Anwendungsgewahre Ressourcenverwaltung in Echtzeitregelungssystemen

Seminarvortrag

Daniel Schiener
20. Juni 2014

LEHRSTUHL FÜR VERTEILTE SYSTEME
UND BETRIEBSSYSTEME

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

Einleitung

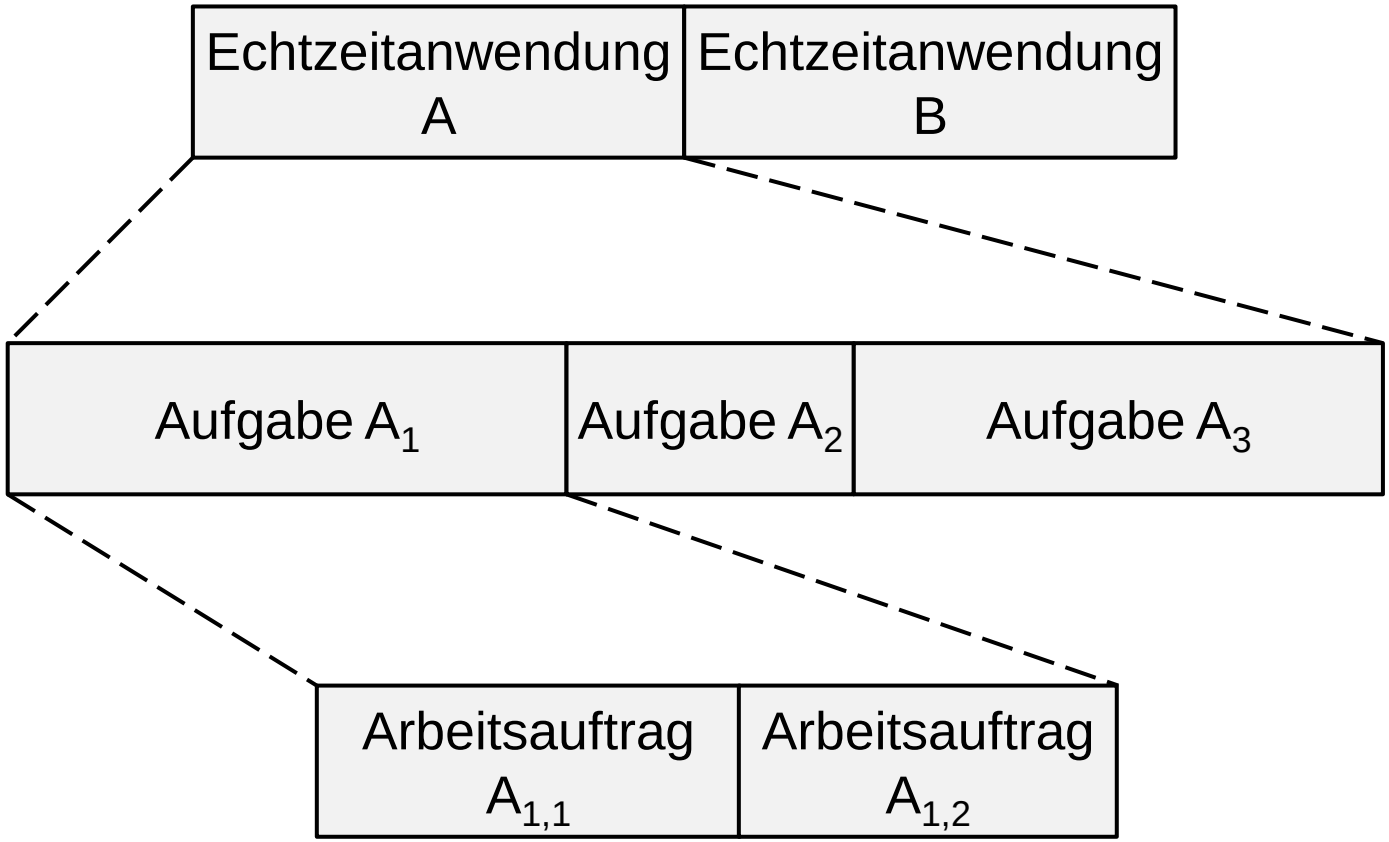
- „Spezialfall“ Echtzeitsystem
  - Echtzeit ≠ Geschwindigkeit
  - nicht-funktionale Anforderungen im Fokus (Rechtzeitigkeit, Vorhersagbarkeit, Ausfallsicherheit)
  - Einsatzgebiet: eingebettete Systeme
  - → Ressourcenbeschränkungen (Rechenzeit, Speicher)
  - Vielzahl bekannter Methoden nicht anwendbar

Agenda

- Die Welt der Echtzeit - Grundlagen
  - Struktureller Aufbau von Echtzeitsystemen
  - Eigenschaften von Echtzeitsystemen (Termin, Hyperperiode)
  - Typen von Echtzeitsystemen
  - naive Rekonfiguration: „one-shot“
- Anwendungsgewahre Ressourcenverwaltung
  - Problemfall Überlast
- Toleranz der Überlast
  - durch Anpassung der Perioden
  - durch Anpassung der Rechenzeit

Die Welt der Echtzeit – Grundlagen (1)

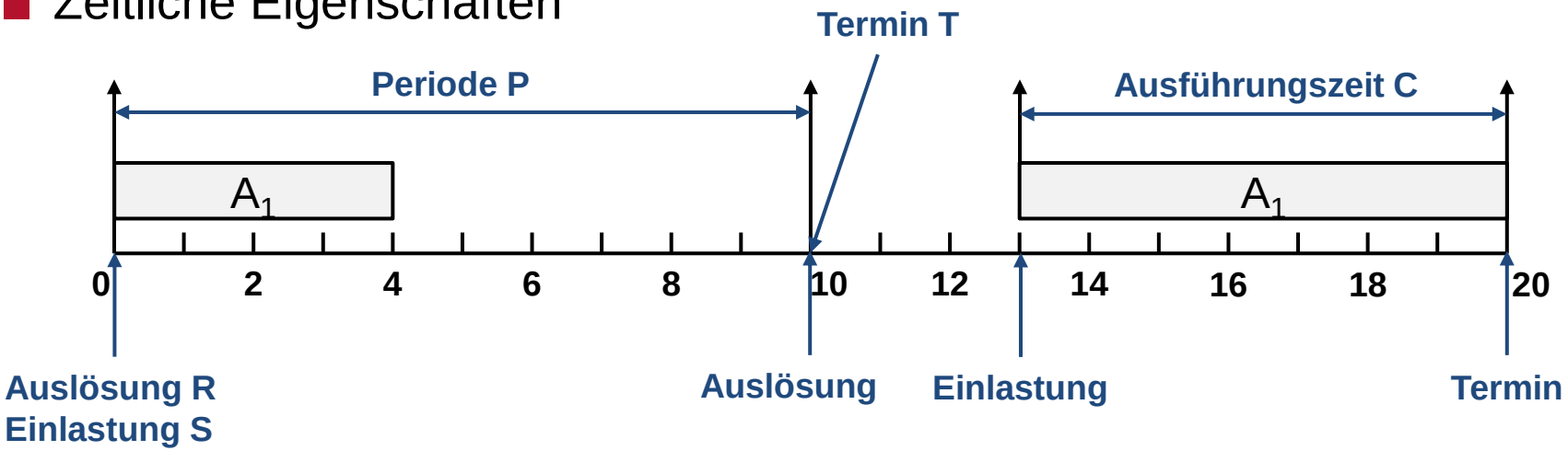
- Struktureller Aufbau



Die Welt der Echtzeit – Grundlagen (2)

- Aufgaben
  - periodisch, aperiodisch, nicht-periodisch
  - Priorität (Verdrängbarkeit)

- Zeitliche Eigenschaften



- Terminarten
  - weich → Terminverletzung hinnehmbar, Datennutzen nimmt ab
  - fest → Terminverletzung hinnehmbar, Daten sofort unbrauchbar
  - hart → Terminverletzung katastrophal (Airbag)

Die Welt der Echtzeit – Grundlagen (3)

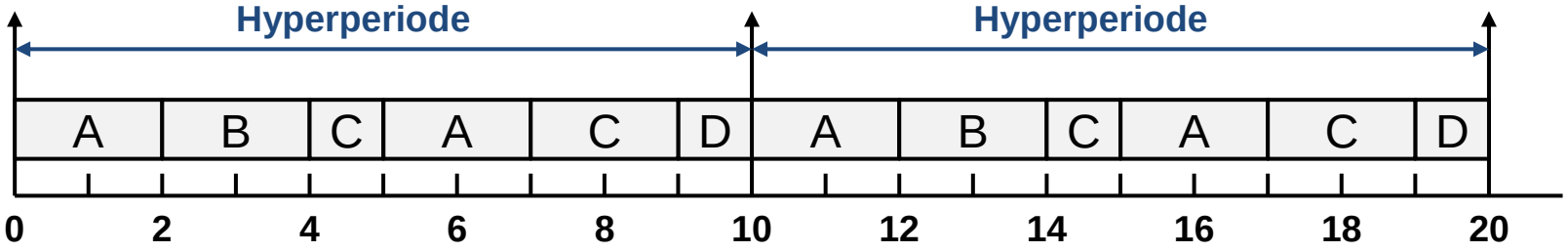
- Taktgesteuerte Echtzeitsysteme
  - statische Ablauftabelle (Auslösezeiten, Ausführungszeiten, Termine, etc.)
  - Aufgabenauslösung durch Zeitgeber
  - keine externen Ereignisse (Unterbrechungen)
  - keine Prioritäten
  - i.d.R. harte Termine

- Ereignisgesteuerte Echtzeitsysteme
  - prioritätsbasierte Ablaufplanung → Verdrängung
  - Aufgabenauslösung durch Ereignisse
    - externe Ereignisse (Unterbrechungen z.B. durch Zeitgeber)
    - interne Ereignisse
  - prioritätsbasierte Ablaufplanungsverfahren (RM, DM, EDF)
    - **R**ate **M**onotonic: je kürzer die Periode desto höher die Priorität
  - weiche bis harte Termine

Die Welt der Echtzeit – Grundlagen (4)

- Hyperperiode
  - Dauer: kgV aller Perioden
  - jede Aufgabe mindestens 1x ausgeführt
  - Ablaufplan/Ablauftabelle beginnt von vorn
  - keine Abhängigkeiten von Betriebsmitteln (Sperrvariablen)
  - Grundzustand (quiescence state)

- Beispiel: Ablaufplan nach RM
  - A: Periode 5
  - B, C, D: Periode 10
  - → Hyperperiode: 10

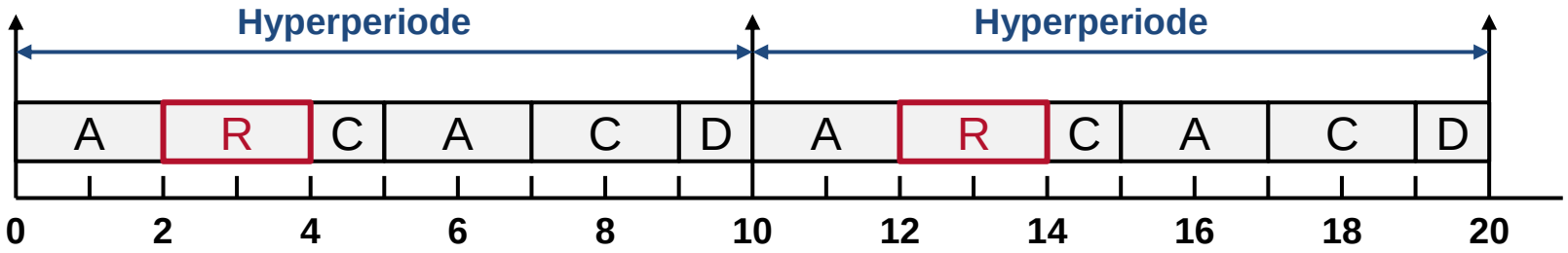


Rekonfiguration: „one-shot“

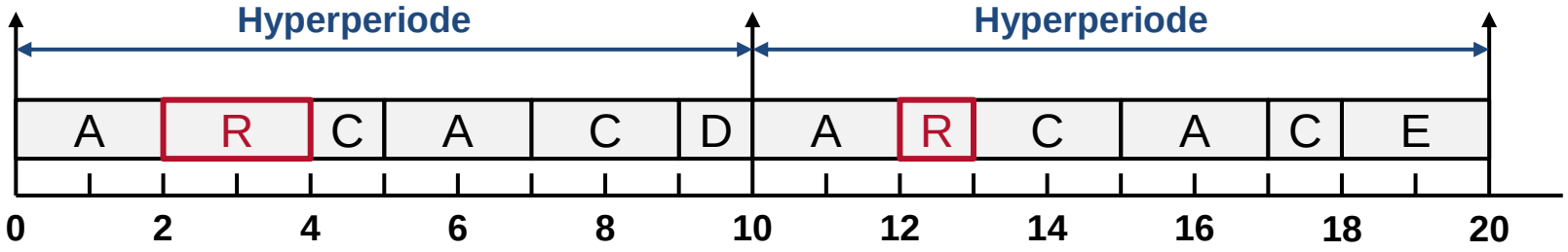
- Szenario
  - Menge von Aufgaben unterschiedlicher Prioritäten und Perioden
  - weiche bis harte Termine
  - abhängige und unabhängige Aufgaben
  - taktgesteuertes oder ereignisgesteuertes System
- Rekonfigurationssituation
  - Wechsel des Betriebsmodus
- Konzept
  - Tausch der Ablaufpläne/Ablauftabellen durch Umschaltung („one-shot“)
  - Ausnutzen der Hyperperiode
    - keine Abhängigkeiten
    - keine Ressourcenkonflikte durch allokierte Betriebsmittel

„one-shot“ – Beispiel: Regelungsanwendung R

- Normalbetrieb



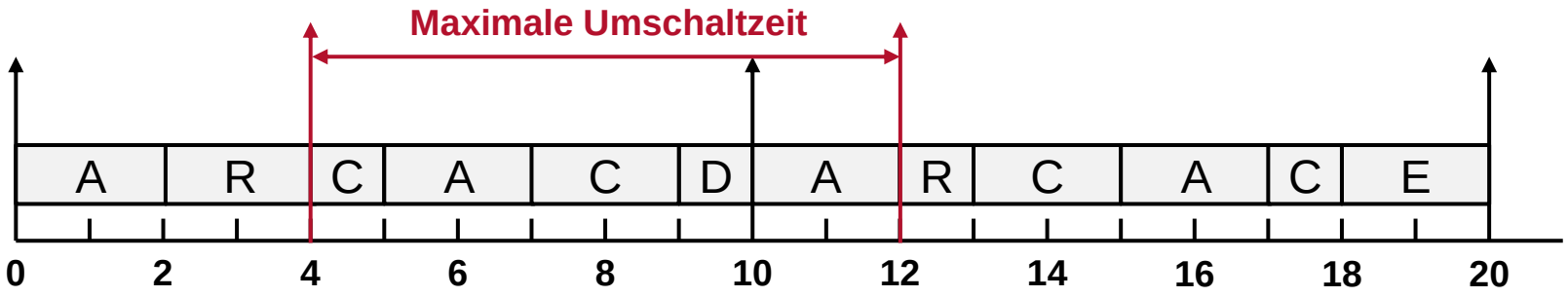
- Umschaltvorgang



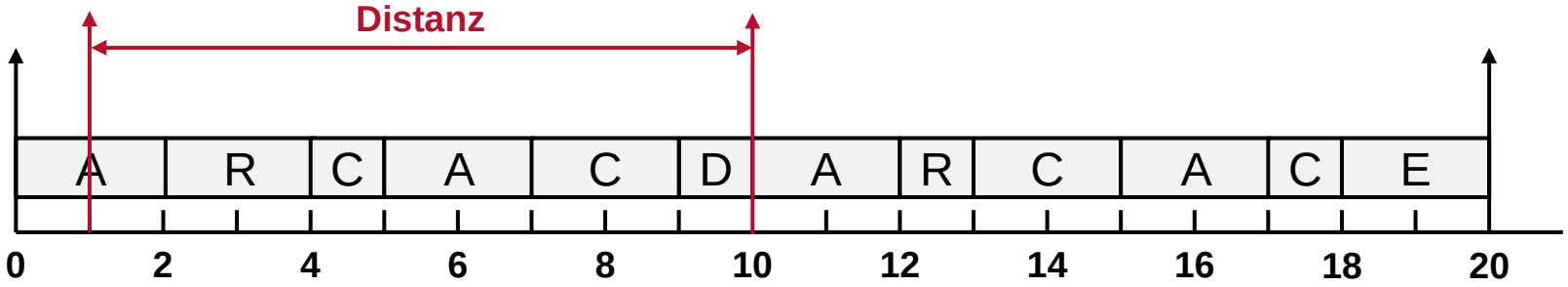
  - Periode von R bleibt gleich
  - Ausführungszeit von R nimmt ab
  - Echtzeitanwendung E ersetzt D

„one-shot“ – Beispiel: Regelungsanwendung R

- **Problem:** Maximale Umschaltzeit
  - hochfrequente Aufgaben in Konflikt mit zeitintensiven Aktualisierungen

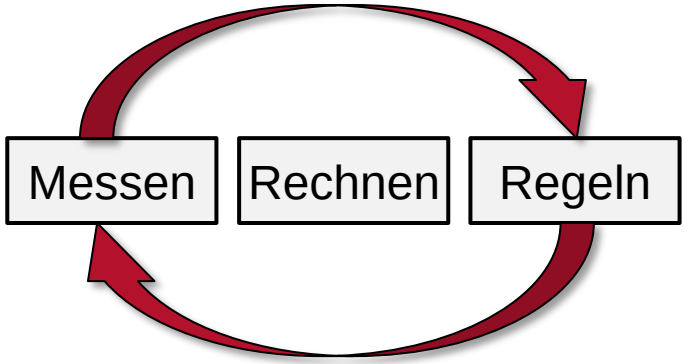


- **Problem:** Distanz zwischen Anforderungs- und Umschaltzeitpunkt
  - Latenz, Trägheit, sehr statisch



Motivation – Anwendungsgewahre Ressourcenverwaltung

- Definition: Echtzeitregelungssystem
  - Regelungsanwendungen häufigste Art von Echtzeitsystemen
  - Einsatzgebiete
    - Automobil
    - Luft- und Raumfahrt
    - Industrieanlagen
  - Regelung typischerweise zyklisch
    - Lesen von Sensordaten
    - Berechnen von Stellwerten
    - Durchführen der Regelung



Motivation – Anwendungsgewahre Ressourcenverwaltung

- Rekonfiguration in Echtzeitregelungssystemen
  - Rekonfiguration überwiegend auf Anwendungsebene
  - **Gründe**
    - Moduswechsel wegen veränderter Umweltbedingungen (Schneefall, Nebel)
    - situationsbedingte Ausnahmefälle (Prozessorausfall)
    - seltener: Aktualisierungen der Anwendung
  - **Probleme**
    - Wiederherstellung des Normalzustands benötigt Zeit
    - Zeit aufgrund der Seltenheit des Fehlerfalls nicht einkalkulierbar
  - **Folgen**
    - Überlast
    - Abnahme der Regelgüte
    - **Systemausfall (Anti-Blockier-System, ABS)**
  - **Ziele**
    - Toleranz der Überlast
    - Gewährleistung der Regelgüte
    - Prävention eines Systemausfalls
- **Fokus:** Einhaltung nicht-funktionaler Eigenschaften

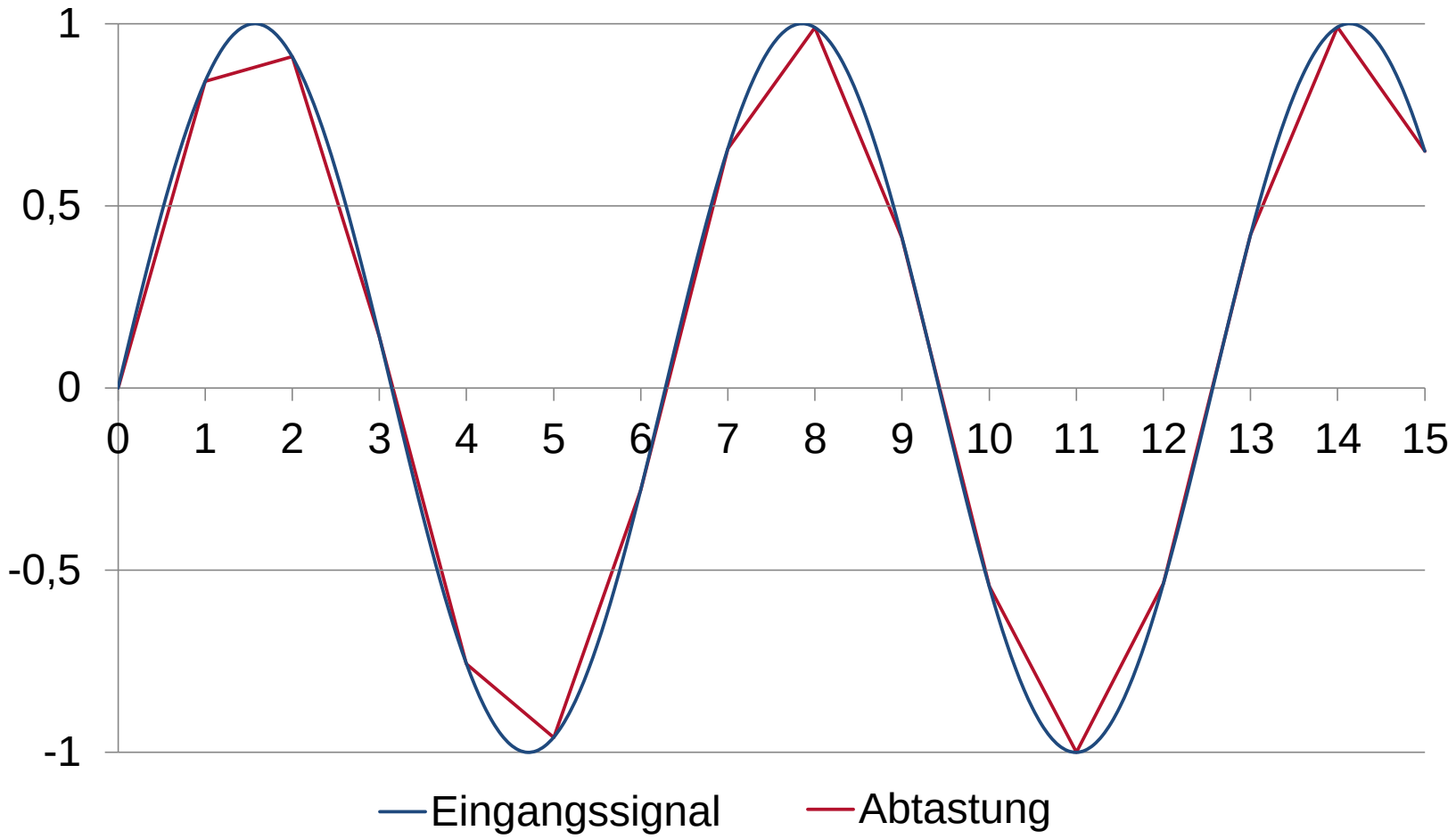
Robustheit der Regelung

- Überlastsituation
  - **Grund:** Ausnahmefälle
    (z.B. Anwendungswiederherstellung nach Prozessorausfall)
  - **Problem:** keine Zeit für die Behandlungsroutine im Ablaufplan
    („woher nehmen wenn nicht stehlen?“)
  - **Lösung:** (Regel-)Aufgaben wird Zeit entzogen
  - **Folgen:**
    - Berechnungszeiten unzureichend
    - Zunahme von Terminverletzungen
    - Abnahme der Regelgüte

- Inhärente Robustheit
  - Überdimensionierung der Regelung
    - Verschlechterung der Abtastung
    - Verschlechterung der Regelung (Periodenausfall)
  - Terminverletzung → Signalverarbeitung funktioniert ggfs. weiter
  - **ABER:** nur begrenzt tolerierbar!

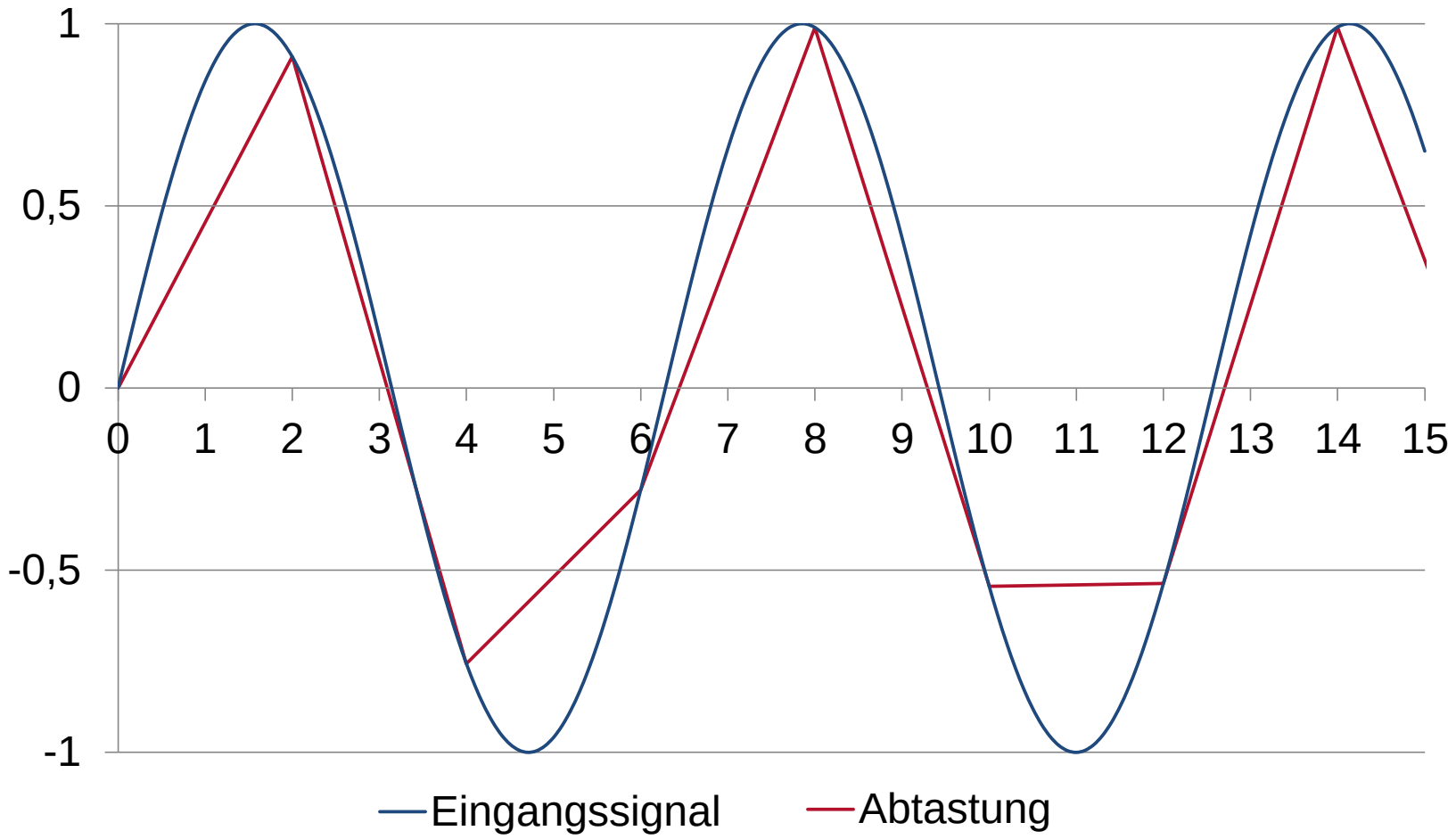
Beispiel: Signalverarbeitung

- Beispiel: Signalabtastung durch Regelaufgabe mit Periode 1



Beispiel: Signalverarbeitung

- Beispiel: Signalabtastung durch Regelaufgabe mit Periode 2



Anpassung der Perioden - Szenario

- Szenario
  - periodische, hochpriore Regelaufgabe
  - Menge von Aufgaben unterschiedlicher Priorität und Periode
  - abhängige und unabhängige Aufgaben
  - Ablaufplan nach RM mit harmonischen Perioden
  - Multiprozessor-System

- Rekonfigurationssituation
  - Wechsel des Betriebsmodus
  - Anwendungswiederherstellung nach Prozessorausfall

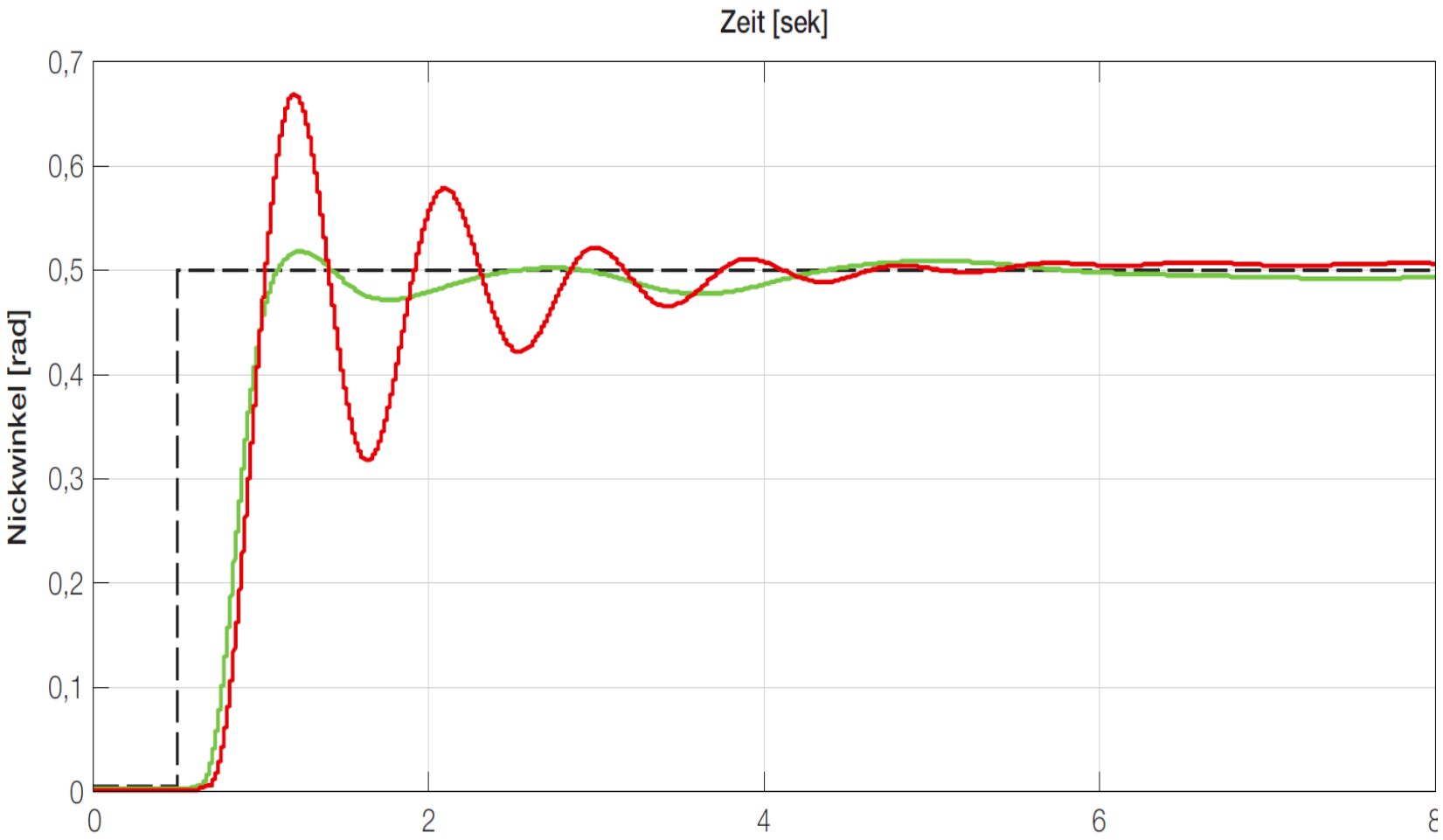
Anpassung der Perioden – Konzept (1)

- Belohnungssystem
  - Aufgaben besitzen…
    - Menge möglicher Perioden
    - Nutzen und „Belohnung“
  - Perioden, Nutzen und „Belohnung“ zur Entwurfszeit bekannt
  - je wichtiger die Aufgabe zur Gewährleistung der Regelgüte…
    - desto höher ist ihr Nutzen
    - desto höher ist ihre „Belohnung“
    - desto höher ist ihr Anteil an Rechenzeit
  - → verringert sich Anteil an Rechenzeit (Periode verlängert sich), so erhöht sich die „Belohnung“ / der Nutzen

→ **Ziel:** Maximierung der „Belohnung“ bzw. des Gesamtnutzen

→ Zeitgewinn durch dynamische Anpassung der Perioden

Anpassung der Perioden – Beispiel: Nutzen/Regelgüte

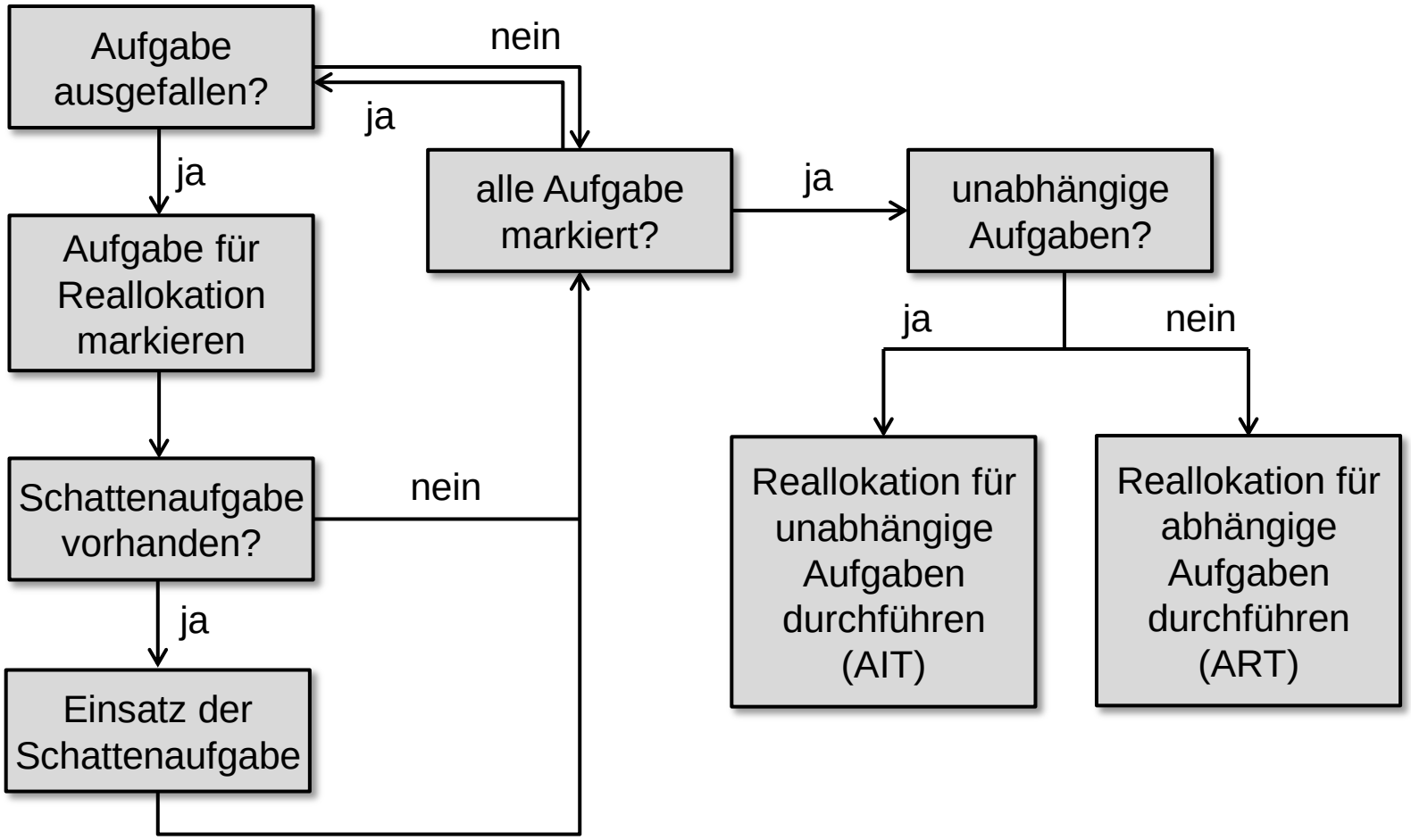


Anpassung der Perioden – Konzept (2)

- Performanz-Index PI (performance index) = Regelgüte
  - Ermittlung durch Simulation
  - spezifisch für jede Regelungsanwendung
  - repräsentiert die Kosten für das System, um…
    - eine Aufgabe mit bestimmter Periode auszuführen bzw.
    - eine Schattenaufgabe auszuführen

- Schattenaufgaben (shadow task)
  - laufen mit maximaler Periode
  - laufen nie zusammen mit Basisaufgabe auf einem Prozessor
  - laufen nie zusammen mit anderen Schattenaufgaben der Basisaufgabe auf einem Prozessor
  - → schnelle Reaktion auf Überlast
  - → Gewährleistung der minimalen Regelgüte im Rekonfigurationsfall

Anpassung der Perioden – Beispiel: Prozessorausfall



Anpassung der Perioden – Reallokationsalgorithmen

- Reallokation für unabhängige Aufgaben (algorithm for independent tasks, AIT)
  - Rucksackproblem → NP-Vollständigkeit → Heuristischer Algorithmus
  - Rucksack: verfügbare, ungenutzte Gesamtrechenleistung (CPU 0 + CPU 1 + CPU 2 …)
  - Zielfunktion: Maximiere Gesamtnutzen
  - → System soll mit den wichtigsten Aufgaben optimal ausgelastet sein
  - Nebenbedingungen
    - maximale CPU-Auslastung 100 %
    - Prozessor Haupttask != Prozessor Schattentask
  - Lösung wird auf Prozessoren verteilt

- Reallokation für abhängige Aufgaben (algorithm for related tasks, ART)
  - analog AIT
  - zusätzliche Nebenbedingung
    - gemeinsame Anpassung der Perioden abhängiger Tasks

Anpassung der Perioden – Fazit

- Fazit
  - + keine Anpassung des Algorithmus erforderlich
  - + Berücksichtigung von Abhängigkeiten
  - – Ermittlung des PI durch Simulation fragwürdig
  - – Anwendbarkeit eingeschränkt
    - – Laufzeit
    - – Komplexitätsgrad des regelungstechnischen Entwurf
  - – Eignung nur für Multiprozessor-Systeme
  - – Komplexitätsgrad der Abhängigkeiten in der Praxis üblicherweise sehr hoch (Bremse)

Anpassung der Rechenzeit - Szenario

- Szenario
  - implizit dreigeteilte Regelaufgabe
    - Abtastung/Eingabe, Berechnung, Regelung/Ausgabe
  - Menge von Aufgaben unterschiedlicher Prioritäten und Perioden
  - unabhängige Aufgaben
  - Singlecore-System

- Rekonfigurationssituation
  - Wechsel des Betriebsmodus

Anpassung der Rechenzeit – Konzept (1)

- Zusteller
  - führt Regelaufgabe aus
  - bandweite-bewahrend: garantiert Ausführungsbudget
  - variable Bandweite, abhängig von Systemauslastung und Regelgüte

→ Zeitgewinn durch dynamische Anpassung der Bandweite

- Berechnungsmodell erlaubt…
  - variable aber **bekannte** Abtastungs- und Regelungszeitpunkte
  - Abschätzungen für den „schlimmsten Fall“
  - gelegentliche Terminverletzungen
  - gelegentliche Berechnungsausfälle

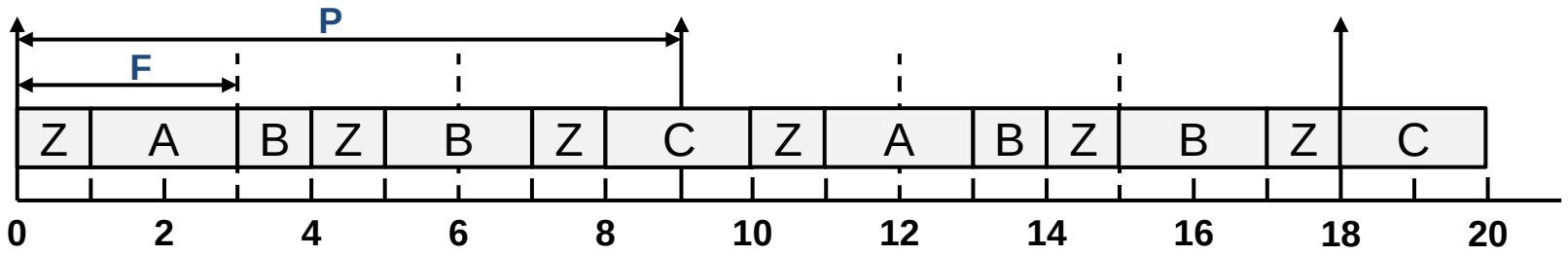
Exkurs: Zusteller

- Zusteller
  - spezielle periodische Aufgabe
  - erlaubt Ausführung nicht-periodischer Aufgaben im eigenen Kontext
  - → vereinfacht Planbarkeitstest
  - → einzige Möglichkeit nicht-periodische Aufgaben in taktgesteuerten Systemen abzuarbeiten
- Auffüllperiode F
  - Erneuerungszeitpunkt für Ausführungsbudget
- Ausführungsbudget Q
  - Ausführungszeit in einer Auffüllperiode
- Bandweite B
  - $B = \frac{Q}{F} = \frac{Ausführungsbudget}{Auffüllperiode}$

Exkurs: Zusteller

- Varianten
  - abfragender Zusteller:
    verwirft Ausführungsbudget für den Fall, dass
    Auslösezeitpunkt n.-p. Aufgabe > Auslösezeitpunkt Zusteller
  - bandweite-bewahrender Zusteller:
    garantiert Ausführungsbudget unabhängig vom Auslösezeitpunkt

- Beispiel: bandweite-bewahrender Zusteller Z
  - R = 0; C = 3; P = 9; Q = 1; F = 3



Anpassung der Rechenzeit – Berechnungsmodell

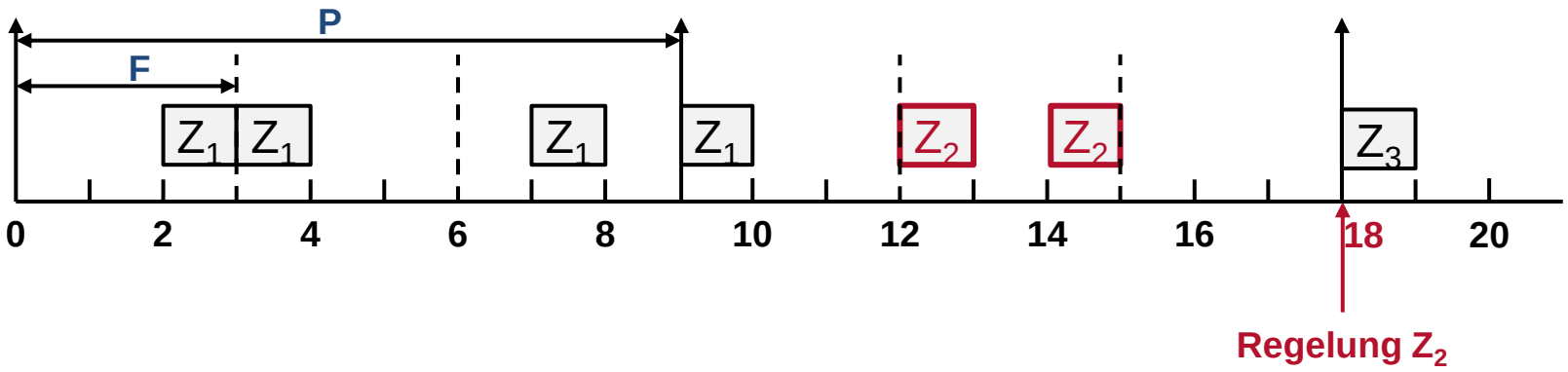
- Annahmen
  - T = R + P (Termin entspricht dem Periodenende)
  - P = n * F (Auffüllperiode ist ganzzahliger Teiler der Periode)

- Berechnungsmodell
  - **Fall 1:** Regelaufgabe endet **vor** ihrem Termin
    Folge: Regelung erfolgt zum Termin
  - **Fall 2:** Regelaufgabe endet **nach** ihrem Termin
    Folge: Regelung erfolgt am Ende der aktuellen Auffüllperiode
  - **Fall 3:** Regelaufgabe endet **nach** ihrem Termin und überschreitet Grenzwert (Grenzwert: 1x Periode)
    Folge: aktuelle Regelaufgabe wird abgebrochen
  - **Fall 4:** Abtastung erfolgt **immer** zusammen mit Regelung der vorhergehenden Aufgabe

Anpassung der Rechenzeit – Beispiel (1)

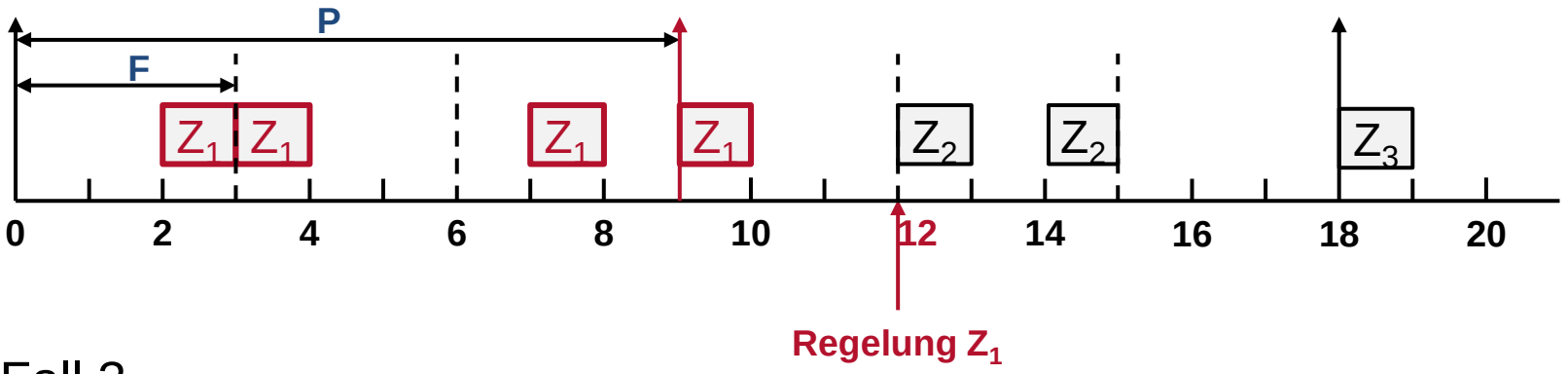
- Beispiel
  - R = 0, P = 9, F = 3, C/Q = variabel
  - Zusteller Z_k mit k gleich der k-ten Ausführung

- Fall 1
  - Regelaufgabe endet **vor** ihrem Termin
  - **Folge:** Regelung erfolgt zum Termin
  - **Ziel:** **Vermeidung von Schwankungen bei der Regelung/Ausgabe**



Anpassung der Rechenzeit – Beispiel (2)

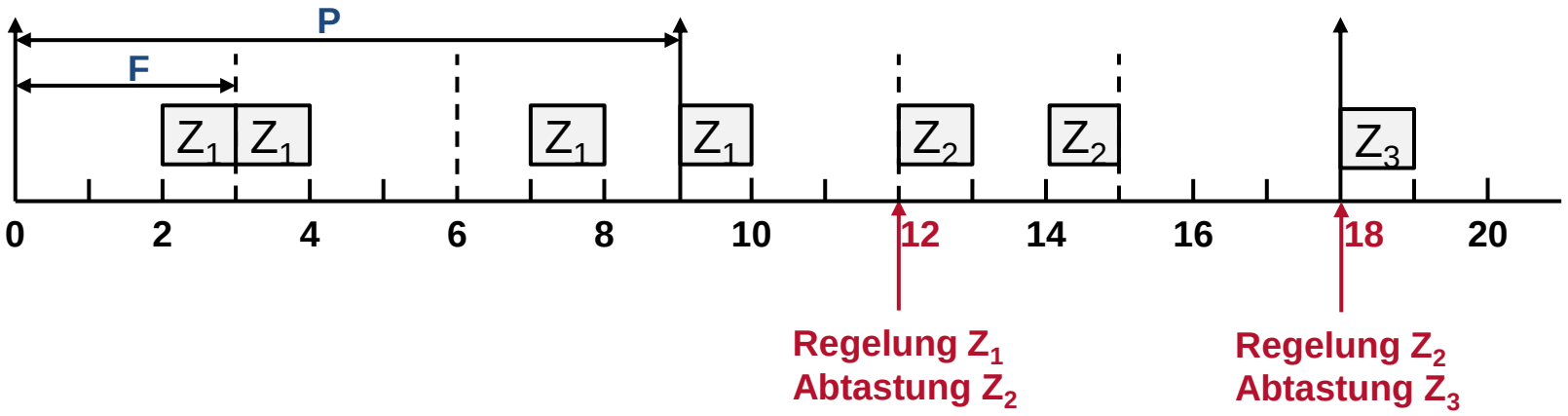
- Fall 2
  - Regelaufgabe endet **nach** ihrem Termin
  - **Folge:** Regelung erfolgt am Ende der aktuellen Auffüllperiode
  - **Ziel:** **Vermeidung variabler Zeitpunkte für Regelung/Ausgabe**



- Fall 3
  - Regelaufgabe endet **nach** ihrem Termin
    und überschreitet Grenzwert (Grenzwert: 1x Periode)
  - **Folge:** aktuelle Regelaufgabe wird abgebrochen
  - **Ziel:** **Vermeidung von Verzögerungen der Regelung/Ausgabe unbekannter Länge**

Anpassung der Rechenzeit – Beispiel (3)

- Fall 4
  - Abtastung erfolgt **immer** zusammen mit Regelung der vorhergehenden Aufgabe
  - **Ziele**
    - **Vermeidung variabler Zeitpunkte für Abtastung**
    - **Aktualisierung der Eingabedaten**



Anpassung der Rechenzeit – Fazit

- Fazit
  - \+ flexibles Modell im Vergleich zu „one-shot“
  - \+ Abschätzbarkeit des schlimmsten Falls (Terminverletzung / Berechnungsausfall)
  - \+ Prävention von Verzögerungen/Schwankungen unbekannter Dauer
  - \+ Auffrischung der Eingabedaten
  - – keine Berücksichtigung von Abhängigkeiten
  - – keine Berücksichtigung der Allokation von Betriebsmitteln (Ressourcenkonflikte?)
  - – keine Berücksichtigung von Abtastung/Regelung (nicht modelliert)
  - – Ermittlung des Referenzmaßes unklar

Danke für eure Aufmerksamkeit!