A C-space sensitivity analysis of Earliest Deadline First scheduling

Jean-François Hermant[*] and Laurent George[*]

[*]ECE/LACSC
53 rue de Grenelle
75007 Paris, France

{hermant, lgeorge}@ieee.org

Abstract. This paper presents a sensitivity analysis for the dimensioning of real-time systems in which sporadic tasks are executed according to the preemptive Earliest Deadline First (EDF) scheduling policy. The timeliness constraints of the tasks are expressed in terms of late termination deadlines. A general case is considered, where the task deadlines are independent of the task sporadicity intervals (also called periods). New results for EDF are shown, which enable us to determine the C-space feasibility domain, such that any task set with its worst-case execution times in the C-space domain is feasible with EDF. We show that the C-space domain is convex, a property that can be used to reduce the number of inequalities characterizing the C-space domain.

1 Introduction

This paper considers the problem of correctly dimensioning real-time systems. The correct dimensioning of a real-time system strongly depends on the determination of the task Worst-Case Execution Times (WCETs). Based on the WCETs, a Feasibility Condition (FC) (1), (10), (5) can be established to ensure that the timeliness constraints of all the tasks are always met, regardless of their release times, when they are scheduled by either a fixed or a dynamic priority-driven preemptive scheduling algorithm. The timeliness constraints are expressed in terms of late termination deadlines imposed on the completion times of the tasks. The task model is the classical sporadic model. A sporadic task set $\tau = \{\tau_1, ..., \tau_n\}$ is composed of n sporadic tasks, where a sporadic task τ_i is defined by:

- x_i: its worst-case execution time (WCET).
- T_i: its minimum inter-arrival time (also called, by extension, the period).
- D_i: its relative deadline (a task released at time t must be executed by its absolute deadline $t + D_i$).

In the sequel, we assume the general case where deadlines and periods are independent.

A recent research area called sensitivity analysis aims at providing interesting information on the feasibility of a system when changing task WCETs, task periods (4), or task deadlines (9). This permits, for example, finding a feasible task set, if the current one is not feasible, by modifying the task parameters (WCETs, periods, or deadlines) or determining the impact of an architecture change on the feasibility of a task set (WCET change). In this paper, we are interested in the sensitivity on WCETs. We want to determine the C-space feasibility domain as defined by Bini and Buttazzo (3) when tasks are scheduled with non preemptive Earliest Deadline First. The C-space is a region of n dimensions where each dimension denotes the WCET of a task such that for any vector $X = \{x_1, \ldots, x_n\}$ in the C-space, task set τ is feasible.

In the case of Fixed Priority scheduling, when deadlines are less than or equal to periods, Bini and Di Natale (4) have shown how to compute the maximum expansion factor α applied to all the WCETs of the tasks to remain in the C-space at a reasonable cost (see Section 3), such that $\forall \tau_i \in \tau$, $i = 1 \ldots n$, the WCET of τ_i is $\alpha\, x_i$. They finally propose a parametric equation of the C-space, detailed in Section 3. In the general case, when deadlines and periods are independent, α can be computed by successive iterations, where each iteration requires a pseudo-polynomial time complexity. In this paper, we show how to derive from an analysis of EDF in a time interval $[min(D_1, \ldots, D_n), lcm(T_1, \ldots, T_n))$ the C-space region parametric equation.

The rest of the paper is organized as follows. Section 2 reviews classical concepts for uniprocessor scheduling. Section 3 presents the state of the art in real-time scheduling with a focus on Fixed Priority (FP) scheduling sensitivity analysis and EDF scheduling. In Section 4, we introduce new results on EDF scheduling that can be useful for a sensitivity analysis of the WCETs. We then establish the C-space feasibility region parametric equations. In Section 5, we show in an example how to determine the C-space domain. Finally, we give some conclusion.

2 Concepts and notations

We recall classical results in the uniprocessor context for real-time scheduling.

- A task is said to be non-concrete if its request time is not known in advance. In this paper, we only consider non-concrete request times, since the activation request times are supposed to be unpredictable.
- Given a non-concrete task set, the synchronous scenario corresponds to the scenario where all the tasks are released at the same time.
- EDF is the preemptive version of Earliest Deadline First non-idling scheduling. EDF schedules tasks according to their absolute deadlines: the task with the shortest absolute deadline has the highest priority. Ties are broken arbitrarily.
- FP is a preemptive Fixed-Priority scheduling according to an arbitrary priority assignment.
- For FP, $hp(i)$ denotes the subset of tasks with a priority higher than or equal to that of τ_i except τ_i.

- A task set is said to be valid with a given scheduling policy if and only if no task occurrence ever misses its absolute deadline with this scheduling policy.
- $U = \sum_{i=1}^{n} \frac{x_i}{T_i}$ is the processor utilization factor, i.e., the fraction of processor time spent in the execution of the task set (8). If $U > 1$, then no scheduling algorithm can meet the task deadlines.
- $W_i(t) = x_i + \sum_{\tau_j \in hp(i)} \left\lceil \frac{t}{T_j} \right\rceil x_i$.
- $W(t) = \sum_{j=1}^{n} \left\lceil \frac{t}{T_j} \right\rceil x_j$.
- The processor demand $h(t)$ is the amount of processing time requested by all tasks, whose release times and absolute deadlines are in time interval $[0, t]$ in the synchronous scenario (1), where $\lfloor x \rfloor$ returns the integer part of x. We have for a given task set τ: $h(t) = \sum_{j=1}^{n} h_j(t) x_j$ where $h_j(t) = Max \left\{ 0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\}$
- D_{min} is the minimum deadline $(D_{min} = Min\{D_1, \ldots, D_n\})$.
- P is the least common multiple of the task periods ($P = LCM\{T_1, \ldots, T_n\}$).
- The synchronous scenario corresponds to the scenario where all the tasks are released at the same time (at time 0).

3 State of the art

For Fixed-Priority (FP) scheduling, necessary and sufficient FCs have been proposed, based on the computation of the task worst-case response times (6), (10). The worst-case response time is obtained in the worst-case synchronous scenario and is computed by successive iterations. A task set is then declared feasible if the worst-case response time of any task in the synchronous scenario is less than or equal to its deadline.

In the case of deadlines less than or equal to periods for all tasks, the worst-case response time r_i of a task τ_i is obtained in the synchronous scenario for the first release of τ_i at time 0 and is the solution of the equation (6) $r_i = W_i(r_i)$. r_i is computed by successive iterations and the number of iterations is bounded by $1 + \sum_{\tau_j \in hp(i)} \left\lfloor \frac{D_i}{T_j} \right\rfloor$. The FC has been revisited by Bini and Buttazzo (3), who show that a necessary and sufficient feasibility condition for a task set is: $\exists t \in S$, such that $W_i(t)/t \leq 1$, where $S = \cup_{\tau_j \in hp(i)} \{kTj, k \in N\} \cap [0, D_i]$. For any task τ_i, the times to check correspond to the arrival times of the tasks of higher priority than τ_i in time interval $[0, D_i]$. This feasibility has been improved by Bini and Buttazzo (3), who show how to reduce S. For any task τ_i, they show how to significantly reduce the number of times to check in time interval $[0, D_i]$ to at most 2^{i-1} times instead of $1 + \sum_{\tau_j \in hp(i)} \left\lfloor \frac{D_i}{T_j} \right\rfloor$ times. The C-space can be obtained at an acceptable complexity for a reasonable number of tasks and can be used for a sensitivity analysis. This result can be used to determine the C-space (n dimensions) feasibility region for the WCETs of a sporadic task set such that any vector $X = \{x_1, \ldots, x_n\}$ of WCETs in the C-space region leads to a feasible task set. The C-space region is then defined as follows:

Theorem 1. *(3) Let $\tau = \tau_1, \ldots, \tau_n$ be a set of periodic taks indexed by decreasing priorities. The C-space region when $\forall i, D_i \leq T_i$ is defined as the region such that $\forall X = \{x_1, \ldots, x_n\} \in {\mathbb{R}^+}^n$:*
$\forall i = 1 \ldots n, \exists t \in \mathcal{P}_{i-1}(D_i), x_i + \sum_{j=1}^{i-1} \left\lceil \frac{t}{T_j} \right\rceil x_j$ where $\mathcal{P}_i(t)$ is defined by the recurrent equation:

$$\begin{cases} \mathcal{P}_0(t) = t \\ \mathcal{P}_i(t) = \mathcal{P}_{i-1}(\left\lfloor \frac{t}{T_i} \right\rfloor T_i) \cup \mathcal{P}i-1(t) \end{cases}$$

When deadlines and periods are independent, Tindell et al. (10) show that the worst-case response time of a sporadic task τ_i is not necessarily obtained for the first activation request of τ_i at time 0. The number of activations to consider is $1 + \left\lfloor \frac{L_i}{T_i} \right\rfloor$, where L_i is the length of the worst-case level-τ_i busy period defined by Lehoczky (7) as the longest period of processor activity running tasks of priority higher than or equal to τ_i in the synchronous scenario. It can be shown that $L_i = \sum_{\tau_j \in hp(i) \cup \tau_i} \left\lceil \frac{L_i}{T_j} \right\rceil x_j$. From its definition, L_i is bounded by U.P (5).
In that case, the complexity depends on L_i leading to a pseudo-polynomial time complexity. In such a context, the characterization of the C-space might be very costly. α is computed by iterations, but the computation becomes more and more costly. Indeed, when α increases, the length of the level-τ_i busy period tends towards P as the load utilization tends towards 1. As a conclusion for FP scheduling, sensitivity analysis can be proposed in the case of deadlines less than or equal to periods but not in the general case, because increasing the task WCETs requires the recomputation of the lengths of the busy periods which tend towards P, a potentially exponential length.

For EDF scheduling, Baruah et al. (1) show that a necessary and sufficient feasibility condition is $\forall t \in [0, L), h(t) \leq t$, where L is the length of the first busy period in the synchronous scenario. When $U \leq 1$, L can be computed by successive iterations and is a solution of $L = W(L)$. With this feasibility test, we have the same drawback as with FP in the general case of independent periods and deadlines, as the value of L increases and tends towards P when we compute α by increasing iterations.
We notice that, in both approaches, the dimensioning strongly depends on the values of the task WCETs. We now introduce new results for EDF to determine the C-space feasibility domain.

4 Sensitivity analysis for EDF

This section is divided into two subsections. In Section 4.1, we revisit the classical feasibility condition for EDF based on processor demand and establish new results for the feasibility of a sporadic task set scheduled with EDF. In Section 4.2, we show how to determine the C-space feasibility domain. The C-space region is expressed with parametric equation.

4.1 Revisiting the feasibility condition for EDF

The following lemma is an adaption of (1):

Lemma 1. *Let τ be a task set. τ feasible with preemptive EDF $\Leftrightarrow$*
$Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} \leq 1.$

Proof. The necessary and sufficient feasibility condition for EDF is as follows: Task set τ is feasible with preemptive EDF if and only if $\forall t \in \mathbb{R}^+$, $h(t) \leq t$, which is equivalent to $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} \leq 1$. The condition $U \leq 1$ is clearly necessary as $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} \leq 1 \Rightarrow U \leq 1$. Indeed, $lim_{t \to \infty} \left(Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} \right) = U$

□

We now prove the following theorem, showing how to compute $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\}$:

Theorem 2. $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} = Max \left\{ U, Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \right\}$.

Proof. **1** ◇ Firstly, we show that: $Max \left\{ U, Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \right\} \leq Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\}$.

By definition, we have: $lim_{t \to +\infty} \left\{ \frac{h(t)}{t} \right\} \leq Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\}$, i.e. $U \leq Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\}$.

Furthermore, we have: $Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \leq Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\}$.

It follows that: $Max \left\{ U, Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \right\} \leq Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\}$.

2 ◇ Secondly, we show that: $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} \leq Max \left\{ U, Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \right\}$.

Given that $h(t)$ returns 0 for all $t \in [0, D_{min})$, we have: $\forall t \in [0, P), h(t) \leq Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \times t$.

Furthermore, we have: $\forall (t_1, t_2) \in \mathbb{R}^+ \times \mathbb{R}^+,\ t_2 \geq t_1,\ h(t_2, A) - h(t_1, A) \leq W(t_2 - t_1, A)$.

Consequently, for all $(t, k) \in [0, P) \times \mathbb{N}$, we have: $\frac{h(t+kP,A)}{t+kP} \leq \frac{h(t)+W(kP,A)}{t+kP}$.
Hence, $\frac{h(t+kP,A)}{t+kP} \leq \frac{Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t,A)}{t} \right\} t}{t+kP} + \frac{U\,kP}{t+kP}$, and $\frac{h(t+kP,A)}{t+kP} \leq Max \left\{ U, Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \right\}$.
It follows that $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} \leq Max \left\{ U, Sup_{t \in [D_{min}, P)} \left\{ \frac{h(t)}{t} \right\} \right\}$.

□

We therefore have the following theorem:

Theorem 3. *A task set τ is feasible with premptive EDF* $\Leftrightarrow$ $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\} = Max \left\{ U, Sup_{t \in \mathcal{M}} \left\{ \frac{h(t)}{t} \right\} \right\} \leq 1$*, where* $\mathcal{M} = \bigcup_{j=1}^{n} \left\{ D_j + k_j\, T_j,\ 0 \leq k_j \leq \left\lceil \frac{P - D_j}{T_j} \right\rceil - 1 \right\}$.

Proof. Straightforward from Lemma 1 and Theorem 2.
Set $\mathcal{M} = \bigcup_{j=1}^{n} \left\{ D_j + k_j\, T_j,\ 0 \le k_j \le \left\lceil \frac{P-D_j}{T_j} \right\rceil - 1 \right\}$ corresponds to the deadlines of the tasks in time interval $[D_{min}, P)$ where function $h(t)$ varies. □

Notice that this test is valid for any WCET configuration and will be used to characterize the C-space domain in Section 4.2.

4.2 C-space feasibility domain for EDF

Theorem 4. *C-space domain $\mathcal{D}^{EDF}(\tau) \subset \mathbb{R}^{+n}$ of $X = (x_1, \ldots, x_n)$ is defined as the subset of $X \in \mathbb{R}^{+n}$ such that:*

$$Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{1}{t} \sum_{j=1}^{n} h_j(t) x_j \right\} \le 1,$$

This defines all the task sets feasible with EDF. We have:

$$\mathcal{D}^{EDF}(\tau) = \left\{ X \in \mathbb{R}^{+n}, Max \left(Sup_{t \in \mathcal{M}} \{ \frac{1}{t} \sum_{j=1}^{n} h_j(t) x_j \}, \sum_{j=1}^{n} \frac{x_j}{T_j} \right) \le 1 \right\}$$

where:

$$\mathcal{M} = \bigcup_{j=1}^{n} \left\{ D_j + k_j\, T_j,\ 0 \le k_j \le \left\lceil \frac{P - D_j}{T_j} \right\rceil - 1 \right\}.$$

Proof. Straightforward from theorem 2.

□

From Theorem 3, C-space feasibility domain $\mathcal{D}^{EDF}(\tau)$ is defined by a set of $m+1$ constraints. The first m constraints come from the set of times in $\mathcal{M}$, the $(m+1)^{th}$ constraint comes from the load utilization. We now show how to reduce the times to consider in $\mathcal{M}$, to extract from the first m constraints, the subset of times in $\mathcal{M}$ representing the most constrained inequalities, i.e. times where $Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{h(t)}{t} \right\}$ is obtained. For any time t_i, starting from time t_1 downto t_m, we show how to determine if time t_i should be considered or can be removed from $\mathcal{M}$. We formalize the problem as a linear programming problem P_i for each time t_i we try to maximize the objective function $\sum_{j=1}^{n} h_j(t_i)\, x_j$ taking into account the $m-1$ constraints, $k \neq i$, $\sum_{j=1}^{n} h_j(t_k)\, x_j \le t_k$. We then check if for time t_i, $\sum_{j=1}^{n} h_j(t_i)\, x_j < t_i$. If it is the case, then adding the constraint $\sum_{j=1}^{n} h_j(t_i)\, x_j \le t_i$ for time t_i will bring the same result, i.e $\sum_{j=1}^{n} h_j(t_i)\, x_j < t_i$. Hence t_i can then be removed from $\mathcal{M}$. Otherwise, time t_i must be kept, indeed, $\sum_{j=1}^{n} h_j(t_i)\, x_j \ge t_i$. The constraint $\sum_{j=1}^{n} h_j(t_i)\, x_j \le t_i$ must be taken into account.
We use the $simplex$ algorithm to solve for any time the maximization problem. The simplex algorithm must be applied on convex regions. We can therefore apply it step by step on the times of $\mathcal{M}$ provided that the C-space region obtained for any time t_i is convex (we show this property in this section).

Linear Programming problem

We now give in terms of a linear programming problem, the problem of determining if a time t_i in $\mathcal{M}$ should be kept or not. To solve this linear programming problem, we maximize step by step every function $h(t_i) = \sum_{j=1}^{n} h_j(t_i)\, x_j$, representing an objective function under the following linear constraints:

$$\bigcup_{\substack{k=1 \\ k \neq i}}^{m} \{h(t_k) \leq t_k\}\,.$$

The linear programming problem can be expressed by means of a matrix of m lines and n rows where the value at line p and row q is $h_p(t_q)$ except for line $p = i$ where it is equal to 0. We multiply this matrix with a times vector $X = \{x_1, \ldots, x_n\}$ and check if the result is less than a time vector such that any line $j \neq i$, equals to t_j and line i equals to 0 (the constraint for line i is always met). The linear problem P_i associated to time t_i is as follows:

$$(P_i)\quad \left\{ \begin{array}{l} Maximize\ \sum_{j=1}^{n} h_j(t_i)\, x_j, \\ \begin{bmatrix} h_1(t_1) & h_2(t_1) & \cdots & h_n(t_1) \\ \vdots & \vdots & \vdots & \vdots \\ h_1(t_{i-1}) & h_2(t_{i-1}) & \cdots & h_n(t_{i-1}) \\ 0 & 0 & 0 & 0 \\ h_1(t_{i+1}) & h_2(t_{i+1}) & \cdots & h_n(t_{i+1}) \\ \vdots & \vdots & \vdots & \vdots \\ h_1(t_m) & h_2(t_m) & \cdots & h_n(t_m) \end{bmatrix}_{m,n} \begin{pmatrix} x_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ x_n \end{pmatrix}_n \leq \begin{pmatrix} t_1 \\ \vdots \\ t_{i-1} \\ 0 \\ t_{i+1} \\ \vdots \\ t_m \end{pmatrix}_m \\ x_1 \geq 0,\ x_2 \geq 0,\ \ldots,\ x_n \geq 0. \end{array} \right.$$

Linear Problem 1. Optimisation problem P_i associated to time t_i

C-space domain convexity

Let $\mathcal{E}_i \subset {\mathbb{R}^+}^n$ be the closed region of $X = (x_1, \ldots, x_n)$ meeting the following property :

$$\frac{1}{t_i} \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t_i - D_j}{T_j} \right\rfloor\right\} x_j \leq 1.$$

Hence :

$$\mathcal{E}_i = \left\{ X \in {\mathbb{R}^+}^n,\ \frac{1}{t_i} \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t_i - D_j}{T_j} \right\rfloor\right\} x_j \leq 1 \right\},\ t_i \in \mathcal{M}.$$

Lemma 2.
The set $\mathcal{E}_i \in {\mathbb{R}^+}^n$ is convex. *That is :*

$$\forall (X, X') \in \mathcal{E}_i^2,\ \forall \lambda \in [0,1],\ \lambda\, X + (1-\lambda)\, X' \in \mathcal{E}_i.$$

Proof. By definition, we have:

$$X \in \mathcal{E}_i \Leftrightarrow \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t_i - D_j}{T_j} \right\rfloor\right\} x_j \leq t_i,$$
$$X' \in \mathcal{E}_i \Leftrightarrow \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t_i - D_j}{T_j} \right\rfloor\right\} x'_j \leq t_i.$$

Furthermore, we have $\lambda \in [0, 1]$. It follows:

$$\lambda \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t_i - D_j}{T_j} \right\rfloor\right\} x_j + (1-\lambda) \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t_i - D_j}{T_j} \right\rfloor\right\} x'_j \leq t_i.$$

Hence, we have:

$$\sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t_i - D_j}{T_j} \right\rfloor\right\} \left(\lambda\, x_j + (1-\lambda)\, x'_j\right) \leq t_i.$$

Finally, we have:

$$\lambda\, X + (1-\lambda)\, X' \in \mathcal{E}_i.$$

□

The C-space region $\mathcal{E} \subset \mathbb{R}^{+n}$ of $X = (x_1, \ldots, x_n)$ meeting :

$$Lim_{t \to +\infty} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor\right\} x_j \right\} \leq 1,$$

is denoted:

$$\mathcal{E} = \left\{ X \in \mathbb{R}^{+n},\ Lim_{t \to +\infty} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor\right\} x_j \right\} \leq 1 \right\}.$$

Lemma 3. *The C-space $\mathcal{E} \in \mathbb{R}^{+n}$ is* convex. *That is :*

$$\forall (X, X') \in \mathcal{E}^2,\ \forall \lambda \in [0,1],\ \lambda\, X + (1-\lambda)\, X' \in \mathcal{E}.$$

Proof. By definition, we have:

$$Lim_{t \to +\infty} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor\right\} x_j \right\} = \sum_{j=1}^{n} \frac{x_j}{T_j}.$$

Furthermore, we have: $X \in \mathcal{E} \Leftrightarrow \sum_{j=1}^{n} \frac{x_j}{T_j} \leq 1$ and $X' \in \mathcal{E} \Leftrightarrow \sum_{j=1}^{n} \frac{x'_j}{T_j} \leq 1$.
By hypothesis, we have:$\lambda \in [0, 1]$.It follows: $\lambda \sum_{j=1}^{n} \frac{x_j}{T_j} + (1-\lambda) \sum_{j=1}^{n} \frac{x'_j}{T_j} \leq 1$.
Hence, we have:

$$\sum_{j=1}^{n} \frac{\lambda\, x_j + (1-\lambda)\, x'_j}{T_j} \leq 1.$$

Finally, we have:

$$\lambda\, X + (1-\lambda)\, X' \in \mathcal{E}.$$

□

The C-space feasibility domain $\mathcal{D}^{EDF}(\tau) \subset \mathbb{R}^{+n}$ of $X = (x_1, \ldots, x_n)$ meeting the following inequation:

$$Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max \left\{ 0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\} \leq 1,$$

is denoted:

$$\mathcal{D}^{EDF}(\tau) = \left\{ X \in \mathbb{R}^{+n},\ Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max \left\{ 0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\} \leq 1 \right\}.$$

Corollary 1. *The C-space region* $\mathcal{D}^{EDF}(\tau) \in \mathbb{R}^{+n}$ *is* convex. *That is :*

$$\forall (X, X') \in \mathcal{D}^{EDF}(\tau) \times \mathcal{D}^{EDF}(\tau),\ \forall \lambda \in [0,1],\ \lambda\, X + (1-\lambda)\, X' \in \mathcal{D}^{EDF}(\tau).$$

Proof. From theorem 4, we have:

$$Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max \left\{ 0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\}$$

$$=$$

$$Sup_{t \in [D_{min},\, P[\cup \{+\infty\}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max \left\{ 0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\}$$

$$=$$

$$Max \left\{ Sup_{t \in \mathcal{M}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max \left\{ 0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\}, \sum_{j=1}^{n} \frac{x_j}{T_j} \right\}.$$

By definition, we have:

$$\mathcal{D}^{EDF}(\tau) = \left\{ X \in \mathbb{R}^{+n},\ Sup_{t \in \mathbb{R}^{+*}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max \left\{ 0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\} \leq 1 \right\}.$$

It follows:

$$\mathcal{D}^{EDF}(\tau) = \left\{ X \in \mathbb{R}^{+n}, Sup_{t \in \mathcal{M}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\} \leq 1 \wedge \sum_{j=1}^{n} \frac{x_j}{T_j} \leq 1 \right\}.$$

Therefore we have:

$$\mathcal{D}^{EDF}(\tau) = \left(\bigcap_{i=1}^{m} \mathcal{E}_i \right) \bigcap \mathcal{E}.$$

The intersection of a finite number of convex C-space regions in $\mathbb{R}^{+n}$ is a convex region in $\mathbb{R}^{+n}$.

□

Corollary 2. *The C-space feasibility domain* $\mathcal{D}^{EDF}(\tau)$ *is the intersection of a finite number convex and closed regions. It is therefore a* convex polytope *in* $\mathbb{R}^{+n}$.
Furthermore, the C-space domain $\mathcal{D}^{EDF}(\tau)$ *is a closed* convex polytope. *It is therefore* convex polyhedra *in* $\mathbb{R}^{+n}$.

5 Numerical applications

Let us consider a sporadic task set $\tau = \{\tau_1, \tau_2, \tau_3\}$, composed of three non concrete tasks where for any task τ_i, T_i and D_i are fixed and $x_i \in \mathbb{R}^+$ the WCET of the task is a parameter.
$\tau_1 : (x_1, T_1, D_1) = (x_1, 7, 5)$, $\tau_2 : (x_2, T_2, D_2) = (x_2, 11, 7)$ and $\tau_3 : (x_3, T_3, D_3) = (x_3, 13, 10)$.
We have with this example: $P = 1001$ and $D_{min} = 5$. From Theorem 3, we must consider the set $\mathcal{M}$ of times for the computation of $h(t)$ in time interval $[5, 1001[$ where $\mathcal{M}$ is given by:

$$\begin{aligned}\mathcal{M} = &\{5 + 7\,k_1,\ k_1 \in \{0, \ldots, 142\}\} \cup \\ &\{7 + 11\,k_2,\ k_2 \in \{0, \ldots, 90\}\} \cup \\ &\{10 + 13\,k_3,\ k_3 \in \{0, \ldots, 76\}\}.\end{aligned}$$

In this example, we have $m = Card(\mathcal{M}) = 281$ times in $\mathcal{M}$.
We recall that the C-space feasibility domain $\mathcal{D}^{EDF}(\tau)$ for EDF is defined by a set of $m + 1$ linear constraints:

$$\mathcal{D}^{EDF}(\tau) = \left\{ X \in \mathbb{R}^{+n},\ Sup_{t \in \mathcal{M}} \left\{ \frac{1}{t} \sum_{j=1}^{n} Max\left\{0, 1 + \left\lfloor \frac{t - D_j}{T_j} \right\rfloor \right\} x_j \right\} \leq 1 \wedge \sum_{j=1}^{n} \frac{x_j}{T_j} \leq 1 \right\}.$$

Applying the simplex algorithm on the Linear Problem 1, for any time $t_i \in \mathcal{M}$, starting from time t_1 down to time t_m, we obtain the following subset S_1 of times in $\mathcal{M}$ maximizing h(t) for any vector of durations $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^{+n}$. We solve this problem with the classical simplex algorithm implemented in the Maple 11 computer algebra system.

$$S_1 = \{5, 7, 10, 12, 19, 40, 62\} \subseteq \mathcal{M}.$$

We have:

$$Sup_{t\in\mathbb{R}^{+*}}\left\{\frac{h(t)}{t}\right\} = Max\left\{\frac{x_1}{5}, \frac{x_1+x_2}{7}, \frac{x_1+x_2+x_3}{10}, \frac{2\,x_1+x_2+x_3}{12}, \frac{3\,x_1+2\,x_2+x_3}{19}, \frac{6\,x_1+4\,x_2+3\,x_3}{40}, \frac{9\,x_1+6\,x_2+5\,x_3}{62}\right\}.$$

As:

$$\begin{cases} x_1+x_2 \le 7 \\ 2\,x_1+x_2+x_3 \le 12 \end{cases} \Rightarrow\ 3\,x_1+2\,x_2+x_3 \le 19,$$

$$\begin{cases} x_1+x_2+x_3 \le 10 \\ 2\,x_1+x_2+x_3 \le 12 \\ 6\,x_1+4\,x_2+3\,x_3 \le 40 \end{cases} \Rightarrow\ 9\,x_1+6\,x_2+5\,x_3 \le 62.$$

We can still reduce the times to test to S_2, valid for any configuration of X in the C-space domain where $S_2 = \{5, 7, 10, 12, 40\} \subseteq S_1$.
Finally, if we consider $h(t)/t$ for every times in S_2, and from Theorem 3 we have:
$Sup_{t\in\mathbb{R}^{+*}}\left\{\frac{h(t)}{t}\right\} = Max\left\{\frac{x_1}{5}, \frac{x_1+x_2}{7}, \frac{x_1+x_2+x_3}{10}, \frac{2\,x_1+x_2+x_3}{12}, \frac{6\,x_1+4\,x_2+3\,x_3}{40}, \frac{x_1}{7}+\frac{x_2}{11}+\frac{x_3}{13}\right\}$.

Hence,

$$\mathcal{D}^{EDF}(\tau) = \left\{X \in \mathbb{R}^3,\ Sup_{t\in\mathbb{R}^{+*}}\left\{\frac{h(t\,;\tau)}{t}\right\} \le 1\right\},$$

$$\mathcal{D}^{EDF}(\tau) = \begin{cases} 0 \le |x_1| \le 5, \\ 0 \le |x_2| \le 7 - |x_1|, \\ 0 \le |x_3| \le \\ Min\left\{10-|x_1|-|x_2|, 12-2\,|x_1|-|x_2|, \frac{1}{3}(40-6\,|x_1|-4\,|x_2|), \frac{1}{77}(1001-143|x_1|-91|x_2|)\right\}. \end{cases}$$

We show in figure 1 a graphical representation of the C-space obtained with our example.

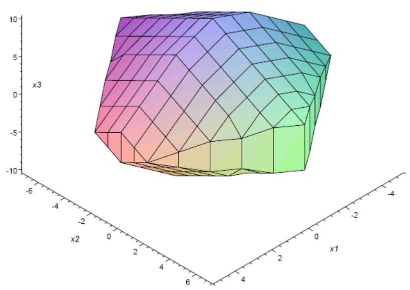

FIG. 1 – *C-space feasibility domain* $\mathcal{D}^{EDF}(\tau)$

6 Conclusion

In this paper, we have presented new results for a sensitivity analysis of preemptive EDF. We have considered sporadic tasks with independent periods and deadlines. Our goal was to express the C-space feasibility domain with parametric equations. We have also shown that the C-space domain can be obtained from an analysis of EDF in a time interval of duration bounded by the least common multiple of the task periods. From this analysis, a linear programming problem is identified and solved with the simplex algorithm. We have shown on an example that this enables us to significantly reduce the complexity of the C-space domain equation.

References

[1] S. Baruah, R. Howell, and L. Rosier. Algorithms and complexity concerning the preemptive scheduling of periodic real-time tasks on one processor. *Real-Time Systems*, Vol. 2, pp. 301-324, 1990.

[2] S. Baruah, A. K. Mok, and L. Rosier. Preemptively scheduling hard real-time sporadic tasks on one processor. *Proceedings of the 11th Real-Time Systems Symposium*, pp. 182-190, 1990.

[3] E. Bini and G. Buttazzo. Schedulability Analysis of Periodic Fixed Priority Systems. *IEEE Transactions On Computers, Vol. 53, No. 11*, Nov.2004.

[4] E. Bini, M. Di Natale, and G. Buttazzo. Sensitivity Analysis for Fixed-Priority Real-Time Systems. *Proceedings of the 18th Euromicro Conference on Real-Time Systems (ECRTS'06)*, Dresden, Germany July 5-7, 2006.

[5] L. George, N. Rivierre, and M. Spuri. Preemptive and non-preemptive scheduling real-time uniprocessor scheduling. *INRIA Research Report*, No. 2966, September 1996.

[6] M. Joseph and P. Pandya. Finding response times in a real-time system. *BCS Comp. Jour.*, 29(5), pp. 390-395,, 1986.

[7] J. Lehoczky. Fixed priority scheduling of periodic task sets with arbitrary deadlines. *Proceedings 11th IEEE Real-Time Systems Symposium*, pp 201-209, Dec. Lake Buena Vista, FL, USA, 1990.

[8] L. C. Liu and W. Layland. Scheduling algorithms for multi-programming in a hard real time environment. *Journal of ACM*, Vol. 20, No 1, pp. 46-61, January 1973.

[9] I. R. Patricia Balbastre and A. Crespo. Optimal deadline assignment for periodic real-time tasks in dynamic priority systems. *Proceedings of the 18th Euromicro Conference on Real-Time Systems (ECRTS'06)*, Dresden, Germany July 5-7, 2006.

[10] K. Tindell, A. Burns, and A. J. Wellings. Analysis of hard real-time communications. *Real-Time Systems*, Vol. 9, pp. 147-171, 1995.