On Strong and Weak Sustainability, with an Application to Self-Suspending Real-Time Tasks

Felipe Cerqueira, Geoffrey Nelissen, Björn Brandenburg

ECRTS’18, Barcelona
July 6, 2018

Partially funded by Deutsche Forschungsgemeinschaft (DFG) — 391919384

http://prosa.mpi-sws.org
This Paper in a Nutshell

On Strong and Weak Sustainability, with an Application to Self-Suspending Real-Time Tasks

Felipe Cerqueira
Max Planck Institute for Software Systems (MPI-SWS), Kaiserslautern, Germany
felipec@mpi-sws.org

Georey Nelissen
CISTER Research Centre, ISEP, Polytechnic Institute of Porto (IPP), Porto, Portugal
grrpn@isep.ipp.pt

Björn B. Brandenburg
Max Planck Institute for Software Systems (MPI-SWS), Kaiserslautern, Germany
bbb@mpi-sws.org

Abstract
Motivated by an apparent contradiction regarding whether certain scheduling policies are sustainable, we revisit the topic of sustainability in real-time scheduling and argue that the existing definitions of sustainability should be further clarified and generalized. After proposing a formal, generic sustainability theory, we relax the existing notion of (strongly) sustainable scheduling policy to provide a new classification called weak sustainability. Proving weak sustainability properties allows reducing the number of variables that must be considered in the search of a worst-case schedule, and hence enables more efficient schedulability analyses and testing regimes even for policies that are not (strongly) sustainable. As a proof of concept, and to better understand a model for which many mistakes were found in the literature, we study weak sustainability in the context of dynamic self-suspending tasks, where we formalize a generic suspension model using the Coq proof assistant and provide a machine-checked proof that any JLFP scheduling policy is weakly sustainable with respect to job costs and variable suspension times.

2012 ACM Subject Classification
Software and its engineering
Real-time schedulability

Keywords and phrases
real-time scheduling, sustainability, self-suspending tasks, machine-checked proofs

Digital Object Identifier
10.4230/LIPIcs.ECRTS.2018.26

Funding
This work was funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) — 391919384, and partially supported by National Funds through FCT (Portuguese Foundation for Science and Technology) within the CISTER Research Unit (CEC/04234).

What Really is Sustainability?
Since the seminal paper by Liu and Layland [14], the analysis and certification of real-time systems has often relied on the fundamental notion of sustainability [6], which at a high level expresses the idea that “if a system is proven to be safe under extreme conditions, then it will remain safe if the conditions improve at runtime.” By allowing system designers to focus on such extreme scenarios (rather than the entire state space of the system), sustainability plays a fundamental role in the design, prototyping, analysis, and validation of real-time systems. One common application of this principle is to determine the schedulability of the system by identifying worst-case scheduling scenarios. For example, any schedulability analysis for uniprocessor fixed-priority (FP) scheduling of sporadic tasks [15] that assumes that jobs

© Felipe Cerqueira, Georey Nelissen and Björn B. Brandenburg; licensed under Creative Commons License CC-BY

Editor: Sebastian Altmeyer; Article No. 26; pp. 26:1–26:21
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany
Identify an ambiguity in the definition of sustainability, a central notion in real-time scheduling theory and practice
This Paper in a Nutshell

Identify an ambiguity in the definition of sustainability, a central notion in real-time scheduling theory and practice

Precise, formal theory that covers all types of sustainability
This Paper in a Nutshell

Identify an ambiguity in the definition of sustainability, a central notion in real-time scheduling theory and practice

Precise, formal theory that covers all types of sustainability

New notion of weak sustainability that enables more efficient analysis for policies that are not sustainable
This Paper in a Nutshell

Identify an ambiguity in the definition of sustainability, a central notion in real-time scheduling theory and practice

Precise, formal theory that covers all types of sustainability

New notion of weak sustainability that enables more efficient analysis for policies that are not sustainable

Uniprocessor JLFP scheduling of self-suspending tasks is weakly weakly sustainable w.r.t. execution times and variable suspension times
This Talk
This Talk

Ambiguity in the Current Definitions
This Talk

Ambiguity in the Current Definitions

Strong and Weak Sustainability
This Talk

Ambiguity in the Current Definitions

Strong and Weak Sustainability

Weak Sustainability of Self-Suspending Tasks
This Talk

Ambiguity in the Current Definitions

Strong and Weak Sustainability

Weak Sustainability of Self-Suspending Tasks
What really is sustainability?
Sustainability of Sporadic Tasks

Original Job Set

$T_1$ $T_2$ $T_3$
Sustainability of Sporadic Tasks

Original Job Set

Job set misses no deadline!
Sustainability of Sporadic Tasks

Original Job Set

Reducing cost of $T_3$

Assume $T_3$ completes one unit earlier
Sustainability of Sporadic Tasks

Original Job Set

Reducing cost of $T_3$

Assume $T_3$ completes one unit earlier

Job set misses no deadline!
Sustainable w.r.t. Execution Times

Uniprocessor scheduling of sporadic tasks is sustainable with respect to execution times.

Definition of Sustainable Policy

[Burns and Baruah, 2008]

Definition 1 A scheduling policy and/or a schedulability test for a scheduling policy is sustainable if any system deemed schedulable by the schedulability test remains schedulable when the parameters of one or more individual tasks are changed in any, some, or all of the following ways: (i) decreased execution requirements; (ii) larger periods; (iii) smaller jitter; and (iv) larger relative deadlines.
Definition of Sustainable Policy

[Burns and Baruah, 2008]

Definition 1 A scheduling policy and/or a schedulability test for a scheduling policy is sustainable if any system deemed schedulable by the schedulability test remains schedulable when the parameters of one or more individual tasks are changed in any, some, or all of the following ways: (i) decreased execution requirements; (ii) larger periods; (iii) smaller jitter; and (iv) larger relative deadlines.

If a system is schedulable with original parameters, then it remains schedulable when assigned better parameters.

Why Does Sustainability Matter?
Why Does Sustainability Matter?

Analysis

Minimize/maximize parameters to reduce the search space.

Validation & Certification

It suffices to check only the extreme execution scenarios, without compromising safety.
Why Does Sustainability Matter?

Analysis

Minimize/maximize parameters to reduce the search space.

Liu and Layland in 1973 already assumed maximum execution times

(A4) **Run-time for each task is constant** for that task and does not vary with time. Run-time here refers to the time which is taken by a processor to execute the task without interruption.

Why Does Sustainability Matter?

### Analysis

Minimize/maximize parameters to reduce the search space.
Why Does Sustainability Matter?

<table>
<thead>
<tr>
<th>Analysis</th>
<th>Minimize/maximize parameters to reduce the search space.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design</td>
<td>Engineers can assume worst-case parameters for design space exploration.</td>
</tr>
</tbody>
</table>
## Why Does Sustainability Matter?

<table>
<thead>
<tr>
<th>Analysis</th>
<th>Minimize/maximize parameters to reduce the search space.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Design</strong></td>
<td>Engineers can assume worst-case parameters for design space exploration.</td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td>Ensure system safety at runtime under non-worst-case conditions.</td>
</tr>
</tbody>
</table>
Definition of Sustainability is Ambiguous
EDF Scheduling of Self-Suspending Tasks is **Not Sustainable** w.r.t. Execution Times

[Example by Abeddaiim and Masson, 2012]
EDF Scheduling of Self-Suspending Tasks is **Not Sustainable** w.r.t. Execution Times

[Example by Abdeddaïm and Masson, 2012]

Original Job Set

Job set misses no deadline
EDF Scheduling of Self-Suspending Tasks is Not Sustainable w.r.t. Execution Times

[Example by Abdeddaïm and Masson, 2012]

Assume $T_3$ completes one unit earlier
EDF Scheduling of Self-Suspending Tasks is Not Sustainable w.r.t. Execution Times

[Example by Abbeddaïm and Masson, 2012]

Original Job Set

Reducing cost of $T_3$

Assume $T_3$ completes one unit earlier

$T_2$ suspends earlier and interferes more with $T_1$
EDF Scheduling of Self-Suspending Tasks is Not Sustainable w.r.t. Execution Times

[Example by Abdeddaïm and Masson, 2012]

Original Job Set

Reducing cost of $T_3$

Assume $T_3$ completes one unit earlier

Interference causes deadline miss for $T_1$
Abdeddaïm and Masson: EDF scheduling of self-suspending tasks is **not sustainable** w.r.t. execution times.

**Deadline miss!**

**Assume T₃ completes one unit earlier**

**Interference causes deadline miss for T₁**
What About Suspension Times?

[Example by Abdellaïm and Masson, 2012]
What About Suspension Times?

[Example by Abdeddaïm and Masson, 2012]

Original Job Set

Job set misses no deadline
What About Suspension Times?

[Example by Abeddaïm and Masson, 2012]

Original Job Set

Reducing Suspension Time of $T_2$

Assume $T_2$ suspends for only one time unit
What About Suspension Times?

[Example by Abdeddaïm and Masson, 2012]

Original Job Set

Reducing Suspension Time of T2

Assume T2 suspends for only one time unit

T2 suspends less and interferes more with T1
What About Suspension Times?

[Example by Abdeddaïm and Masson, 2012]

Original Job Set

Reducing Suspension Time of $T_2$

Assume $T_2$ suspends for only one time unit

Interference causes deadline miss for $T_1$
Abdeddaïm and Masson: JLFP scheduling of self-suspending tasks is not sustainable with respect to both execution and suspension times.

Assume T_2 suspends for only one time unit.
Is the System Originally Schedulable?

The new job set has the original execution times but misses a deadline.
Is the System Originally Schedulable?

Is the system really schedulable with original execution times?

Reducing Suspension Time of T₂

The new job set has the original execution times but misses a deadline
### Two Possible Interpretations

**If a system is schedulable with original parameters**, then it remains schedulable when assigned better parameters.

<table>
<thead>
<tr>
<th>Baseline System</th>
<th>Sustainability w.r.t. Execution Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>Burns and Baruah, 2008</td>
<td>&quot;For any schedulable job set…&quot;</td>
</tr>
<tr>
<td>Baker and Baruah, 2009</td>
<td>&quot;For any job set of a schedulable task set…”</td>
</tr>
</tbody>
</table>
Did We Pick a Bad Example?
Did We Pick a Bad Example?

NO!
Definitions Need Clarification

**Definition 1 (Sustainable scheduling policy)** Let $A$ denote a scheduling policy. Let $\tau$ denote any sporadic task system that is $A$-schedulable. Let $J$ denote a collection of jobs generated by $\tau$. Scheduling policy $A$ is said to be sustainable if and only if $A$ meets all deadlines when scheduling any collection of jobs obtained from $J$ by changing the parameters of one or more individual jobs in any, some, or all of the following ways: (i) decreased execution requirements; (ii) larger relative deadlines; and (iii) later arrival times with the restriction that successive jobs of any task $\tau_i \in \tau$ arrive at least $T_i$ time units apart.
Definitions Need Clarification

**Definition 1 (Sustainable scheduling policy)** Let $A$ denote a scheduling policy. Let $\tau$ denote any sporadic task system that is $A$-schedulable. Let $J$ denote a collection of jobs generated by $\tau$. Scheduling policy $A$ is said to be sustainable if and only if $A$ meets all deadlines when scheduling any collection of jobs obtained from $J$ by changing the parameters of one or more individual jobs in any, some, or all of the following ways: (i) decreased execution requirements; (ii) larger relative deadlines; and (iii) later arrival times with the restriction that successive jobs of any task $\tau_i \in \tau$ arrive at least $T_i$ time units apart.

1) Assume arbitrary job sets or job sets from a schedulable task set?
Definitions Need Clarification

Definition 1 (Sustainable scheduling policy) Let $A$ denote a scheduling policy. Let $\tau$ denote any sporadic task system that is $A$-schedulable. Let $J$ denote a collection of jobs generated by $\tau$. Scheduling policy $A$ is said to be sustainable if and only if $A$ meets all deadlines when scheduling any collection of jobs obtained from $J$ by changing the parameters of one or more individual jobs in any, some, or all of the following ways: (i) decreased execution requirements; (ii) larger relative deadlines; and (iii) later arrival times with the restriction that successive jobs of any task $\tau_i \in \tau$ arrive at least $T_i$ time units apart.

1) Assume arbitrary job sets or job sets from a schedulable task set?

2) Should other parameters remain fixed, or they are allowed to vary?
Definitions Need Clarification

Definition 1 (Sustainable scheduling policy) Let \( A \) denote a scheduling policy. Let \( \tau \) denote any sporadic task system that is \( A \)-schedulable. Let \( J \) denote a collection of jobs generated by \( \tau \). Scheduling policy \( A \) is said to be sustainable if and only if \( A \) meets all deadlines when scheduling any collection of jobs obtained from \( J \) by changing the parameters of one or more individual jobs in any, some, or all of the following ways: (i) decreased execution requirements; (ii) larger relative deadlines; and (iii) later arrival times with the restriction that successive jobs of any task \( \tau_i \in \tau \) arrive at least \( T_i \) time units apart.

1) Assume arbitrary job sets or job sets from a schedulable task set?
2) Should other parameters remain fixed, or they are allowed to vary?
3) Why are we limited to a specific set of parameters?
This Talk

Ambiguity in the Current Definitions

Strong and Weak Sustainability

Weak Sustainability of Self-Suspending Tasks
This Talk

Ambiguity in the Current Definitions

Strong and Weak Sustainability

Weak Sustainability of Self-Suspending Tasks
Strongly Sustainable w.r.t. Parameter P
Strongly Sustainable w.r.t. Parameter P

If a job set is schedulable, then it remains schedulable for better values of parameter P, assuming all parameters other than P remain constant.
Strongly Sustainable w.r.t. Parameter P

If a job set is \textit{schedulable}, then it \textit{remains schedulable} for better values of parameter P, assuming all parameters other than P remain constant.

This matches the result by Abeddaïm and Masson. The examples show that EDF scheduling of self-suspending tasks is \textit{not strongly sustainable} w.r.t. both execution and suspension times.
What we Know about Sustainability

Strongly Sustainable w.r.t. Parameter $P$

- Allows fixing worst-case values for parameter $P$

vs.

Unsustainable

- All combinations of parameter values must be checked by the analysis
If all parameters are strongly sustainable, the worst-case scenario lies in a single point of the parameter space.
Strong Sustainability: Ideal but Hard to Prove

If all parameters are strongly sustainable, the worst-case scenario lies in a single point of the parameter space.

Simplifies analysis greatly, but may be impossible to prove.

If all parameters are strongly sustainable, the worst-case scenario lies in a single point of the parameter space.
Unsustainability: Difficult to Analyze

All possible combinations of parameters must be checked to determine the worst-case scenario.
All possible combinations of parameters must be checked to determine the worst-case scenario.
How to Find a Middle Ground?
How to Find a Middle Ground?

Core idea: make explicit which parameters are allowed to vary.

- Sustainable parameter $P$
- Set of varying parameters $V$
How to Find a Middle Ground?

Core idea: make explicit which parameters are allowed to vary.

[Diagram: Sustainable parameter P + Set of varying parameters V]

If we have fewer varying parameters, the sustainability result is stronger.
Weakly Sustainable w.r.t. Parameter P and Variable Parameters V
Weakly Sustainable w.r.t. Parameter \( P \) and Variable Parameters \( V \)

If a job set is *schedulable* for all possible combinations of parameter values in \( V \), then it *remains schedulable* for *better values of parameter* \( P \), assuming *all parameters other than* \( P \) *and those in* \( V \) *remain constant.*
Unsustainability: Difficult to Analyze

Interarrival Times
Suspension Times
Execution Times
Weak Sustainability: Search Space Reduction

We can maximize/minimize the sustainable parameter \( P \), as long as the analysis covers all possible values in \( V \).
Weak Sustainability: Search Space Reduction

We can maximize/minimize the sustainable parameter $P$, as long as the analysis covers all possible values in $V$.

Allows for more efficient analysis for policies that are not strongly sustainable.

Interarrival Times

Suspension Times ($V$)

Execution Times ($P$)
This Talk

Ambiguity in the Current Definitions

Strong and Weak Sustainability

Weak Sustainability of Self-Suspending Tasks
This Talk

Ambiguity in the Current Definitions

Strong and Weak Sustainability

Weak Sustainability of Self-Suspending Tasks
EDF scheduling of self-suspending tasks is not (strongly) sustainable w.r.t. both execution and suspension times.
The Policy is Actually Sustainable

EDF scheduling of self-suspending tasks is not (strongly) sustainable w.r.t. both execution and suspension times.

We proved that uniprocessor JLFP scheduling of self-suspending tasks is weakly sustainable w.r.t. execution times and variable suspension times.
The Policy is Actually Sustainable

EDF scheduling of self-suspending tasks is not (strongly) sustainable w.r.t. both execution and suspension times.

We proved that uniprocessor JLFP scheduling of self-suspending tasks is weakly sustainable w.r.t. execution times and variable suspension times.

machine-checked with

formally proven schedulability analysis | PROSA

prosa mpi-sws.org/releases/sustainability
What Else is in the Paper?
What Else is in the Paper?

Formal theory of sustainability
What Else is in the Paper?

Formal theory of sustainability

What really is a job parameter? How to define “better” or “worse”?

We formalize sustainable policy, sustainable analysis and self-sustainable analysis.
What Else is in the Paper?

Formal theory of sustainability
What Else is in the Paper?

Formal theory of sustainability

Composition rules for weak and strong sustainability

\[ A + B = C \]
How to combine sustainability proofs with different values of P and V?

We need certain assumptions on the parameters!
What Else is in the Paper?

- Formal theory of sustainability
- Composition rules for weak and strong sustainability
What Else is in the Paper?

- Formal theory of sustainability
- Composition rules for weak and strong sustainability
- Proof strategy for weak sustainability

Schedule construction with two proof obligations:
(a) service invariant + (b) validity of the new schedule
Takeaways
With the rigor imposed by a proof assistant, we were able to clarify an important concept in real-time scheduling.
Takeaways

With the rigor imposed by a proof assistant, we were able to clarify an important concept in real-time scheduling.

We proposed weak sustainability, which enables more efficient analysis for policies that are not strongly sustainable.
Takeaways

With the rigor imposed by a proof assistant, we were able to clarify an important concept in real-time scheduling.

We proposed weak sustainability, which enables more efficient analysis for policies that are not strongly sustainable.

We proved that uniprocessor JLFP scheduling of self-suspending tasks is weakly sustainable w.r.t. execution times and variable suspension times.

For more details, visit prosa.mpi-sws.org/releases/sustainability