

Rechnernetze - Media Networking (WiSe 2023/2024)

Übung 09

Aufgabe 1

Mit dem aus dem IoT-Umfeld bekannten Routing-Protokoll RPL (Routing Protocol for Low Power and Lossy Networks) kann auch ein Multicast-Routing realisiert werden. An welche am Anfang des Semesters vorgestellten Multicast-Routing-Verfahren erinnern die dabei verwendeten Lösungsansätze

a) im Storing-Mode bzw.

b) im Non-Storing-Mode? Kurze Begründung.

Das Routing Protocol for Low-power and Lossy Networks (RPL) eignet sich gut für den Einsatz in einem 802.15.4 Netzwerk und ist besonders für eine DODAG-Topologie (DODAG: Destination-Oriented Directed Acyclic Graph) geeignet.

In diesem Netzwerk fungiert ein Border Router als Wurzelknoten, und es existieren mehrere Pfade von der Wurzel zu den Blättern.

Um die Pfade im DODAG zu bestimmen, werden zwei Arten von Steuerpaketen verwendet: DODAG Information Objects (DIO).

Diese Pakete werden vom Border Router zu den anderen Knoten gesendet, um ihnen bei der Pfadfindung zum Border Router zu helfen.

Beim Multicast-Routing in RPL teilen Sensorknoten ihre Adresse in den DAO-Paketen mit. Dadurch kann der Border Router feststellen, welche Knoten existieren und wie man sie erreichen kann. Diese Adressmitteilung kann einfach eine Multicast-Adresse sein, zusätzlich zur normalen Unicast-Adresse. Diese Pakete werden von den einzelnen Knoten zum Border Router gesendet und etablieren die Wege vom Border Router zu den Knoten.

Es gibt zwei Modi für die Verteilung von DAO-Paketen in RPL:

a)

Erstens den sogenannten Storing Mode.

Im Unterschied zu einer Funkzelle haben wir es im DODAG mit einer Topologie zu tun, die nicht automatisch alle Nachrichten an alle Knoten weiterleitet. Die Umsetzung im Storing Mode ist in gewisser Weise trivial, da sich die Zwischenknoten die Adressen merken und somit die Gruppenmitgliedschaften im Falle einer Multicast-Adresse kennen. Der Border Router muss die Multicast-Pakete nur an die entsprechenden Zweige verteilen, und von dort werden sie über die Zwischenknoten an die Empfänger weitergeleitet.

Im Storing Mode erfolgt die Kommunikation beispielhaft wie folgt.

Naja, es ist schon immer noch ein Funknetz, aber die Knoten werden explizit adressiert und die Pakete nur weitergeleitet, wenn man für den jeweiligen Knoten „zuständig“ ist.

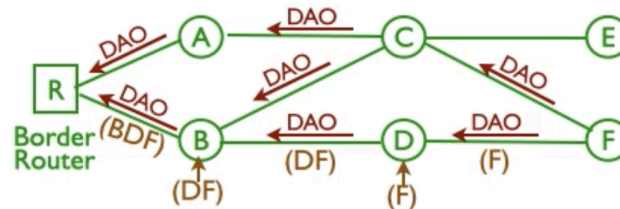


Abbildung 1

Im Beispiel in der Abbildung 1 würde z.B. der Knoten F ein DAO Paket an seine übergeordneten Knoten schicken. Diese merken sich, dass über diesen Weg der Knoten F erreichbar ist. Sie schicken das DAO dann weiter an den nächsten Knoten Richtung Border Router. Knoten B merkt sich daraufhin, dass auf diesem Weg D und F erreichbar sind. Er schickt es weiter Richtung Border Router und dieser weiß nun, dass auf dem Pfad B, D und F erreichbar sind. Er kann also Pakete die an F gehen sollen, einfach an B weiterleiten, da B sich ja gemerkt hat das er über D den Knoten F erreichen kann.

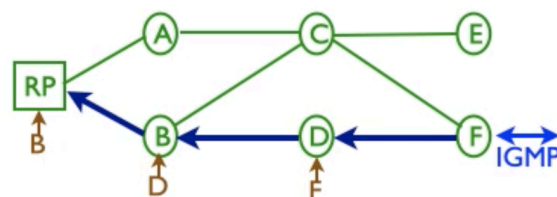


Abbildung 2

Das hat besonders bei der Anmeldung Ähnlichkeiten zu dem bekannten zentralisierten Verfahren PIM-SM. Das soll besonders die Nachteile von DVMRP und MOSPF umgehen, weil dort zu viel unnötiger Verkehr bei weit verstreuten Gruppen entsteht. Gelöst werden soll dieses Problem durch die Einrichtung einer expliziten Zentrale (Rendezvous-Point) im Netz pro Gruppenadresse. Der Sender schickt hierbei seine Pakete zur Zentrale. Die Empfänger melden sich per Unicast möglichst vorher bei der Zentrale an und bei der Gelegenheit wird der SPT der Zentrale zu den einzelnen Empfängern berechnet. Wenn F eine IGMP Meldung von einem Empfänger bekommt, dass er sich für eine bestimmte Gruppe interessiert, schickt F dies auf seinem Weg zum Rendezvous-Point weiter. D merkt sich, dass F sich für diese Gruppe interessiert, B bemerkt, dass D sich für diese Gruppe interessiert und der Rendezvous-Point weiß nun, dass B sich dafür interessiert. Bei RPL im Storing Mode wird also die Anmeldung von Knoten beim Border Router durchgeführt. Prinzipiell ähnlich wird beim Multicast Routing mit PIM-SM die Anmeldung vom Empfänger Routern beim Rendezvous-Point durchgeführt.

b)

Im Non-Storing-Mode von RPL erfolgt die Verteilung von Multicast-Nachrichten durch die Verwendung von DAOs. In diesem Modus teilen Gruppenmitglieder beim Eintritt nicht nur ihre Multicast-Adresse, sondern auch erneut ihre Unicast-Adresse mit. Zwischenknoten, die solche DAO-Pakete empfangen, verteilen die Informationen blind weiter, ohne sich die Adressen oder Gruppenmitgliedschaften zu merken.

Die Wurzel, in diesem Fall der Border Router, spielt eine entscheidende Rolle. Sie verteilt Multicast-Pakete mit einer Source-Routen-Angabe, die den Weg zu den Gruppenmitgliedern enthält. Anders als im Storing-Mode, wo dazwischenliegende Knoten sich die Adressen und Gruppenmitgliedschaften merken und gezielt an entsprechende Zweige weiterleiten, erfolgt im Non-Storing-Mode eine verteilte Weiterleitung ohne spezifisches Wissen über Gruppenmitglieder.

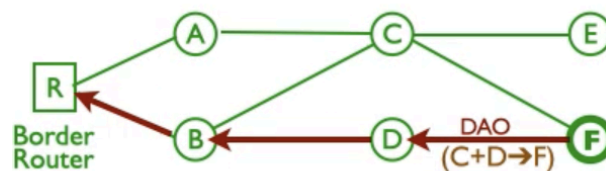


Abbildung 3

Auch in diesem Fall würde F ein sogenanntes DAO-Paket in Richtung Border Router schicken. Allerdings gibt er an, dass seine Elternknoten in Richtung Border Router C und D sind. Diese Information hat er zuvor mithilfe eines anderen Steuerpakets erhalten. D hingegen merkt sich diese Information nicht, sondern leitet das Paket einfach weiter in Richtung Border Router. Gleichzeitig sendet D jedoch selbst DAO-Pakete mit seinen Informationen über Elternknoten an den Border Router. Dadurch kennt der Border Router auch in diesem Fall alle möglichen Wege, um Knoten zu erreichen. Da die Zwischenknoten diese Informationen nicht gespeichert haben, muss der Border Router nun mit Source Routing arbeiten. Das bedeutet, er schreibt in die Pakete, die in Richtung F gehen, dass sie über B und D geleitet werden sollen.

Der Non-Storing-Mode von RPL weist gewisse Ähnlichkeiten mit dem "BIER" (Bit Index Explicit Replication)-Verfahren auf. BIER ermöglicht in einer Multicast-Kommunikation das Durchqueren einer Routing-Domain, ohne dass jeder einzelne Router unterwegs sich alle Gruppenzugehörigkeiten merken muss, ähnlich wie es im Non-Storing-Mode von RPL abläuft.

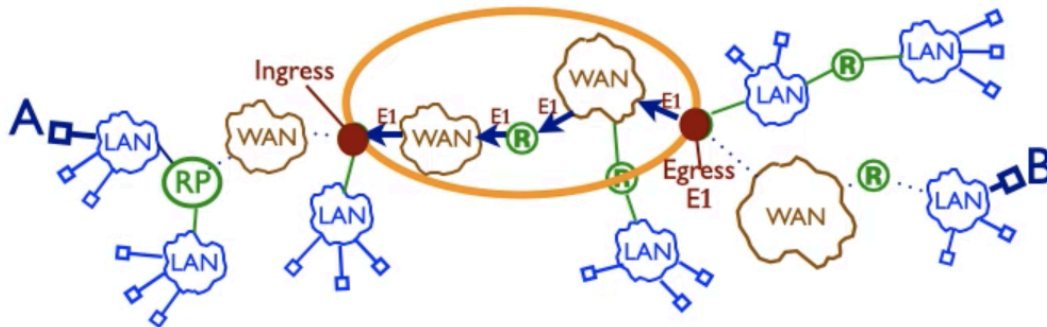


Abbildung 4

Im Beispiel (Abbildung 4) hat A als Sender eine Gruppenkommunikation angestoßen, und B hat sich als interessierter Empfänger bei seinem Router angemeldet. Die Anmeldung wird zum Rendezvous Point geschickt. Unter Nutzung von BIER wird diese Anmeldung um die Information ergänzt, dass die Datenpakete beim Egress Point E1 aus dieser Routing-Domäne ausgeführt werden müssen.

Ähnlich zum Non-Storing-Mode müssen sich die Zwischenknoten diese Information nicht merken, sondern nur der Ingress Point.

Wenn Datenpakete vom Sender zum Empfänger fließen, setzt der Ingress Point eine Source Route zum Egress Point E1 auf, sodass alle Router unterwegs wissen, wie sie das Paket weiterleiten sollen. Dabei wird der IPv6 Routing Header verwendet.

Obwohl es sich nicht um eine senderbasierte Gruppenkommunikation handelt, da der Sender A nicht wissen muss, dass B ein Empfänger der Gruppe ist, wird die Source-Routing-Funktionalität nur innerhalb der markierten Routing-Domäne verwendet. Wenn es einen anderen Empfänger wie C gibt, der über einen anderen Egress Point aus der markierten Routing-Domäne erreicht werden kann, muss die Source Route den anderen Egress Point benennen. Dies kann kompakt durch die Identifizierung aller Border Router in der Source Route über einen Bitvektor erfolgen, sodass jeder mögliche Egress Point durch ein einzelnes Bit kodiert wird. Dadurch kann dasselbe Paket an beide Empfänger im Sinne einer Multicast-Verteilung geschickt werden.

Der Non-Storing-Mode von RPL und das "BIER" (Bit Index Explicit Replication)-Verfahren weisen also mehrere Gemeinsamkeiten auf. In beiden Ansätzen erfolgt die Verteilung von Multicast-Nachrichten durch die Verwendung spezieller Steuerpakete, wie zum Beispiel DAOs (Destination Advertisement Objects) bei RPL. Diese Pakete enthalten Informationen über Gruppenmitgliedschaften und Routen.

Ein wesentliches Merkmal beider Modelle ist die verteilte Weiterleitung von Multicast-Nachrichten, bei der Zwischenknoten die erhaltenen Informationen ohne Zwischenspeicherung blind weiterleiten. Sowohl im Non-Storing-Mode von RPL als auch in "BIER" spielt die Wurzel (wie der Border Router in RPL) oder der Ingress Point eine entscheidende Rolle. Diese Instanzen verteilen Multicast-Pakete mit

Source-Routen-Angaben, um den Weg zu den Gruppenmitgliedern oder Empfängern festzulegen.

Beide Ansätze nutzen Source Routing, bei dem der Pfad zu den Gruppenmitgliedern oder Empfängern bereits in den Paketen angegeben wird. Das ermöglicht Multicast-Kommunikation durch das Durchqueren einer Routing-Domain, ohne dass jeder einzelne Router alle Gruppenzugehörigkeiten speichern muss.

Ein weiterer gemeinsamer Aspekt besteht darin, dass die Zwischenknoten sich die Informationen über Gruppenmitgliedschaften oder Routen nicht merken müssen.

2 Punkte

Aufgabe 2

Ein fiktiver Anwendungsdienst arbeitet mit UTF8-Kodierung. Ein Empfänger erhält folgenden Bytestrom (entgegen der in Netzen üblichen LSB-first-Bitorder in MSB-first angegeben): 10111100 01110000 01110000 01101001 01100111

Ist die Nachricht vollständig empfangen worden? Kurze Begründung. Was wollte der Sender vermutlich mitteilen?

binary	symbol
11000011 10111100 11000010 10111100	ü ¼
10111100	¼ (ISO 8859-1)
01110000	p
01110000	p
01101001	i
01100111	g

"üppig"; der Absender wollte wahrscheinlich das Wort "üppig" senden. Dies bedeutet, dass bei der Übertragung 1 Byte verloren gegangen ist. Hintergrund ist, dass die Umlaute in UTF-8 durch 2 Bytes dargestellt werden. Die entsprechende Kodierung für den jeweiligen Umlaut ist unten aufgeführt¹²³:

¹ <https://php-de.github.io/jumpto/utf-8/>

² <https://www.utf8-chartable.de/unicode-utf8-table.pl?utf8=bin>

³ <https://stevemorse.org/hebrew/utf8.php?charset=1>

ä - 11000011 10100100

ö - 11000011 10110110

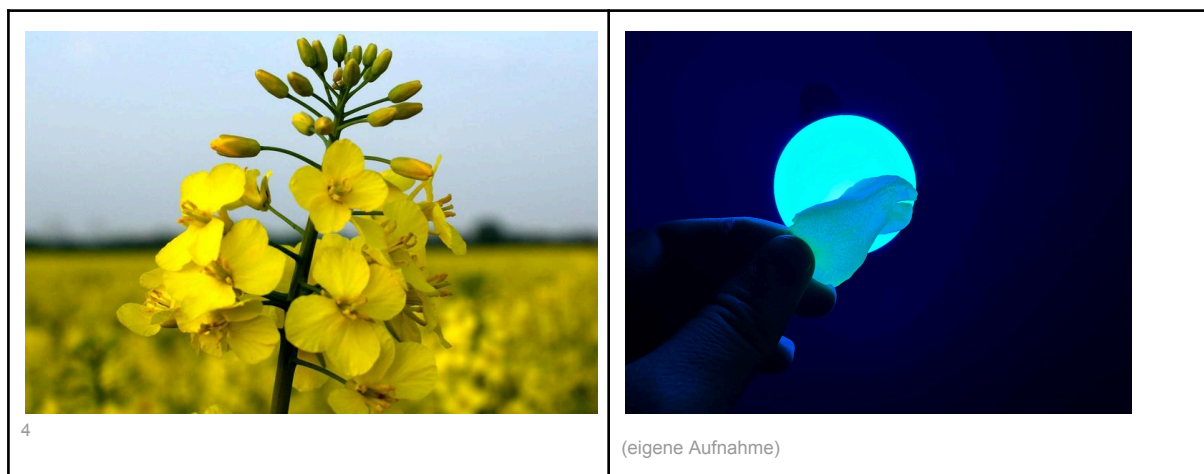
ü - 11000011 10111100

Das zweite Byte von ü ist genau das Byte, das vom Empfänger empfangen wurde. In der UTF-8-Kodierung ist dies nur in Kombination mit anderen Bytes gültig und wird in allen anderen Fällen nicht angezeigt (ignoriert) bzw. als ? oder \blacklozenge angezeigt. Eine andere gültige Kombination wäre z. B. mit 11000010 10111100, was das Zeichen $\frac{1}{4}$ ergeben würde. In diesem Kontext macht das Zeichen jedoch wenig Sinn.

1 Punkt

Aufgabe 3

Welche Farbe haben Rapsblüten in blauem Licht? Kurze Begründung

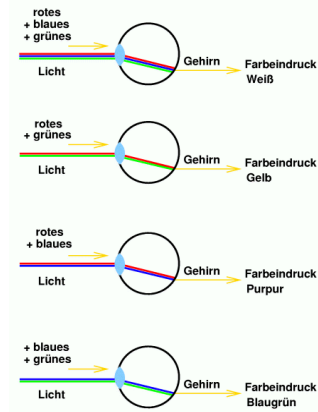


Die Gelbfärbung der Blütenblätter ist auf ihren Gehalt an Karotinoidpigmenten zurückzuführen, die Wellenlängen von ~450 nm absorbieren. Durch die Absorption des blauen Lichts reflektieren sie eine Mischung aus grünem und rotem Licht, das wir als gelb wahrnehmen (das Bild auf der linken Seite). Gelb ergibt sich durch das Mischen von rot und grün. Wenn gelb nun auf blau trifft, dann haben wir eben das erste Szenario in der rechten Abbildung, nämlich die Mischung aus rot + grün (gelb) und blau, also entsteht ein weißer Farbeindruck (siehe Bild unten⁵).

Da die Rapsblüten blaues Licht absorbieren, würden sie in blauem Licht sehr dunkel erscheinen (im Extremfall schwarz)

⁴ <https://www.transgen.de/lexikon-nutzpflanzen/1902.raps.html>

⁵ <https://www.elmar-baumann.de/fotografie/fotobuch/node22.html>



0,5 Punkte

Aufgabe 4

Ein Fax-Gerät erhält folgenden (stark gekürzten) zweidimensionalen Bild-Code — vereinfachend nicht binär, sondern über Codeworte angegeben. Jede Zeile beschreibt eine Rasterzeile.

```
H 2W 3S V(0)
VL(1) VL(3) V(0)
V(0) V(0) V(0)
VR(1) VR(2) V(0)
VR(2) VR(1) V(0)
V(0) V(0) V(0)
VL(3) VL(1) V(0)
P V(0)
```

- Wie sieht das zugehörige Bild aus? Kurze Begründung.
- Was passiert, wenn eine Übertragungsstörung die zweite Zeile wie folgt verändert?
Kurze Begründung.

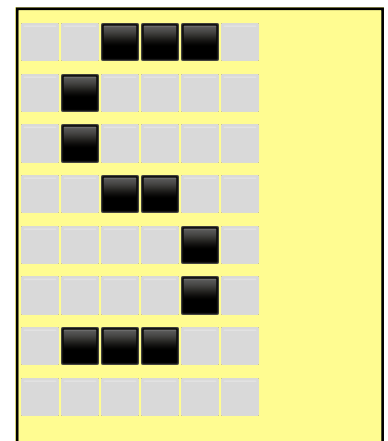
```
VL(1) VR(3) V(0)
```

- ohne ECM
- mit ECM

(2 Punkte)

a)

		b1		b2			
0	0	1	1	1	0	0	H 2W 3S V(0)
0	1	1	0	0	0	0	VL(1) VL(3) V(0)
0	1	1	0	0	0	0	V(0) V(0) V(0)
0	0	1	0	1	0	0	VR(1) VR(2) V(0)
0	0	0	0	1	1	0	VR(2) VR(1) V(0)
0	0	0	0	1	1	0	V(0) V(0) V(0)
0	1	0	0	1	0	0	VL(3) VL(1) V(0)
0	0	0	0	0	0	0	P V(0)



Nee, es würde eher wie ein S aussehen

Durch die Möglichkeit von $|a_1b_1| \leq 3$ beginnt ab der zweiten Zeile schon der Vertikalmodus, wo von unserem Bezugspunkt b_1 um einen nach links, also früher mit dem Farbwechsel von weiß auf schwarz begonnen wird ($a_1=b_1-1$). Dem gleichen Prinzip folgend wird b_2 zu b_2-3 Felder nach links, der Rest wird mit weiß aufgefüllt $V(0)$ bzw. wird die Zeile mit einem weißen Ende abgeschlossen wird und $a_1 = b_1$. Die dritte Zeile ist gleich wie die davor. Die vierte: $a_1=b_1+1$ und b_2 mit b_2+2 . Diesem Prinzip folgend hat sich das Bild entwickelt.

[illegible]

Grundsätzlich findet eine Verschiebung nach rechts statt, was aber dazu führt, dass die schwarzen Lauflängen entsprechend 6 Pixel länger werden

i) ECM steht für "Error Correction Mode" und ist eine Funktion in Faxgeräten, die dazu dient, die Übertragung von Faxnachrichten zu verbessern und Fehler während der Übertragung zu korrigieren. Ohne ECM würde die Übertragungsstörung zu einer fehlerhaften Darstellung führen, und das empfangene Bild würde die Änderungen aufgrund der Störung widerspiegeln.

Die Geräte der Telefax Gruppe 3 haben zunächst - obwohl sie zweidimensional kodieren können, ca. nach 0.5 mm eine eindimensionale Zeile eingefügt. Der Grund dafür war ein fehlendes Fehlersicherungsprotokoll, wobei z.B. durch einen Bitkipper im Code - bei einer durchgehend zweidimensionalen Kodierung - eine Fortpflanzung eines solchen Fehlers über die ganze Seite passieren konnte, da zweidimensionale Kodierung relativ zur Vorgängerzeile

sein kann. Mittlerweile verwenden Faxgeräte ECM, sodass auch eine durchgehend zweidimensionale Kodierung möglich wird. Grundlage für diese Fehlersicherung ist die Aufteilung des bei der Kodierung entstandenen Bitstroms auf HDLC ähnlichen Frames, die jeweils einen CRC Wert enthalten.

Jede Seite wird in 256-Byte-Frames unterteilt, sodass ein Fax über mindestens 64 KiB Speicherplatz verfügen muss. Auf der Senderseite werden Prüfsummen berechnet und an die Frames angehängt. Die Frames werden fortlaufend gesendet und beim Empfänger gespeichert. Danach werden die CRC-Werte vom Empfänger überprüft. Fehlerhafte Frames werden durch "selective reject" herausgefiltert. Die Frames werden erneut gesendet. Dieser Vorgang wird so lange wiederholt, bis alle Frames korrekt sind.

1 Punkt