



Vortrag am Donnerstag,
den 19. November 2009



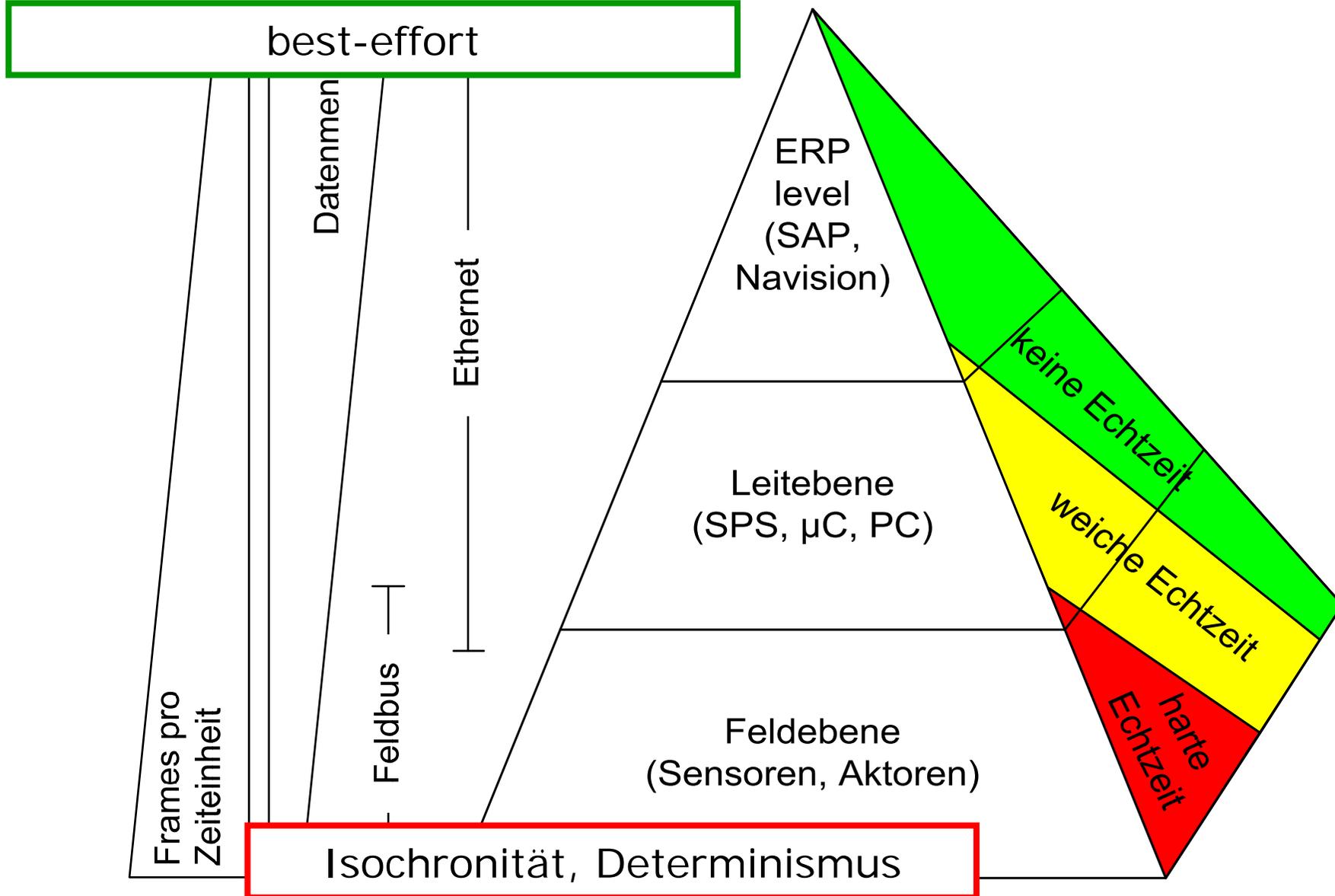
Thema aus der Dissertation
an der Universität Siegen,
FB12 - Betriebssysteme und verteilte Systeme
(Prof. Dr. Roland Wismüller)

Ein Framework für echtzeitfähige Ethernet-Netzwerke

Referent

Dr. Frank Dopatka, GFU Cyrus AG

Motivation: Idee der Vertikalen Integration



Ziel: Ethernet echtzeitfähig machen! Feldbusse verdrängen und/oder integrieren



- minimale Verzögerung durch Hubs
- zentrale Schedule, zentraler Manager
- jedes Gerät muss Protokoll beachten
- geschlossenes Subnetz



- Schieberegister-ähnliche Übertragung in einzelnen großen Ethernet-Frames
- zentralisiert
- jedes Gerät muss Protokoll beachten
- geschlossenes Subnetz



- offene Infrastruktur
- dezentrale Kommunikation
- (proprietäre Nicht-Standard-)Switches

- **jeder Ansatz ist ein fester Kompromiß**
- **Aufgabe: Kompromisse in einem Framework unter ein Dach bringen**

Konkretere Ziel-Formulierung:

- Erstellung eines flexiblen, formal beründeten Frameworks, das von einer speziellen Implementierung unabhängig ist

Ausgangspunkt:

- Gegebene Netzwerk-Infrastruktur
- Gegebene IRT-Übertragungen

Vorgehensweise:

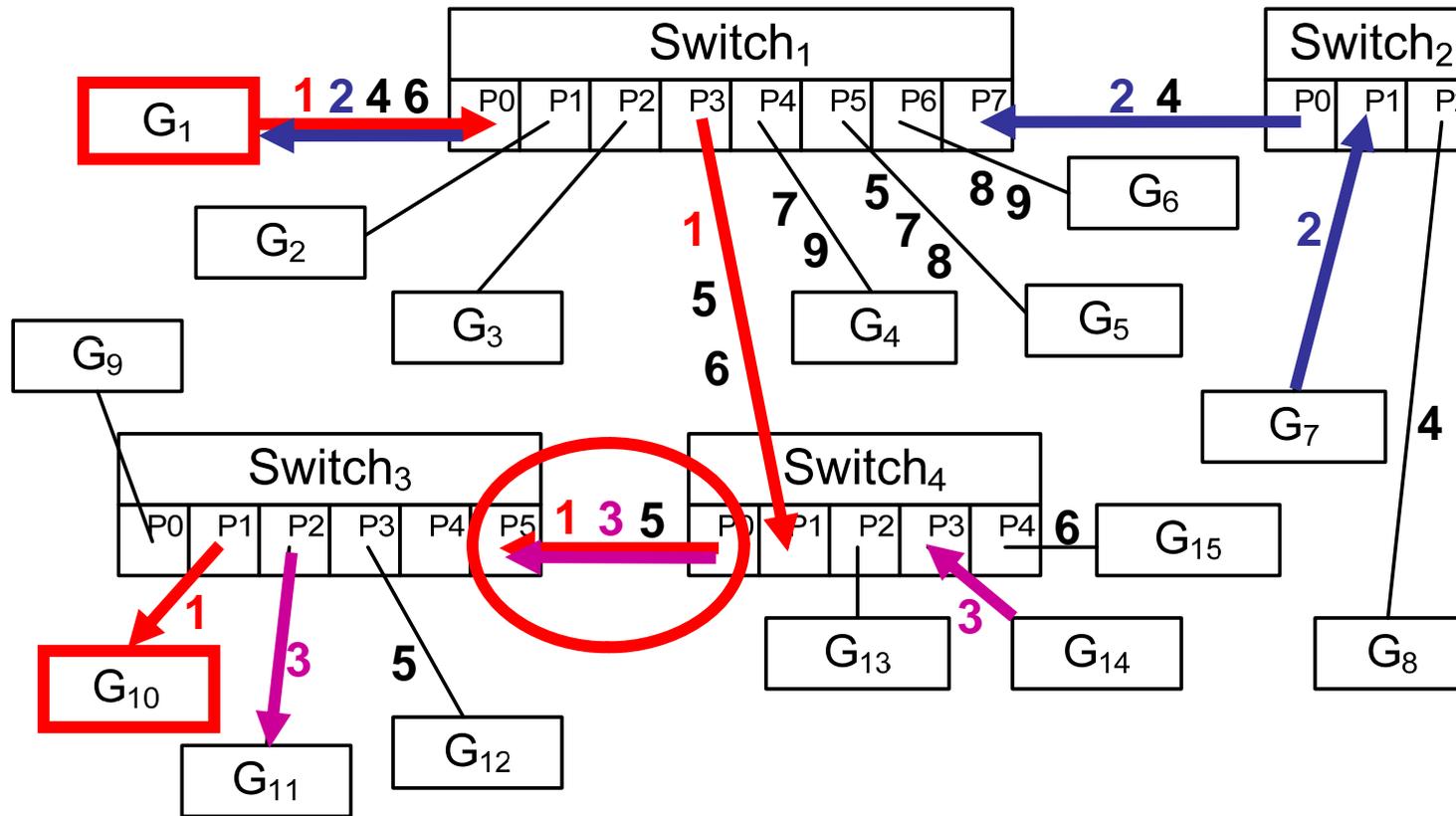
- Vom vereinfachten Modell zur realitätsnahen Anwendung
- Zerlegung in Teilprobleme und algorithmische Lösung

erstes Beispiel im Folgenden:

- Idealisierte Unicast (1:1) Vollduplex-Übertragungen
- Keine Berücksichtigung von Verzögerungszeiten
- Einheitliche (Mindest-)Framegröße
- Verwendung von Switches

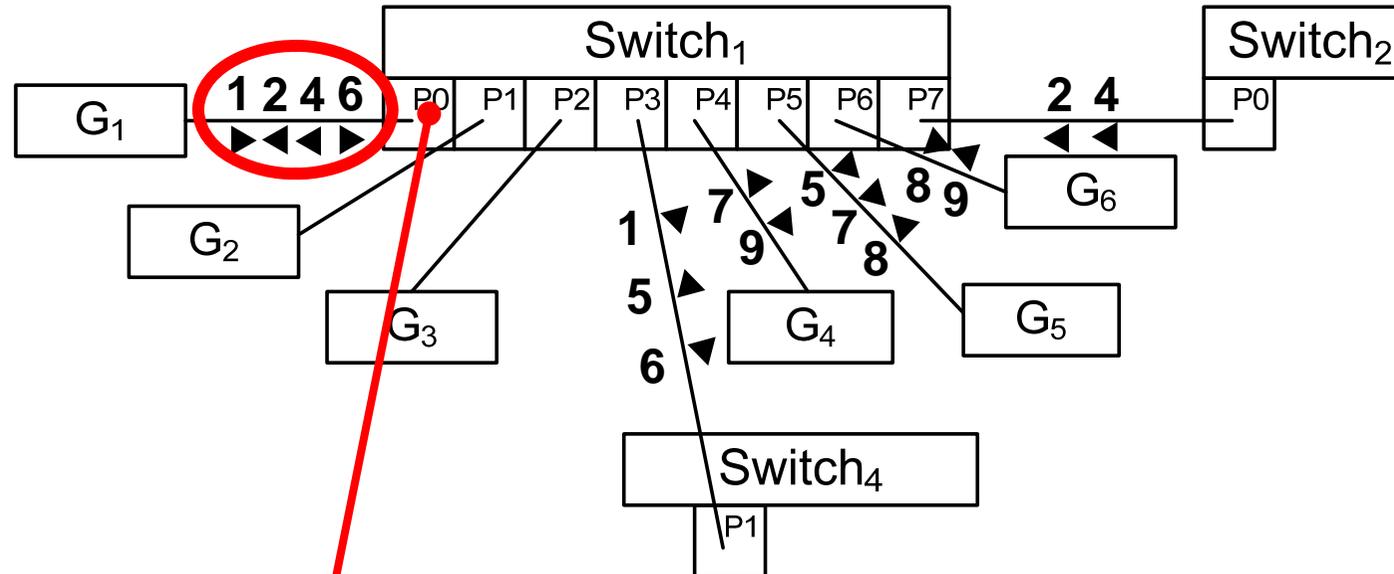


Idealisierte Vollduplex-Verbindungen



Gerichtete Übertragungen erzeugen „Konflikte“:
 Gefahr der Zwischenspeicherung/Verwerfung bei identischer Übertragungsrichtung → Scheduling!

Betrachtung von Switch 1: Vollduplex



An jedem Port gibt es
eingehende und ausgehende
Übertragungen!

Ansatz dieser Arbeit:

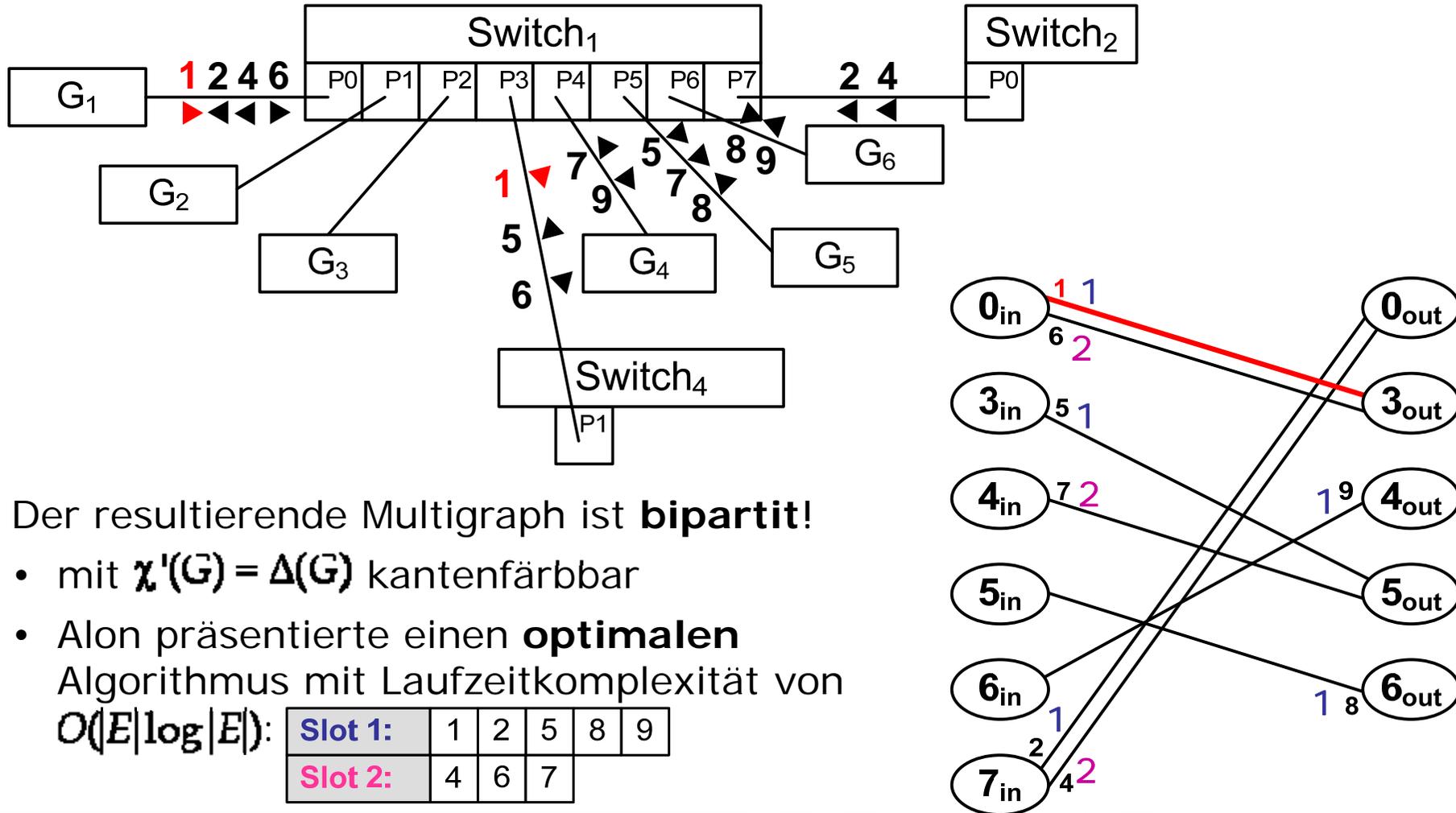
jeden Switch einzeln betrachten, Konfliktgraphen bilden
und diesen färben...

Ziel:

Konflikte zeitlich entzerren → TDMA-Schedule

- Eingangs- und Ausgangs-Teil eines Ethernet-Ports
→ je ein Knoten im Graph
- Eine Unicast-Übertragung verläuft von genau einem Eingangs-Teil zu genau einem Ausgangs-Teil eines Ports
→ eine Kante im Graph
- Färbung: Jede Kante mit einer Farb-Nummer versehen
- zwei Kanten mit einem gemeinsamen Knoten
→ Konflikt! (Benutzung eines gemeinsamen Port-Teils)
→ verschiedene Farb-Nummern
→ verschiedene Zeit-Slots der Ausführung
- Minimierung der Anzahl der vergebenen Farb-Nummern
→ optimale Schedule
→ minimale Zykluszeit

Betrachtung von Switch 1



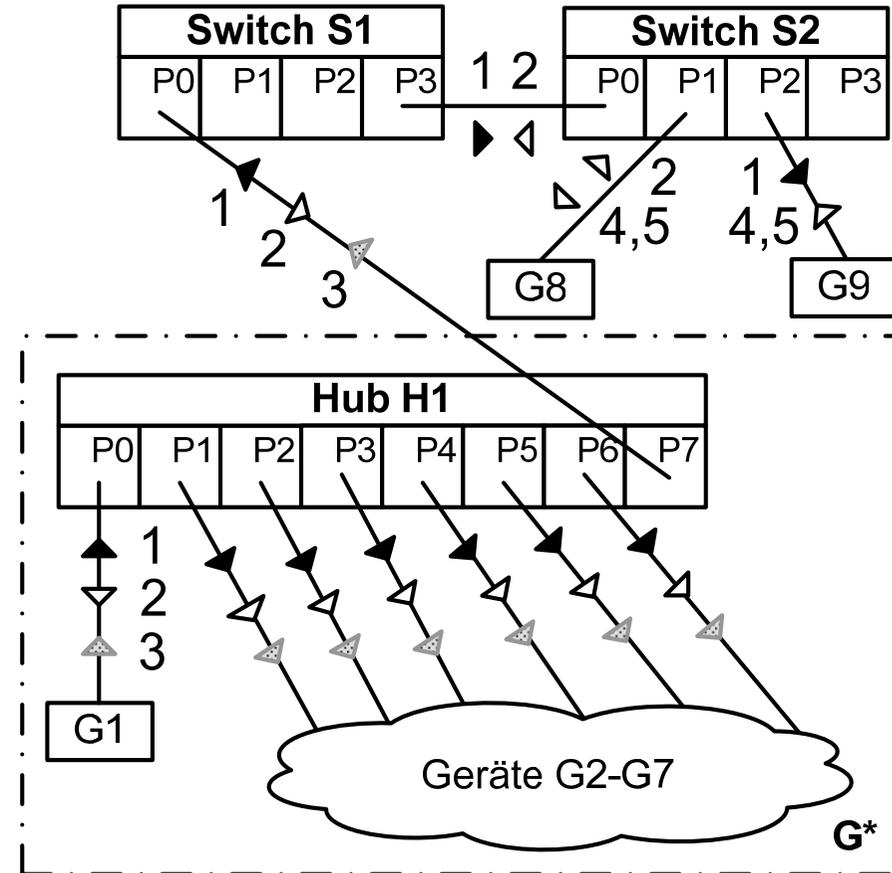


Integration von Hubs

Ein realistisches Netzwerk mit einem Hub, 8 Achsen und zwei Switches

Übertragung Nummer...

- 1 von Gerät G1 (Master-Achse) zu Gerät G9 (SPS)
- 2 von Gerät G8 (SPS) zu Gerät G1
- 3 von Gerät G1 zu Geräten G2 bis G7 (Slave-Achsen)
- 4 und 5 von Gerät G9 zu Gerät G8 (Sync. der SPSen)



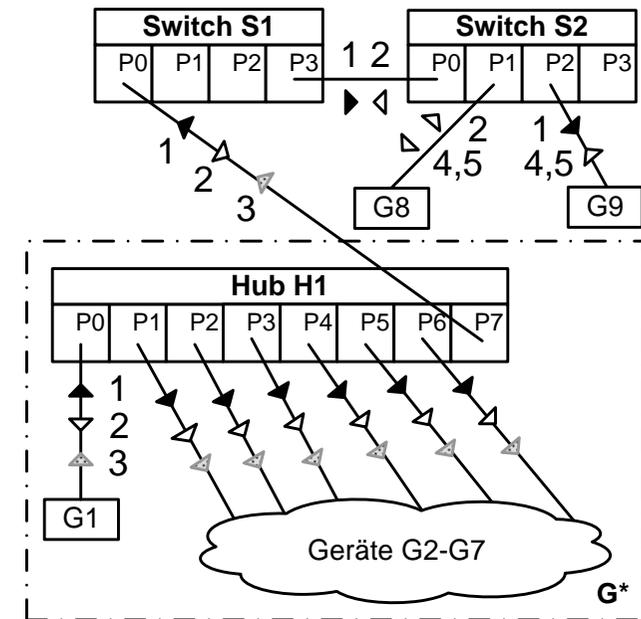
- Die Übertragungen 1 und 2 stehen an Switch 2 und an Port 3 von Switch 1 nicht in Konflikt!
- Am Hub und an Port 0 von Switch 1 stehen die Übertragungen 1 und 2 aber in Konflikt zueinander!
 → Halbduplex-Betrieb → nur 1 Übertragung zeitgleich erlaubt!

→ gesamtes Netz betrachten:
 Knoten im Graph: Übertragungen
 Kanten im Graph: Konflikte

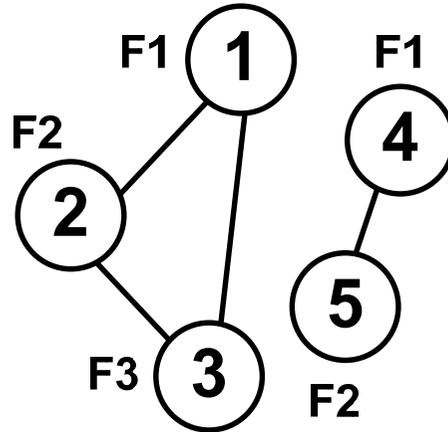
Knotenfärbung:

zwei Knoten mit einer gemeinsamen Kante

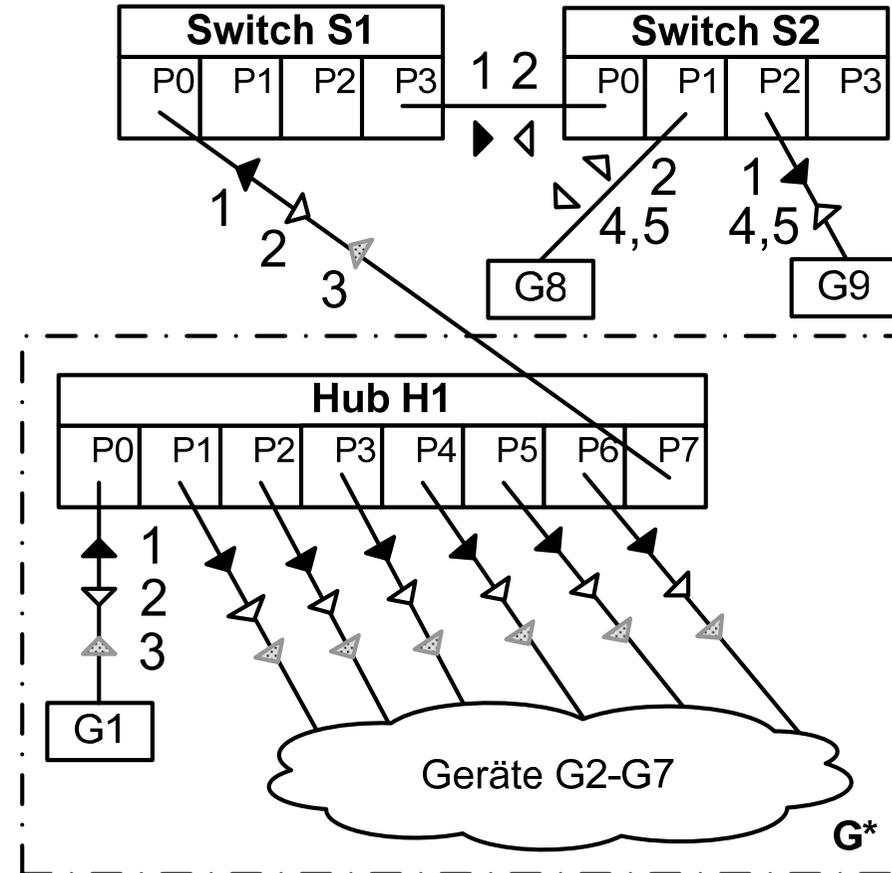
- Konflikt
- Knoten-Färbung anwenden
- Schedule für das gesamte Netz



Vom Graph zur Schedule...



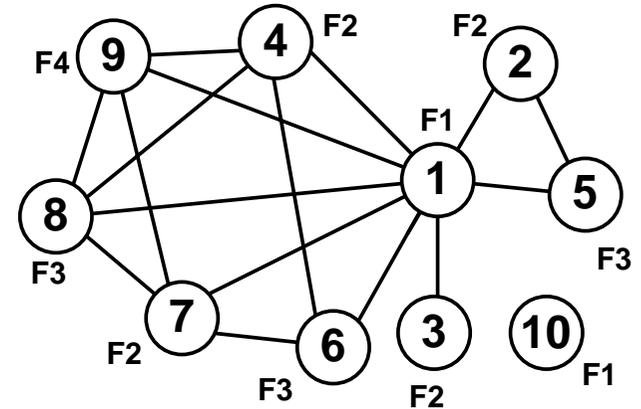
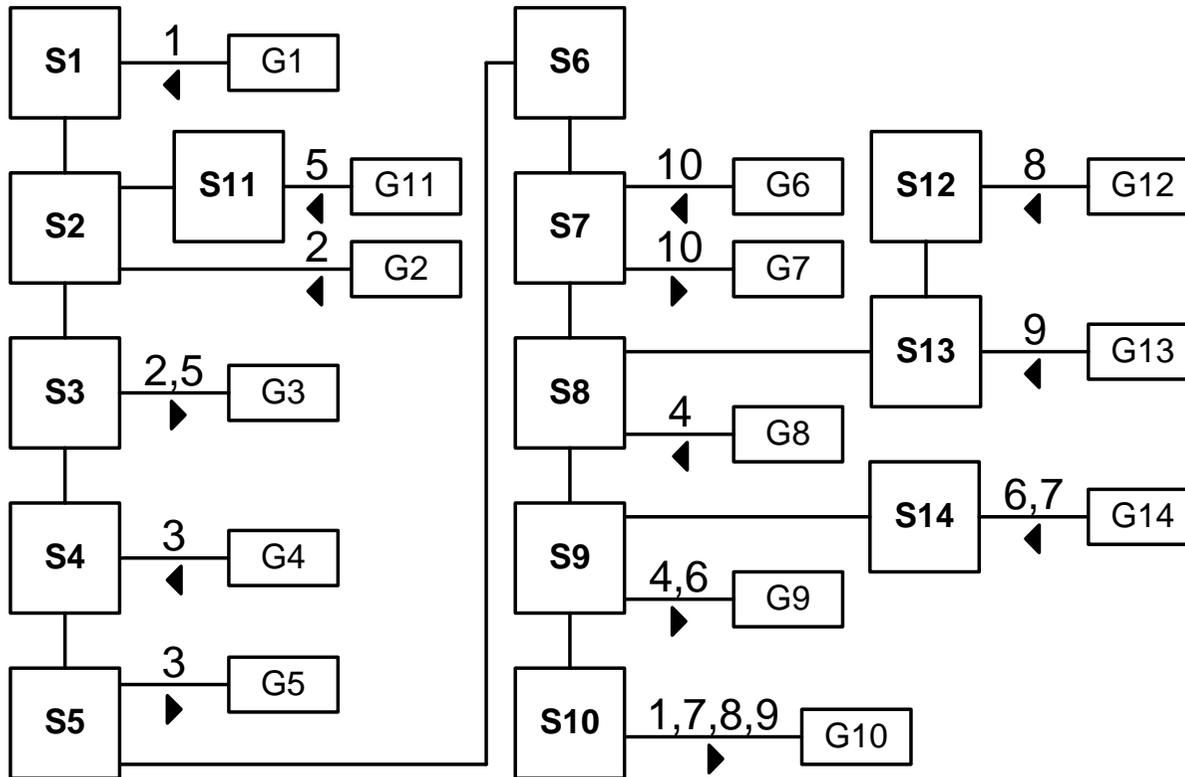
S1	1	2	3	→ t
	F1	F2	F3	
	4	5		
S2	1	2	3	→ t
	F1	F2	F3	
	Farbe/Slot			





Berücksichtigung von Verzögerungszeiten

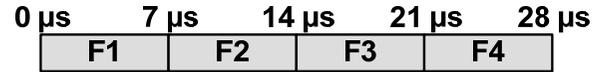
Eine „lange“ Übertragung und Schedule ohne Verzögerung...



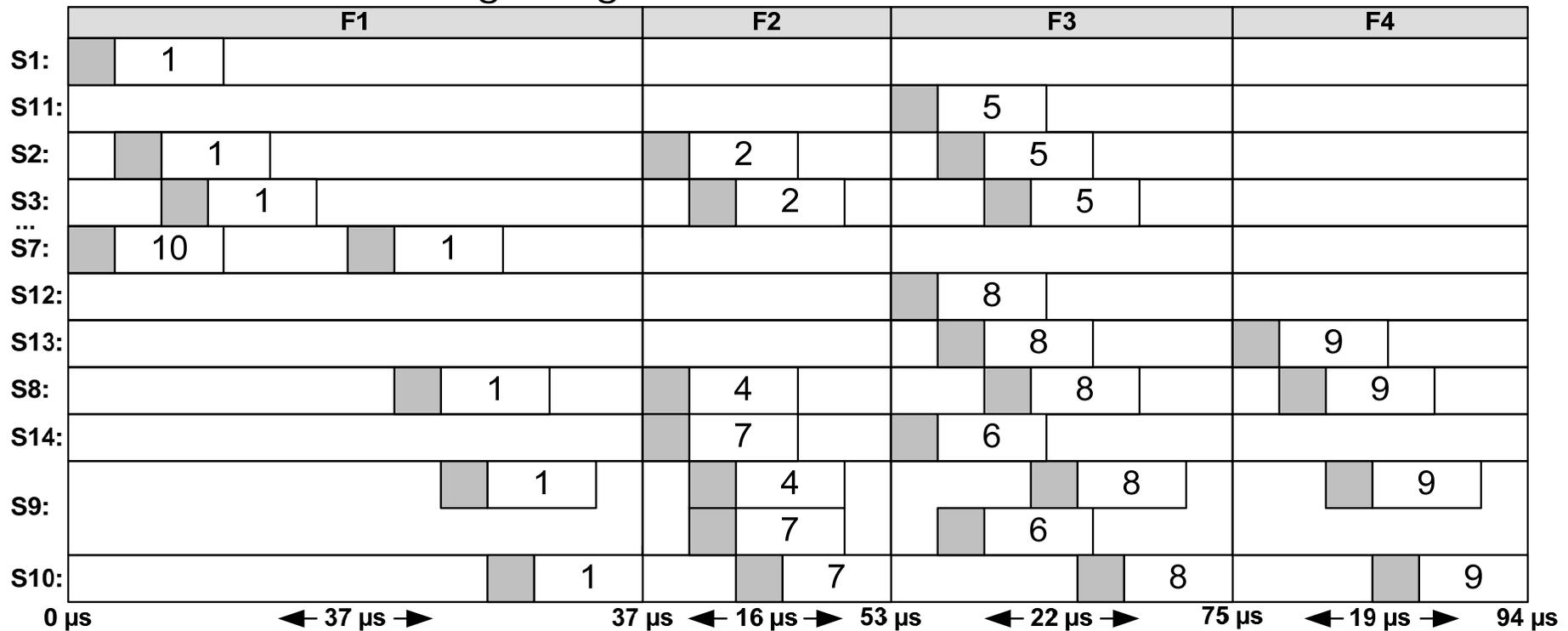
S _x		2		
		3	5	
	10	4	6	
	1	7	8	9
	F1	F2	F3	F4
	Farbe/Slot			
				t

Berücksichtigung einer Verzögerung von $3\mu\text{s}/\text{Switch}$ (ProfiNet ERTEC 400)

ohne Verzögerung ergibt sich eine Frame-Laufzeit von ca. $7\mu\text{s}/\text{Switch}$:



mit addierten Verzögerungen:





Berücksichtigung von Verzögerungszeiten

- Grundlagen für ein **formales Framework** gelegt zur Integration bestehender RT-Ansätze
- **Graphen-Färbung & List-Scheduling** geben Impulse für eine Schedule-Berechnung
- **Simulierbarkeit** der Integration nachgewiesen
- **Erweiterbares Modell** bis hin zu realitätsnahen Anwendung unter Berücksichtigung von Verzögerungszeiten
- Ggf. **Fortsetzung der Forschung:**
 - Netzwerk-Simulator
 - weitere Integration (Sercos, EtherCAT)
 - eigener RT-Switch



**Vielen Dank für
Ihre Aufmerksamkeit!**

Noch Fragen?