
Multicore in Echtzeitsystemen (1)

Liyuan Zhang

Hauptseminar AKSS im SS 2009

Ausgewählte Kapitel der Systemsoftware: Multicore- und Manycore-Systeme



Überblick

- Einleitung
- Uniprozessor in Echtzeitsystemen
- Multiprozessor in Echtzeitsystemen
- Ein komplettes Fallbeispiel
- Fazit



Einleitung - Echtzeitsysteme

- *Echtzeitbetrieb ist ein Betrieb eines Rechensystems, bei dem Programme zur Verarbeitung anfallender Daten ständig betriebsbereit sind derart, dass die Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sind.*
 - *DIN 44300*
- *Geschwindigkeit liefert keine Garantie, um rechtzeitig Ergebnisse abzuliefern.*
- *Wichtige Maßnahme : Gültige und effektive Ablaufplanung.*



Uniprozessor in Echtzeitsystemen

- Gebräuchliche Schedulingverfahren
 - Taktgesteuerte Verfahren
 - Vorranggesteuerte Verfahren
 - Reihum gewichtete Verfahren
- Zugriffskontrolle
 - Unkontrollierte Prioritätenumkehr <-> Prioritätsvererbung
 - Verklemmung <-> Prioritätsobergrenze



Uniprozessor in Echtzeitsystemen

- **Dynamische Prioritäten**
 - EDF (engl. earliest deadline first)
 - Je früher der Termin, desto höher die Priorität
 - LRT (engl. latest release-time first)
 - Je später die Auslösezeit, desto höher die Priorität
 - LST (engl. least slack-time first)
 - Je kürzer die Schlupfzeit, desto höher die Priorität
- **Statische Prioritäten**
 - DM (engl. deadline monotonic)
 - Je kürzer der relative Termin, desto höher die Priorität
 - RM (engl. rate monotonic)
 - Je kürzer die Periode, desto höher die Priorität



Multiprozessor in Echtzeitsystemen

- **Aufgaben**
 - Partitionieren und Zuordnungen von Aufgaben
- **Prozessoren**
 - Interprozessor Synchronisation
- **Betriebsmittel**
 - Betriebsmittelvergabe zwischen Prozessoren
- **Scheduling**
 - Ablaufplanung der Unteraufgaben auf jeden Prozessor
 - Multiprocessor Priority-Ceiling Protocol, MPCP
 - Ablaufplanung für End-to-End Aufgaben Modell



Ein komplettes Fallbeispiel

- **Problemstellung:** Erstellen eine gültige Ablaufplanung in einem abstrakten Multiprozessorsystem mit festgelegter Anzahl von Aufgaben.



- Systembeschreibung

- Quad Core Prozessor
- Betriebsmittel (Speicher, Geräte, Puffer...), S_1 , S_2 , S_3
- Feste oder dynamische Prioritäten
- Geschachtelter Betriebsmittelzugriff
- Max. zwei zugeteilte Aufgaben per Prozessor



Aufgabe

- **Aufgabenbeschreibung**

- Fünf periodische Aufgaben/Tasks T_i mit unterschiedlichen Perioden und Ausführungszeiten.
- $T_i := \{ J_{i,j} \}$, Aufgabe := { Arbeitsaufträge/Jobs }
- J_i [Ausführungszeit, Periode(auch Deadline)]
- J_1 [6,10], J_2 [7.5,12], J_3 [8,15], J_4 [8.5,16], J_5 [10,20]



Scheduling

- Anforderungen

- Gültige Ablaufplanung für alle Aufgaben, nämlich alle Deadlines einhalten
- Minimal nötige Prozessoren
- Begrenzte Blockierungszeit für alle Aufgaben



Entwicklungsphase



1. Prioritätenvergabe
2. Aufgabenzuordnung
3. Anwendung des MPCPs
4. Resultatanalyse



- Statische Prioritäten

- **RM** (engl. rate monotonic), je kürzer die Periode, desto höher die Priorität
- Ausführungszeiten von Arbeitsträgen vor Laufzeit bekannt
- J_1, J_2, J_3, J_4, J_5 nach Perioden
- $P(J_1) > P(J_2) > P(J_3) > P(J_4) > P(J_5)$
- Priorität von J_1 gleich 1, Priorität J_2 von gleich 2, ...
- $J_1(S_1), J_2(S_1), J_3(S_1, S_2), J_4(S_1, S_2, S_3), J_5(S_1, S_3)$
- J_4 besitzt geschachtelten Betriebsmittelzugriff zu S_2 und S_3



Entwicklungsphase

1. Prioritätenvergabe
2. Aufgabenzuordnung
3. Anwendung des MPCPs
4. Resultatanalyse



Aufgabenzuordnung

- Anforderungen

- Gemeinsame Daten zwischen Aufgaben
- Kommunikation zwischen Prozessoren
- Kommunikation zwischen Prozessoren und Betriebsmittel
- „Overhead“ Kosten für Kontextwechsel von Jobs
- Und viel mehr Faktoren
- NP-Hartes Problem

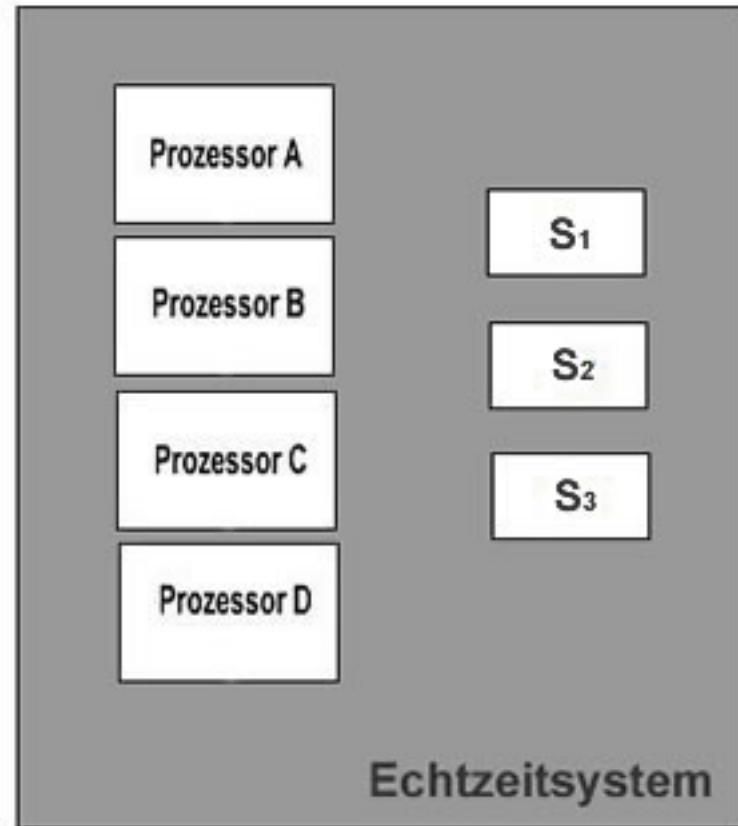
- Entwicklungsmethode

- Graphentheorie, Netzwerk-Partitionieren
 - System als Netzwerk
 - Aufgaben und Prozessoren als Knoten
 - Interprozessor Kommunikation als Kanten, zwischen Prozessoren und Prozessoren
 - Ausführungszeit als Kanten, zwischen Aufgaben und Prozessoren



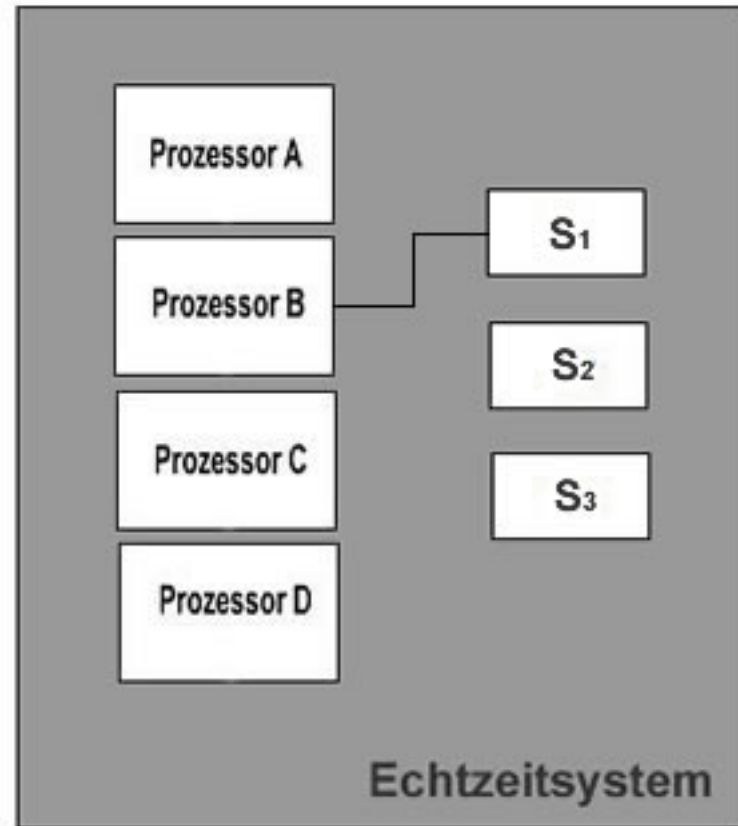
Aufgabenzuordnung (Forts.)

T ₁	S ₁
T ₂	S ₁
T ₃	S _{1,2}
T ₄	S _{1,2,3}
T ₅	S _{1,3}

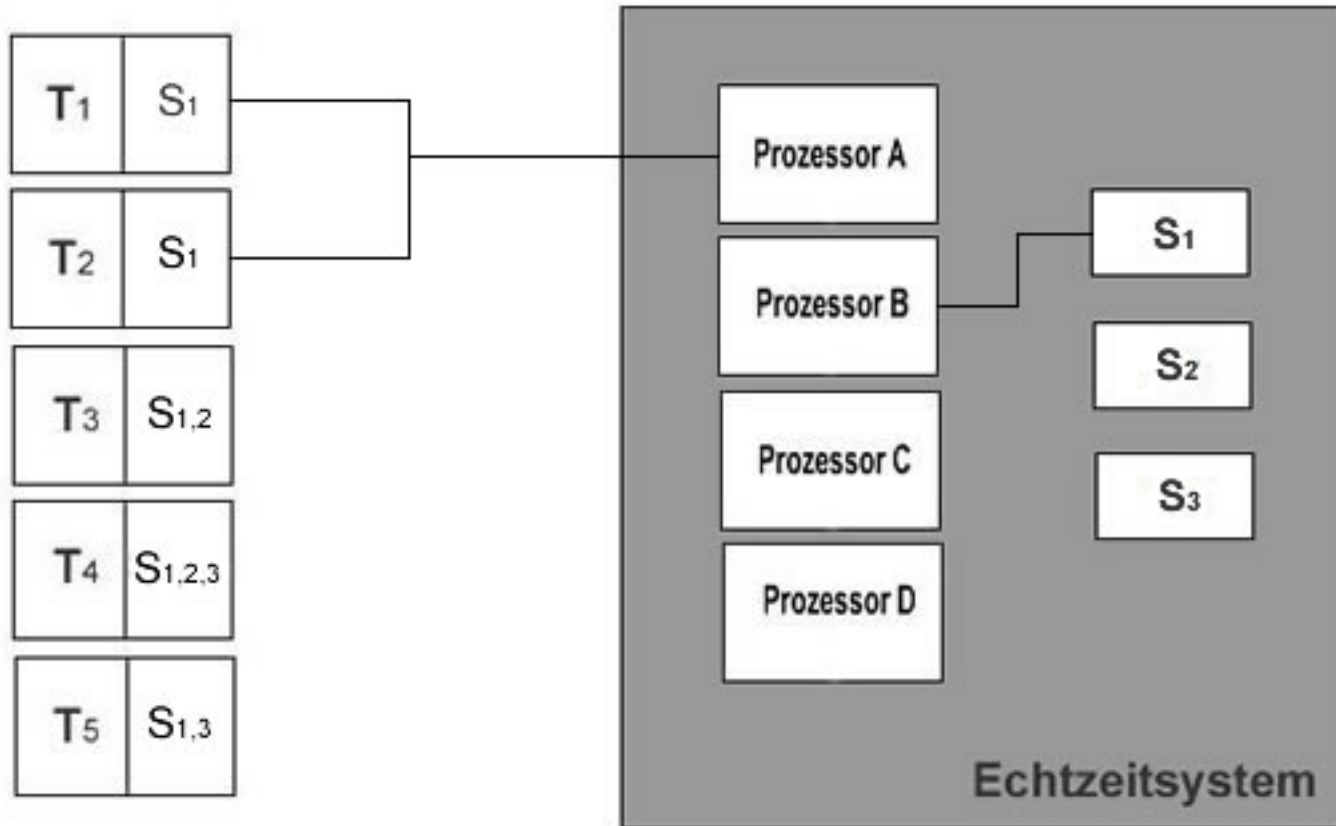


Aufgabenzuordnung (Forts.)

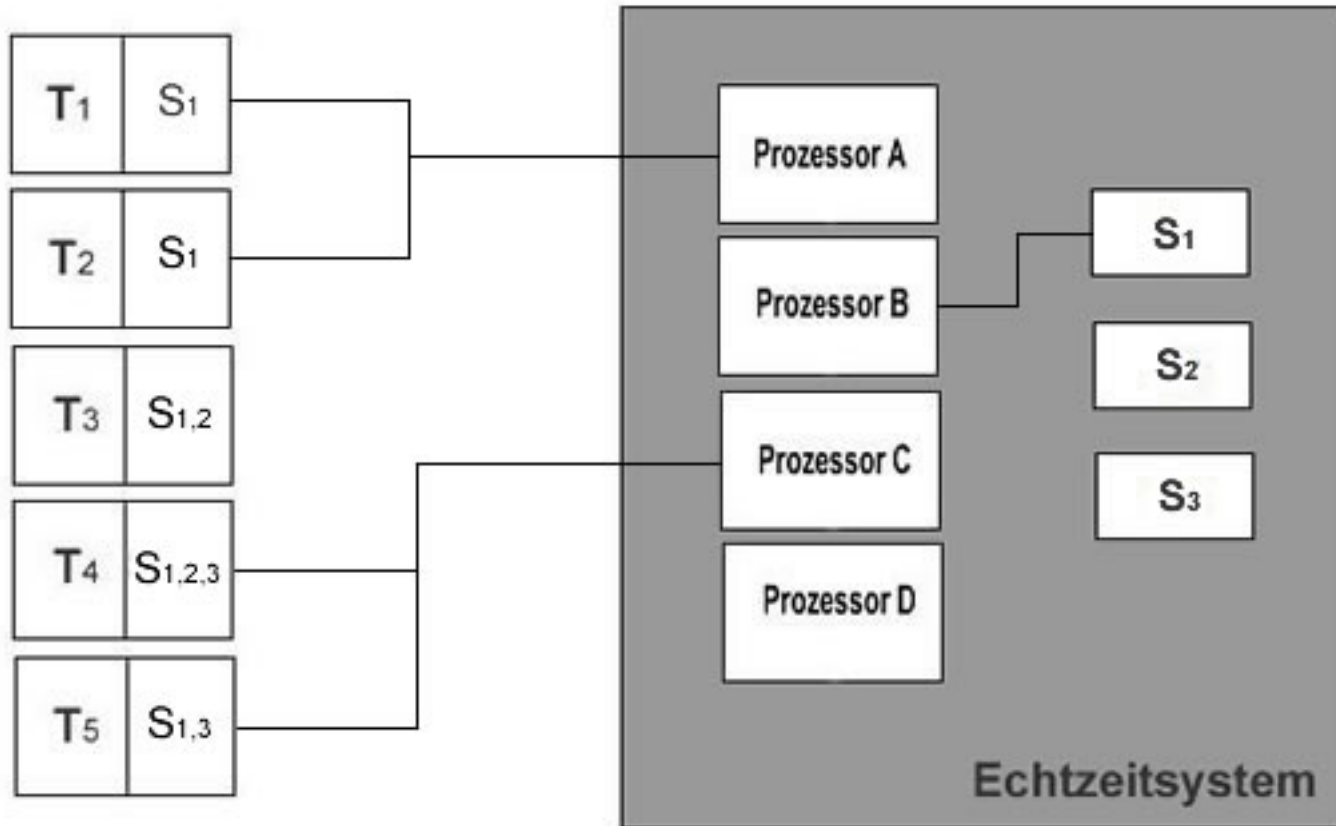
T ₁	S ₁
T ₂	S ₁
T ₃	S _{1,2}
T ₄	S _{1,2,3}
T ₅	S _{1,3}



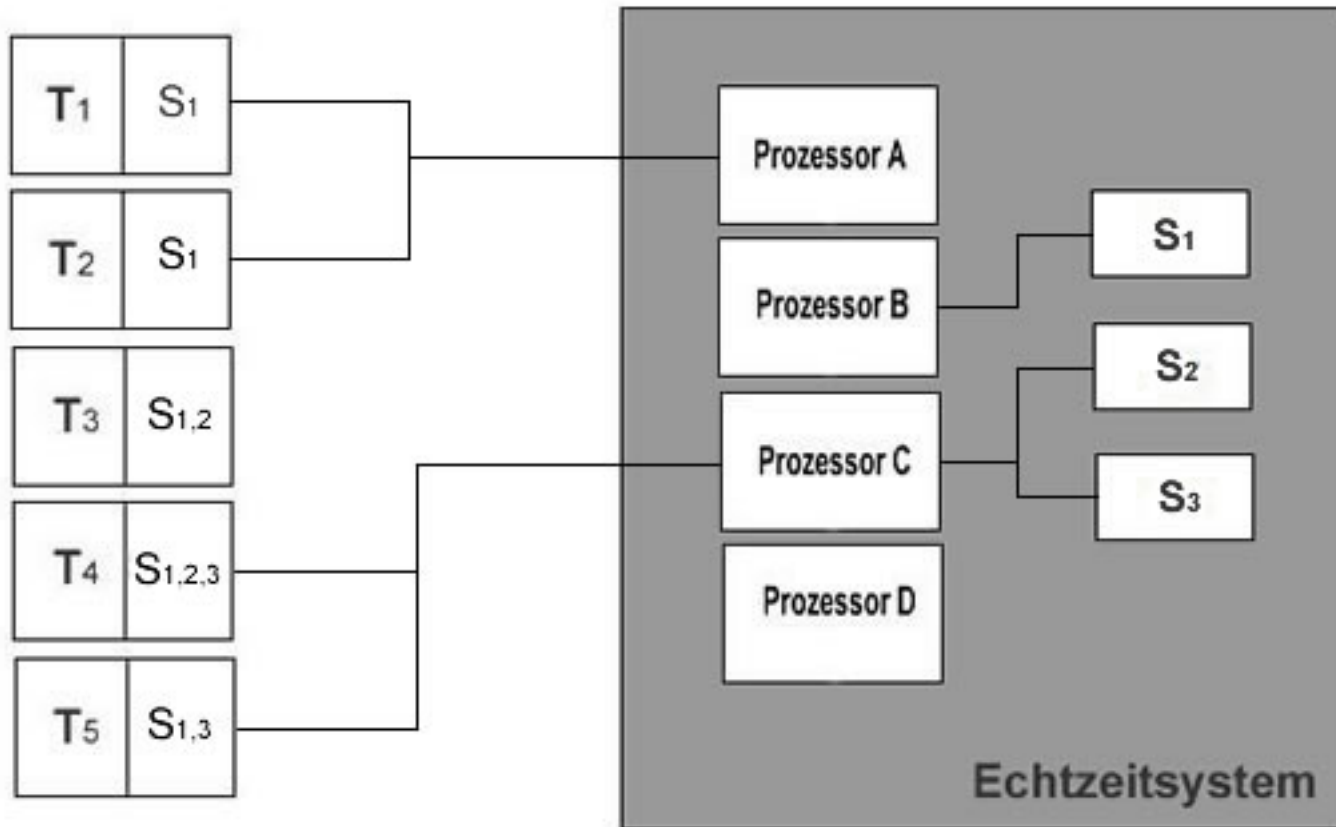
Aufgabenzuordnung (Forts.)



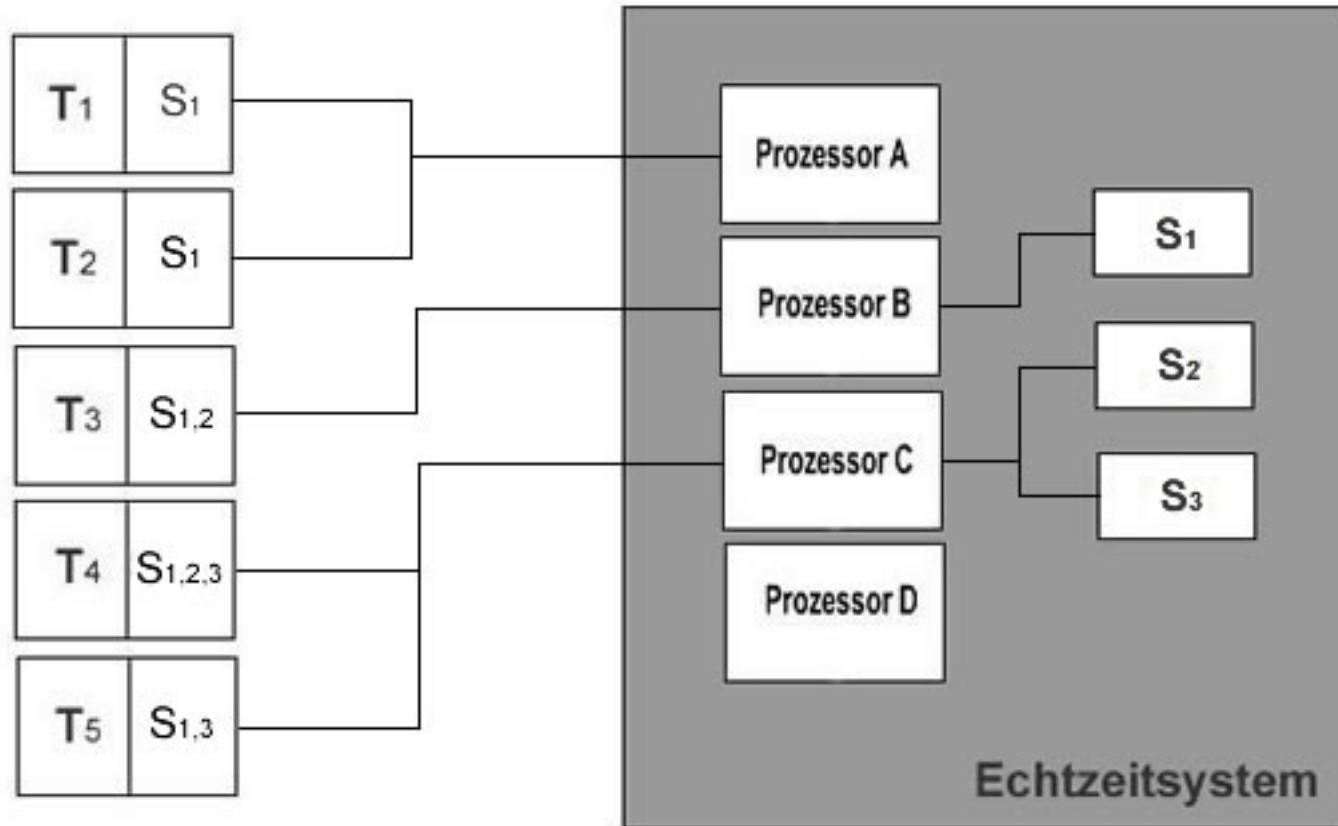
Aufgabenzuordnung (Forts.)



Aufgabenzuordnung (Forts.)



Aufgabenzuordnung (Forts.)



Entwicklungsphase

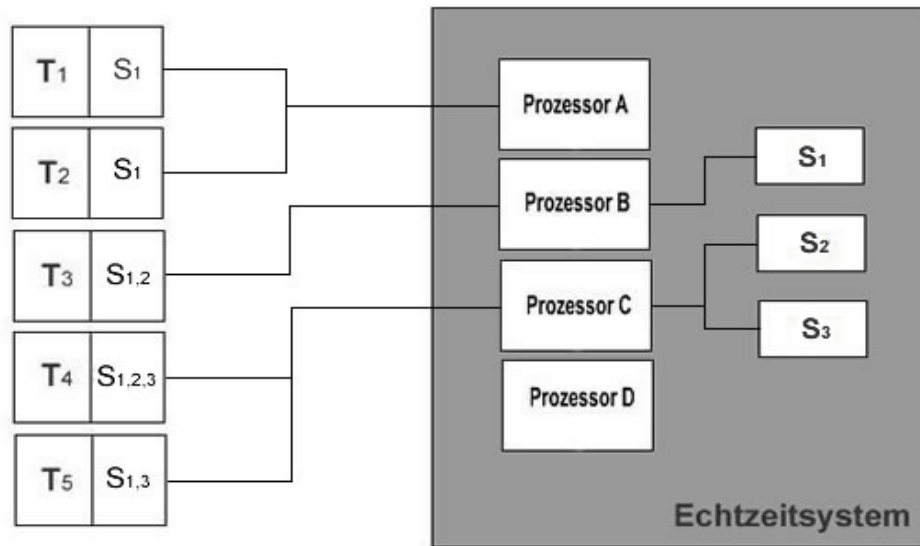
1. Prioritätenvergabe
2. Aufgabenzuordnung
3. Anwendung des MPCPs
4. Resultatanalyse



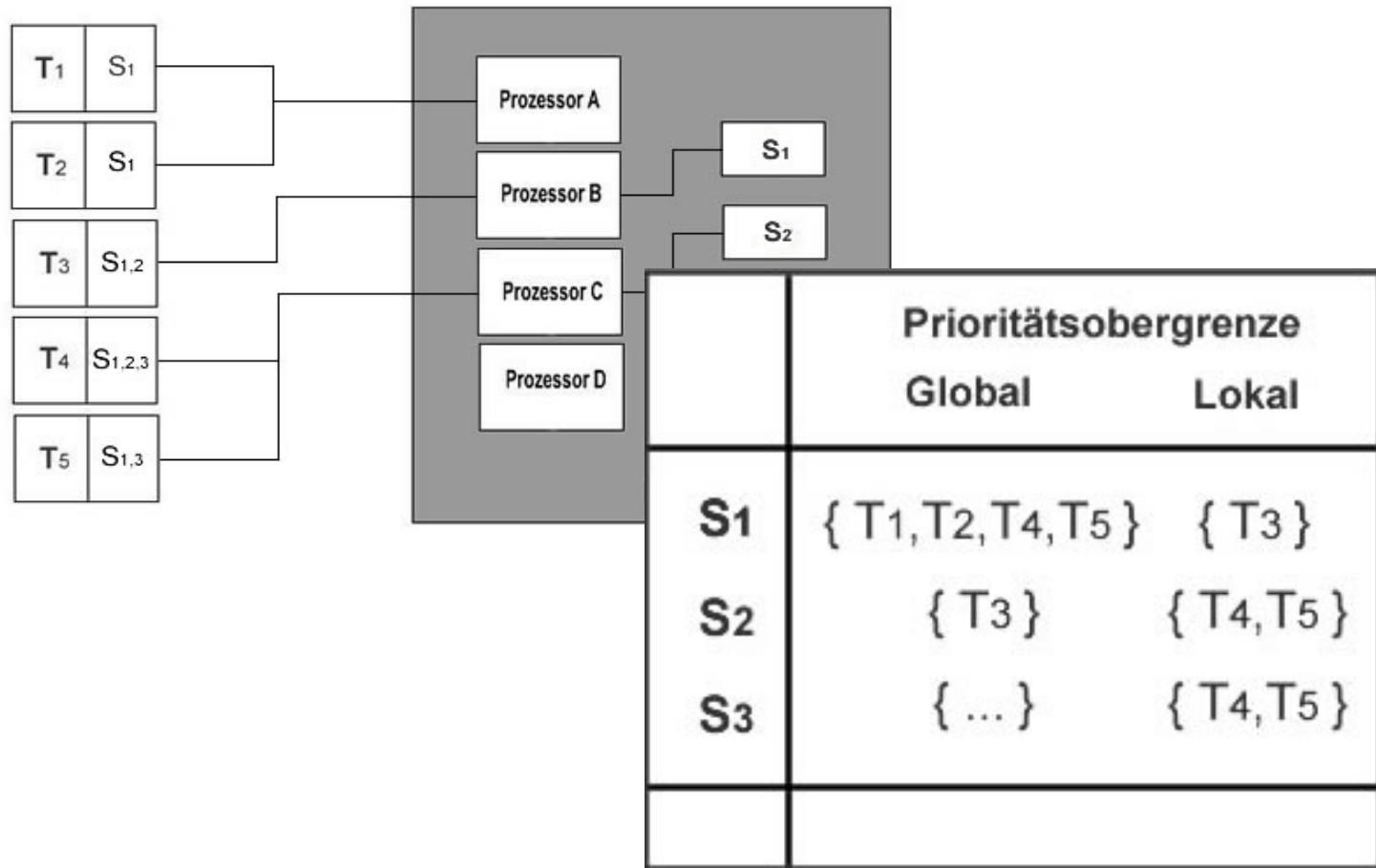
- Multiprocessor Priority-Ceiling Protocol
 - Erweiterung der Prioritätsobergrenzen
 - Lokale und globale Prioritätsobergrenze
 - Vergabe der Prioritätsobergrenzen
 - $P_{i,L}(S_k) := \mathbf{MAX}\{ P(J_i) \}$
 - $P_{i,G}(S_k) := \mathbf{MAX}\{ P(J_i) - P_{\text{Grenze}} \}$



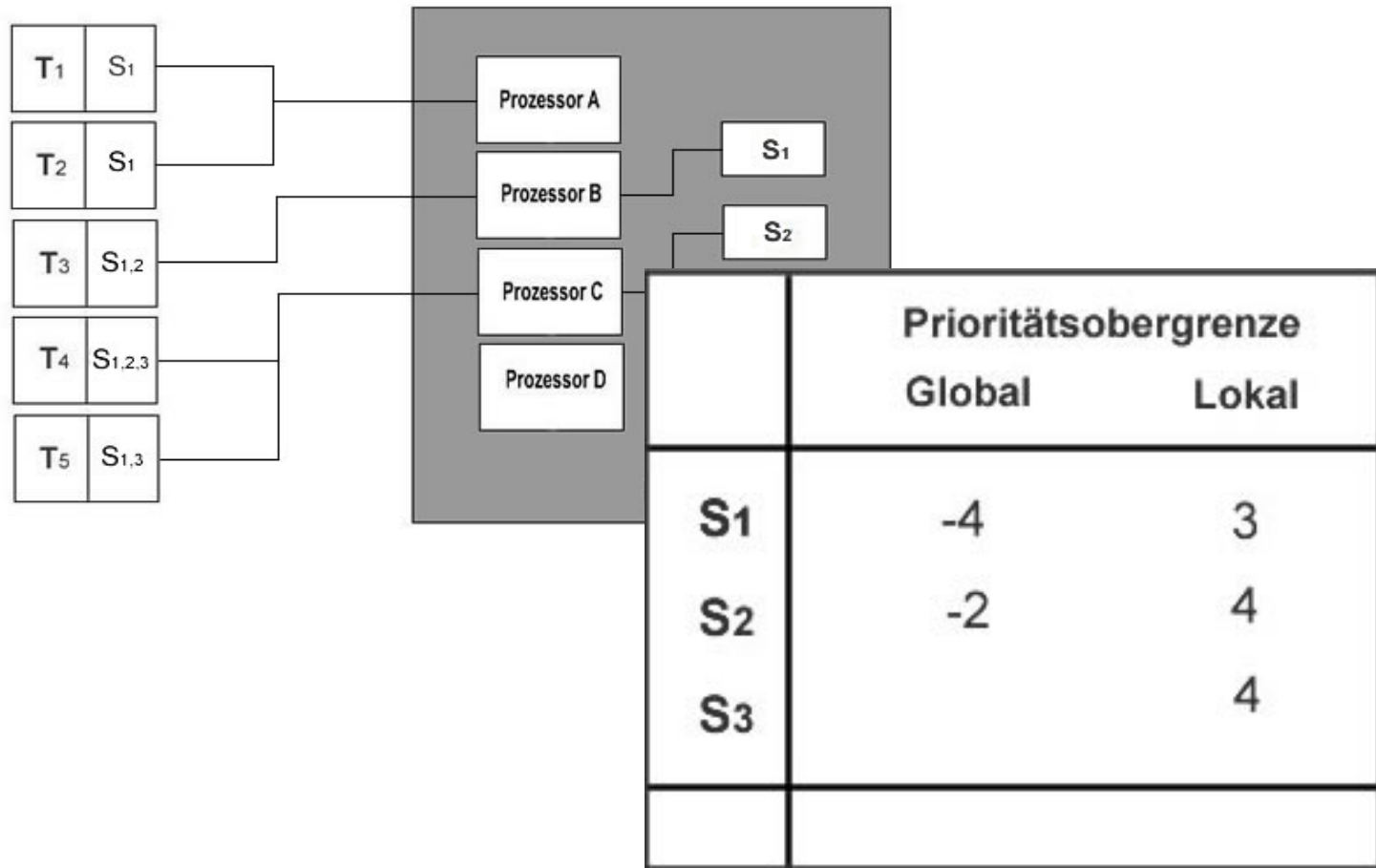
MPCP (Forts.)



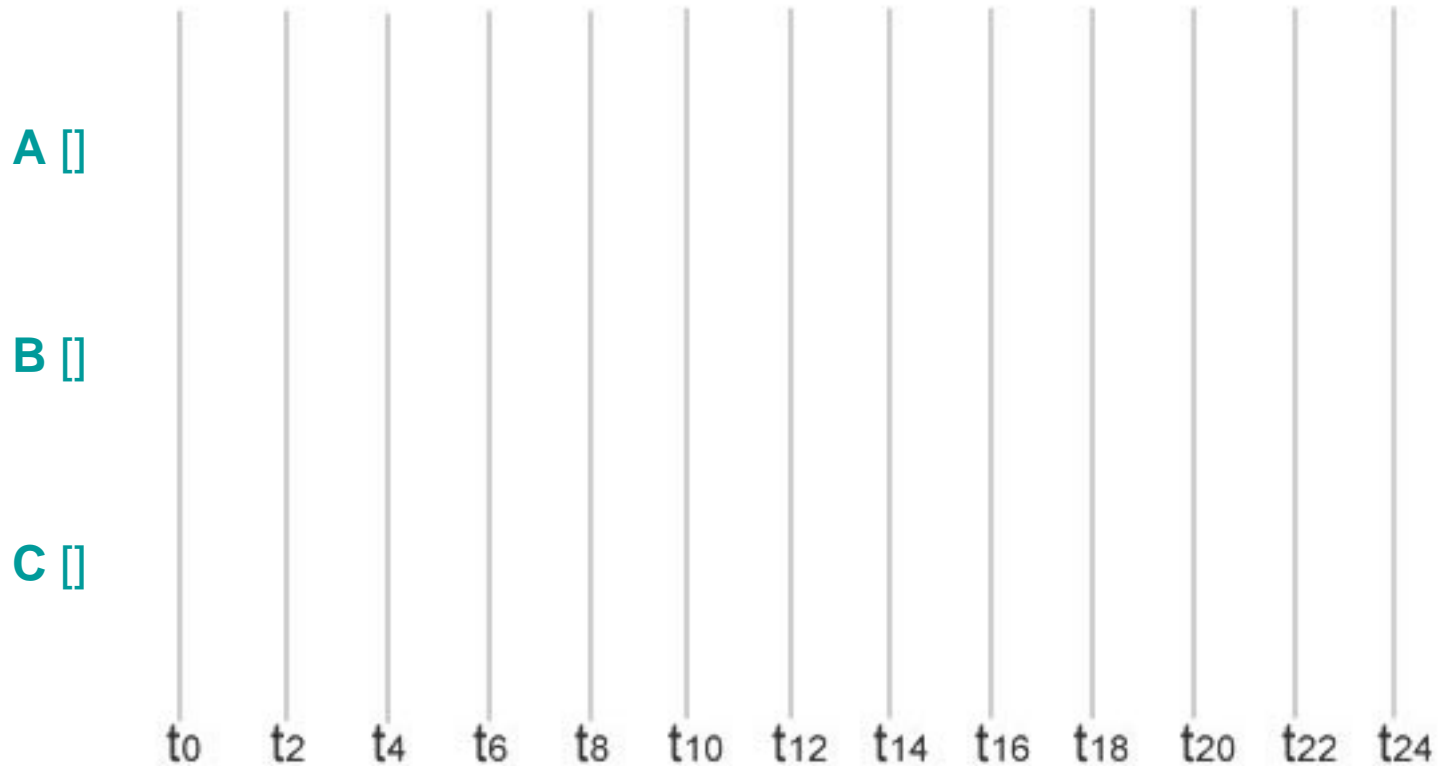
MPCP (Forts.)



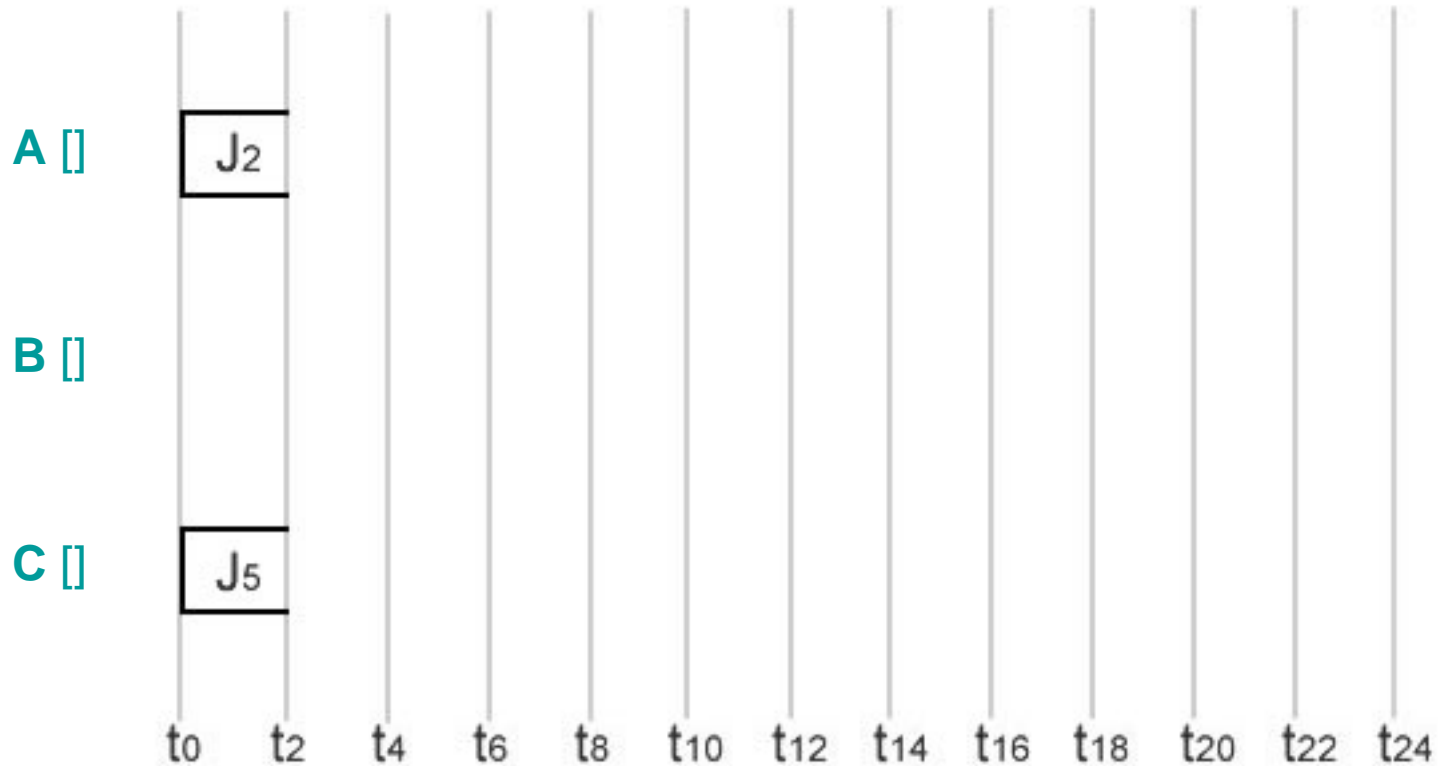
MPCP (Forts.)



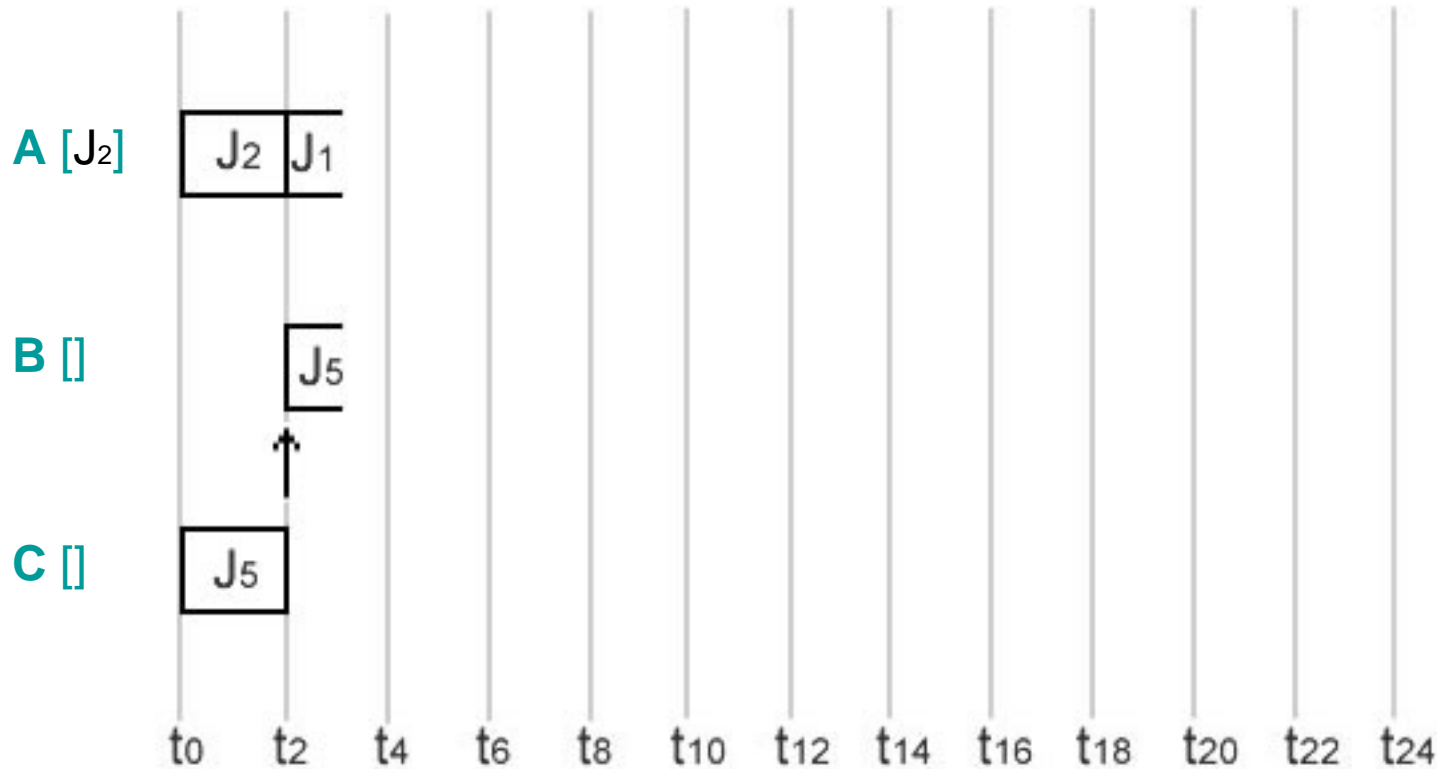
MPCP (Forts.)



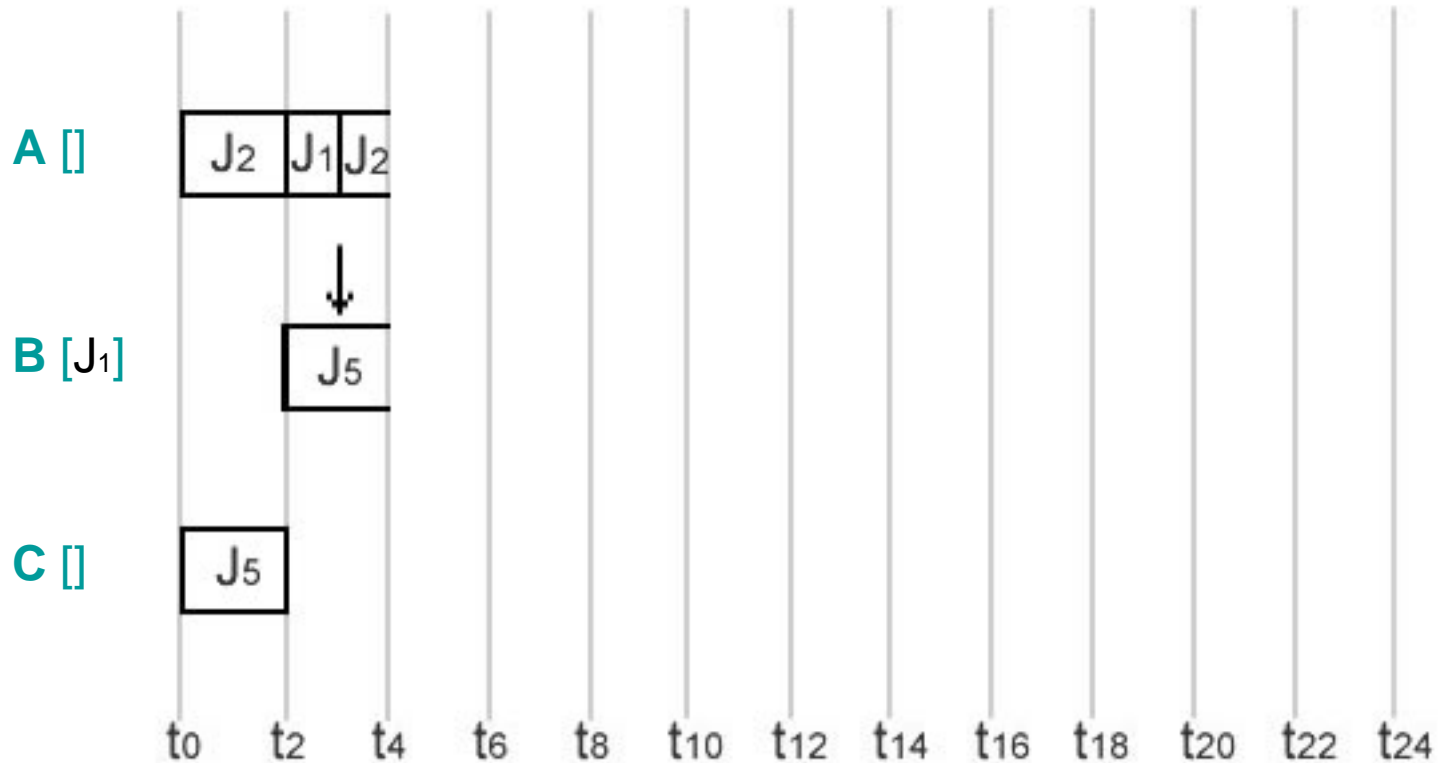
MPCP (Forts.)



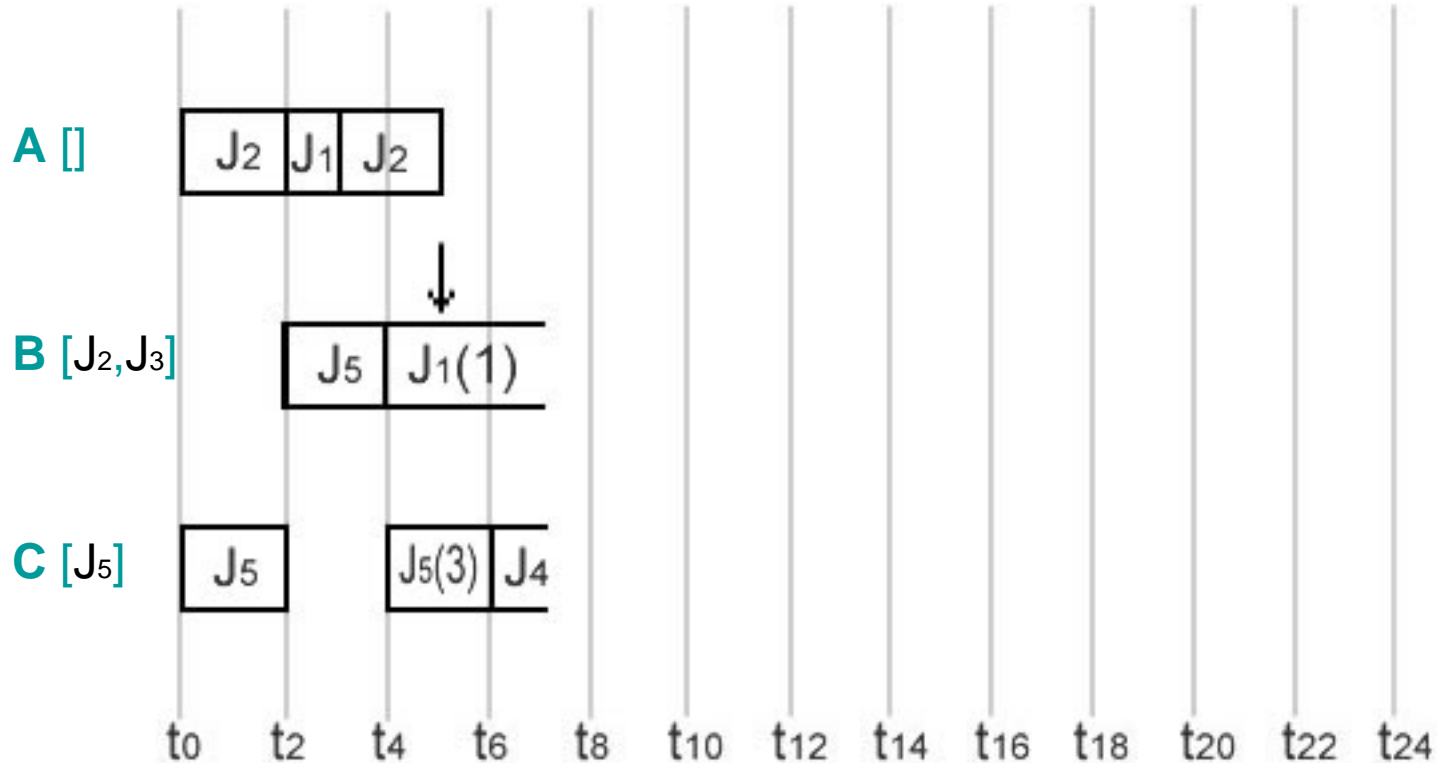
MPCP (Forts.)



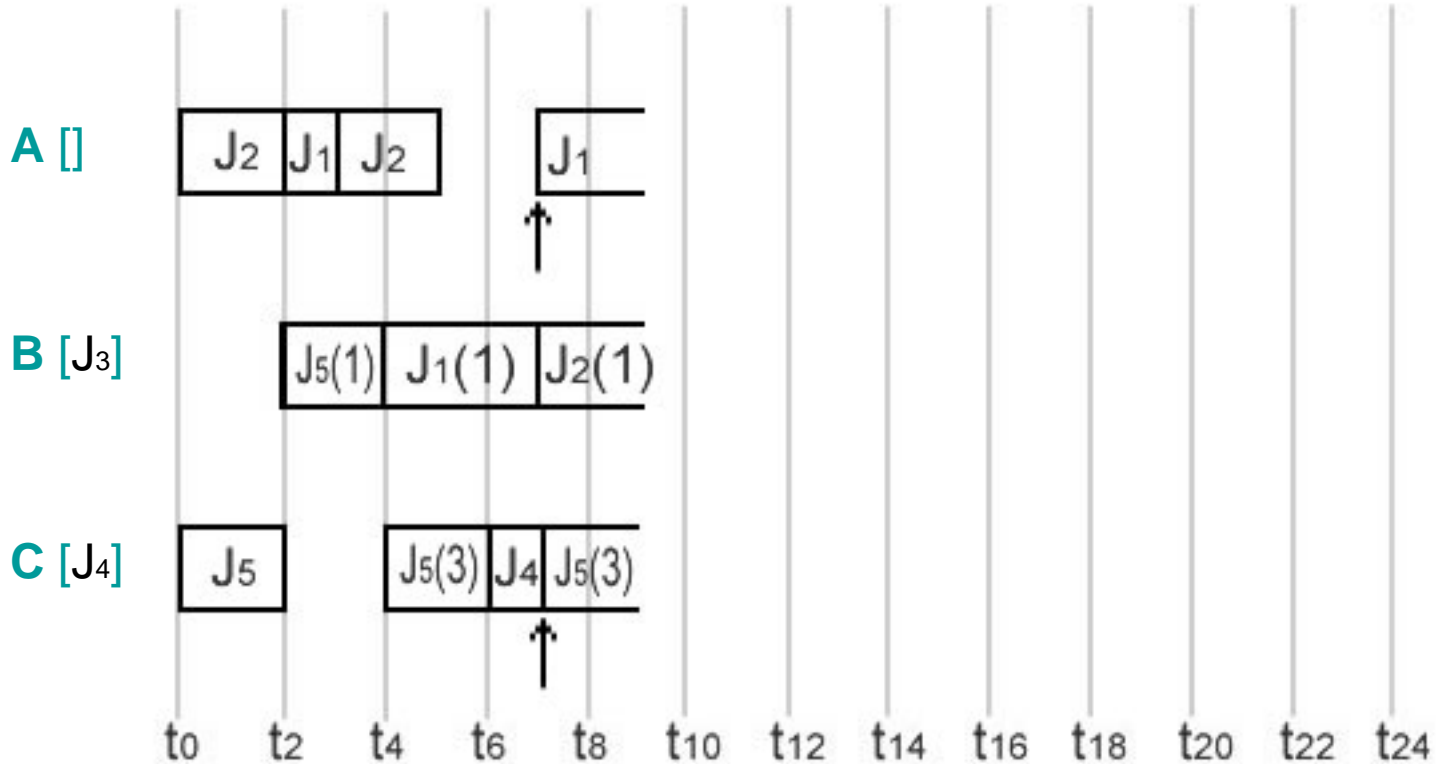
MPCP (Forts.)



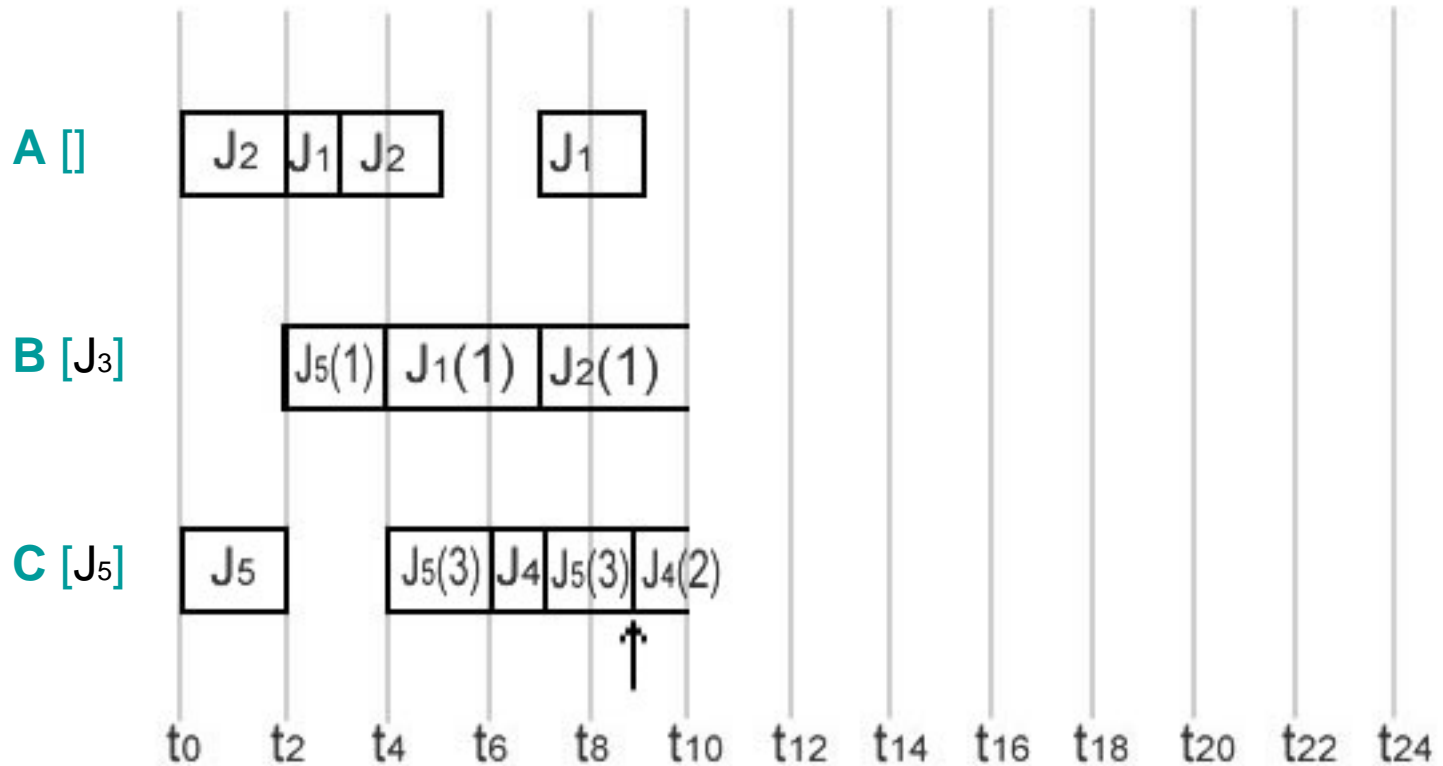
MPCP (Forts.)



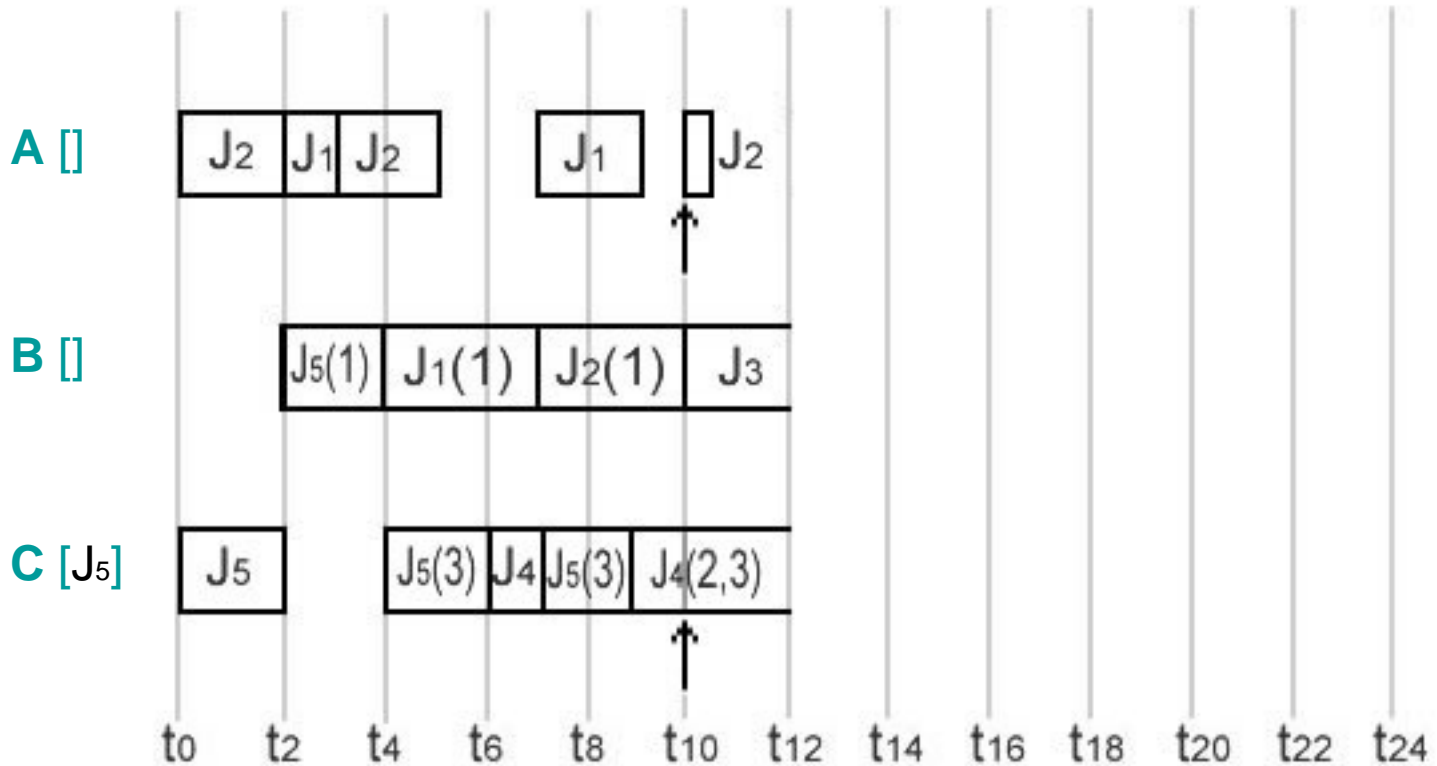
MPCP (Forts.)



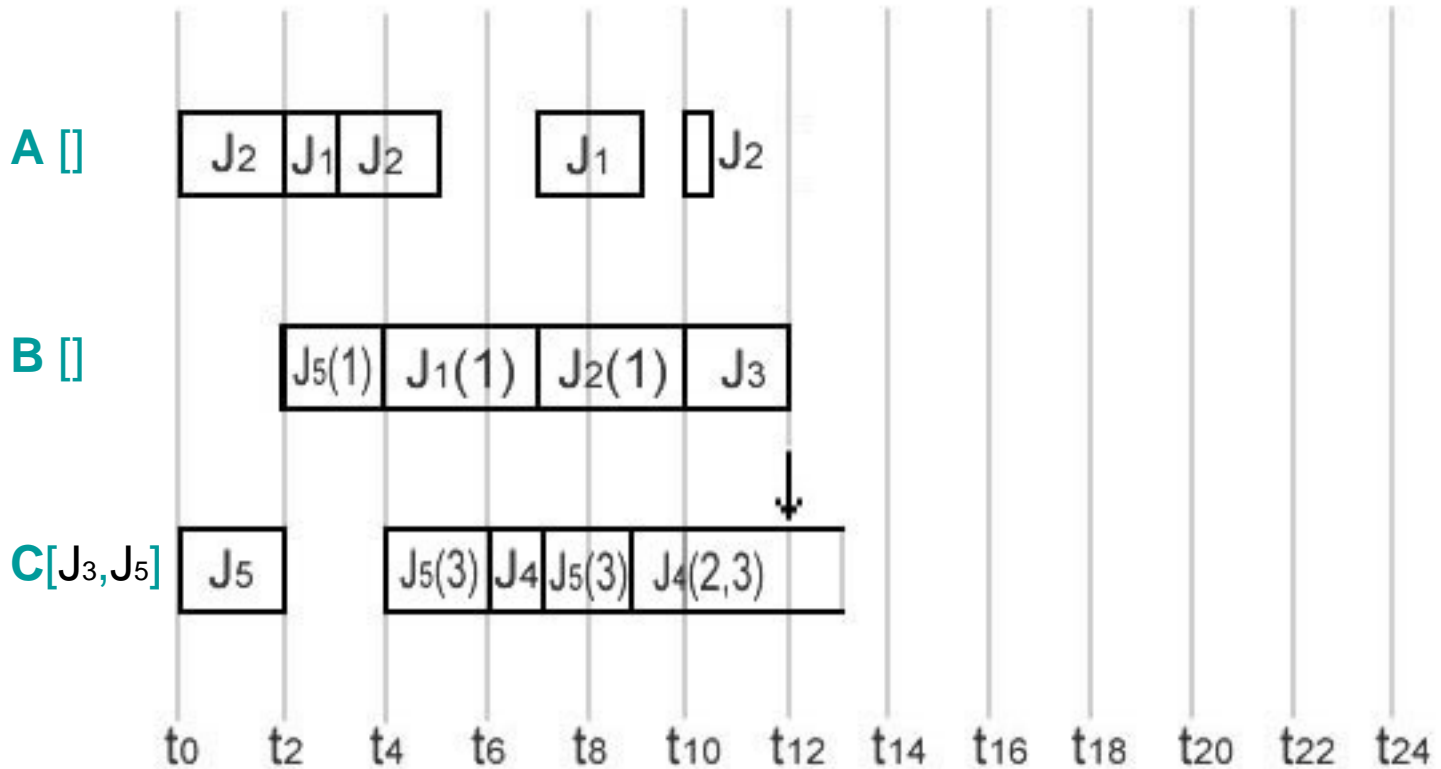
MPCP (Forts.)



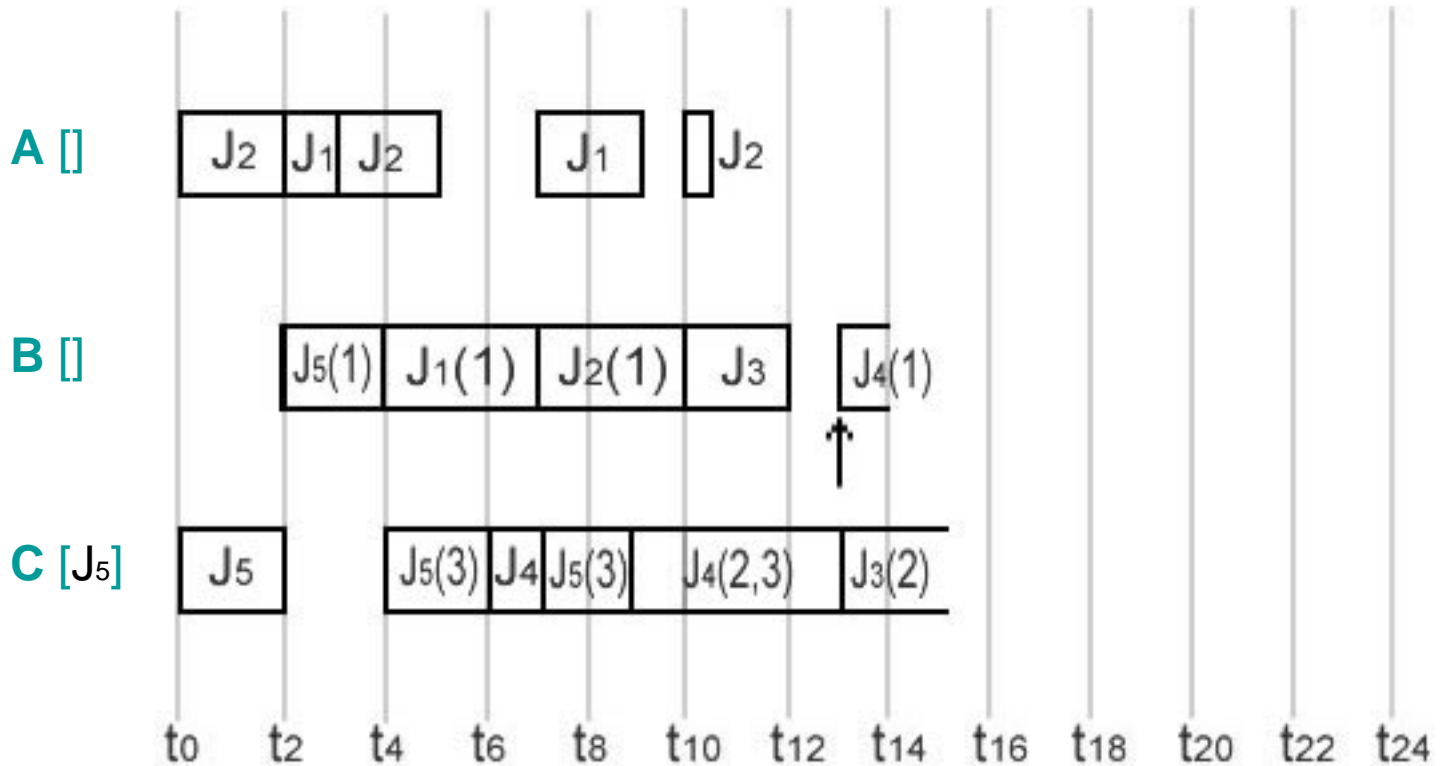
MPCP (Forts.)



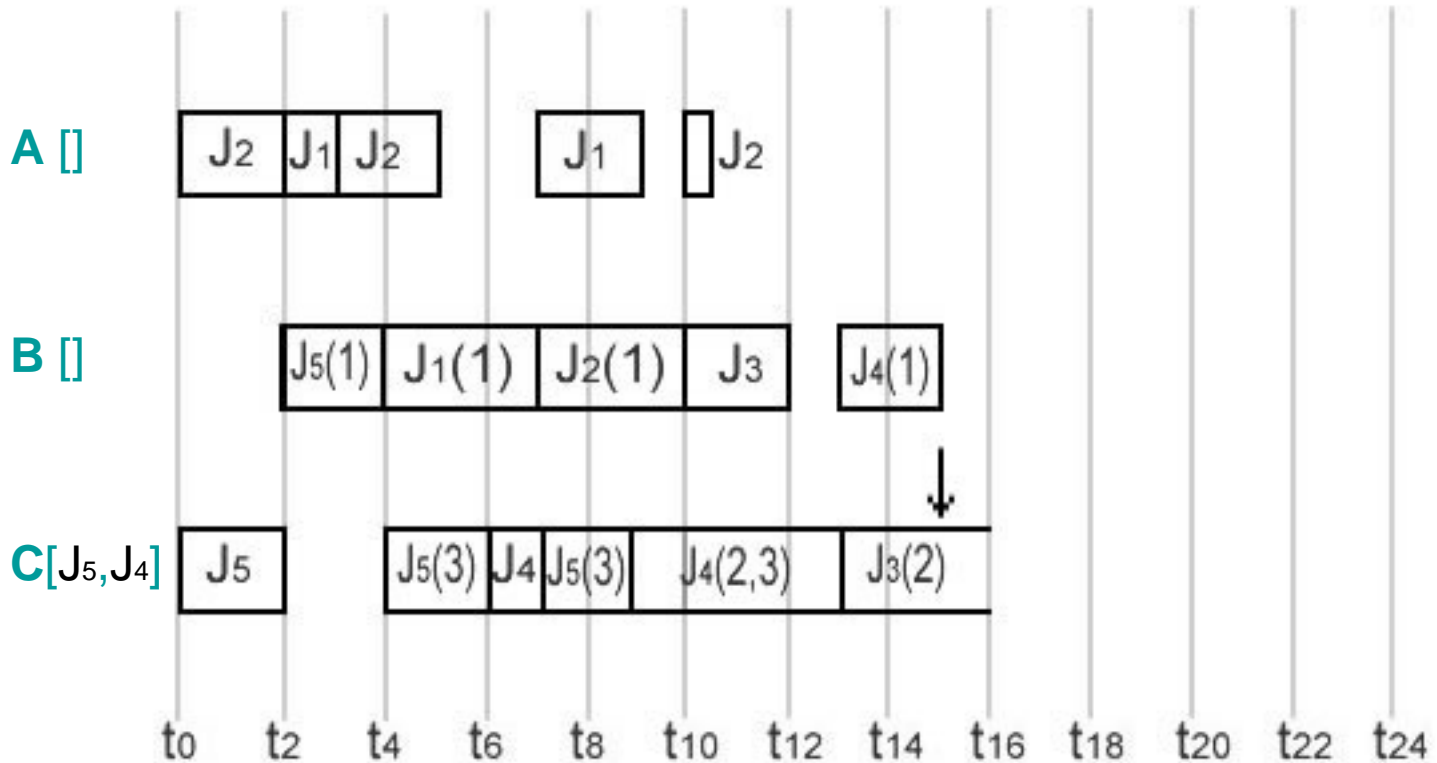
MPCP (Forts.)



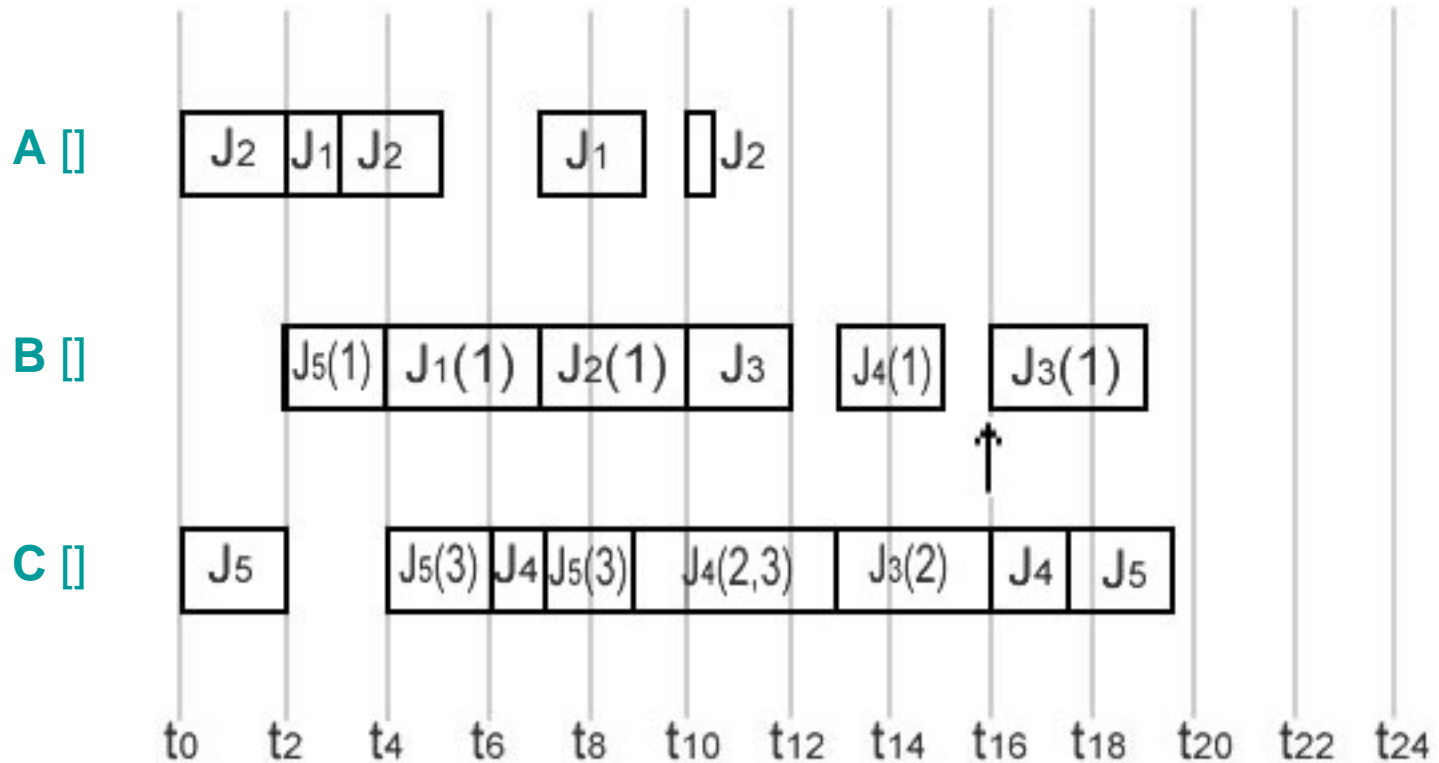
MPCP (Forts.)



MPCP (Forts.)



MPCP (Forts.)



Entwicklungsphase

1. Prioritätenvergabe
2. Aufgabenzuordnung
3. Anwendung des MPCPs
4. Resultatanalyse



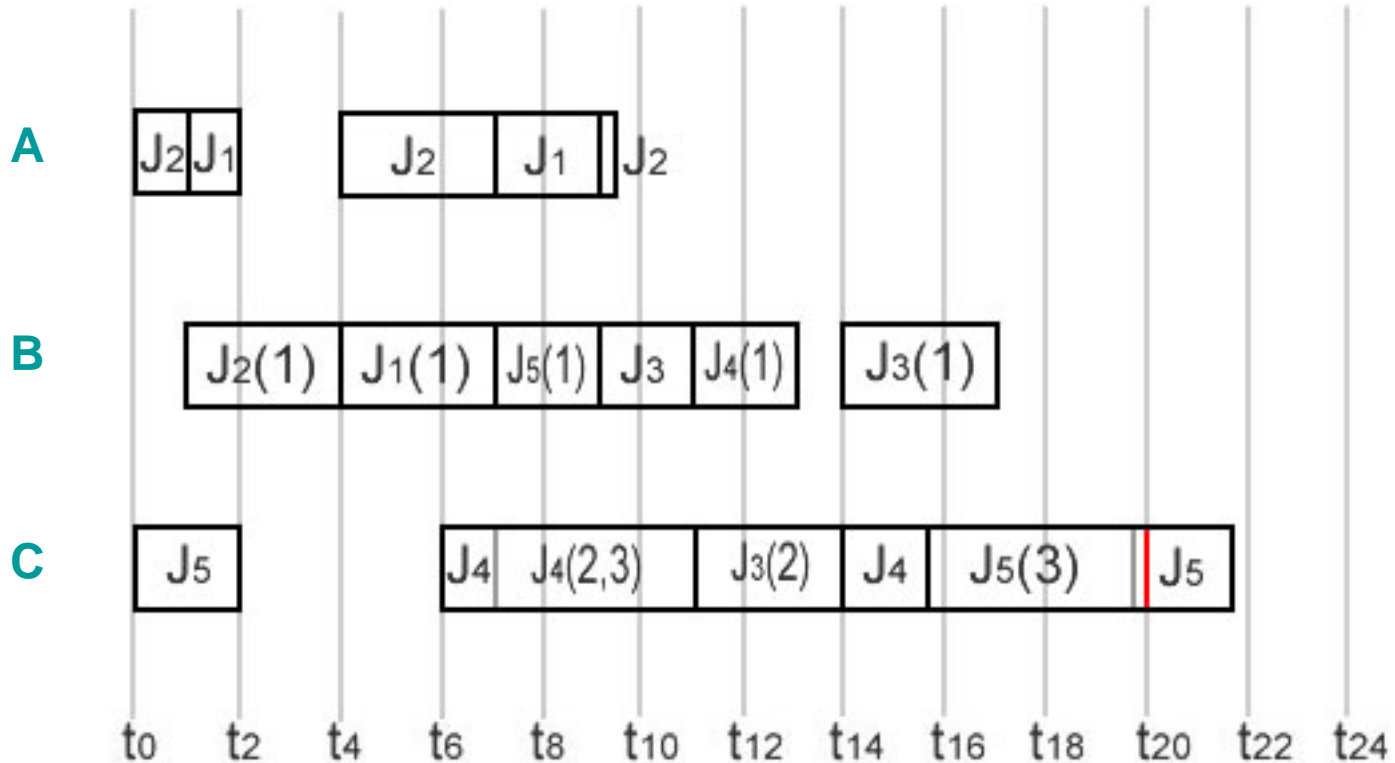
Resultatanalyse

- **Totale Blockierungszeit B_i**
 - Lokale Blockierungszeit
 - Lokale Verdrängungsverzögerung
 - Ferne Blockierungszeit
 - Ferne Verdrängungsverzögerung
 - Aufschiebbare Blockierungszeit

- **Gültigkeit und Optimalität**
 - „First-Fit“ in der Praxis
 - Nicht mehr gültig durch kleine Änderungen
 - Auslösezeit von J_1 ist nun T_1 statt T_2
 - J_2 verlangt S_1 zum Zeitpunkt T_1 statt T_5



Resultatanalyse



Resultatanalyse

- Mögliche Maßnahme
 - Ablaufplanung für End-to-End Aufgaben Modell statt MPCP
 - Dynamische Prioritäten, z. B. **LRT** statt **RM**
 - Probleme wieder auftauchen, z. B. unkontrollierbare Prioritätsumkehr



Fazit

- Ablaufplanung in Multiprozessorsystemen
 - Anforderungen
 - Algorithmen
 - Probleme

- Optimaler Schedulingverfahren
 - Allgemeine Beurteilung
 - Unterschiedliche Eigenschaften von Systemen und Aufgaben
 - Geeignete Ablaufplanung auswählen



Fragen?

Vielen Dank für Ihre Aufmerksamkeit!

