Hierarchical State Routing (HSR)

- Реактивни маршрутизиращи протоколи (*on-demand)*

  Ad-hoc On-demand Distance Vector (AODV), Dynamic Source Routing (DSR), Temporary Ordered Routing Algorithm (TORA), Location-aided Routing (LAR)

- Хибридни протоколи за маршрутизация

  Zone Routing Protocol (ZRP) - Intra-zone Routing Protocol (IARP) (локален проактивен) и Inter-zone Routing Protocol (IERP) (реактивен)

Изисквания към Протоколите за маршрутизация

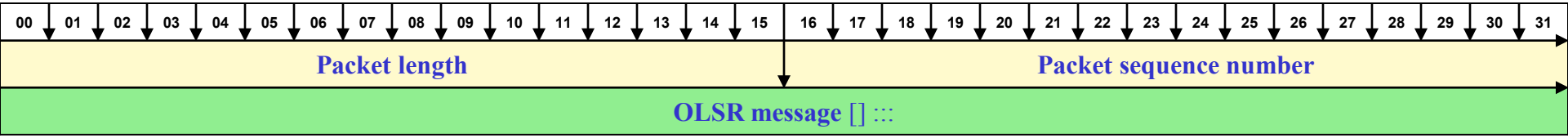
- Достатъчно прости и лесни за конфигуриране
- Осигуряване надеждна и стабилна работа на мрежата

  - Алтернативни пътища, „Релета - Буфери” - препращат трафика към другите у-ва
- Реактивни, Проактивни и Хибридни протоколи - зависи от гъстота и степен на мобилност на възлите
  - по–голяма гъстотата -> по– малко “релета” са нужни
  - по-голяма мобилност – реактивни протоколи
- **Сигурност** при маршрутизирането

Optimized Link State Routing (OLSR)
[Network Working Group - Request for Comments: 3626]

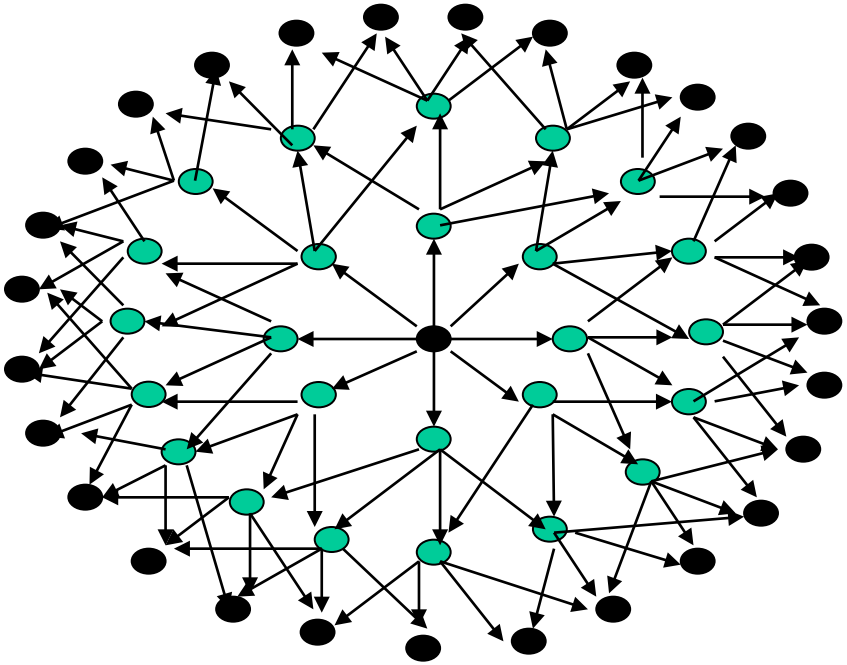
- OLSR – проактивен протокол за маршрутизация
- Разработен от **Internet Engineering Task Force (IETF)**, Project Hipercom, INRIA, (T. Clausen & P. Jacquet) Октомври 2003
- **Толелира загуба на контролни съобщения** (спасителни операции)

OLSR header:



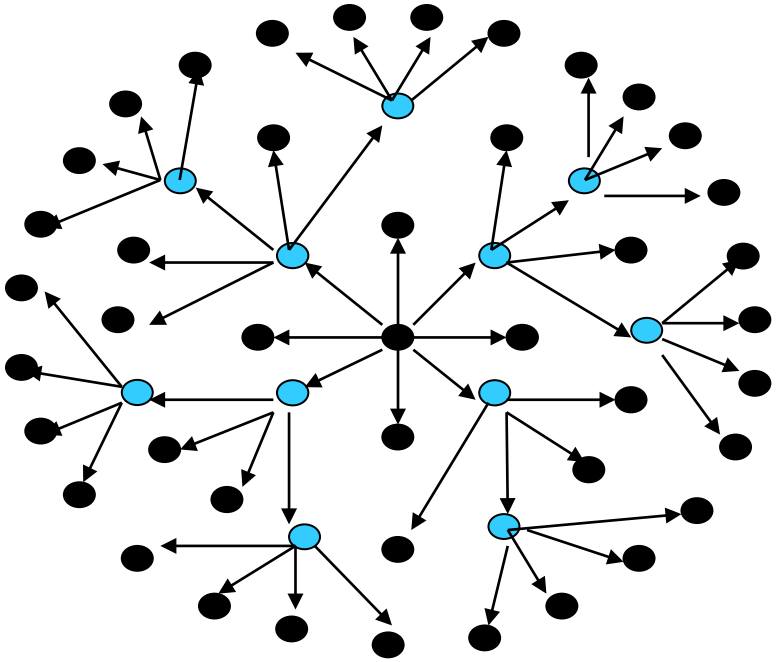
Контролна информация

OLSR опимизира **LSR** чрез намаляване на „междинните релета”



24 препредавания (релета) за 3 hops

LSR препредаващо у-во (реле)

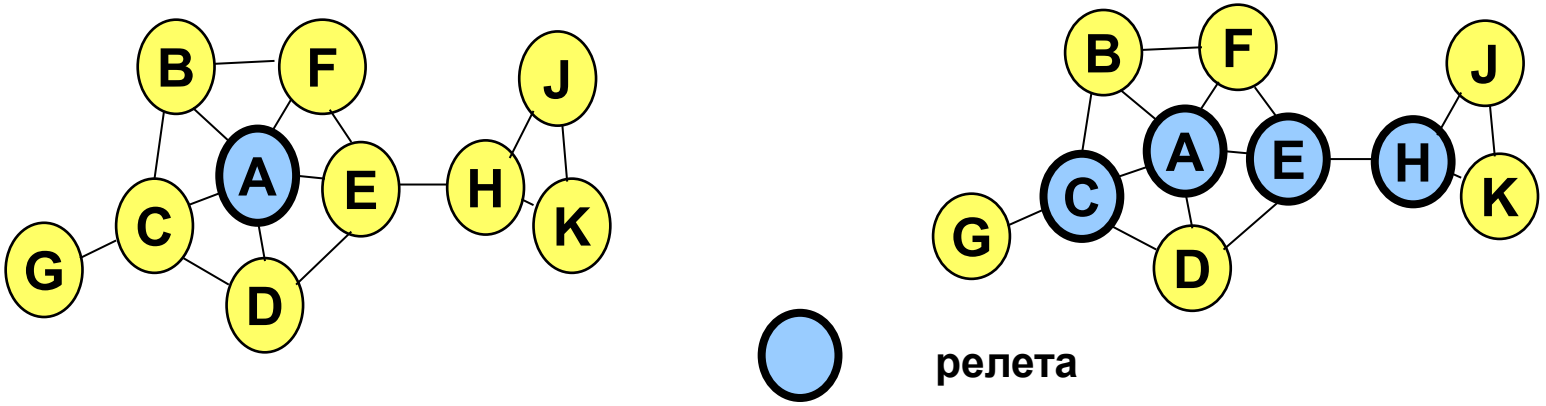


11 препредавания за 3 hops - пести честотната лента

OLSR препредаващо у-во (реле)

OLSR Пример

- Възли C и E са релета (MRP) за възел A
  - Релета на A са тези съседи, които осигуряват връзка на A със съседите 2-hop
  - Възлите разменят списък на съседите и така знаят 2-hop съседите (известни на 1-hop съседа)
  - Избират релета

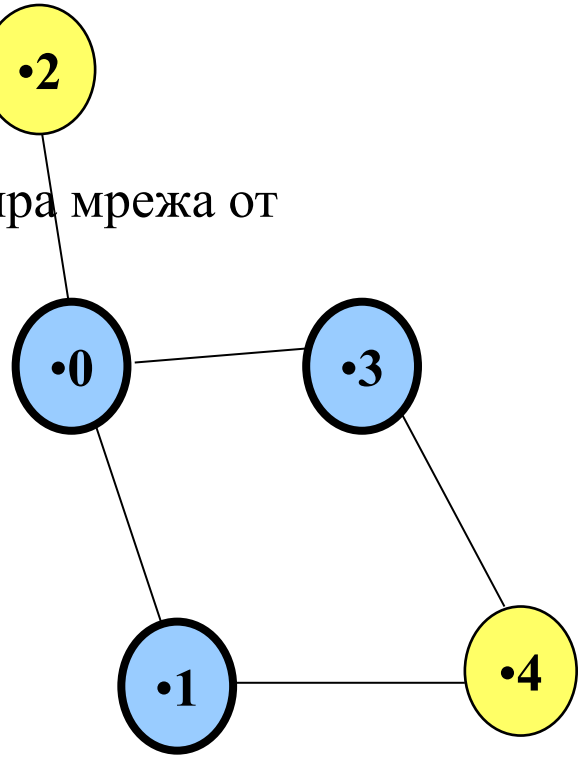


- Възли C и E препращат данните изпратени от A

Откриване на Съседи и Избор на Релета (локално)

✓ Всяко у-во периодично излъчва Hello beacon:

- -записва списък със съседи, които имат директна връзка – 1 hop
- -записва съседи с индиректна връзка, т.е. известни на съседа – 2 hop

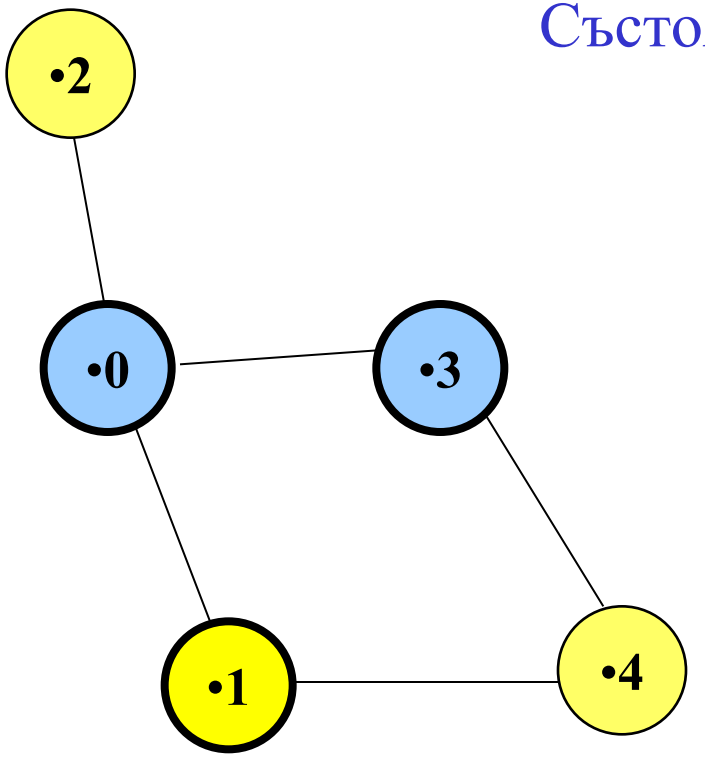


✓ на базата на Hello beacons всяко у-во избира мрежа от MPR

- Node_ID = 2 => MRP 0
- Node_ID = 4 => MRP 3 или 1

- Зависи от Евристики, като:
- - Node_Willingness (батерия)
- - D(y) – брой симетрични съседи на y
- - Достигане на повече 2-hop

Състояние Връзки, Таблица за Топологията и Матрица на Връзките



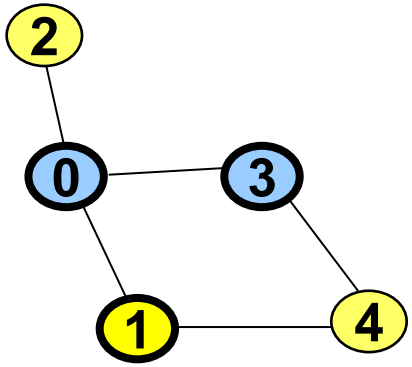
У-во	1 Hop Съсед	2 Hop Съсед	MRPs
0	1, 2, 3	4	3
2	0	1, 3	0

•**1_Willingness** =WILL_HIGH

•**3_Willingness** = WILL_ALWAYS

•**ДиректенРадиоОбхват[N][N]**

	1	2	3	4	5
1	-1	1	1	1	0
2	1	-1	1	0	0
3	1	1	-1	1	1
4	1	0	1	-1	0
5	0	0	1	0	-1



Пример за Таблица - Съседи

Всеки запис в таблицата има timestamp (текущото време + HTime (0.5 sec). След изтичане на времето, записът не е валиден.

Node_ID = 0

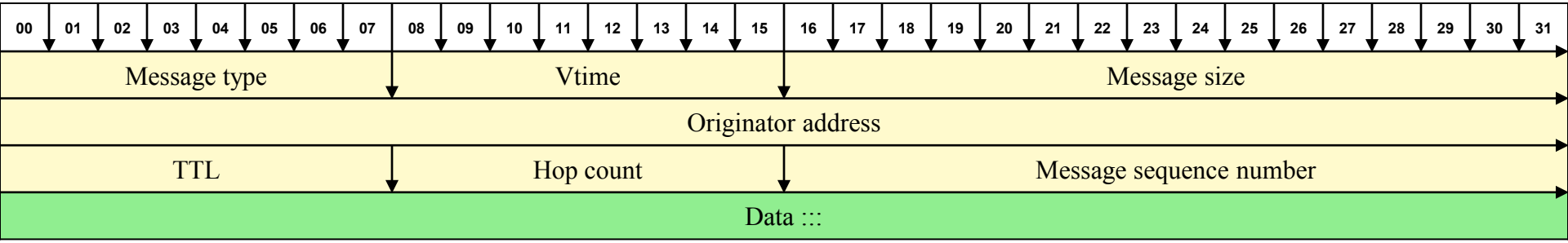
One-hop neighbors

Neighbor's id	State of Link
1	Bidirectional
2	Unidirectional
3	MPR
…	…

Two-hop neighbors

Neighbor's id	Access through
4	3

OLSR message



Message type: 8 bits

1 HELLO, 2 TC (topology control) min 96 bits (32+32+32)

Vtime: 8 bits

Колко дълго след получаване възелът да счита информацията в съобщението за валидна

Message size: 16 bits.

Дължина на message header и data в байтове

TTL, time to live. 8 bits

Съдържа максимума брой hops съобшението да се препрати

Hop count. 8 bits.

Съдържа броя hops, които съобщението преминава. Преди да се препрати, Hop Count трябва да ес увеличи с 1

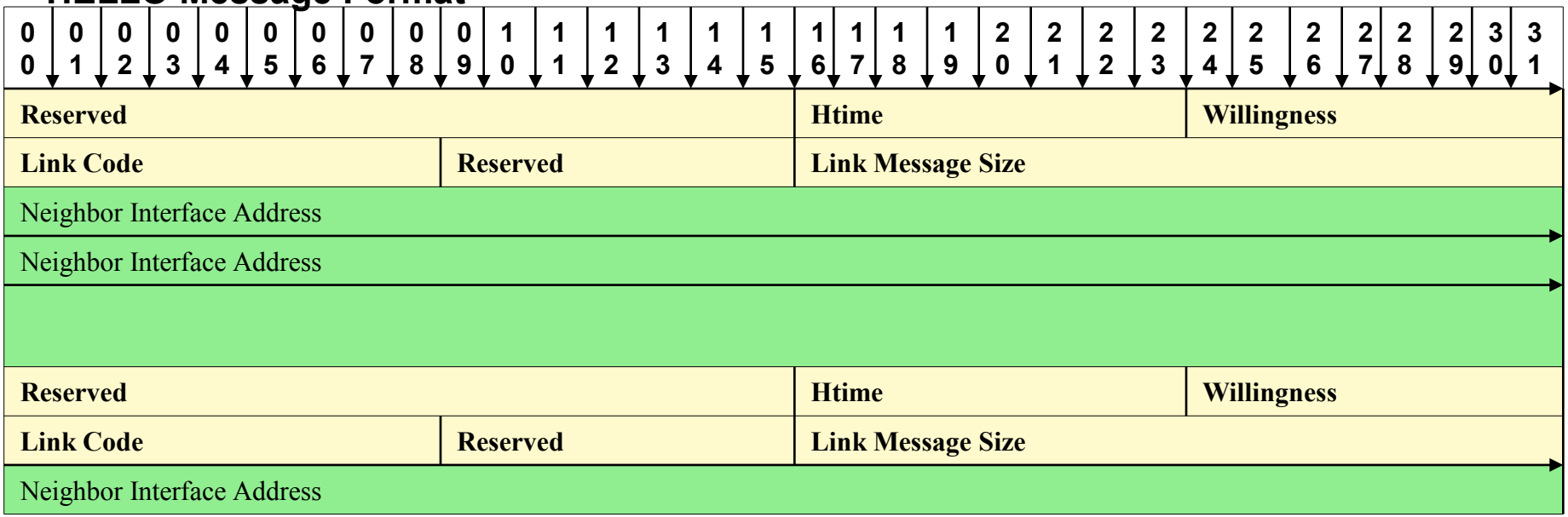
Message sequence number. 16 bits.

номер за последователност – уникален идентификационен номер за всяко съобщение. Увеличава се с 1 (едно) за всяко съобщение на сорс-възела. Осигурява, че съобщението няма да се предаде повече от един път.

HELLO Message Format

Task of advertising the link sensity, neighbor detection and MRP signaling

HELLO Message Format



Willingness - the willingness of a node to carry and forward traffic for other nodes

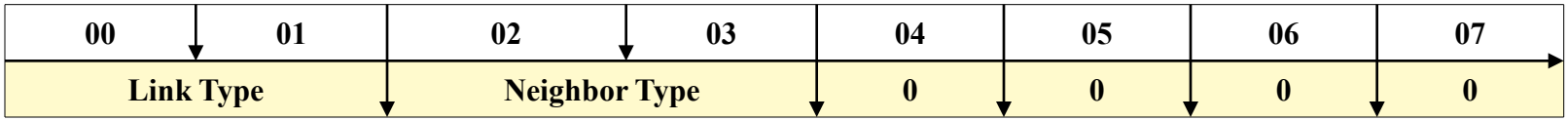
- WILL_NEVER = 0
- WILL_LOW = 1
- WILL_DEFAULT = 3
- WILL_HIGH = 6
- WILL_ALWAYS = 7

HELLO Message Format (2)

HTime - specifies the HELLO emission interval, i.e., the time before the transmission of the next HELLO

HTime= 0.5 **sec**

Link Code

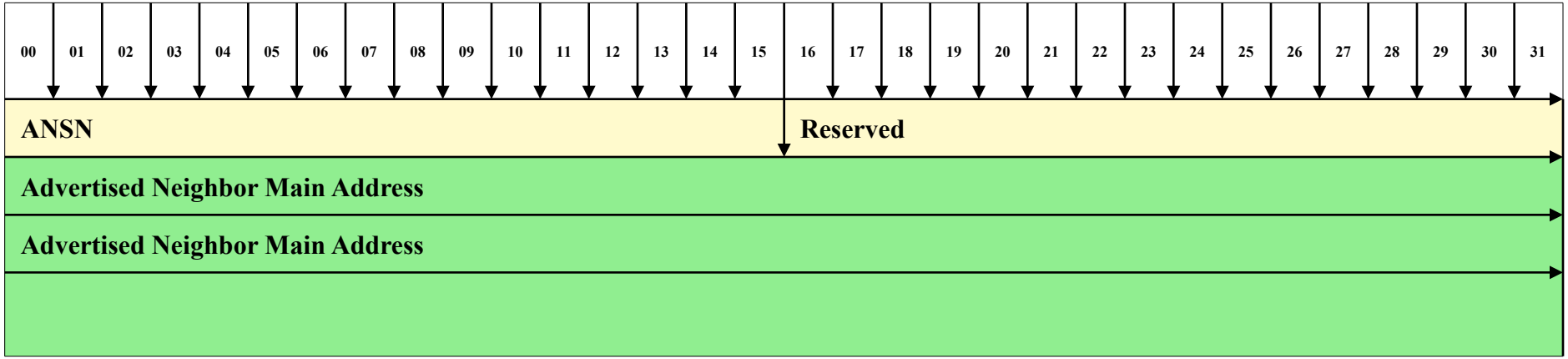


UNSPEC_LINK
ASYM_LINK
SYM_LINK
LOST_LINK
SYM_NEIGH
MRP_NEIGH
NOT_NEIGH

Topology Control Message Format

Целта е да се рекламират връзките със съседите (broadcast) – всяко у-во знае валидните пътища до всеки един възел

TC Message Format



Advertised Neighbor Sequence Number (ANSN)- номер за последователност**, който се асоцира с рекламирания набор от съседи. Всеки път когато се установи промяна, този номер се увеличава с 1**.

Advertised Neighbor Main Address – съдържа адреса на всички съседи, които са избрали този възел за MRP

AODV: Ad-Hoc On-Demand Distance Vector

- Реактивен протокол за маршрутизиране - Маршрутът се създава по поискване - когато възел се нуждае да изпрати пакет до друг. За насочване на съобщенията се използва информацията в маршрутизиращата таблица
- Ако възел-сорс се нуждае да изпрати пакет до възел-дестинация се изпращат контролни съобщения - route request и route reply messages
- Междинните у-ва поддържат маршрутизираща информация за всеки активен път, който минава през тях. Така те могат да препратят информацията към крайната дестинация
- Всяко у-во поддържа свързаност със съседите си по актуалните маршрути през него посредством периодични контролни съобщения.
- Локализиращ ефект при смяна на топлогията - ако има промяна някъде по пътя се изпраща route error съобщение и отпадналата предишна информацията се изтрива.
- Минимизира служебна информация (overhead), която се включва в заглавната част на всеки пакет (за разлика от DSR протокола)

AODV Характеристики

- Протокол свободен от цикли при намиране на маршрута:

  - предотвратяване на цикли и разпознаване на по-нови пътища - номер за последователност (Sequence #)
- За насочване на съобщенията се използва информацията в маршрутизиращата таблица- активните пътища в MANETs през това у-во

  Destination IP address, Destination Sequence #, Hop Count, Next Hop, List of Precursors (препращат пакетите), Lifetime, Routing Flags

AODV Характеристики (2)

- Използват се таймери, след изтичането на които информацията отново се изтрива без значение дали пътя е активен или не. Поправяне на маршрута, TTL време
- Мащабируемост - до 10,000 възела
- Лошо качество при предаване на данни при гъсти мрежи

Модификации

- Multicast AODV protocol (MAODV)

няколко next hops, групов лидер променя номера за последователност

- Ad-hoc On-demand Multipath Distance Vector (AOMDV)

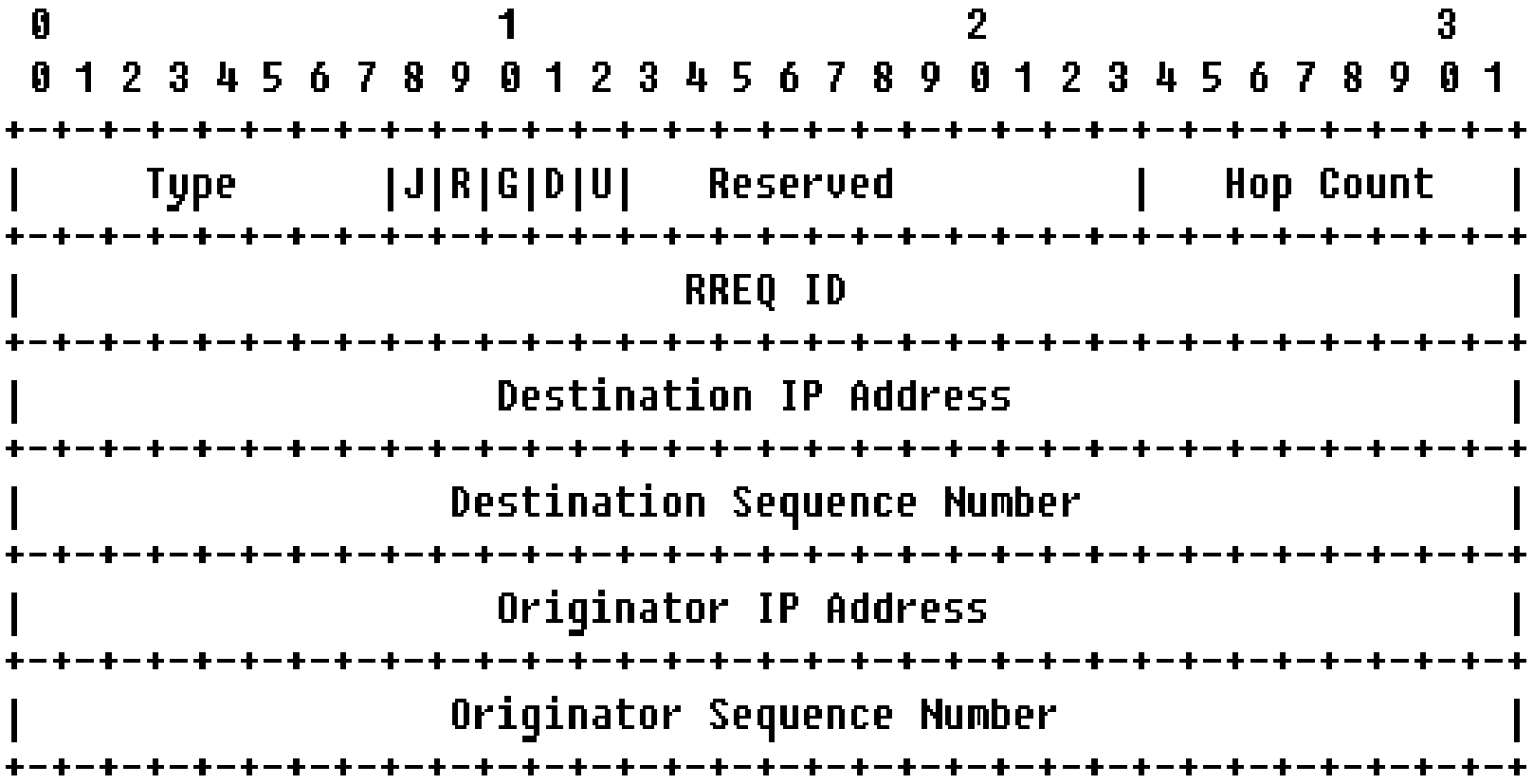
позволява използването на алтернативни пътища за намаляване на броя на процедурите за откриване на нов път

AODV Откриване на маршрут

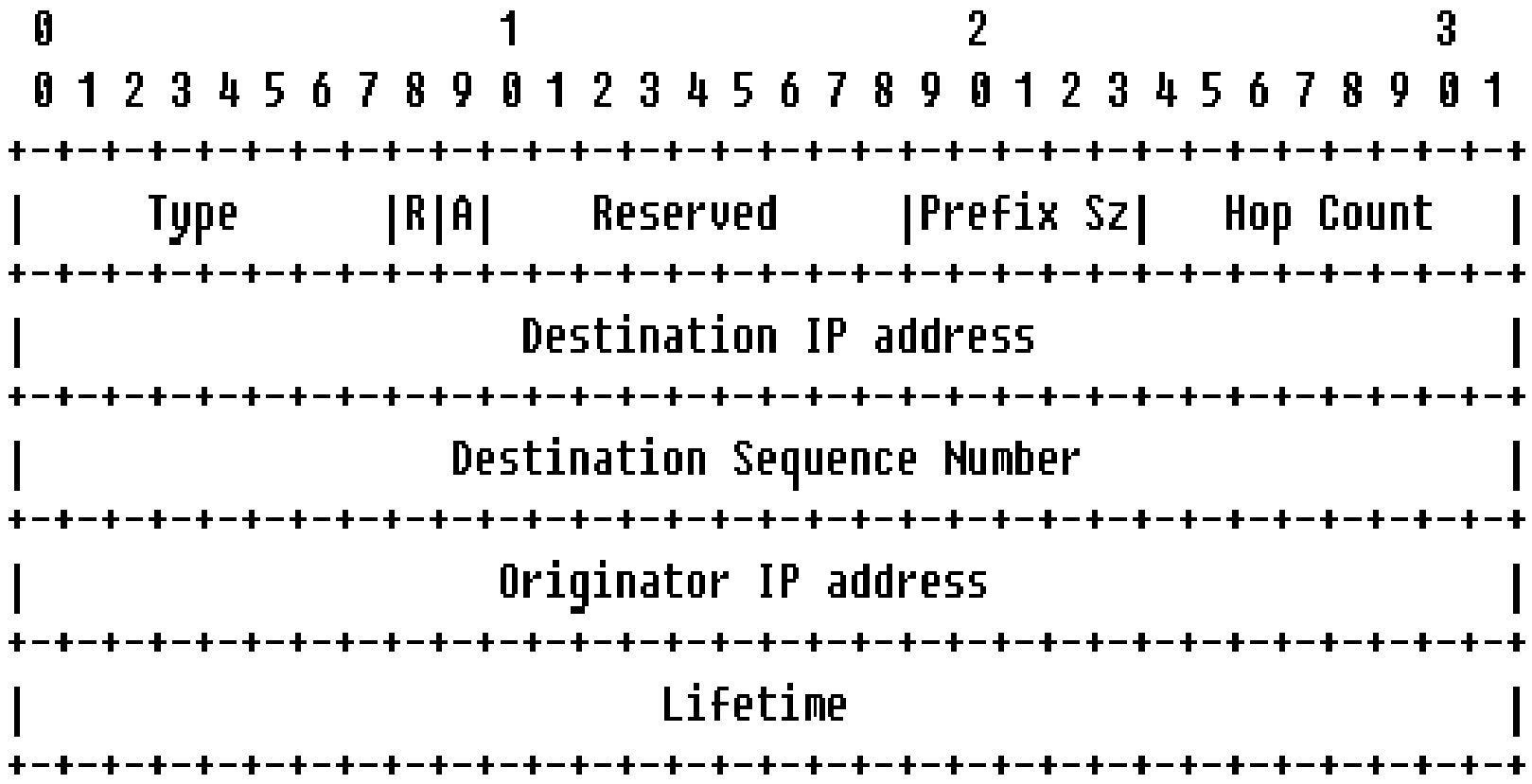
- RREQ (route request) - broadcast
  - По пътя се записва обратния път до източника
  - RREQ message < RREQ ID; Destination IP address; Destination Sequence #; Source IP address; Source Sequence # ; hop_count >
- RREP (route reply) - unicast back
  - От дестинацията

  или
  - От междинен възел, ако той знае актуален път до дестинацията
  - Всеки възел кешира пътя обратно до възела, поискал маршрут
- RERR (route error)
  - Възлите бързо реагират на прекъснати връзли и промяна в топлогията на мрежата, тъй като следят статуса на next-hop по активния път
  - Когато се прекъсне връзка, AODV информира засегната мрежа от възли да девалидират всички пътища до тази дестинация (които използват прекъсната връзка)

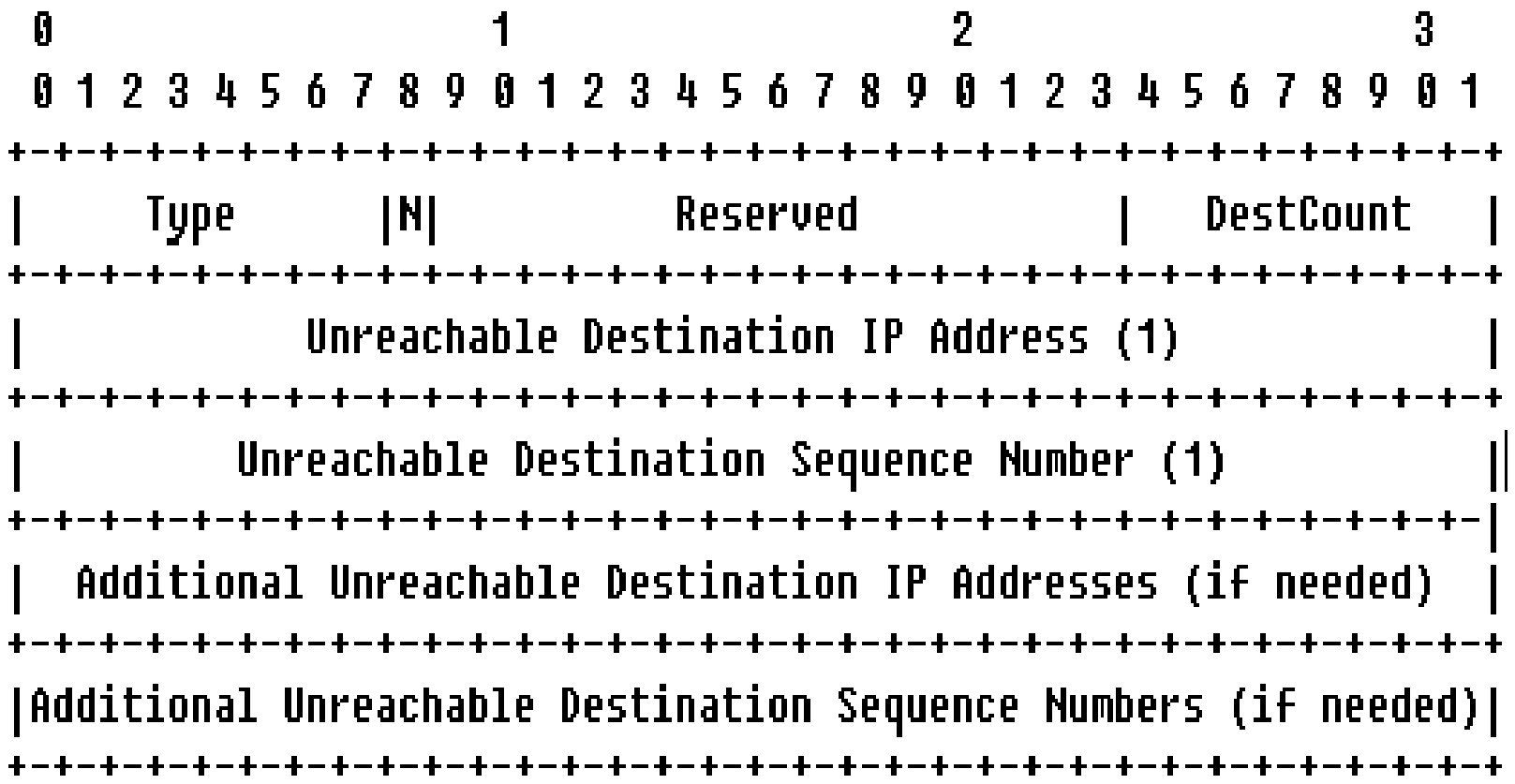
Route Request (RREQ) Message Format



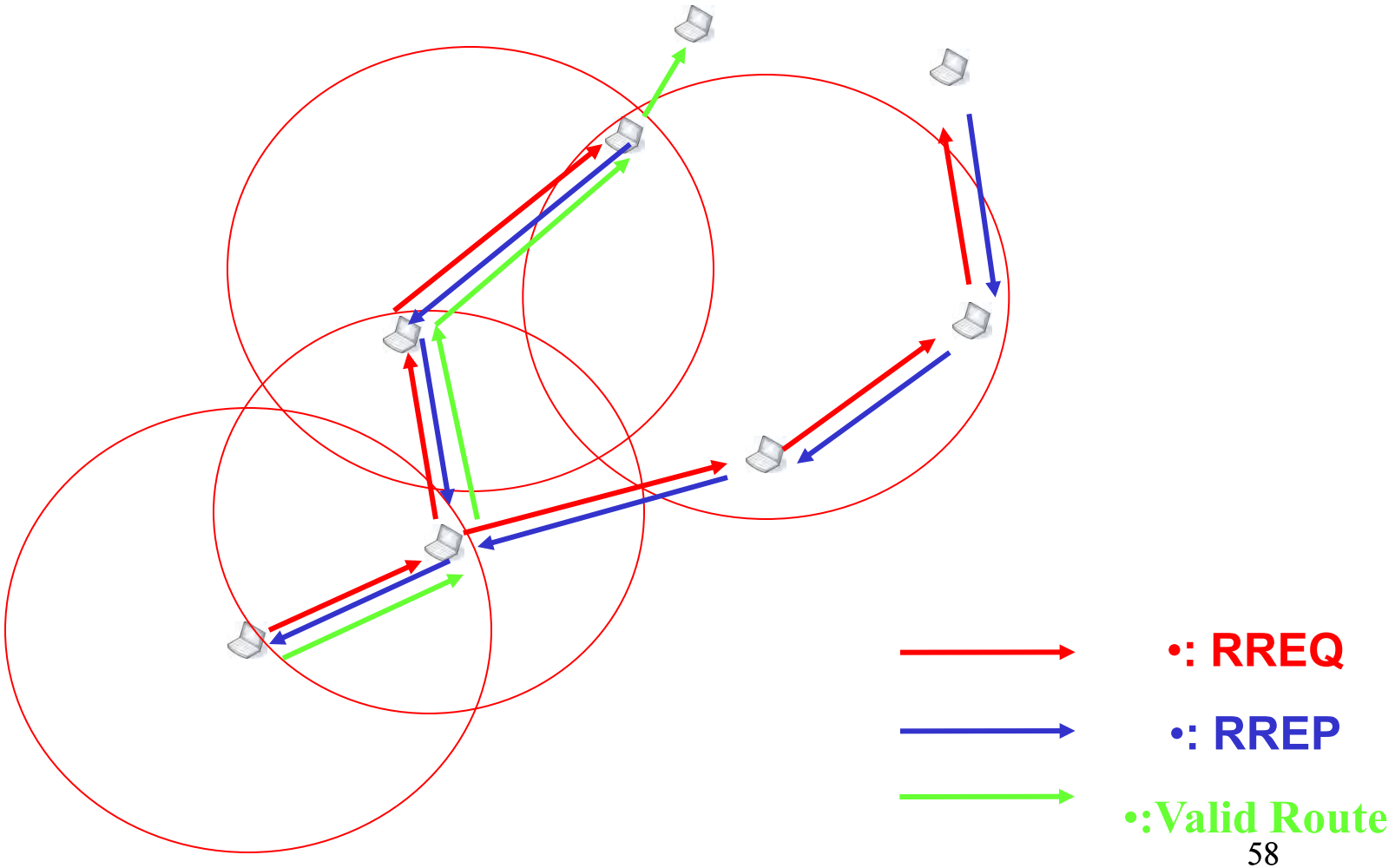
Route Reply (RREP) Message Format



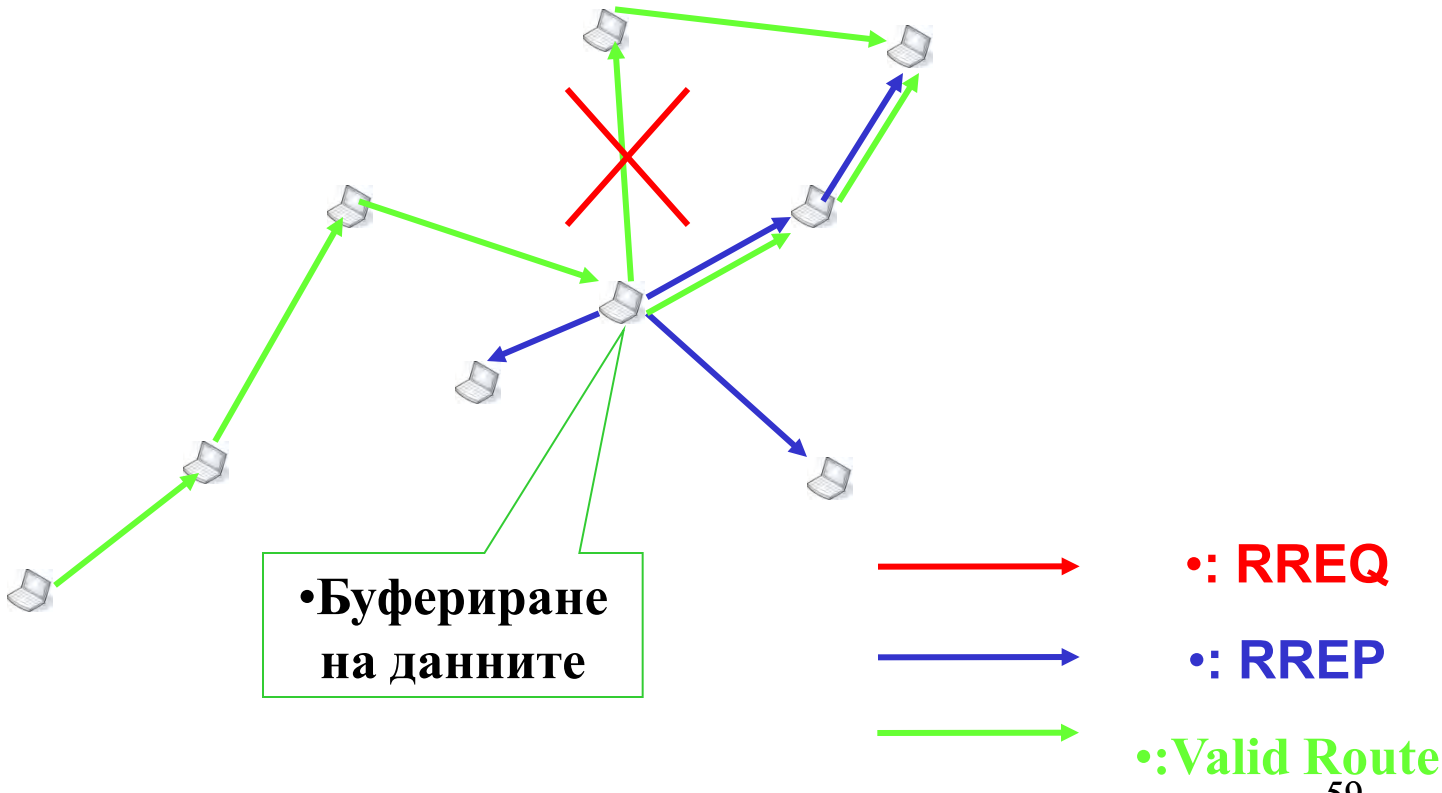
Route Error (RERR) Message Format

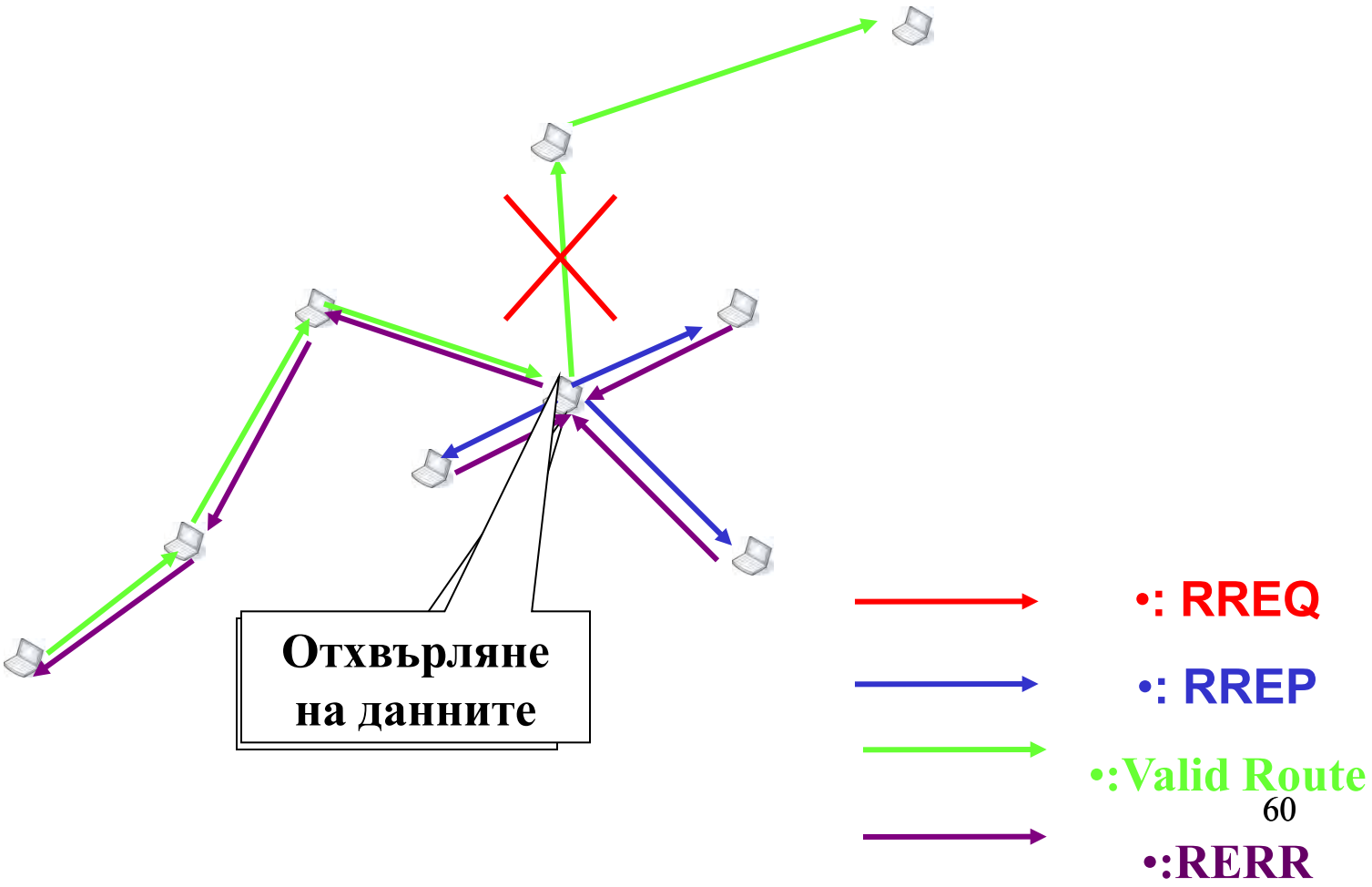


AODV Откриване на път



AODV Локално Поправяне на Пътя




Отхвърляне
на данните
•: RREQ
•: RREP
•:Valid Route
•:RERR

Симулиране на MANETs

- Има различни среди за анализиране на протоколите за маршрутизация в MANETs. Едно от решенията е използване на симулатори като OPNET, GloMoSim , NS2/3
- UPPAAL - Toolbox for modeling and verification of real-time systems
- Защо в UPPAAL? – формалното моделиране позволява откриване на скрити грешки и гранични случаи чрез изследване на всички възможни състояния, тъй като UPPAAL проверява всики възможни състояния на модела
- NS3 и другите дискретни симулатори откриват грешки, но могат да пропуснат граничните случаи и не е сигурно, че системата се държи според очакванията

Моделиране, симулиране и верификация на MANETs в UPPAAL

- UPPAAL – софтуер с дескриптивен и графичен интерфейс за описание на системни модели
- Wibling първи моделира MANET като мрежа от автомати в UPPAAL
- Разработва мобилност, локално broadcast и unicast.
- Формалното моделиране в UPPAAL е процес за верифициране на ранен етап от дизайна на мрежата
- Премахват се скрити грешки чрез тестване на всички възможни състояния на модела за определен сценарий

UPPsala University in Sweden
AALborg University in Denmark

- Първа версия - 1995
- Приложен за MANETs - 2005
- Подходящ за системи, които могат да бъдат моделирани като мрежа от времеви автомати разширени с цели променливи, таймери с реално време, структури от данни и синхронизиращи канали за предаване на данни

- Текуща версия V.4.1.15

UPPAAL – Развойна Среда

Uppaal състои се от три главни части

- **Системен Редактор и Език за Моделиране (description language)**

  -език за моделиране, който описва поведението на системата като мрежа от автомат с краен брой състояния и транзакции, разширени с таймери

  -гардове и команден език с типове данни (интервали за цели числа, масиви и др.)

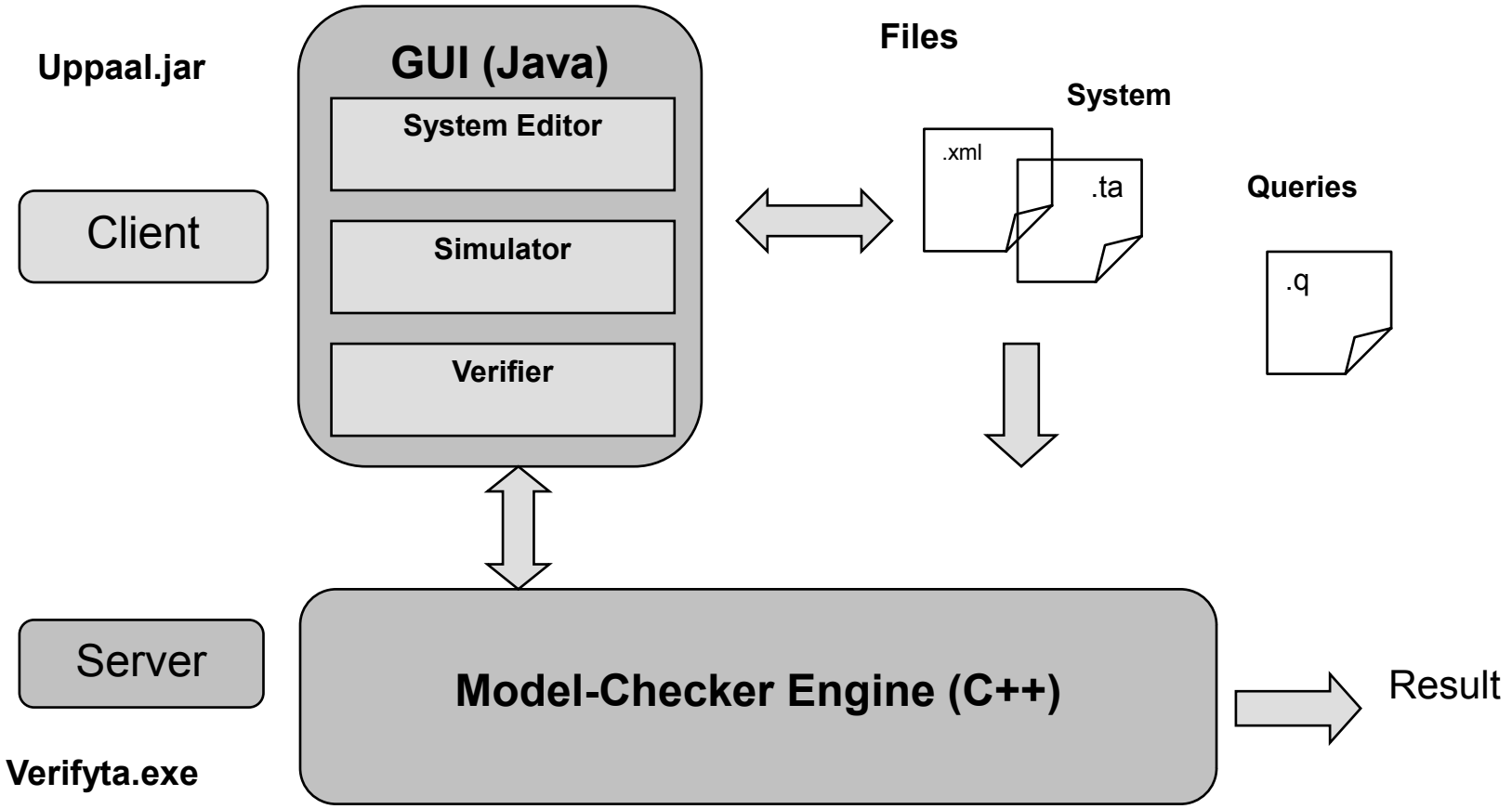
- **Симулатор (simulator)**

  -осигурява изследване на възможните динамични работни състояния на системата на ранен етап на проектирането

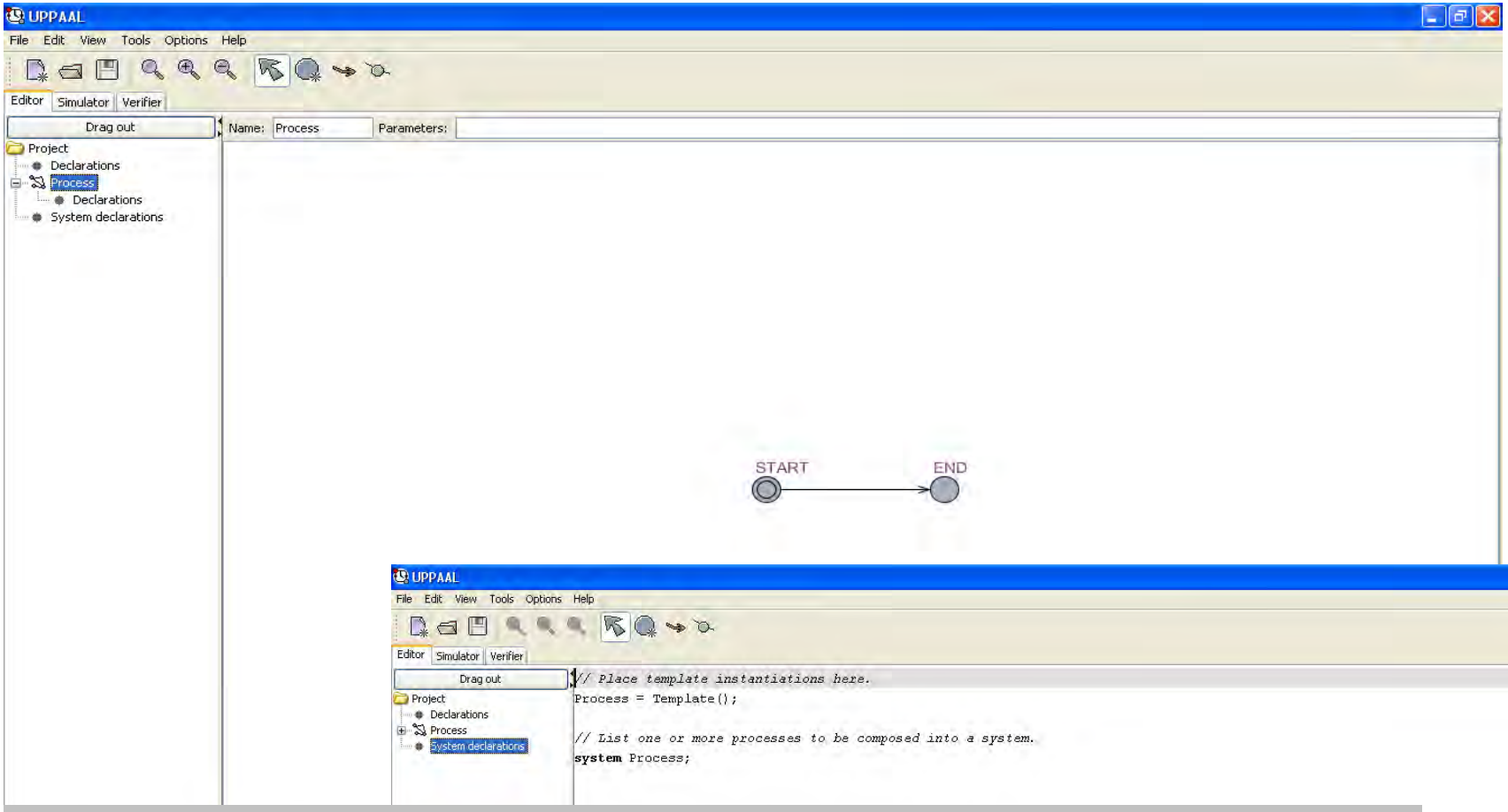
- **Верифициране на модела (model-checker)**

  -проверява инварианти и вероятността за достигане на определени свойства чрез изследване на пространство на състоянията, т.е. осъществява се крос-продукт (cross product) по всички локации и транзакции

UPPAAL Архитектура



UPPAAL Tool Box – GUI Editor



Компонентите на системата се описват с template (наречен “process **“)**

- Състои се от 2 локации и една транзакция

- **template** се дефинира с група параметри (global or local), които могат да бъдета от всеки тип (int, chan, clock)

UPAALL Локации (Locations)

Invariants

- Прогресивно (invariant) условие, което изразява ограничения върху таймерите (часовник)
- Контролира колко дълго да се остане в дадена локация

Типове Локации:

- initial or initial with invariants – начало на процеса
- normal or normal with invariants
- urgent – актуалния процес да направи транзакция без закъснение
- committed – локацията трябва да бъде напусната веднага

Locations

initial urgent committed normal


invariant x<1

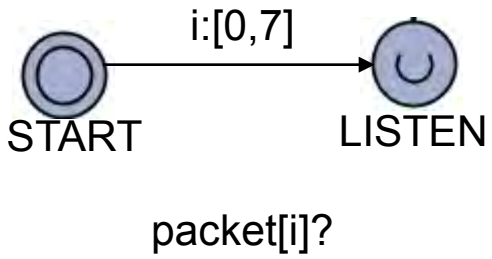
UPPAAL Транзакции - Edges

- Edge – линията между две локации

Може да бъде от типа:

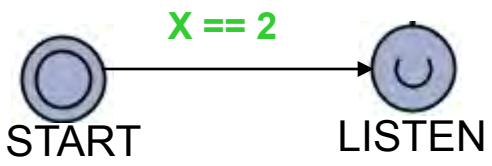
selections, *guards*, *synchronization channels* и *updates*

- ***Selections*** неопределено присвоява стойност от зададен интервал



UPPAAL Транзакции – Edges (2)

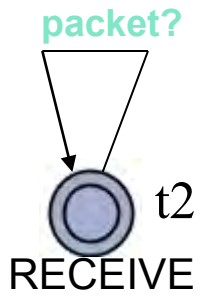
- ***Guards***
  - Условие за стойността на часовника
  - Условие за стойността на целите или булеви променливи (ако бъдат изпълнени, транзакцията се осъществява)

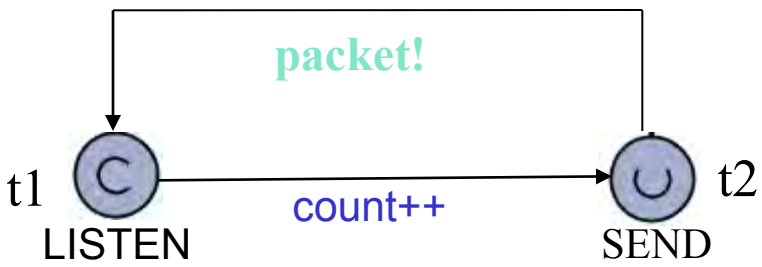


UPPAAL Транзакции – Edges (3)

Synchronizations

- Два процеса променят своята локация на една и съща симулационна стъпка (t2)
- Синхринизацията се осъществява чрез канали – unicast, broadcast и urgent
- За синхронизиране на два процеса – маркиране на канала (channel variable) с “!” или “?”





Процес2 Процес1

UPPAAL Транзакции – Edges (4)

- ***Updates***

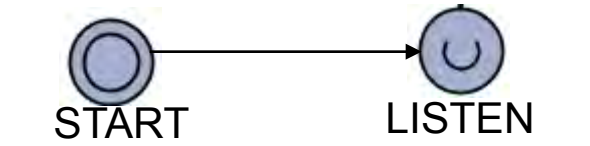
- при преминаване на транзакцията update expression се изчислява (оценява)

Expressions:

- number of clock resets – рестартиране на време

- assignments to integer variables- присвояване цяло число на променлива

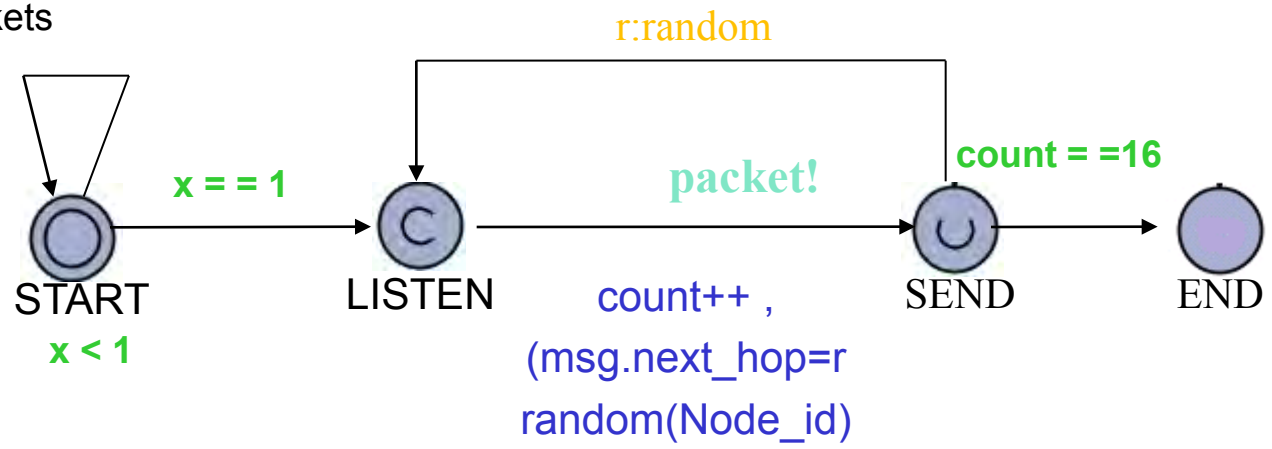
- functions (пример: WSN shortest tree protocol)



UPAALL Примерна Система Transmitter/Receiver

```
clock x ;
broadcast channel packet ;
int count ;   //number of sent packets
typedef int[0,2] Receiver_id;
```

Transmitter model



Receiver models

Global Listing declarations

```
//Analysis of RECEIVER/TRANSMITTER Using UPPAAL
const int N = 4; //number of receivers
const int M = 1; //number of  transmitters

broadcast chan packet;
bool RadioBusy = false;//0 is false

typedef int[0,N-1] Receiver_id;
typedef int[0,M-1] Transmitter_id;
typedef int[0, N-1] random;
meta struct{
      Receiver_id next_hop ;
            } msg ;
```

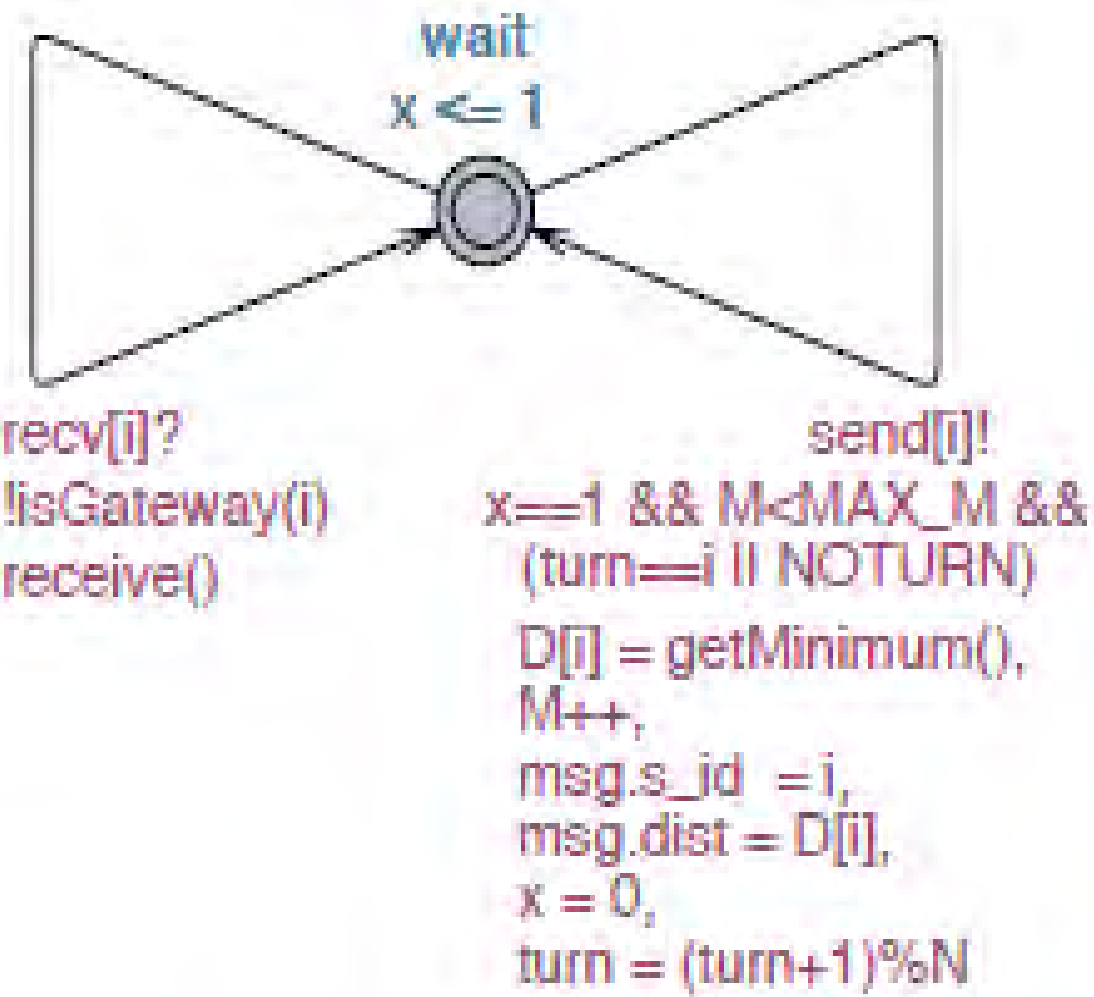
Local Listing declarations

Name: Transmitter

Parameters:const Transmitter_id i

```
clock x;
int count ;   //number of sent packets
```

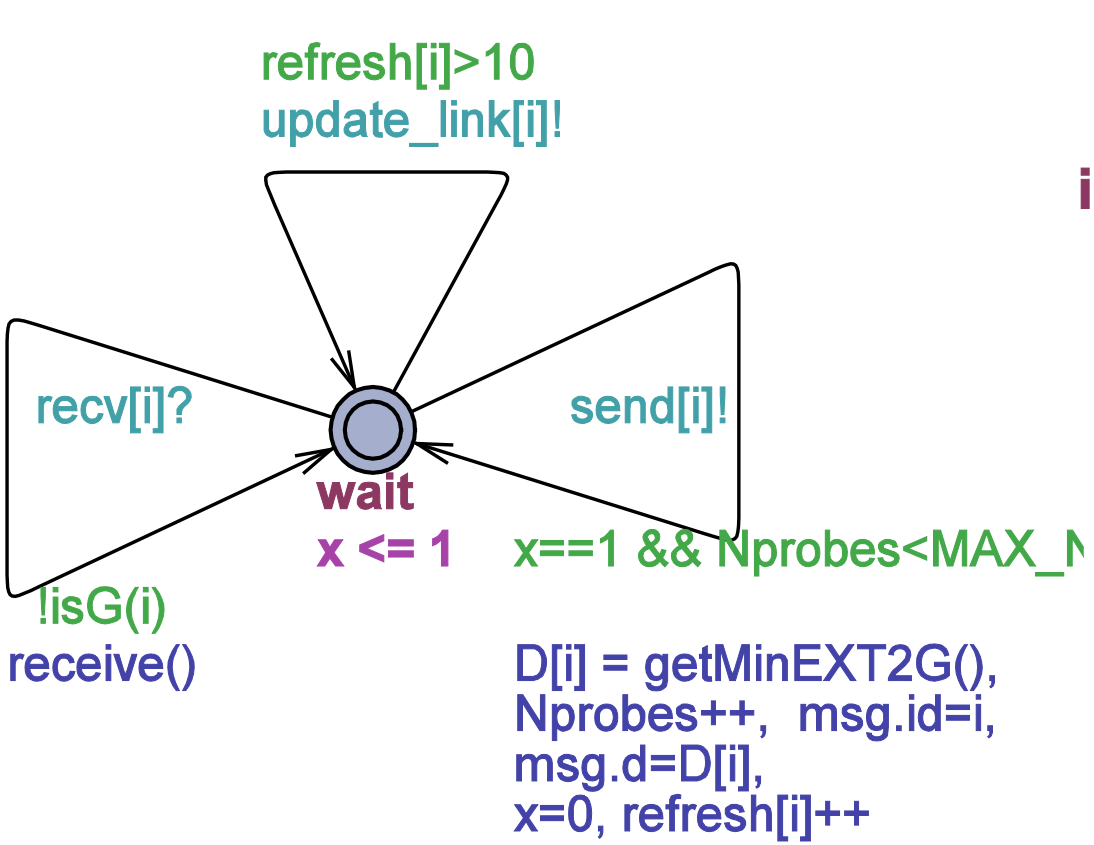
Пример: Анализ на WSN shortest tree protocol в UPPAAL

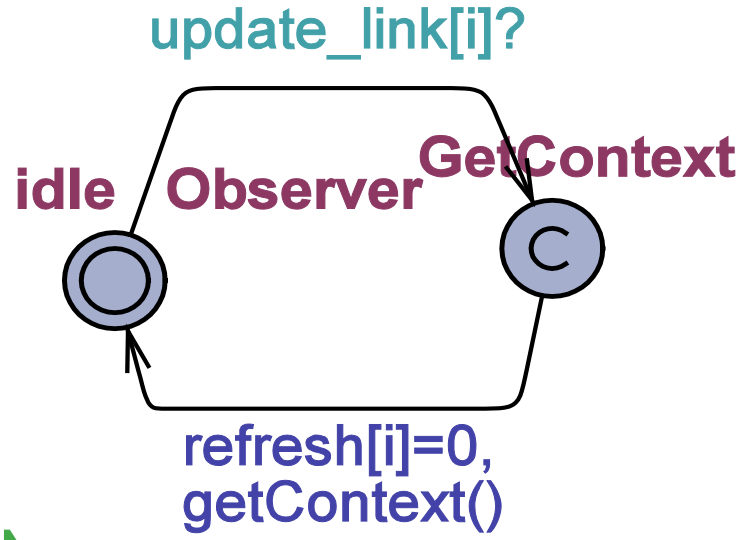


Local Listing declarations

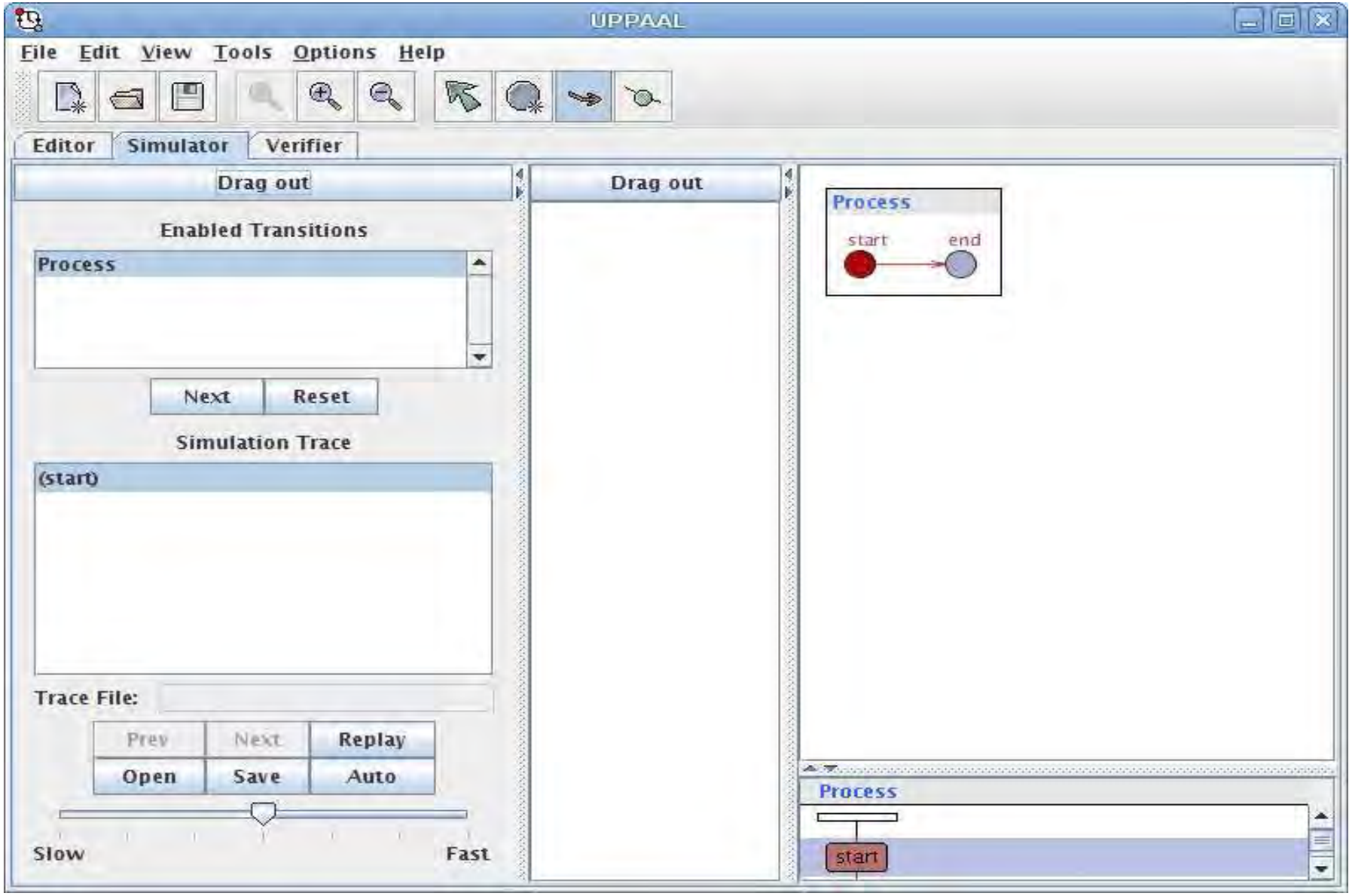
```
//Function returns the min distance to the gateway
int getMinimum(){
meta int minval = MAX_DIST; // To hold the min  found so f a r
meta int try; // To hold the next v a l u e
if (isGateway(i) ) return 0 ; // Minimum dist to G is 0
if ( M == 0 ) return MAX_DIST; // First round returns MAX DIST
for( j : Node_id ){
        if ( R[j] > 0 && j != i && D[j] < MAX_DIST ){
                try = M/R[j] + D[j] ;
                if ( (M % R[j] ) >= (R[j] / 2) ) try++; // Round to nearest int
                if ( try <= minval ){
                minval= try;
                parent = j ;
}}}
return minval;
                }
```

Още UPAALL Примери

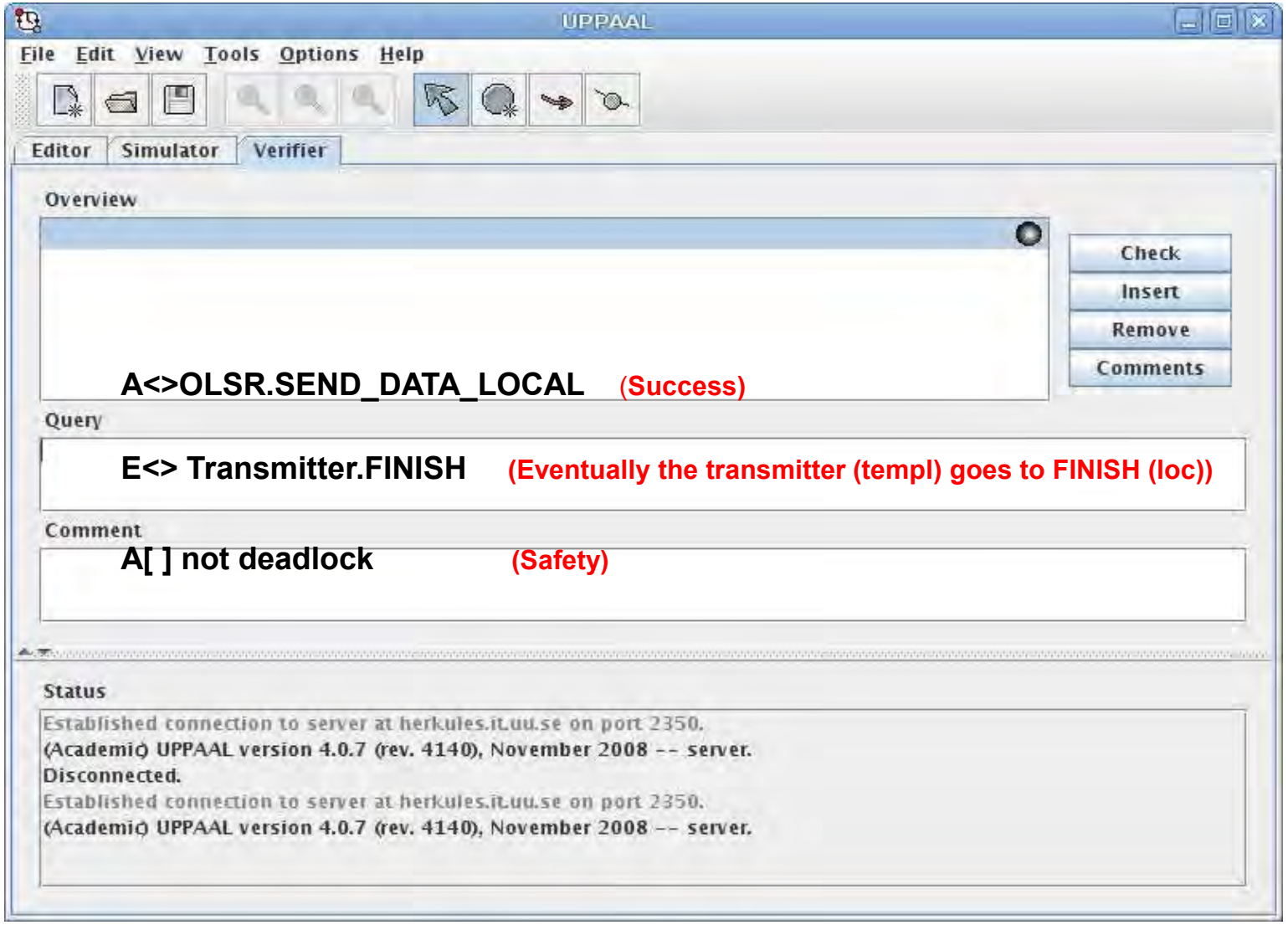




UPPAAL Tool Box – GUI Simulator



UPPAAL Toolbox – GUI Verifier

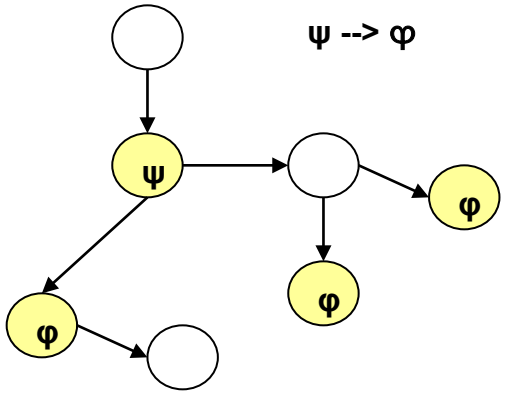


Query Types - Задаване на Заявки към Системата

- A[ ] - свойство за сигурност (always)
  - A[ ] not deadlock – най-общо, изисква много памет
- E<> - свойство за достижимост (дали се достига до дадено състояние)
- Liveness Properties – истина за всички възможни симулации
  - “ leads to” или “response property”

ψ --> ϕ : ако ψ е изпълнено,

то евентуално ϕ ще бъде изпълнено



e.g. whenever a message is sent, then eventually it will be received:

- Node(4).SEND_DATA --> Node(3).SEND_DATA_LOCAL

Конкретни Изследвания, проведени в ИСИР-БАН
R-OLSR

Разработен, моделиран, симулиран и верифициран в UPPAAL е нов модел на протокол за повишаване сигурността на маршрутизацията в MANETs - R-OLSR

- Локален мониторинг и изолиране на мобилни устройства, които застрашават маршрутизирането в MANETs

Разпределено Локално Наблюдение:

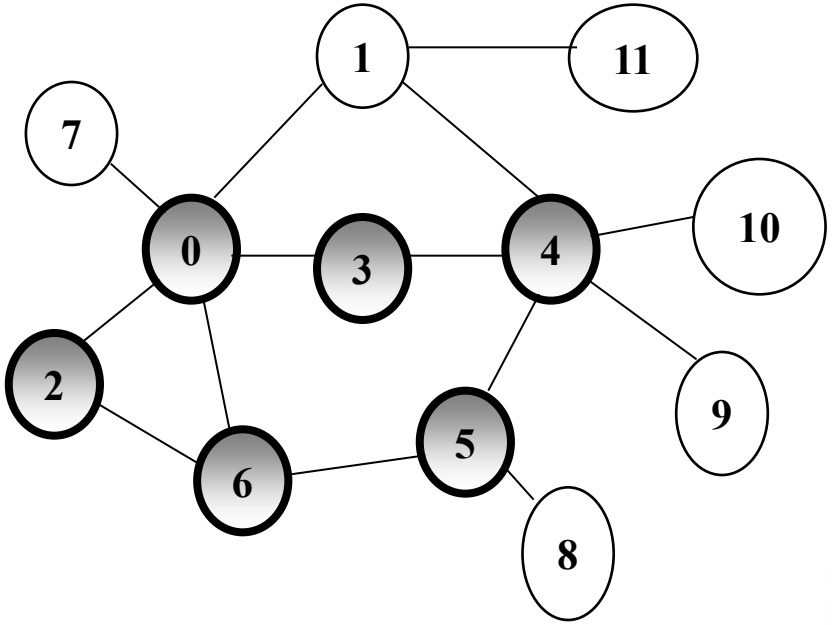
- ✓ Базирано на евристика за типа на трафика на съседите
- ✓ Само MRP анализират трафика за няколко определени последователни интервала и променят параметрите на радиовръзките на устройството или го изключват като съсед

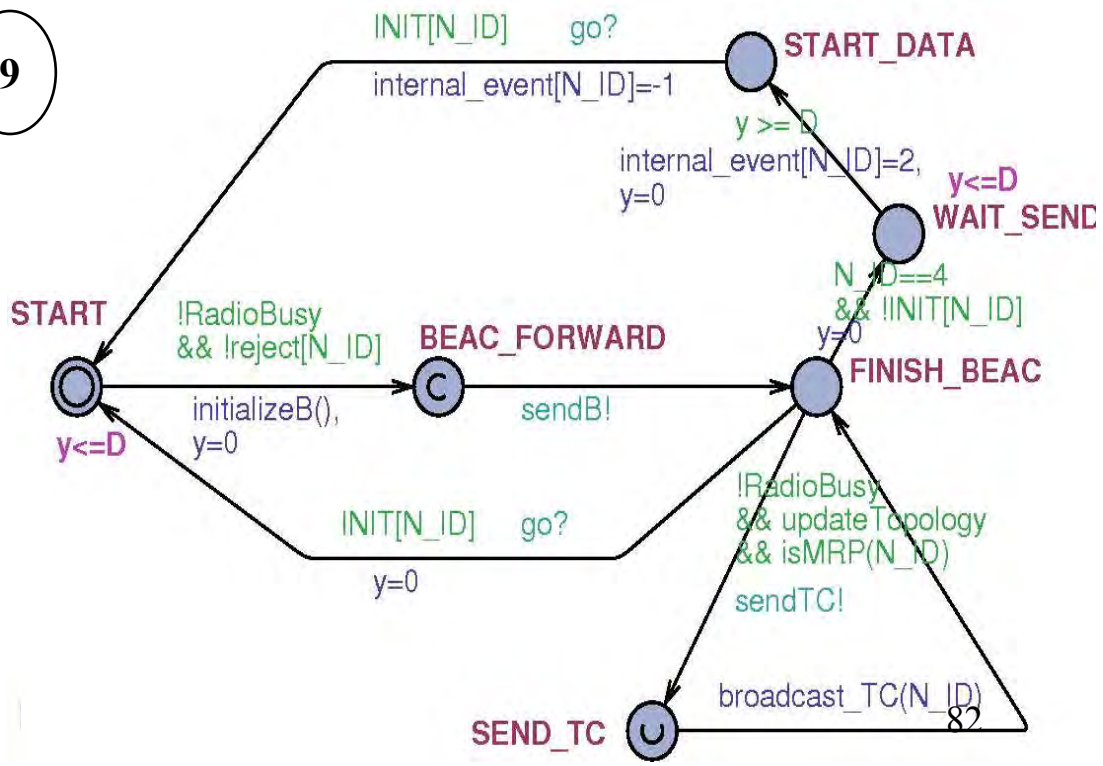
  N_ID(3)_Willingness = NEVER

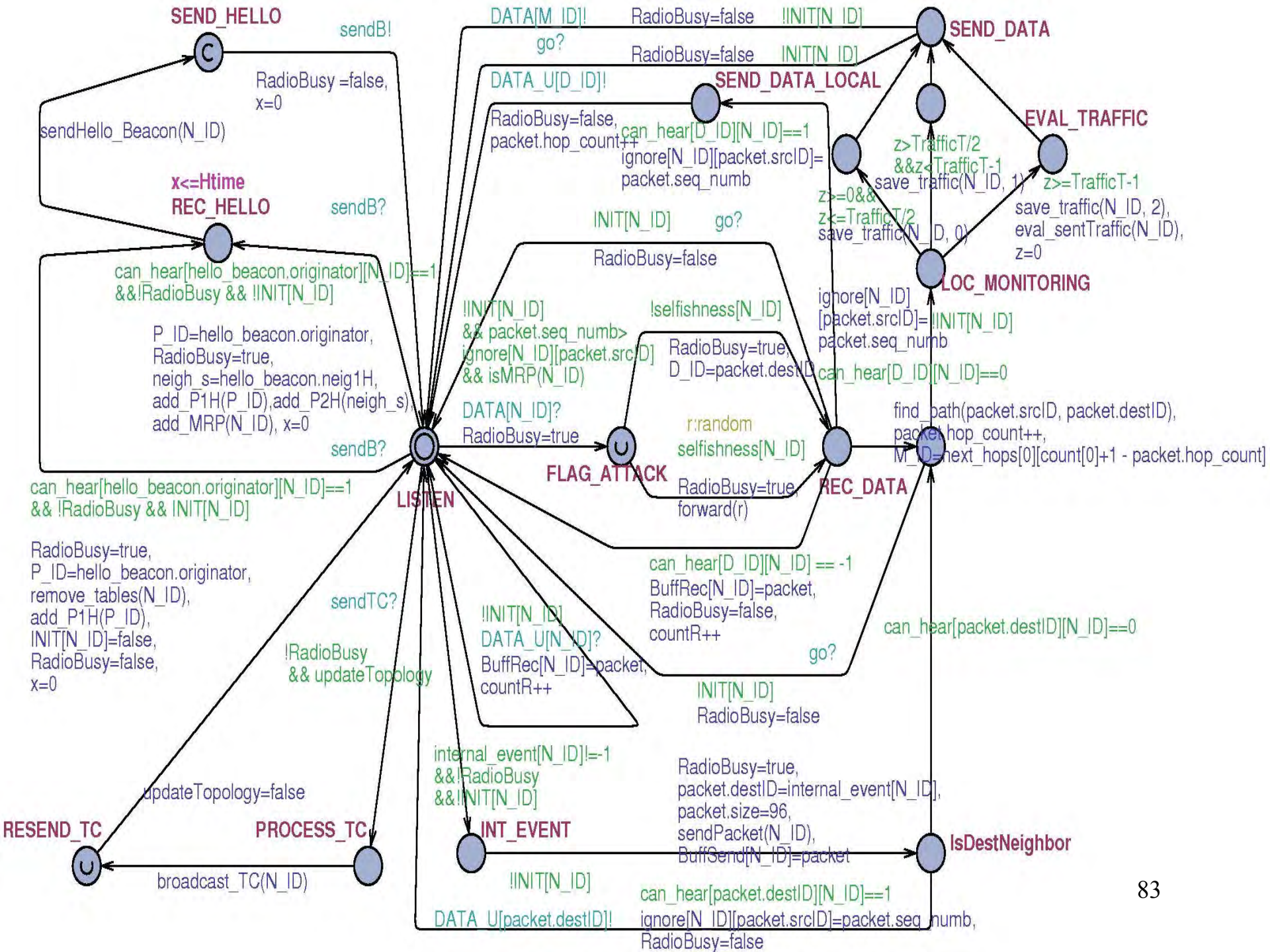
  N_ID(1)_Willingness = WILL_ALWAYS - MRP

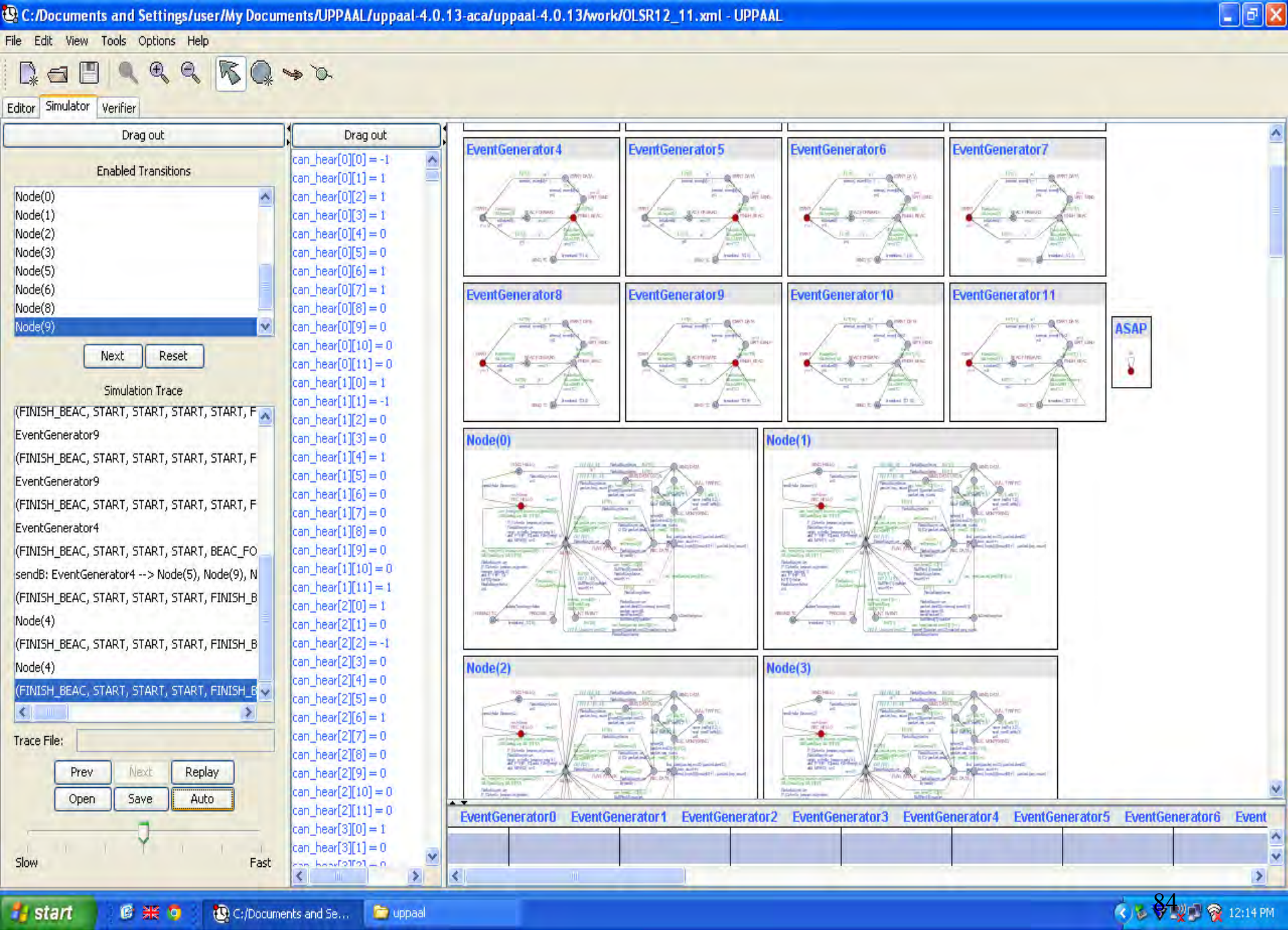
- ✓ Преодолява недостатъците на локалните схеми, описани в литературата, които са базирани на потвърждаване (acknowledgment) от у-вото, което получава пакетите

UPPAAL Модел за Event Generator

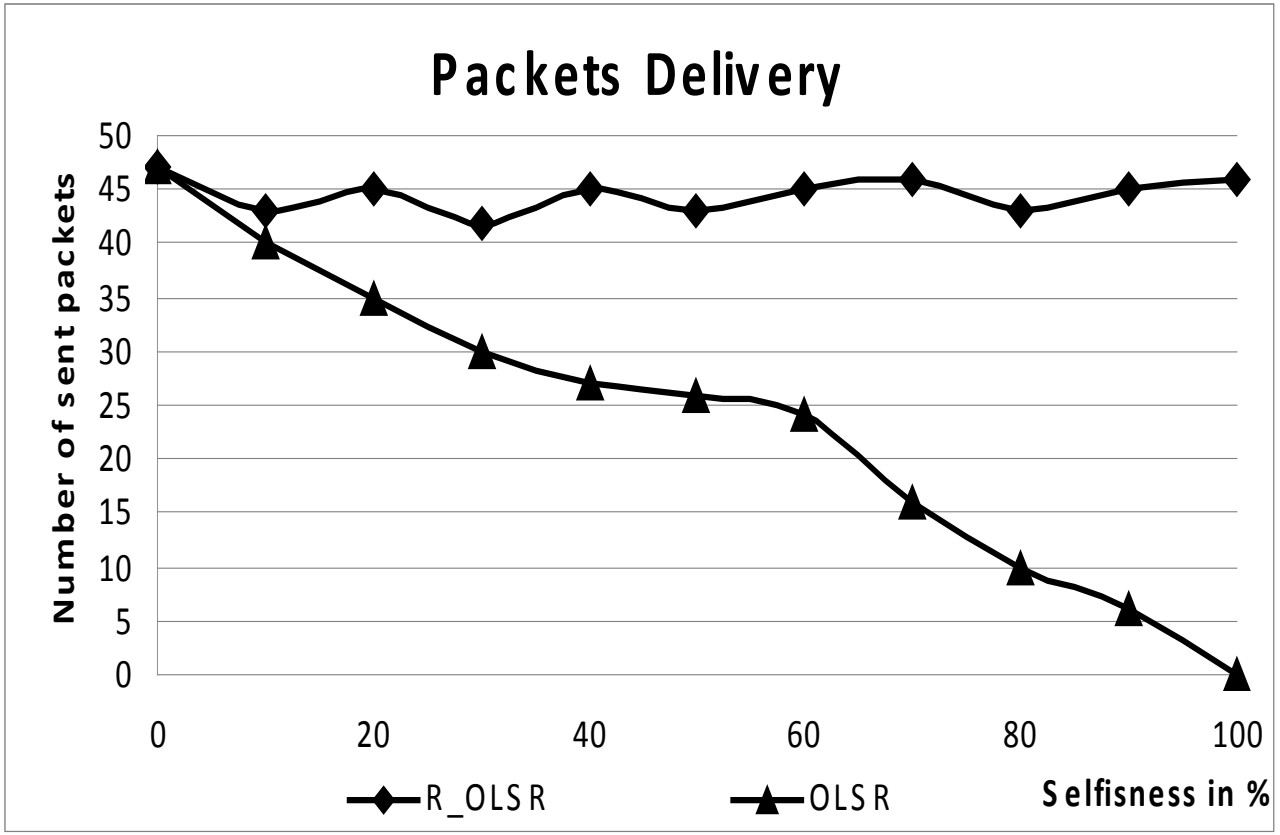





SEND_HELLO
sendB!
RadioBusy =false,
x=0
sendHello_Beacon(N_ID)
x<=Htime
REC_HELLO
sendB?
can_hear[hello_beacon.originator][N_ID]==1
&&!RadioBusy && !INIT[N_ID]
P_ID=hello_beacon.originator,
RadioBusy=true,
neigh_s=hello_beacon.neig1H,
add_P1H(P_ID),add_P2H(neigh_s),
add_MRP(N_ID), x=0
sendB?
can_hear[hello_beacon.originator][N_ID]==1
&& !RadioBusy && !INIT[N_ID]
RadioBusy=true,
P_ID=hello_beacon.originator,
remove_tables(N_ID),
add_P1H(P_ID),
INIT[N_ID]=false,
RadioBusy=false,
x=0
LISTEN
DATA[M_ID]!
go?
RadioBusy=false
!INIT[N_ID]
SEND_DATA
RadioBusy=false
INIT[N_ID]
DATA_U[D_ID]!
SEND_DATA_LOCAL
RadioBusy=false,
packet.hop_count++
can_hear[D_ID][N_ID]==1
ignore[N_ID][packet.srcID]=
packet.seq_numb
INIT[N_ID]
go?
RadioBusy=false
!INIT[N_ID]
&& packet.seq_numb>
ignore[N_ID][packet.srcID]
&& isMRP(N_ID)
DATA[N_ID]?
RadioBusy=true
!selfishness[N_ID]
RadioBusy=true,
D_ID=packet.destID
r:random
selfishness[N_ID]
RadioBusy=true,
forward(r)
FLAG_ATTACK
REC_DATA
EVAL_TRAFFIC
z>TrafficT/2
&&z<TrafficT-1
save_traffic(N_ID, 1)
z>=TrafficT-1
z>=0&&
z<=TrafficT/2
save_traffic(N_ID, 0)
save_traffic(N_ID, 2),
eval_sentTraffic(N_ID),
z=0
LOC_MONITORING
ignore[N_ID]
[packet.srcID]=!INIT[N_ID]
packet.seq_numb
can_hear[D_ID][N_ID]==0
find_path(packet.srcID, packet.destID),
packet.hop_count++,
M_ID=next_hops[0][count[0]+1 - packet.hop_count]
can_hear[D_ID][N_ID] == -1
BuffRec[N_ID]=packet,
RadioBusy=false,
countR++
go?
can_hear[packet.destID][N_ID]==0
sendTC?
!RadioBusy
&& updateTopology
!INIT[N_ID]
DATA_U[N_ID]?
BuffRec[N_ID]=packet,
countR++
INIT[N_ID]
RadioBusy=false
internal_event[N_ID]!=-1
&&!RadioBusy
&&!INIT[N_ID]
RadioBusy=true,
packet.destID=internal_event[N_ID],
packet.size=96,
sendPacket(N_ID),
BuffSend[N_ID]=packet
updateTopology=false
RESEND_TC
PROCESS_TC
INT_EVENT
IsDestNeighbor
broadcast_TC(N_ID)
!INIT[N_ID]
DATA_U[packet.destID]!
can_hear[packet.destID][N_ID]==1
ignore[N_ID][packet.srcID]=packet.seq_numb,
RadioBusy=false

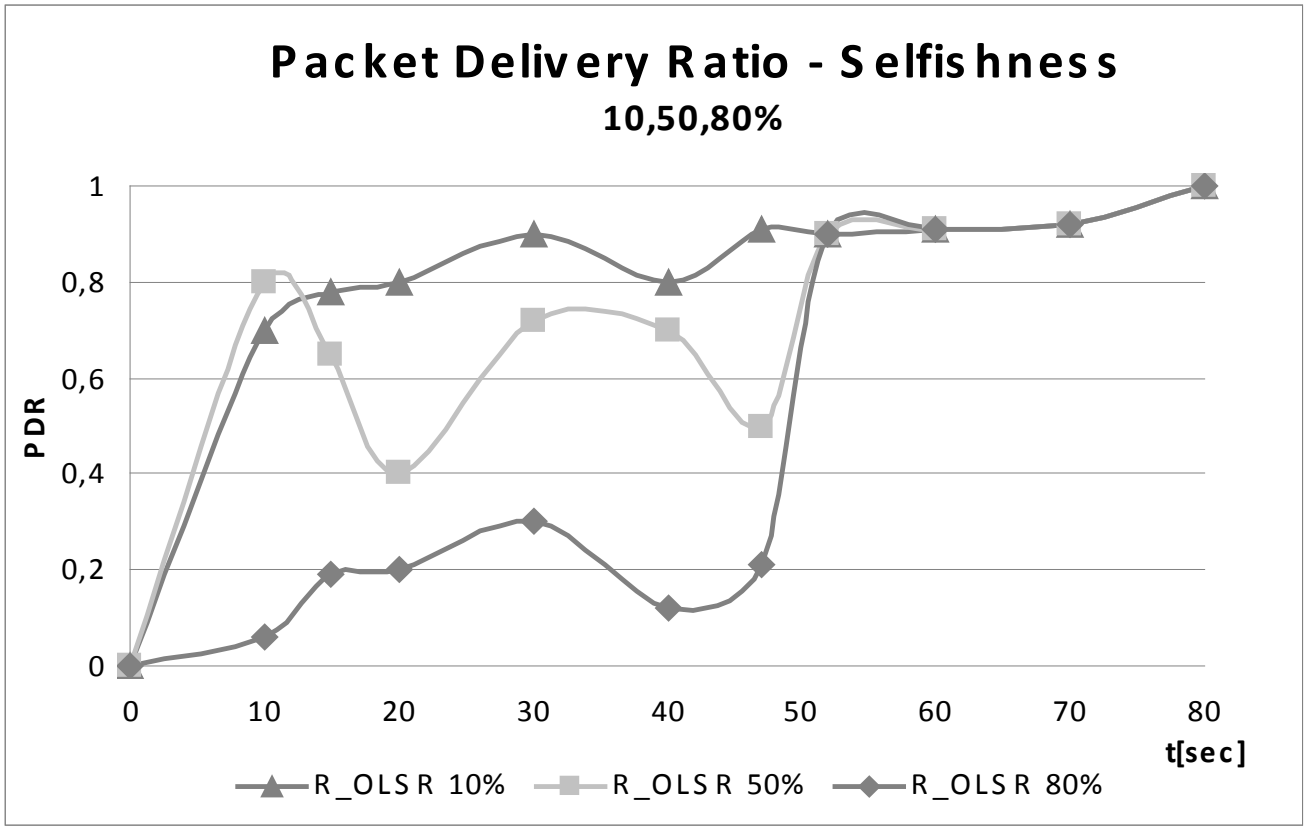

C:/Documents and Settings/user/My Documents/UPPAAL/uppaal-4.0.13-aca/uppaal-4.0.13/work/OLSR12_11.xml - UPPAAL
File Edit View Tools Options Help
Editor Simulator Verifier
Drag out
Enabled Transitions
Node(0)
Node(1)
Node(2)
Node(3)
Node(5)
Node(6)
Node(8)
Node(9)
Next Reset
Simulation Trace
(FINISH_BEAC, START, START, START, START, F
EventGenerator9
(FINISH_BEAC, START, START, START, START, F
EventGenerator9
(FINISH_BEAC, START, START, START, START, F
EventGenerator4
(FINISH_BEAC, START, START, START, BEAC_FO
sendB: EventGenerator4 --> Node(5), Node(9), N
(FINISH_BEAC, START, START, START, FINISH_B
Node(4)
(FINISH_BEAC, START, START, START, FINISH_B
Node(4)
(FINISH_BEAC, START, START, START, FINISH_B
Trace File:
Prev Next Replay
Open Save Auto
Slow Fast
Drag out
can_hear[0][0] = -1
can_hear[0][1] = 1
can_hear[0][2] = 1
can_hear[0][3] = 1
can_hear[0][4] = 0
can_hear[0][5] = 0
can_hear[0][6] = 1
can_hear[0][7] = 1
can_hear[0][8] = 0
can_hear[0][9] = 0
can_hear[0][10] = 0
can_hear[0][11] = 0
can_hear[1][0] = 1
can_hear[1][1] = -1
can_hear[1][2] = 0
can_hear[1][3] = 0
can_hear[1][4] = 1
can_hear[1][5] = 0
can_hear[1][6] = 0
can_hear[1][7] = 0
can_hear[1][8] = 0
can_hear[1][9] = 0
can_hear[1][10] = 0
can_hear[1][11] = 1
can_hear[2][0] = 1
can_hear[2][1] = 0
can_hear[2][2] = -1
can_hear[2][3] = 0
can_hear[2][4] = 0
can_hear[2][5] = 0
can_hear[2][6] = 1
can_hear[2][7] = 0
can_hear[2][8] = 0
can_hear[2][9] = 0
can_hear[2][10] = 0
can_hear[2][11] = 0
can_hear[3][0] = 1
can_hear[3][1] = 0
EventGenerator4
EventGenerator5
EventGenerator6
EventGenerator7
EventGenerator8
EventGenerator9
EventGenerator10
EventGenerator11
ASAP
Node(0)
Node(1)
Node(2)
Node(3)
EventGenerator0 EventGenerator1 EventGenerator2 EventGenerator3 EventGenerator4 EventGenerator5 EventGenerator6 Event
start
C:/Documents and Se...
uppaal
12:14 PM

Резултати от Симулацията



- **Зависимост на доставените пакети данни от**
- **степента на егоистичност на безжичните устроствата**

Резултати от Симулацията (2)



- **Динамична характеристика на изпратените пакети данни в зависимост от степента на егоистичност на безжичните устройствата, на които е инсталиран R-OLSR протокола**

Инсталиране на UPPAAL-current version 4.1.15 released April 19th, 2013

UPPAAL

www.uppaal.org

RELATED TOOLS: TIMES | CORA | TRON | TIGA | SMC | COVER | PORT | PRO

UPPAAL

Home

Home | About | Documentation | Download | Examples | Web Help | Bugs

UPPAAL is an integrated tool environment for modeling, validation and verification of real-time systems modeled as networks of timed automata, extended with data types (bounded integers, arrays, etc.).

The tool is developed in collaboration between the Department of Information Technology at Uppsala University, Sweden and the Department of Computer Science at Aalborg University in Denmark.

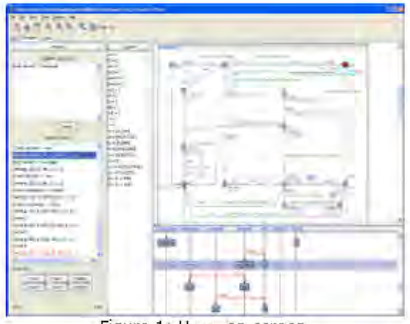


Figure 1: UPPAAL on screen.

Download

News: The current official release is UPPAAL 4.0.13 (Sep 27, 2010). Compared to version 3, the 4.0 release is the result of over 2.5 years of additional development, and many new features and improvements are introduced (see also this release note and the web help section new features). To support models created in previous versions of UPPAAL, version 4.0 can convert most old models directly from the GUI (alternatively it can be run in 3.4 compatibility mode by defining the environment variable `UPPAAL_OLD_SYNTAX`, see also item 2 of the FAQ).

Since Feb 26 2008, we also distribute a development snapshot of the forthcoming UPPAAL 4.2. The current development snapshot version is 4.1.15 released April 19th, 2013.

UPPSALA UNIVERSITET

AALBORG UNIVERSITY

License

The UPPAAL tool is free for non-commercial applications in academia **only**. For commercial applications a commercial license is required. Please see the Download section or www.uppaal.com for more information.

To find out more about UPPAAL, read this short introduction. Further information may be found at this web site in the pages About, Documentation, Download, and Examples.

Mailing Lists

UPPAAL has an open discussion forum group at Yahoo!Groups intended for users of the tool. To join or post to the forum, please refer to the information at the discussion forum page. Bugs should be reported using the bug tracking system. To email the development team directly, please use uppaal(at)list(dot)it(dot)uu(dot)se.

Localization

In our ongoing work to localize the UPPAAL GUI we would like to acknowledge contributions by the following external people:

- Hiroshi Fujimoto (JA)
- Line Juhl (DK)
- Shuhao Li (ZH)
- Marius Mikucionis (LT)

start | uppaal-4.0.13 | Microsoft PowerPoint ... | UPPAAL - Google Chr... | readme - Notepad | 18:40

Курсова работа 1

1. Да се опише функционалната схема на Dynamic Source Routing (DSR) протокола - source routing реактивен маршрутизиращ протокол.

2. Да се посочат неговите преимущества пред останалите реактивни протоколи и за колко големи мрежи и с каква мобилност работи добре.

Курсова работа 2

1. Да се моделира в UPPAAL система от 1 трансмитер и 2 приемника, брой получени съобщения - 20, стартиращ таймер - 3 единици време. Пакетите да бъдат предназначени само за един от двата приемника на случаен принцип.

2. Да се симулира системата и се проследи дали всяко съобщение се получава от верния приемник.

3. Да се верифицира системата за deadlock.

Литература

1. Peterson L, B. Davie, Computer Networks – A System Approach, Morgan Kaufmann Publishers, 2003
2. Behrmann G., A. David, K. Larsen. A tutorial on UPPAAL. In Marco Bernardo and Flavio Corradini, editors, Formal Methods for the Design of Real-Time Systems, SFM-RT 2004, number 3185 in LNCS, pages 200-236. Springer-Verlag, 2004
3. Wibling O., J. Parrow, A. Pears. Automatized verifcation of ad hoc routing protocols. In FORTE, 2004, 343-358
4. Clausen T, P. Jacquet, A. Laouati, P. Minet, P. Muhltahler, A.Qayyum, & L. Viennot, “Optimized Link State Routing Protocol,” (2003) IETF RFC 3626
5. Wireless Ad-Hoc Networks,Edited by Hongbo Zhou, ISBN 978-953-51-0896-2, Hard cover, 164 pages, Publisher: InTech, Published: December 12, 2012 under CC BY 3.0 license,DOI: 10.5772/3438
6. Mobile Ad-Hoc Networks: Applications, Edited by Xin Wang, ISBN 978-953-307-416-0, Hard cover, 514 pages, Publisher: InTech, Published: January 30, 2011 under CC BY-NC-SA 3.0 license
7. Wireless Ad-Hoc Networks, MOBILE AD HOC NETWORKS: PROTOCOL DESIGN, Edited by Xin Wang
8. David B. Johnson David A. Maltz Josh Broch, ***DSR***: *The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks*, http://www.monarch.cs.rice.edu/monarch-papers/dsr-chapter00.pdf

Благодаря за вниманието !

Не се притеснявайте да ми задавате въпроси на:

ALEKOVA@ISER.BAS.BG