Fixed-Priority Schedulability of Sporadic Tasks on Uniprocessors is NP-hard

Pontus Ekberg & Wang Yi

Uppsala University

RTSS 2017
**Overview**

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p \text{ unrelated}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arbitrary utilization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Joseph and Pandya, 1986  
Liu and Layland, 1973  
Lehoczky, 1990
**Overview**

<table>
<thead>
<tr>
<th>Utilization bounded by a constant $c$</th>
<th>Implicit deadlines $(d = p)$</th>
<th>Constrained deadlines $(d \leq p)$</th>
<th>Arbitrary deadlines $(d, p$ unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

(*) Joseph and Pandya, 1986

Liu and Layland, 1973

Lehoczky, 1990

Joseph and Pandya, 1986
### Overview

<table>
<thead>
<tr>
<th>Utilization bounded by a constant $c$</th>
<th>Implicit deadlines $(d = p)$</th>
<th>Constrained deadlines $(d \leq p)$</th>
<th>Arbitrary deadlines $(d, p$ unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arbitrary utilization</td>
<td>Pseudo-poly. time algorithm</td>
<td>Pseudo-poly. time algorithm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Polyn. time for $c \leq \ln 2$ and RM priorities</td>
<td>Polynomial time algorithm</td>
<td>Pseudo-poly. time algorithm</td>
</tr>
</tbody>
</table>

$(*)$ Joseph and Pandya, 1986

$(†)$ Liu and Layland, 1973
<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ($d = p$)</th>
<th>Constrained deadlines ($d \leq p$)</th>
<th>Arbitrary deadlines ($d, p$ unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arbitrary utilization</td>
<td>Pseudo-poly. time algorithm</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td>Utilization bounded by a constant $c$</td>
<td>Polynomial time for $c \leq \ln 2$ and RM priorities</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
</tbody>
</table>

(*) Joseph and Pandya, 1986

(†) Liu and Layland, 1973

(‡) Lehoczky, 1990
## Overview

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p \text{ unrelated}))</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td>Pseudo-poly. time algorithm</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time for (c \leq \ln 2) and RM priorities</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Overview

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Pseudo-poly. time algorithm</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td></td>
<td>Polynomial time for (c \leq \ln 2) and RM priorities</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td></td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>

Joseph and Pandya, 1986

Liu and Layland, 1973

Lehoczky, 1990
## Overview

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td>Pseudo-poly. time algorithm</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><strong>Utilization</strong></td>
<td>Polynomial time for (c \leq \ln 2) and RM priorities</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td><strong>Utilization</strong></td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td><strong>Constrained</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Arbitrary</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Utilization</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(c\) bounded by a constant \(c\)

---

\(d = p\)

\(d \leq p\)

\((d, p\) unrelated)

Joseph and Pandya, 1986

Liu and Layland, 1973

Lehoczky, 1990

Pontus Ekberg
### Overview

<table>
<thead>
<tr>
<th>Utilization</th>
<th>Implicit deadlines ($d = p$)</th>
<th>Constrained deadlines ($d \leq p$)</th>
<th>Arbitrary deadlines ($d, p$ unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FP</td>
<td>Pseudo-poly. time algorithm</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td>Polynomials bounded by a constant $c$</td>
<td>Polynomial time for $c \leq \ln 2$ and RM priorities</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td>EDF</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Polynomials bounded by a constant $c$</td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for $0 &lt; c &lt; 1$</td>
<td>Weakly coNP-complete for $0 &lt; c &lt; 1$</td>
</tr>
</tbody>
</table>

- **Utilization**
  - Arbitrary
  - FP: Pseudo-poly. time algorithm
  - EDF: Weakly coNP-complete for $0 < c < 1$

- **Deadlines**
  - Implicit: $d = p$
  - Constrained: $d \leq p$
  - Arbitrary: $d, p$ unrelated

- **Complexity Classes**
  - FP: Weakly NP-hard
  - EDF: Weakly coNP-complete

- **Priorities**
  - FP: Pseudo-poly. time algorithm
  - EDF: Polynomial time

References:
- Joseph and Pandya, 1986
- Liu and Layland, 1973
- Lehoczky, 1990

*Pontus Ekberg*
### Overview

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p \text{ unrelated}))</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Arbitrary utilization</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FP</td>
<td>Weakly NP-hard</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time (\text{for } c \leq \ln 2 \text{ and RM priorities})</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td>EDF</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>

\(\text{FP}\) Joseph and Pandya, 1986

\(\text{EDF}\) Liu and Layland, 1973

\(\text{EDF}\) Lehoczky, 1990

**Polynomial time**

**Strongly coNP-complete** for \(0 < c < 1\)

**Weakly coNP-complete** for \(0 < c < 1\)

**Weakly NP-hard**

**Pseudo-poly. time algorithm**

**Exponential time algorithm**
### Overview

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td>Weakly (NP)-hard</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><strong>Utilization</strong></td>
<td>Polynomial time for (c \leq\ln 2) and RM priorities</td>
<td>Polynomial time</td>
<td>Polynomial time</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td>Polynomial time</td>
<td>Strongly (coNP)-complete</td>
<td>Strongly (coNP)-complete</td>
</tr>
<tr>
<td><strong>Utilization</strong></td>
<td>Weakly (coNP)-complete for (0 &lt; c &lt; 1)</td>
<td>Strongly (coNP)-complete</td>
<td>Weakly (coNP)-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>

\(d = p\): Deadlines are implicit and match the processor speed.
\(d \leq p\): Deadlines are constrained by the processor speed.
\(d, p\) unrelated: Deadlines and processor speed are unrelated.

- **FP** (First Fit)
- **EDF** (Earliest Deadline First)

Joseph and Pandya, 1986
Liu and Layland, 1973
Lehoczky, 1990

**Utilization bounded by a constant** \(c\)

- Polynomial time for \(c \leq\ln 2\) and RM priorities
- Weakly \(coNP\)-complete for \(0 < c < 1\)
# Overview

<table>
<thead>
<tr>
<th>Implicit deadlines $(d = p)$</th>
<th>Constrained deadlines $(d \leq p)$</th>
<th>Arbitrary deadlines $(d, p$ unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Weakly NP-hard</td>
<td>Pseudo-poly. time algorithm</td>
</tr>
<tr>
<td>Utilization bounded by a constant $c$</td>
<td>Polynomial time for $c \leq \ln 2$ and RM priorities</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Utilization bounded by a constant $c$</td>
<td>Polynomial time for $0 &lt; c &lt; 1$</td>
<td>Strongly coNP-complete</td>
</tr>
</tbody>
</table>

- Weakly NP-hard
- Pseudo-poly. time algorithm
- Exponential time algorithm
- Strongly coNP-complete
- Polynomial time
- Weakly coNP-complete for $0 < c < 1$
## Overview

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ($d = p$)</th>
<th>Constrained deadlines ($d \leq p$)</th>
<th>Arbitrary deadlines ($d, p$ unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td>Weakly NP-hard</td>
<td>Pseudo-poly. time algorithm</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><em>Arbitrary utilization</em></td>
<td>Polynomial time for $c \leq \ln 2$ and RM priorities</td>
<td>Weakly NP-hard for $0 &lt; c &lt; 1$</td>
<td>Exponential time algorithm</td>
</tr>
<tr>
<td><em>Utilization bounded by a constant $c$</em></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td><em>Arbitrary utilization</em></td>
<td>Weakly coNP-complete for $0 &lt; c &lt; 1$</td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>Utilization bounded by a constant $c$</em></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**
- Joseph and Pandya, 1986
- Liu and Layland, 1973
- Lehoczky, 1990

**Utilization bounded by a constant $c$:**
- Polynomial time for $c \leq \ln 2$ and RM priorities
- Weakly NP-hard for $0 < c < 1$
- Strongly coNP-complete for $0 < c < 1$
# Overview

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arbitrary utilization</td>
<td>Weakly NP-hard</td>
<td>Weakly NP-hard</td>
<td>Weakly NP-hard</td>
</tr>
<tr>
<td>FP</td>
<td>Polynomial time for (c \leq \ln 2) and RM priorities</td>
<td>Weakly NP-hard for (0 &lt; c &lt; 1)</td>
<td>Weakly NP-hard for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td>EDF</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>

Joseph and Pandya, 1986

Liu and Layland, 1973

Lehoczky, 1990
## Overview

<table>
<thead>
<tr>
<th>Implicit deadlines ( (d = p) )</th>
<th>Constrained deadlines ( (d \leq p) )</th>
<th>Arbitrary deadlines ( (d, p \text{ unrelated}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-complete</td>
</tr>
<tr>
<td>Polynomial time for ( c \leq \ln 2 ) and RM priorities</td>
<td>Weakly NP-complete for ( 0 &lt; c &lt; 1 )</td>
<td>Weakly NP-hard for ( 0 &lt; c &lt; 1 )</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Polynomial time</td>
<td>Weakly coNP-complete for ( 0 &lt; c &lt; 1 )</td>
<td>Weakly coNP-complete for ( 0 &lt; c &lt; 1 )</td>
</tr>
<tr>
<td>Utilization bounded by a constant ( c )</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Utilization bounded by a constant ( c )</td>
<td>Weakly coNP-complete for ( 0 &lt; c &lt; 1 )</td>
<td>Weakly coNP-complete for ( 0 &lt; c &lt; 1 )</td>
</tr>
</tbody>
</table>

\( \text{Joseph and Pandya, 1986} \)

\( \text{Liu and Layland, 1973} \)

\( \text{Lehoczky, 1990} \)
A Tale of Two Reductions

EDF-schedulability

- Constrained deadlines
- Bounded utilization

FP-schedulability

- Implicit deadlines
  —or—
- Constrained deadlines
- Bounded utilization
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines
  —or—
  - Constrained deadlines
  - Bounded utilization

YES
NO

coNP-hard

NP-hard
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines
- Constrained deadlines
- Bounded utilization

YES $\rightarrow$ YES $\rightarrow$ YES
NO $\rightarrow$ NO $\rightarrow$ NO

coNP-hard

Pontus Ekberg
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines
- Constrained deadlines
- Bounded utilization

YES → YES → YES → YES
NO → NO → NO → NO

coNP-hard
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines
  —or—
- Constrained deadlines
- Bounded utilization

\[
\begin{array}{ccc}
\text{YES} & \rightarrow & \text{YES} \\
\text{NO} & \rightarrow & \text{NO} \\
\text{coNP-hard} & \rightarrow & \text{coNP-hard}
\end{array}
\]
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines
- Constrained deadlines
- Bounded utilization

Yes
No

coNP-hard

Yes
No

coNP-hard

Yes
No

NP-hard
A Tale of Two Reductions

**EDF-schedulability**
- Constrained deadlines
- Bounded utilization

**EDF-schedulability**
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

**FP-schedulability**
- Implicit deadlines
- or
- Constrained deadlines
- Bounded utilization

\[ \begin{align*}
\text{YES} & \rightarrow \text{YES} \\
\text{NO} & \rightarrow \text{NO}
\end{align*} \]

\[ \begin{align*}
\text{coNP-hard} & \rightarrow \text{coNP-hard} \\
\text{NP-hard} & \rightarrow \text{NP-hard}
\end{align*} \]
EDF-schedulability $\leadsto$ FP-schedulability
EDF-schedulability $\leadsto$ FP-schedulability

\[ dbf(\tau_1) \]

\[ dbf(\tau_2) \]

\[ dbf(\tau_3) \]

Pontus Ekberg
EDF-schedulability $\leadsto$ FP-schedulability

\[
dbf(\tau_1) = \sum dbf(1) + \sum dbf(2) + \sum dbf(3) = \sum (x + 1) = D \equiv D = x + 1
\]
EDF-schedulability $\iff$ FP-schedulability

\[ \text{dfb}(\tau_1) + \text{dfb}(\tau_2) + \text{dfb}(\tau_3) = \sum \text{dfb} \]

\[ D = x + 1 \]
EDF-schedulability $\rightsquigarrow$ FP-schedulability
EDF-schedulability \(\rightsquigarrow\) FP-schedulability

\[
dbf(\tau_1) + \dbf(\tau_2) + \dbf(\tau_3) = D \leq \sum \dbf
\]
EDF-schedulability $\Leftrightarrow$ FP-schedulability

$$
\begin{align*}
&\text{dbf}(\tau_1) + \text{dbf}(\tau_2) + \text{dbf}(\tau_3) = \\
&\sum \text{dbf}
\end{align*}
$$
EDF-schedulability $\leadsto$ FP-schedulability

\[ \text{dbf}(\tau_1) + \text{dbf}(\tau_2) + \text{dbf}(\tau_3) = \sum \text{dbf} \]

\[ \text{rbf}(\tau_1) + \text{rbf}(\tau_2) + \text{rbf}(\tau_3) = \sum \text{dbf} \]

Pontus Ekberg
EDF-schedulability $\leadsto$ FP-schedulability

\[
\sum \text{dbf} = D \iff D = \sum \text{rbf} + x
\]
EDF-schedulability $\leadsto$ FP-schedulability

$$
\text{dbf}(\tau_1) + \text{dbf}(\tau_2) + \text{dbf}(\tau_3) = \sum \text{dbf} = \text{dbf}(\tau_1) + \text{dbf}(\tau_2) + \text{dbf}(\tau_3) = \sum \text{rbf} + x + 1
$$
EDF-schedulability $\leadsto$ FP-schedulability

$$dbf(\tau_1) + dbf(\tau_2) + dbf(\tau_3) = D$$

$$\sum dbf = D \iff \sum \text{dbf}$$

$$\sum rbf(\tau_1) + rbf(\tau_2) + rbf(\tau_3) = x + 1$$
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines—or—
- Constrained deadlines
- Bounded utilization

YES → YES → YES
NO → NO → NO

coNP-hard
coNP-hard
NP-hard
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines
- or
- Constrained deadlines
- Bounded utilization

YES → YES → YES
NO → NO → NO

coNP-hard
coNP-hard
NP-hard
Reducing EDF-schedulability to a Special Case

Bonifaci et al., 2013
Reducing EDF-schedulability to a Special Case

EDF-schedulability with constrained deadlines and bounded utilization

Bonifaci et al., 2013

T

T'
Reducing EDF-schedulability to a Special Case

- EDF-schedulability with constrained deadlines and bounded utilization is coNP-hard.
- With pairwise coprime periods, EDF-schedulability is coNP-hard.
- With harmonic periods, EDF-schedulability is polynomial time.

Bonifaci et al., 2013
Reducing EDF-schedulability to a Special Case

EDF-schedulability with constrained deadlines and bounded utilization

coNP-hard

With pairwise coprime periods
Reducing EDF-schedulability to a Special Case

EDF-schedulability with constrained deadlines and bounded utilization

coNP-hard

coNP-hard?

With pairwise coprime periods

EDF-schedulability with constrained deadlines and bounded utilization

With pairwise coprime periods
Reducing EDF-schedulability to a Special Case

EDF-schedulability with constrained deadlines and bounded utilization

coNP-hard

Polynomial time\(^\dagger\)

With pairwise coprime periods

coNP-hard?

With harmonic periods

\(\dagger\) Bonifaci et al., 2013
Reducing EDF-schedulability to a Special Case

EDF-schedulability with constrained deadlines and bounded utilization

- coNP-hard
- Polynomial time

With pairwise coprime periods

With harmonic periods

\[ \mathcal{T} \]

† Bonifaci et al., 2013
Reducing EDF-schedulability to a Special Case

EDF-schedulability with constrained deadlines and bounded utilization

- coNP-hard
- Polynomial time\(^\dagger\)

With pairwise coprime periods

With harmonic periods

\(^\dagger\) Bonifaci et al., 2013
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number.
2. Add small numbers $i$ to each period so that the periods become pairwise coprime.

- The $i$ can be found in polynomial time.
- The $i$ are so small relative to the parameters that schedulability is unaffected.
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.

\begin{align*}
  e_1 & \quad d_1 & \quad p_1 \\
  e_2 & \quad d_2 & \quad p_2 \\
  e_3 & \quad d_3 & \quad p_3 \\
  \cdots \\
  e_n & \quad d_n & \quad p_n
\end{align*}
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$. 

- The $\kappa$ can be found in polynomial time.
- The $\kappa$ are so small relative to $\kappa$ that schedulability is unaffected.
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.

2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

\[ \kappa e_1 \quad \kappa d_1 \quad \kappa p_1 \quad \kappa e_2 \quad \kappa d_2 \quad \kappa p_2 \quad \kappa e_3 \quad \kappa d_3 \quad \kappa p_3 \quad \ldots \quad \kappa e_n \quad \kappa d_n \quad \kappa p_n \]
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.
2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

The $i$ can be found in polynomial time.

The $i$ are so small relative to that schedulability is unaffected.
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.

2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

\begin{itemize}
\item $\delta_1$ can be found in polynomial time.
\item $\delta_i$ are so small relative to $\kappa$ that schedulability is unaffected.
\end{itemize}
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.
2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

$\begin{align*}
\kappa e_1 & \quad \kappa d_1 & \quad \kappa p_1 \\
\kappa e_2 & \quad \kappa d_2 & \quad \kappa p_2 \\
\kappa e_3 & \quad \kappa d_3 & \quad \kappa p_3 \\
\ldots & \quad \kappa e_n & \quad \kappa d_n & \quad \kappa p_n
\end{align*}$
**Outline of the Reduction**

1. Scale all task parameters uniformly by a *huge* number $\kappa$.

2. Add *small* numbers $\delta_i$ to each period so that the periods become pairwise coprime.

- The $\delta_i$ can be found in polynomial time.
1. Scale all task parameters uniformly by a huge number $\kappa$.

2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

- The $\delta_i$ can be found in polynomial time.
- The $\delta_i$ are so small relative to $\kappa$ that schedulability is unaffected.
The Jacobsthal function $g(n)$ gives the largest gap between numbers that are coprime to $n$. 

Iwaniec, 1978
The Jacobsthal function \( g(n) \) gives the largest gap between numbers that are coprime to \( n \).

Coprime to 100?
The Jacobsthal function $g(n)$ gives the largest gap between numbers that are coprime to $n$. 

Coprime to 100?

0  5  10  15  20  25  30  35
The Jacobsthal function $g(n)$ gives the largest gap between numbers that are coprime to $n$. 

$g(100) = 4$
Some Number Theory

The Jacobsthal function

The Jacobsthal function \( g(n) \) gives the largest gap between numbers that are coprime to \( n \).

Coprime to 100?

\[
g(100) = 4
\]

\[
g(n) \in \Theta(\log^2 n)
\]

Iwaniec, 1978
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.

2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

- The $\delta_i$ can be found in polynomial time.
- The $\delta_i$ are so small relative to $\kappa$ that schedulability is unaffected.
1. Scale all task parameters uniformly by a huge number $\kappa$.

2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

- The $\delta_i$ can be found in polynomial time.
- The $\delta_i$ are so small relative to $\kappa$ that schedulability is unaffected.
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.
2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

- The $\delta_i$ can be found in polynomial time.
- The $\delta_i$ are so small relative to $\kappa$ that schedulability is unaffected.
Schedulability unaffected

Shifted at most $\max_i \overline{H}_i$
Schedulability unaffected

\[ \text{Shifted how much?} \]

\[ \text{Shifted at most} \quad \max_i \quad \text{HP} \]

\[ \quad \implies \quad \text{Schedulability unaffected if} \quad > \quad \max_i \quad \text{HP} \]
Schedulability unaffected

Shifted at most $\max_i i$
Schedulability unaffected

\[ \text{Shifted at most } \max_i \]
Schedulability unaffected

Shifted how much?

≥ \kappa

\text{Shifted at most} \ max_i \ HP

Schedulability unaffected if > \ max_i \ HP
Schedulability unaffected if
\[ \max_i \delta_i \cdot HP \geq K \]
Schedulability unaffected if \( \kappa > \max_i \delta_i \cdot HP \)

Shifted at most \( \max_i \delta_i \cdot HP \)

\( \geq \kappa \)
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.

2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

- The $\delta_i$ can be found in polynomial time.
- The $\delta_i$ are so small relative to $\kappa$ that schedulability is unaffected.
Outline of the Reduction

1. Scale all task parameters uniformly by a huge number $\kappa$.

2. Add small numbers $\delta_i$ to each period so that the periods become pairwise coprime.

- The $\delta_i$ can be found in polynomial time.
- The $\delta_i$ are so small relative to $\kappa$ that schedulability is unaffected.
A Tale of Two Reductions

EDF-schedulability
- Constrained deadlines
- Bounded utilization

EDF-schedulability
- Constrained deadlines
- Bounded utilization
- Pairwise coprime periods

FP-schedulability
- Implicit deadlines
- or-
- Constrained deadlines
- Bounded utilization

YES → YES → YES
NO → NO → NO

coNP-hard → coNP-hard → NP-hard
**Conclusions**

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-hard</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time for (c \leq \ln 2) and RM priorities</td>
<td>Weakly NP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly NP-hard for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>
## Conclusions

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td>Weakly NP-complete</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-hard</td>
</tr>
<tr>
<td><strong>Utilization</strong></td>
<td>Polynomial time (c \leq \ln 2) and RM priorities</td>
<td>Weakly NP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly NP-hard for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td><strong>Arbitrary</strong></td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td><strong>utilization</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>
## Conclusions

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-hard</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time for (c \leq \ln 2) and RM priorities</td>
<td>Weakly NP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly NP-hard for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arbitrary utilization</td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>
## Conclusions

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td>Weakly NP-complete</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-hard</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time for (c \leq \ln 2) and RM priorities</td>
<td>Weakly NP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly NP-hard for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td>Polynomial time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>Utilization bounded by a constant (c)</td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
</tbody>
</table>

**FP**
- Arbitrary utilization
- Weakly NP-complete
- Weakly NP-complete
- Weakly NP-hard

**EDF**
- Arbitrary utilization
- Polynomial time
- Strongly coNP-complete
- Strongly coNP-complete
Conclusions

<table>
<thead>
<tr>
<th></th>
<th>Implicit deadlines ((d = p))</th>
<th>Constrained deadlines ((d \leq p))</th>
<th>Arbitrary deadlines ((d, p) unrelated)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP</strong></td>
<td>Weakly NP-complete</td>
<td>Weakly NP-complete</td>
<td>Weakly NP-hard</td>
</tr>
<tr>
<td>Utilization</td>
<td>Polynomial time</td>
<td>Weakly NP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly NP-hard for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td>bounded by a</td>
<td>Polynomials time</td>
<td>Strongly coNP-complete</td>
<td>Strongly coNP-complete</td>
</tr>
<tr>
<td>constant (c)</td>
<td>for (c \leq \ln 2) and RM priorities</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>EDF</strong></td>
<td>Polynomial time</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
<td>Weakly coNP-complete for (0 &lt; c &lt; 1)</td>
</tr>
<tr>
<td>Utilization</td>
<td>Polynomial time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bounded by a</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>constant (c)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
∀Thank you!

∃Questions?