

FHNW PAPER

Performancevergleich von Zigbee, Thread und Bluetooth Mesh Netzwerken

Cyrill Horath¹ | Raffael Anklin¹ | Robin Bobst¹

¹ Institut für ??, Fachhochschule
Nordwestschweiz, Windisch, Aargau, 5210,
Schweiz

Correspondence

Team Blau, Institut für ??, Fachhochschule
Nordwestschweiz, Windisch, Aargau, 5210,
Schweiz
Email: TeamBlau@email.com

Funding information

The assignment of this bachelor thesis was to build a consistent test framework for all three mesh-networks to benchmark them under realistic conditions. The benchmark is structured in two parts, a slave node and a master which is directly connected to a computer. The master node is responsible for controlling the measurement, whereas the slave nodes send benchmark messages to each other. These benchmark messages collect the necessary information to determine latency, RSSI, throughput and active radio time. For a better comparability an apartment house, an apartment and a labor environment were selected as different test benches. The Thread stack results the best in the different test benches. Because of its automatic routing it is able to adapt himself to the environment, as a result the latency of this stack is in every three benches similarly low. Bluetooth Mesh was able to reach the lowest latency with small payload. The ZigBee network stands out with its constant and low latency within one test bench. As a conclusion all of the three networks perform well in case of a home automation. Due to of their own assets and drawbacks it cannot be said this is the best mesh-stack. It depends on the application which mesh network performs the best.

1 | EINLEITUNG

Im 2.4GHz ISM-Band konkurrieren sich derzeit die drei weit verbreiteten Low Power Mesh Netzwerk Protokolle Bluetooth Mesh, Thread und Zigbee. Alle drei wurden konzipiert für die kabellose Übertragung in sogenannten WSN (Wireless Sensor Networks) oder in Netzen für die Heim Automatisierung. Während Thread und Zigbee den IEEE 802.15.4 Standard als Physical Layer benutzen basiert der BT Mesh Stack auf dem BLE (Bluetooth Low Energy) Standard. Aufgrund der hohen Dichte an Netzwerkprotokollen die das 2.4GHz ISM-Band ebenso nutzen (z.B. Wifi) sind die Störeinflüsse auf die Mesh Protokolle eines der grössten Probleme. Die Protokollstacks begegnen diesem und weiteren Problemen auf unterschiedliche Weise. Diese Unterschiede und schliesslich die Performance der Mesh Netzwerke sollen unter unterschiedlichen Testbedingungen aufgezeigt werden wodurch ein objektiver Vergleich der drei Mesh Protokolle möglich wird. In der nachfolgenden Tabelle ist sind die Hauptmerkmale der drei verschiedenen Mesh-Netzwerke aufgelistet:

Mesh Netzwerke im Vergleich			
	Bluetooth Mesh	Thread	ZigBee
Markt	Beleuchtung und Smart Home	Industrie und Smart Home	Beleuchtung, Haus Automation und Messtechnik
Veröffentlicht	2017	2015	2003
Applikations Layer	Mesh Model System	Verknüpfbar mit allen IPv6 basierten Protokollen	Cluster Bibliothek
IPv6	Nein	Ja	Nein
Netzwerk Zugriff	Smartphone oder Gateway	Border Router	Gateway
Ökosysteme	Ledvance	Google Nest	Ikea, Phillips Hue, Amazon und weitere
Routing	Managed Flooding	Geroutet	Geroutet
Weiteres	Ist direkt mit Smartphone erreichbar	Automatisiertes Verwalten des Netzwerks	Am meisten verbreitet

TABLE 1 Vergleich Mesh Netzwerke

Hauptziel ist ein objektiver Vergleich der drei gängigsten Low Power Mesh Netzwerke Bluetooth Mesh, Thread und Zigbee bezüglich deren Leistungsfähigkeit unter wechselnden Bedingungen. Es soll erkennbar werden, welches Protokoll in welchen Bereichen seine Stärken hat und wie es am besten eingesetzt werden kann.

2 | METHODE

Um die Performance der drei Mesh Stacks zu vergleichen wurde ein einheitliches Benchmark Konzept erarbeitet. Dieses definiert die Mesh Parameter, Testumgebungen, den Ablauf sowie sämtliche Messgrößen und Messreihen.

2.1 | Messablauf

Für den Vergleich der 3 Mesh Netzwerkstacks Bluetooth Mesh (BT Mesh), Thread und Zigbee wird ein vom Mesh Protokoll unabhängiges Testkonzept umgesetzt welches in der Abbildung 1 als Konzeptschema dargestellt ist. Die Benchmark Slave Nodes (BSN) in der Abbildung als Sensoren und Aktoren mit unterschiedlichen Funktionalitäten dargestellt, bilden zusammen mit dem Benchmark Master Node (BMN) das zu testende Mesh Netzwerk. Innerhalb des Netzwerks wird dessen Organisation vom jeweiligen Protokoll sichergestellt. Das Testnetzwerk soll ein realitätsnahes Netzwerk nachbilden. Beispielsweise wird eine Hausautomation in einem Einfamilienhaus als Referenz angenommen in welchem jeweils nur gewisse Nodes untereinander Applikationsdaten austauschen. Ein Lichtschalter kommuniziert nur mit einer Lichtquelle und umgekehrt. Der selbe Lichtschalter tauscht jedoch keine Applikationsdaten mit dem Temperatursensor aus. Trotzdem bilden die Nodes zusammen ein Mesh Netzwerk.

Die Benchmark Management Station (BMS) welche mit dem BMN via USB/UART kommuniziert, ist zuständig für die Verwaltung und Verarbeitung der Benchmarks. Während eines Benchmark Prozesses sollen sämtliche Messungen jedoch unabhängig von der BMS durchgeführt werden damit allfällige Latenzzeiten der USB/UART Verbindung die Resultate nicht verfälschen.

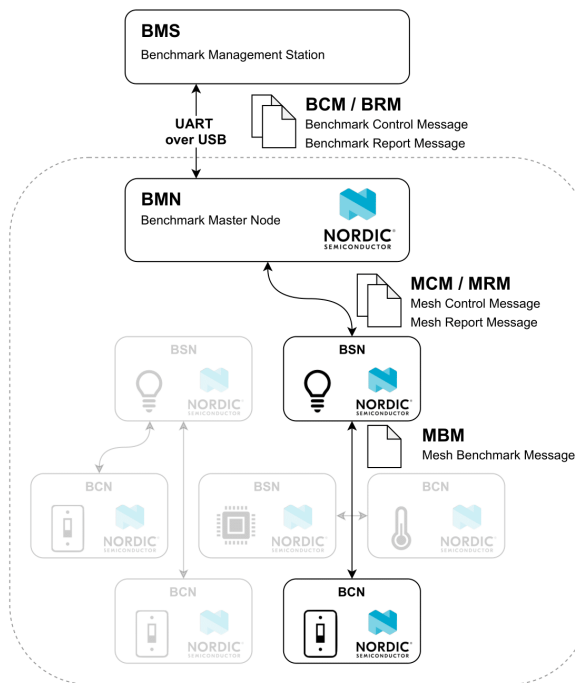


FIGURE 1 Konzeptschema Testablauf

2.2 | Messaufbau

Unterschiedliche Testumgebungen sollen die Benchmarks und schlussendlich den Vergleich der 3 Mesh Protokolle aussagekräftiger machen. Nachfolgende Umgebungen mit den entsprechenden Eigenschaften sollen getestet werden. Die Abbildungen zu den Testumgebungen zeigen jeweils die Platzierung der Nodes sowie deren Funktion und Gruppen Zugehörigkeit. Die Farbe Grün identifiziert den Node als Client Node während Blau für einen Server Nodes steht. Die Nummerierung zeigt welcher Node zu welcher Adressgruppe gehört. Ein Client Node in Gruppe 1 sendet jeweils Nachrichten zu allen Server Nodes in der selben Gruppe.

Labor

Der Laboraufbau ist ein Extremtest welcher die Leistungsgrenzen der Protokollstacks ausloten soll. Dabei werden die Nodes auf einem Raster gemäss Abbildung 2 angeordnet. Die genauen Abmessungen sind der Abbildung zu entnehmen.

- Testaufbau unter Laborbedingungen auf engstem Raum.
- Ausgeglichene Anzahl Sensoren und Aktoren.
- Sehr Hohe Node-Dichte.
- Geringe bis keine Störbeeinflussung durch die Umgebung zu erwarten.
- Die Mesh-Beziehungen werden künstlich bestimmt sodass einfache P2P Verbindungen mit oder ohne Hop entstehen.

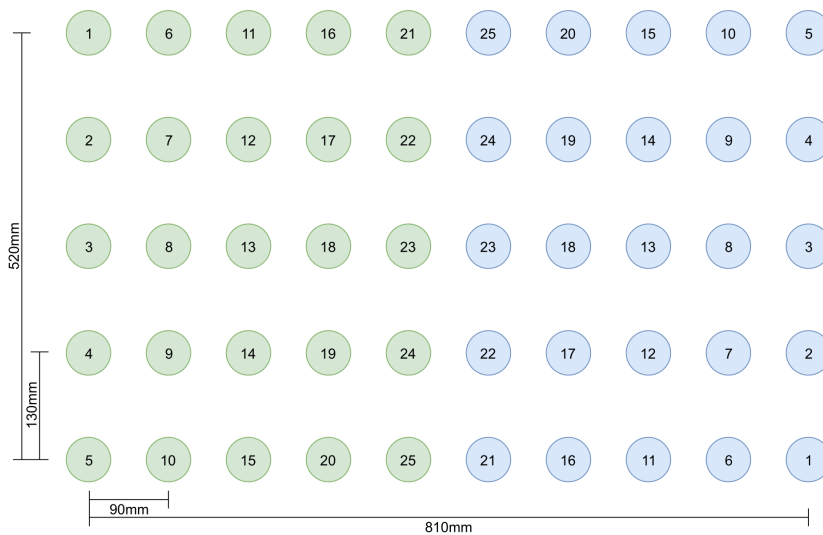


FIGURE 2 Testaufbau Labor

Einfamilienhaus

Die Testgeräte werden in einem Einfamilienhaus installiert und repräsentieren damit eine flächendeckende Heim-Automatisierung. Folgende Eigenschaften soll diese Messung abdecken:

- Einfamilienhaus über mehrere Etagen.
- Anzahl Sensoren und Aktoren vergleichbar gross.
- Node-Dichte relativ gering.
- Kleine Beeinflussung durch Nachbarsysteme sind zu erwarten.

Die Abbildung 3 zeigt den Schnitt des Einfamilienhauses in welchem der Benchmark durchgeführt wurde.

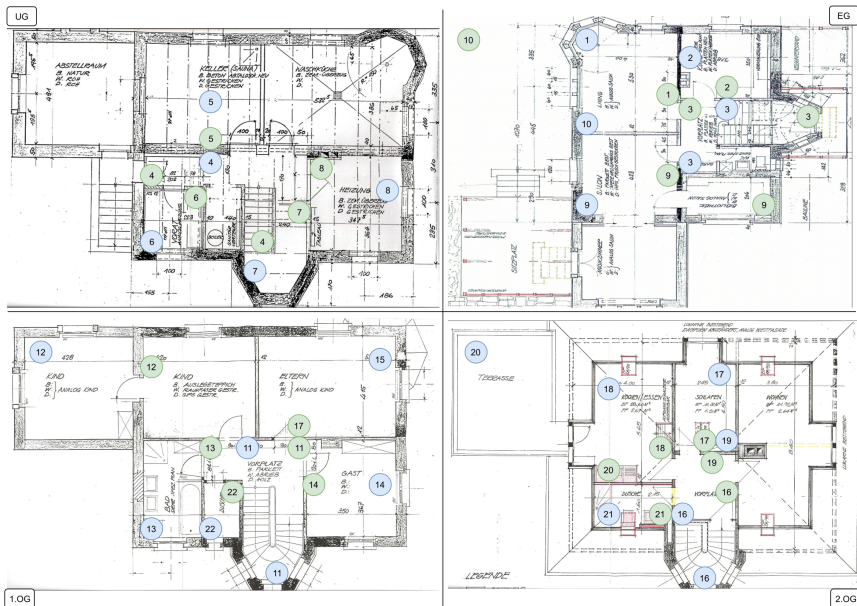


FIGURE 3 Platzierung der Nodes im Einfamilienhaus

Wohnung

Ebenfalls als Heim-Automatisierung gedacht werden die Messungen in einer Wohnung durchgeführt.

- Wohnung über eine Etage in einem Mehrfamilienhaus
- Anzahl Sensoren und Aktoren vergleichbar gross.
- Node-Dichte höher als im Haus.
- Mögliche Störeinflüsse durch andere Systeme von Nachbarn sind zu erwarten.

Bei der Wohnung handelt es sich um eine 3.5 Zimmer Wohnung mit einer Wohnfläche von 122 Quadratmetern. Die genauen Abmessungen sowie die Platzierung der Nodes ist in Abbildung 4 zu sehen.

2.3 | Messerwartung

Wie in der Übersicht ?? bereits erwähnt, konnte im Rahmen der Vorarbeiten zu dieser Thesis der BT Mesh Stack bereits vertieft untersucht und erste Erfahrungen damit gesammelt werden. Bei den beiden anderen Stacks, Thread und Zigbee musste dies erst

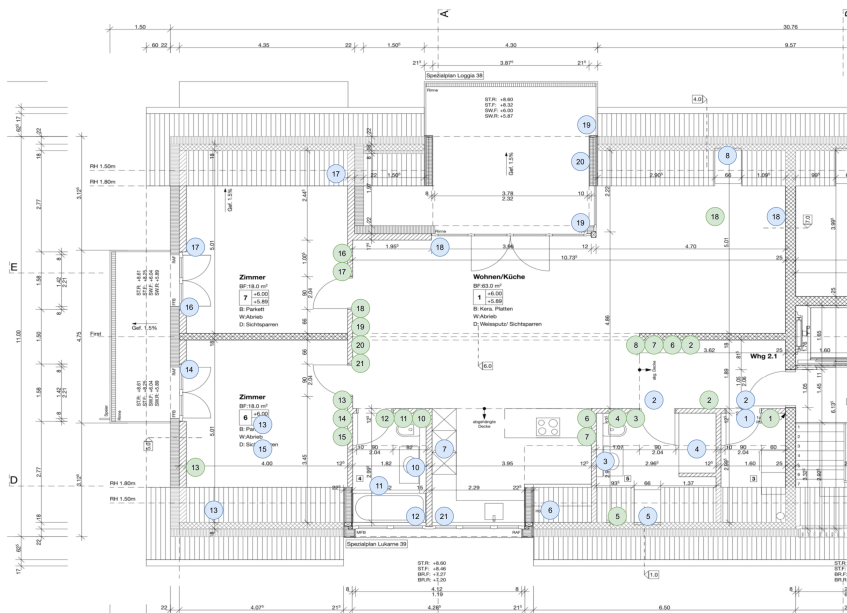


FIGURE 4 Testaufbau Wohnung

noch erfolgen. Bei eben diesen Vorarbeiten konnten bereits deutliche Unterschiede zwischen den Stacks beobachtet werden, hauptsächlich verursacht durch das Routing bei Thread und Zigbee sowieso das Flooding-Mesh Prinzip bei BT Mesh. Aufgrund dieser Erfahrungen wird erwartet, dass die beiden auf dem IEEE 802.15.4 Standard aufbauenden Protokolle klare Vorteile haben werden da die Nachrichtenzustellung durch das Routing effizienter realisiert werden kann. Hingegen könnte BT Mesh Vorteile haben bei kleiner Belastung des Netzes.

Das menschliche Auge ist in der Lage eine Verzögerung festzustellen, wenn die Latenz vom Knopfdruck bis das Licht angeht mehr als 200ms beträgt. In einer modernen Hausautomation darf natürlich keine Verzögerung wahrgenommen werden. Somit wird erwartet, dass die Latenzzeit im Schnitt unter 200ms bleiben soll. [?]

3 | ERGEBNISSE

Die Ergebnisse sollen hier nach verschiedenen Kriterien dargestellt werden (Anzahl Nodes, Anzahl Hops, usw.)

4 | INTERPRETATION

Der Vergleich der Mesh-Netzwerke basiert auf den Messergebnissen der Messreihen, welche in Abschnitt ?? erwähnt wurden. Eine Messung unterscheidet sich durch die unterschiedlichen Messparameter, sowie durch den Messaufbau (Wohnung / Labor / Haus). Um einen Vergleich ziehen zu können, wird daher nach diesen beiden Hauptmerkmalen unterschieden.

4.1 | Vergleich Messreihen

Dieser Vergleich bezieht sich auf die selbe Testumgebung, jedoch mit unterschiedlichen Messreihen. Als Referenz wurde die Umgebung des Labors ausgewählt, da Messergebnisse aller Messreihen vorliegen (siehe Anhang ??). Im Anschluss soll gezeigt werden wie sich die einzelnen Mesh-Netzwerke bei veränderten Messreihen verändern.

Zusammengefasst lassen sich die Messreihen folgendermassen unterscheiden.

- **Payload** Die Payload wurde zwischen den Messreihen 1&3, 2&4 und 7&8 von 8 Byte auf 32 Byte, resp. 50 Byte erhöht.
- **Traffic Generation Mode** Wurde zwischen den Messreihen 1&2 und 3&4 von Random auf Sequentiell geändert.
- **Message Dichte** Wurde zwischen den Messreihen 2&7 und 4&8 von 2.5 M/s auf 0.33 M/s gesenkt (Messages/Sekunde).
- **Disturbance** Wurde von Messung 2 auf Messung 6 eingeschaltet.

4.1.1 | Latenzzeit

Durch die in Abbildung 5 dargestellten Latenzzeiten wird ersichtlich das Bluetooth-Mesh am anfälligsten auf die Erhöhung der Payload reagiert. Eine detaillierte Analyse zu diesem Phänomen wird in Abschnitt ?? durchgeführt. Die Verzögerung bleibt bei Zigbee und Thread trotz Erhöhung der Paketlänge stabil.

Ein Änderung des Traffic Generation Mode von Random auf Sequentiell bewirkt bei Bluetooth-Mesh eine drastische Abnahme der Latenz. Bei Zigbee ist zwischen der Messreihe 1 und 2 ebenfalls eine Abnahme zu verzeichnen, welche jedoch nicht zwischen der Reihe 3 und 4 feststellbar ist. Somit muss ein anderer Einfluss für die Abnahme verantwortlich sein (in Abschnitt ?? untersucht). Die Änderung des Traffic Generation Mode wirkt bei Thread ebenfalls zu einer Verbesserung der Latenz.

Das Senken der Message Dichte führt bei Bluetooth-Mesh zu einer markanten Abnahme der Latenz. Bei Thread und Zigbee bleibt die Latenz nahezu identisch.

Die Einbringung von Störungen hat lediglich bei Bluetooth-Mesh einen negativen Einfluss, wodurch sich die Latenzzeit erhöht.

4.1.2 | Durchsatz

Die in der Abbildung 6 dargestellten Durchsätze wird ersichtlich, dass sich die ähnlichen Resultate ergeben wie bei der Latenzzeit. Bluetooth-Mesh reagiert am anfälligsten auf die Payload-Erhöhung. Wie bereits in Abschnitt 4.1.1 dargelegt wird eine detaillierte Analyse zu diesem Phänomen in Abschnitt ?? durchgeführt. Der Durchsatz steigt bei Zigbee und Thread durch die Erhöhung der Paketlänge stark an. Dies wird in Abschnitt ?? für Thread und Abschnitt ?? für Zigbee genauer untersucht.

Ein Änderung des Traffic Generation Mode von Random auf Sequentiell bewirkt bei Bluetooth-Mesh eine drastische Zunahme des Durchsatzes. Bei Thread ist ebenfalls eine Steigerung der Übertragungskapazität feststellbar. Die Änderung des Traffic Generation Mode wirkt bei Zigbee zu keiner merklichen Verbesserung des Durchsatzes.

Das senken der Message Dichte führt bei Bluetooth-Mesh zu einer Verbesserung des Durchsatzes. Bei Thread und Zigbee kann keine Abhängigkeit festgestellt werden.

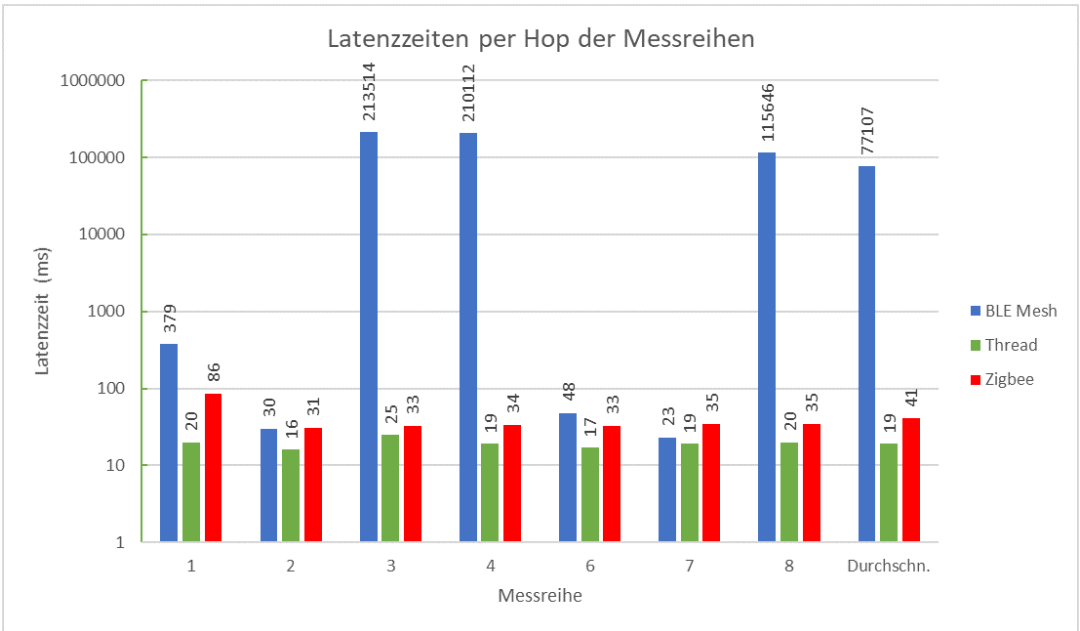


FIGURE 5 Durchschnittliche Latenzzeit per Hop der einzelnen Messreihen im Vergleich

Das einbringen von Störungen hat bei allen Netzwerken einen leichten negativen Einfluss, wodurch der Durchsatz sinkt.

4.1.3 | Paketverlust

Wie in Abbildung 6 ersichtlich ist Bluetooth-Mesh sehr anfällig auf das erhöhen der Paketlänge. Dies wird in Abschnitt ?? genauer untersucht. Zigbee und Thread zeichnen sich über alle Messreihen als sehr zuverlässig aus.

4.1.4 | Energieverbrauch

Durch Abbildung 8 wird ersichtlich das Bluetooth-Mesh den etwas geringeren Energiebedarf aufweist als Thread und Zigbee. Ansonsten zeigen alle Netzwerke ähnliche Resultate unabhängig der Messreihen.

4.2 | Vergleich Testumgebungen

Dieser Vergleich bezieht sich auf die selbe Messreihe, jedoch in unterschiedlichen Testumgebungen. Als Referenz wurde die Messreihe 2 ausgewählt, da diese für alle Testumgebung repräsentative Resultate enthält (siehe Anhang ??). Im Anschluss soll gezeigt werden wie sich die einzelnen Mesh-Netzwerke bei veränderten Testumgebungen verändern.

4.2.1 | Latenzzeit

Die Abbildung 9 zeigt das die Testumgebung Labor für alle Netzwerke die geringste Verzögerung aufweist. Bluetooth-Mesh erfährt bei weiter Streuung der Teilnehmer, wie sie im Haus anzutreffen ist, die grösste Steigerung der Latenz. Zigbee erhält in

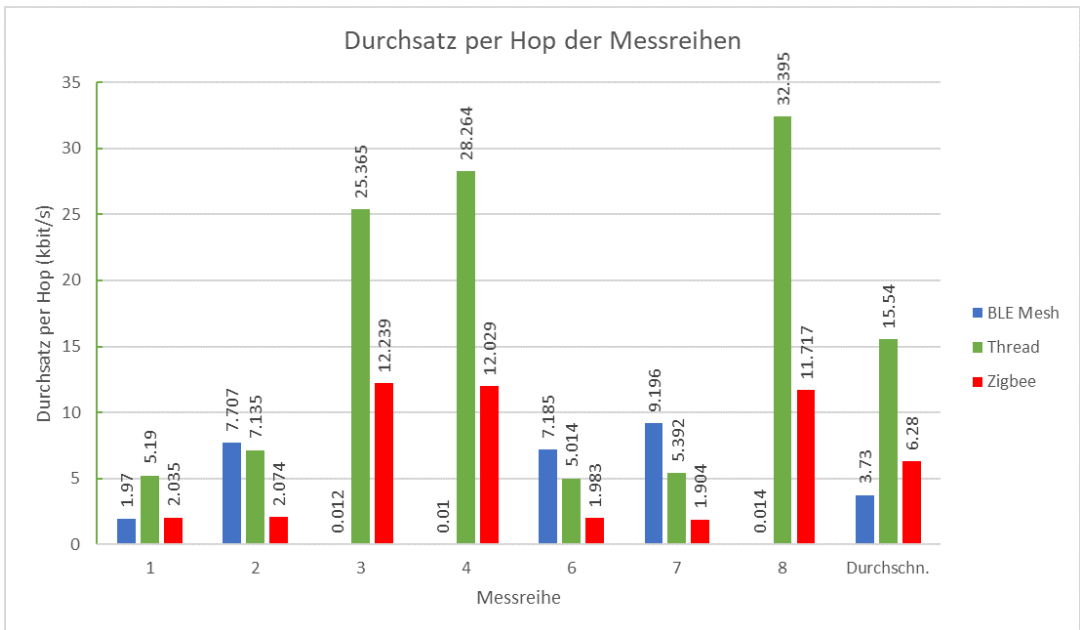


FIGURE 6 Durchschnittlicher Durchsatz per Hop der einzelnen Messreihen im Vergleich

der Wohnung eine kaum relevante Erhöhung der Verzögerung. Thread zeichnet sich als unabhängig bezüglich der Testumgebung aus.

4.2.2 | Durchsatz

Ähnlich zu den Betrachtungen der Latenzzeit, zeigt die Abbildung 10 den höchsten Durchsatz im Testaufbau Labor. Bluetooth-Mesh erfährt den grössten Einfluss durch ein ändern der Topologie.

4.2.3 | Paketverlust

Abbildung 10 zeigt das im Labor-Aufbau alle Netzwerke vollkommen zuverlässig arbeiten. In der Wohnung erlebt Bluetooth-Mesh einen geringen Anstieg des Paketverlusts. Bei Thread ist die Änderung kaum relevant. Alle Netzwerke erfahren bei starker Verzweigung des Netzwerks einen Anstieg der Verlustrate wie es bei der Haus-Topologie zu sehen ist.

4.3 | Fazit

Alle Mesh-Netzwerke mussten diversen Tests standhalten. Die Ergebnisse aus den Messreihen 1 bis 4 aus allen Testumgebungen wurden als Durchschnittswert zusammengefasst, um einen finalen Vergleich zu erzielen. Die Abbildung 12 zeigt alle Ergebnisse auf einen Blick.

Schlussendlich lässt sich Thread als klarer Sieger erkennen. Dieser Network-Stack hat die Tests am besten absolviert. Auf dem zweiten Platz steht Zigbee, welches mit seiner konstanten Latenz und als zuverlässigstes Protokoll die Daten verarbeiten konnte. Dabei gilt zu beachten das bei Zigbee die Hops nicht berücksichtigt wurden (siehe Abschnitt ??). Bluetooth-Mesh kann

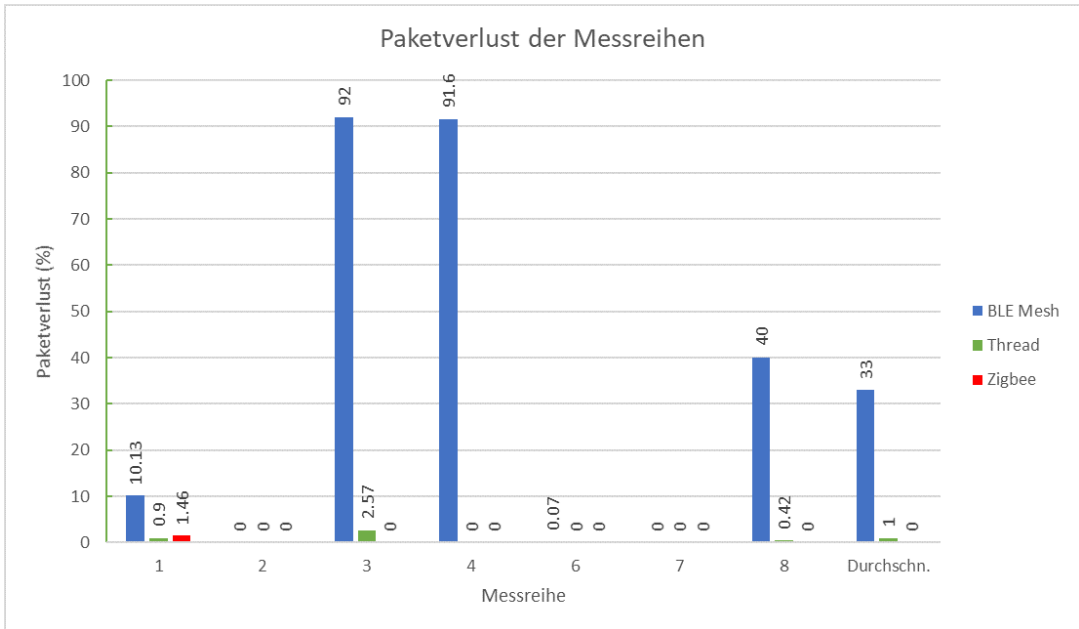


FIGURE 7 Durchschnittlicher Paketverlust der einzelnen Messreihen im Vergleich

durch den niedrigen Energiebedarf brillieren, schneidet jedoch in Sachen Performance deutlich schlechter als seine Konkurrenten ab. Im Anschluss wird jedes Protokoll genauer analysiert und auf die Vor- und Nachteile, sowie mögliche Verbesserungen eingegangen.

4.3.1 | Thread

Wie aus den vorhergehenden Kapiteln ersichtlich hat der Thread Stack die Messungen am besten abgeschlossen. Das bedeutet aber nicht unbedingt, dass der Stack der Beste ist. Es ist klar zu erkennen, dass der Thread Stack dank seiner automatischen Ernennung von Routing-Knoten, sich seiner Umgebung gut anpassen kann. Die Latenzzeit der gesendeten Nachrichten ändert sich in verschiedenen Umgebungen nur minimal. Das bedeutet, dass sich der Stack sehr gut für eine Hausautomation eignet. Wenn sich Sensoren oder Aktoren verschieben, z.B. wenn eine Lampe einen neuen Standort erhält, erkennt dies der Stack und kann das Routing der Knoten anpassen. Da sich das Netz sehr gut erweitern lässt und sich die Latenzzeiten eher tief zeigen, kann das Mesh-Netzwerk ausserdem gut für Industrieanwendungen verwendet werden.

Durch das automatische Routen der Knoten, entsteht auf den einzelnen Nodes jedoch ein Overhead, der zum Beispiel BT-Mesh nicht hat. Aus diesem Grund ist der Energieverbrauch auf den Knoten höher, was sich auch in der Abbildung 8 zeigt. Dies ist keine repräsentative Messung, da nur die aktiven Radio Zeiten gemessen wurden, es könnte trotzdem die vorherige These mit dem Overhead bestätigen.

Anders als bei BT-Mesh können mit dem IEEE 802.15.4 und dem 6LoWPAN Layer grössere Pakete versendet werden. Das bedeutet, dass die Segmentierung im Gegensatz zu BT Mesh später stattfindet. Dies erklärt das Phänomen, dass sich der Durchsatz mit steigender Payload vergrößert. Der Thread Stack erreicht bei manchen Messreihen einen Durchsatz von bis zu 32 kbit pro Sekunde. Dies ist jedoch eher unwahrscheinlich. Da bei den Durchschnittsberechnungen kein Median verwendet wurde, können Extremwerte den Durchschnitt verfälschen. Es gab sehr wenige Latenzmessungen, die fehlerhaft waren und eine Latenz

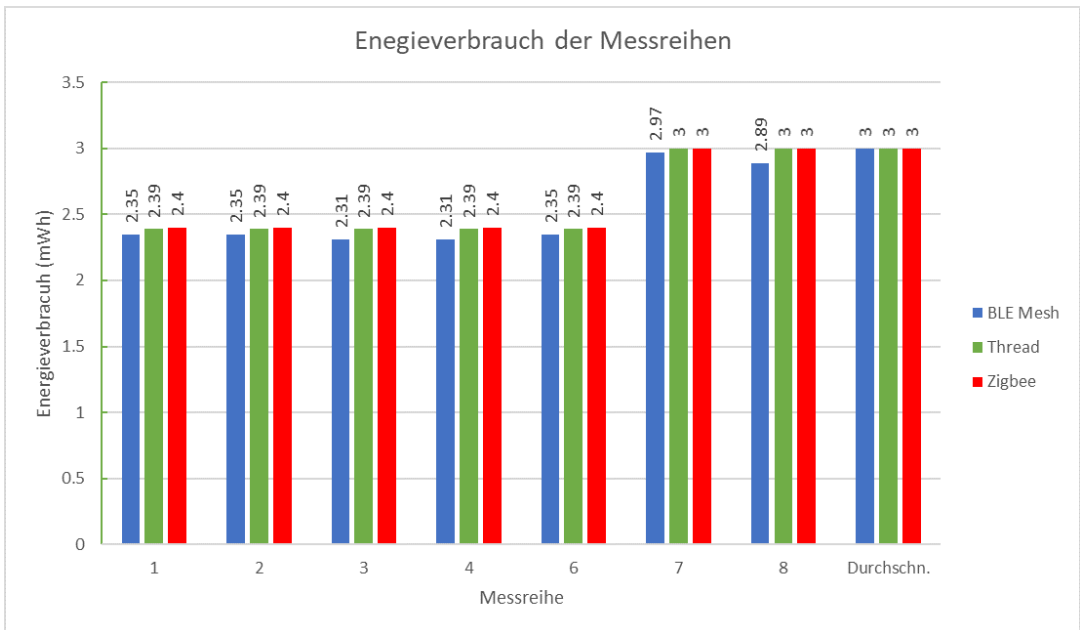


FIGURE 8 Durchschnittlicher Energiebedarf der einzelnen Messreihen im Vergleich

unter 0.5ms aufwiesen. Durch diesen Fehler wurde der Durchsatz in der Auswertung unwahrscheinlich hoch.

4.3.2 | Zigbee

Die Resultate und Vergleiche aus den vorhergehenden Abschnitten zeigen, dass das Zigbee Protokoll in dieser Implementation eine solide Performance abliefert. Auch wenn Zigbee nicht ganz an jene Leistung von Thread herankommt, bestätigen die Ergebnisse wieso es zurzeit das am weitesten verbreiteten WPAN Protokoll ist.

Die durchschnittlichen Latenzzeiten liegen zwischen 30 und 50 Millisekunden. Damit schneidet Zigbee besser ab als Bluetooth jedoch liegt es hinter jenen Durchschnittswerten von Thread. Dies kommt nicht allzu überraschend. Sehr interessant zu beobachten sind die Verteilungen der Latenzzeiten, wie sie beispielsweise in Abbildung ?? zu sehen sind. Anders als beispielsweise Thread weist Zigbee dort deutliche Peaks bei 40ms und 70ms auf. Diese Beobachtung kann bei sämtlichen Messresultaten gemacht werden. Ausreisser nach unten gibt es praktisch keine und solche nach oben nur sehr vereinzelt. Es kann also festgestellt werden, dass bei Zigbee mit einer minimalen Latenzzeit von ca. 30ms gerechnet werden muss, diese aber in den seltensten Fällen deutlich überschreitet. Zigbee wäre also auch für eher zeitkritische Anwendungen geeignet da die Latenzzeiten deterministisch sind. In diesen Zusammenhang nochmals zu erwähnen ist, das Fehlen der Information über die Anzahl Hops die ein Paket während dem Benchmark genommen hat. Es kann davon ausgegangen werden, dass der zweite Peak bei 70ms Latenzzeit diesem Problem verschuldet ist. Wenn also die Anzahl Hops erfasst werden könnte, dürfte sich die Verteilung der Latenzzeiten wohl nochmals reduzieren.

Die Ergebnisse aus Abbildung 5 zeigen eine weitere interessante Eigenschaft von Zigbee. In Messreihe 1 beträgt die durchschnittliche Latenzzeit mit 86ms mehr als das doppelte als in den übrigen Messreihen. Die Ursache dafür kann nicht abschliessend geklärt werden. Jedoch liegt die Vermutung nahe, dass sich das Mesh Netz während dieser Messreihe noch nicht komplett aufgebaut respektive das *Commissioning* und *Route Discovery* noch nicht abgeschlossen war. Dies verursachte

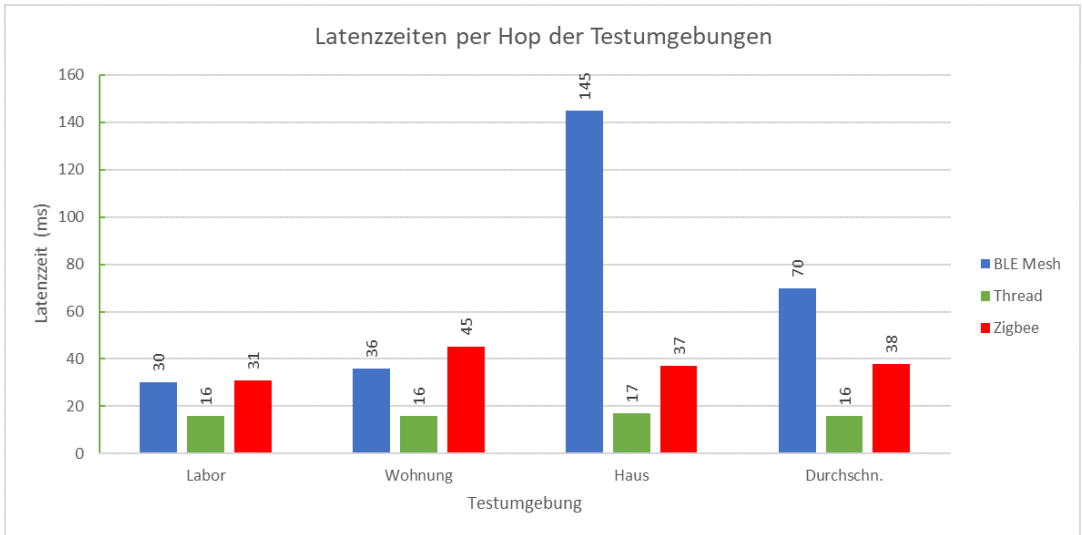


FIGURE 9 Durchschnittliche Latenzzeit per Hop der einzelnen Testumgebungen im Vergleich

zusätzlichen Traffic und beeinflusste den Benchmark.

Auffallend tief ist auch der Paketverlust. In den meisten Fällen lag dieser sogar bei 0 Prozent. Hier macht sich das CSMA/CA des IEEE 802.15.4 MAC Layers bemerkbar. Zudem wirkt sich hier wohl die Unicast Adressierung innerhalb von Zigbee positiv aus. Die Gesamtbelastung des Netzes kann dadurch deutlich minimiert werden.

Die Erhöhung der Payload von 8 auf 50 Byte hat die Latenzzeiten bei den Zigbee Messungen nicht merklich beeinflusst. Somit steigt auch der Durchsatz mit grösserer Payload. Da die Segmentierung dank IEEE 802.15.4 erst ab 127 Byte Framelänge startet ist dies auch nicht weiter verwunderlich. Gegenüber BT Mesh haben Zigbee wie auch Thread den klaren Vorteil grössere Payloads versenden zu können ohne segmentieren zu müssen.

Zigbee brilliert in den Anwendungen in denen es sowieso bereits verbreitet eingesetzt wird. Sei dies eine komplette Hausautomation oder eine Lichtsteuerung, bis zu einer Netzwerkgrösse von ungefähr 200 Nodes mit einigermaßen kleinem Verkehrsaufkommen ist Zigbee die ideale Wahl. Ergänzt durch die *Zigbee Cluster Library* welche die Systeme herstellerunabhängig macht wird Zigbee noch interessanter.

4.3.3 | Bluetooth Mesh

Die Performance von Bluetooth-Mesh ist stark von der Belastung abhängig, wie aus dem Abschnitt 4.1 zu erkennen ist. Der grösste Einfluss auf die Performance hatte die Länge der gesendeten Nachrichten. Dies hat seinen Ursprung das ein einzelnes Bluetooth-Mesh-Paket lediglich 10-Byte aufnehmen kann (siehe Abschnitt ??). Der Stack beginnt mit der Segmentierung der Daten in mehrere kleine Pakete. Bei 32-Byte sind dies bereits 4 Frames. Damit ist der komplette Network-Stack überfordert.

Abbildung 13 zeigt eine Messung mit 32-Byte Payload, einer Message-Dichte von 3 Sekunden / Message welche Sequenziell versendet wurden. Dabei ist zu erkennen, dass die Anzahl der Ongoing Transactions immer weiter ansteigt. Lediglich in der Nachbearbeitungszeit sinkt die Kurve wieder. Dies deutet darauf hin das zu viel Traffic im Netz gesendet wird. Angelehnt an die Ongoing Transactions steigt die Latenz von einer Sekunde auf einige 100 Sekunden an.

Vermutlich entsteht eine zu hohe Message-Dichte, da zu viele Relay-Nodes jede empfangene Message wiederholen. Durch

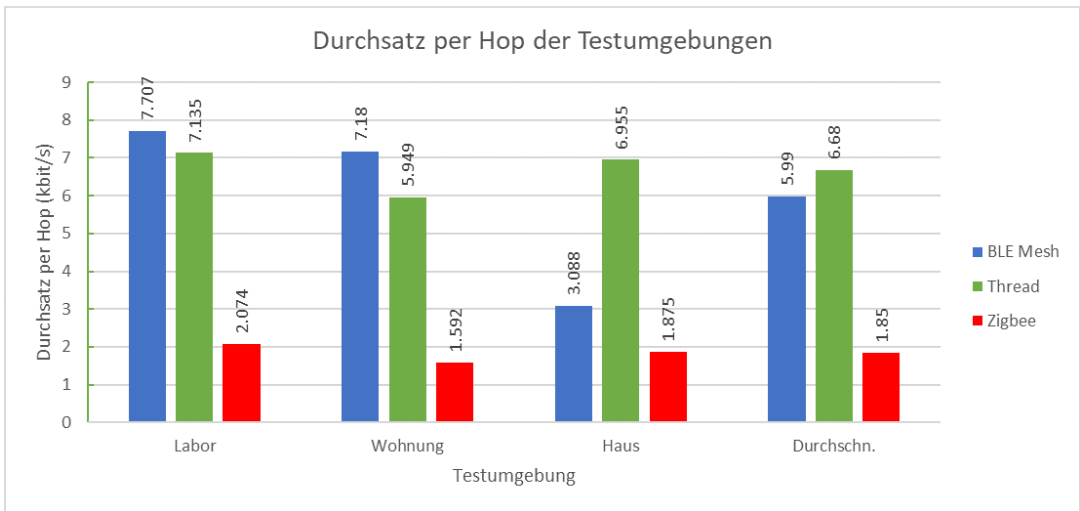


FIGURE 10 Durchschnittlicher Durchsatz per Hop der einzelnen Testumgebungen im Vergleich

deaktivieren einiger Relays könnte der Netzaufbau entlastet werden. Jedoch erfordern solche manuellen Eingriffe Fachwissen und verringern die Ausfallsicherheit des Netzes.

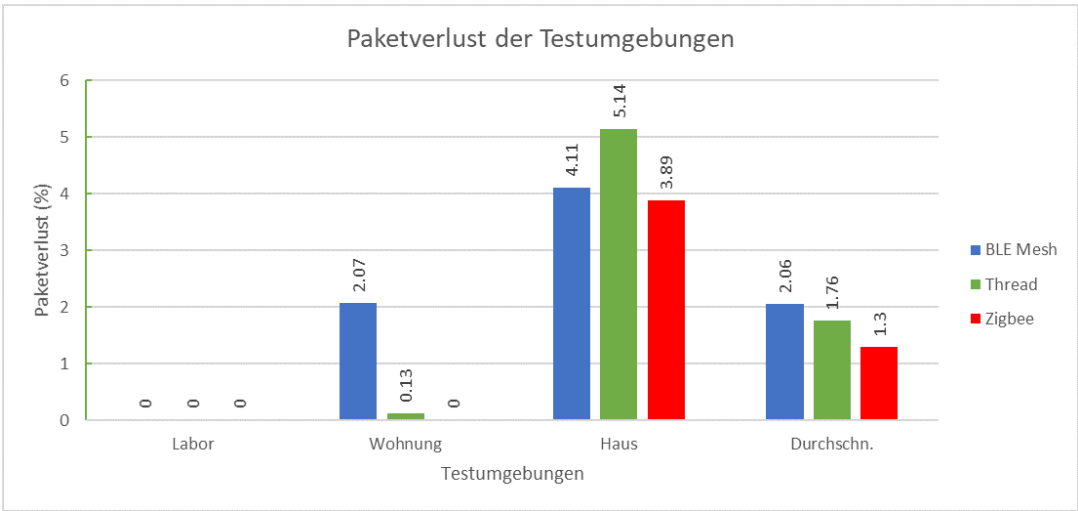


FIGURE 11 Durchschnittlicher Paketverlust der einzelnen Testumgebungen im Vergleich

5 | VALIDIERUNG

Fehlerabschätzung und Vergleich mit Benchmarks von anderen Organisationen

ERGÄNZENDE INFORMATIONEN

Infos die evtl. wichtig sind aber nicht unbedingt in den Kontext gehören

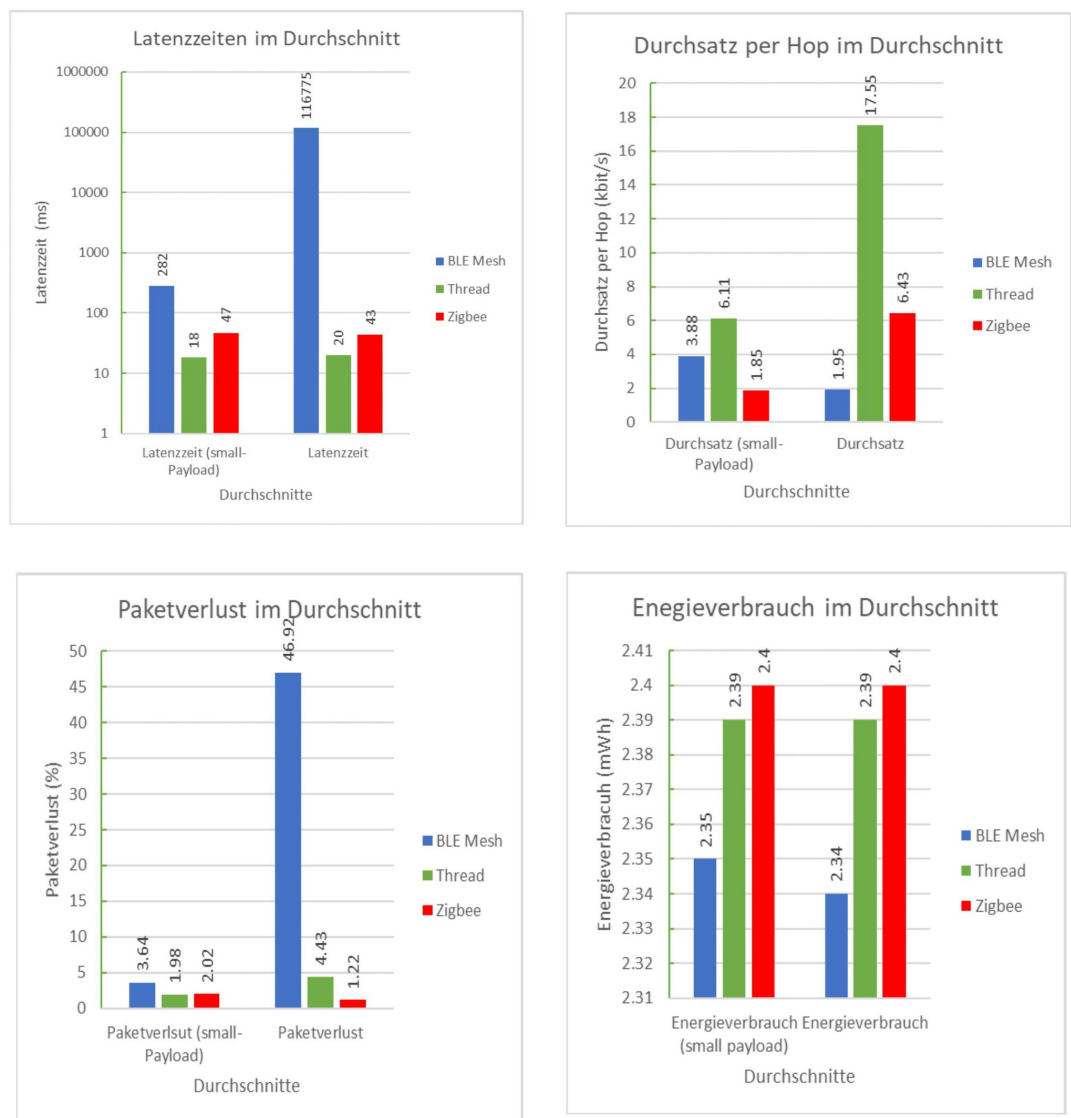


FIGURE 12 Durchschnittliche Messgrößen im Vergleich inkl. separater Betrachtung der Messergebnisse, welche mit 8-Byte Paketlänge erzielt wurden

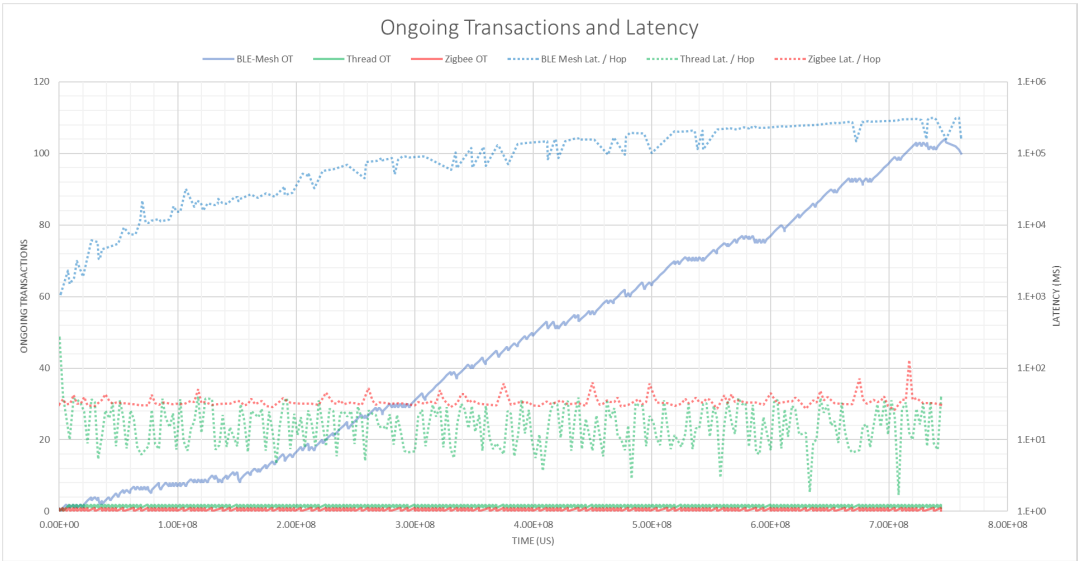


FIGURE 13 Ongoing Transactions und Latenzzeiten einer Messung mit 32Byte Paketlänge