On the Problem of Finding Optimal Harmonic Periods

Morteza Mohaqeqi  Mitra Nasri  Yang Xu, Anton Cervin, Karl-Erik Årzén
Overview: Period Selection

Step Response

- High cost
- Low utilization
- Small task period
- Low (control) cost
- Large task period
- High utilization

Cart position (m) vs. time

T = 0.1
T = 0.01
Overview: Period Selection

- **Utilization**: High 
- **Task Period**: Small 
- **(control) cost**: Low 
- **Period Selection**

Graphs showing cost and utilization over period with different values of M.
Introduction

- Harmonic periods:
  \[ \frac{T_{i+1}}{T_i} \in \mathbb{N} \]

Benefits of Harmonic Periods
- High schedulable utilization
- More predictability
- Small hyperperiod
- Polynomial-time WCRT analysis
Overview

Goal

Finding optimal feasible harmonic periods

Contributions

Two approximation algorithms:
- Polynomial-time complexity
- Error bounds:
  - General case: 2
  - A special case: $2 - 1/n$

Complexity result:
- The problem (with period intervals) is NP-Hard
Agenda

- Introduction
  - Approximation Algorithms
    - A Linear-Time Algorithm
    - A Quadratic-Time Algorithm
  - Evaluation
- Complexity Result
- Conclusion
Introduction

- Cost function: $f_i(T_i)$

Objective

- Minimize $\sum_{i=1}^{n} f_i(T_i)$

Constraints

- Feasibility
- Harmonicity
- Period limits

$C_i$: WCET

$\sum_{i=1}^{n} \frac{C_i}{T_i} \leq 1$

$\frac{T_{i+1}}{T_i} \in \mathbb{N}$

$T_i \in [T_i^{\text{min}}, T_i^{\text{max}}]$
Agenda

- Introduction
- Approximation Algorithms
  - A Linear-Time Algorithm
  - A Quadratic-Time Algorithm
  - Evaluation
- Complexity Result
- Conclusion
Problem Formulation

Problem 1

\[
\min S = \sum_{i=1}^{n} w_i T_i
\]

subject to

\[i. \quad \frac{T_{i+1}}{T_i} \in \mathbb{N}, \quad \text{for } 1 \leq i < n\]

\[\text{ii. } \sum_{i=1}^{n} \frac{c_i}{T_i} \leq 1\]
Problem 1 (Arbitrary Period)

\[
\min S = \sum_{i=1}^{n} w_i T_i
\]

subject to

i. \( \frac{T_{i+1}}{T_i} \in \mathbb{N} \), for \( 1 \leq i < n \)

ii. \( \sum_{i=1}^{n} \frac{c_i}{T_i} \leq 1 \)

Solution ([Cervin et al. 2002])

\[
T_i' = \sqrt{c_i/w_i} \cdot D, \quad \text{where } D = \sum_{k=1}^{n} \sqrt{c_k w_k}
\]

Polynomial-time

\[
\sum_{i=1}^{n} \frac{c_i}{T_i'} = 1
\]
An Approximation Algorithm

**Solution (Unconstrained)**

\[ T'_i = \sqrt{\frac{C_i}{w_i}} D, \quad \text{where} \quad D = \sum_{k=1}^{n} \sqrt{C_k w_k} \]

\[ T_1' = T_1 \]

\[ T_2' = 3T_1' \]

\[ T_2' = 2T_1' \]

\[ T_3' = 2T_2' \]

\[ \frac{T_3'}{T_2'} = 2 \]

\[ \frac{T_2'}{T_1'} = 2.7 \]

\[ \frac{T_3'}{T_2'} = 2 \]

**Procedure**

**Steps:**

1. \( T_1^* = T_1' \)

2. \( T_{i+1}^* = \left[ \frac{T_{i+1}'}{T_i'} \right] T_i^* \quad (2 \leq i \leq n) \)

\( U = 1 \)
Backward Procedure

Solution

Steps:
1. $T_n^* = T'_n$
2. $T_{i-1}^* = T_i^* / \left| \frac{T_i^*}{T_{i-1}'} \right|$
   
   $(2 \leq i \leq n)$
Algorithm 2

**Solution (Unconstrained)**

\[ T_i' = \sqrt{\frac{C_i}{w_i}} D, \quad \text{where} \quad D = \sum_{k=1}^{n} \sqrt{C_k w_k} \]
Summary

Running Time

- Algorithm 1: Linear
- Algorithm 2: Quadratic

Error Bounds

- General case: \( \frac{S^*}{S'} \leq 2 \)
- Equal weights: \( \frac{S^*}{S'} \leq 2 - \frac{1}{n} \)

\[
S^* = \sum_{i=1}^{n} w_i T_i^*
\]
\[
S' = \sum_{i=1}^{n} w_i T_i'
\]
Agenda

- Introduction
- Approximation Algorithms
  - A Linear-Time Algorithm
  - A Quadratic-Time Algorithm
- Evaluation
- Complexity Result
- Conclusion
### Method Evaluation

#### Questions that we want to answer

- How an increase in the number of tasks will affect the relative error of our approach?
- How our algorithms perform if the execution times are selected from wider ranges.

#### Measure

<table>
<thead>
<tr>
<th>Relative Error $= \frac{S^*}{S'}$</th>
</tr>
</thead>
</table>

\[
S^* = \sum_{i=1}^{n} w_i T_i^*, \quad S' = \sum_{i=1}^{n} w_i T_i'
\]

Non-Harmonic
Generation:
The WCET of each task is selected from [1, 500] with uniform distribution

Algorithm 2 is as good as the optimal algorithm
Method Evaluation: The Effect of the Wideness of WCETs

![Graph showing Relative Error vs. K (Wideness of WCETs)]

**Algorithm 2 is as good as the optimal algorithm**

**Generation**
- $C_1$ from $[1, 10]$ with uniform distribution
- $C_i$ from $[C_{i-1}, kC_{i-1}]$ with uniform distribution
Evaluation in a Control Application

- Three plant families:
  - Family I: two poles (stable)
  - Family II: two poles (stable or unstable)
  - Family III: three poles (stable or unstable)

Control costs:

<table>
<thead>
<tr>
<th>Family</th>
<th>I</th>
<th>II</th>
<th>III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-harmonic</td>
<td>6.80</td>
<td>19.67</td>
<td>17.04</td>
</tr>
<tr>
<td>Algorithm 1</td>
<td>3.07</td>
<td>6.78</td>
<td>15.59</td>
</tr>
<tr>
<td>Algorithm 2</td>
<td>2.91</td>
<td>6.71</td>
<td>15.17</td>
</tr>
</tbody>
</table>
Agenda

- Introduction
- Approximation Algorithms
  - A Linear-Time Algorithm
  - A Quadratic-Time Algorithm
  - Evaluation
- Complexity Result
- Conclusion
Problem 2

Objective

• Minimize \( \sum_{i=1}^{n} f_i(T_i) \)

Constraints

• Harmonicity
• Feasibility
• Period limits

\( T_i \in [T_i^{\text{min}}, T_i^{\text{max}}] \)

\[
\text{cost}(T_i) = w_i T_i
\]

\[
\min \sum_{i=1}^{n} \text{cost}(T_i)
\]

\[
\text{gain}(T_i) = \frac{w_i}{T_i}
\]

\[
\max \sum_{i=1}^{n} \text{gain}(T_i)
\]
Optimization Objective

Problem 2

\[
\text{max } S = \sum_{i=1}^{n} \frac{C_i}{T_i}
\]

subject to:

i. \( \frac{T_{i+1}}{T_i} \in \mathbb{N} \), for \( 1 \leq i < n \)

ii. \( \sum_{i=1}^{n} U_i \leq 1 \)

iii. \( T_i \in [T_i^{\text{min}}, T_i^{\text{max}}] \), for \( 1 \leq i \leq n \)
Complexity Result

- **Number Partitioning Problem**
  - \( A = \{a_1, a_2, ..., a_n\} \)
  - Partition \( A \) into \( A_1 \) and \( A_2 \) such that \( \sum_{a_i \in A_1} a_i = \sum_{a_i \in A_2} a_i \)
  - Weakly NP-Hard
Complexity Result

- **Number Partitioning Problem**
  - $A = \{a_1, a_2, \ldots, a_n\}$
  - Partition $A$ into $A_1$ and $A_2$ such that $\sum_{a_i \in A_1} a_i = \sum_{a_i \in A_2} a_i$
  - Weakly NP-Hard

- **Transformation:**

For $1 \leq i \leq n$:

\[
\begin{align*}
C_i &= a_i \\
T_i &\in [1,2]
\end{align*}
\]

\[
\begin{align*}
C_0 &= 0 \\
T_0 &\in [1,1]
\end{align*}
\]

1. Tasks corresponding to $A_1$
2. Tasks corresponding to $A_2$
Agenda

- Introduction
- Approximation Algorithms
  - A Linear-Time Algorithm
  - A Quadratic-Time Algorithm
  - Evaluation
- Complexity Result
- Conclusion
## Conclusion

### Contributions:
- Efficient algorithms with **bounded approx. error**
- A **hardness** results

### Open Problems:

<table>
<thead>
<tr>
<th></th>
<th>$\max U$</th>
<th>$\min \sum w_i T_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restricted Range</td>
<td>Weakly NP-hard</td>
<td>?</td>
</tr>
<tr>
<td>Unrestricted range</td>
<td>Poly. Time algorithm exists.</td>
<td>? (poly. time approximation)</td>
</tr>
</tbody>
</table>
On the Problem of Finding Optimal Harmonic Periods

Thanks!

Morteza Mohaqeqi
Uppsala University, Sweden
morteza.mohaqeqi@it.uu.se

Mitra Nasri
Max Planck Institute, Germany

Yang Xu, Anton Cervin, Karl-Erik Årzén
Lund University, Sweden