Improving the Analysis of GPC in Real-Time Calculus

Yue Tang[1], Yuming Jiang[2], and Nan Guan[1]

[1] The Hong Kong Polytechnic University, Hong Kong
[2] Norwegian University of Science and Technology, Norway

Abstract. Real-Time Calculus (RTC) is a framework for modeling and performance analysis of real-time networked systems. In RTC, workload and resources are modeled as arrival and service curves, and processing semantics are modeled by abstract components. Greedy Processing Component (GPC) is one of the fundamental abstract components in RTC, which processes incoming events in a greedy fashion as long as there are available resources. The relations between inputs and outputs of GPC have been established, which are consistent with its behaviors. In this paper, we first revise the original proof of calculating output curves in GPC, and then propose a new method to obtain tighter output arrival curves. Experiment results show that the precision of output arrival curves can be improved by our method compared with the original calculation and existing work.

Keywords: RTC · GPC · Output arrival curves

1 Introduction

Real-Time Calculus (RTC), originated in Network Calculus, is a framework for modeling and performance analysis of real-time networked embedded systems. RTC uses arrival and service curves to model workload and resource, and the performance analysis is mainly based on min-plus and max-plus algebra. Compared with the traditional real-time scheduling theory, RTC is more expressive and can model a much wider range of realistic systems due to usage of much more general workload and resource models. At the same time, the models and analysis techniques of RTC generate closed-form analytical results, thus having higher analysis efficiency, compared to state-based modeling and analysis techniques such as model checking [2], [7]. All these advantages make RTC draw much attention from both real-time community and industry.

Greedy Processing Component (GPC) is one of the fundamental abstract components in RTC, which processes input events in a greedy fashion in FIFO order, as long as there are available resources. GPCs can be connected into networks to model the behaviors of real-time systems, and one of the typical scenarios is fixed-priority scheduling, where the remaining resource of one GPC related to a higher-priority task is used as input resource of the GPC related to a lower-priority task.

The analysis of GPC has been consistently studied and improved since it was proposed in [1, 5, 6]. The improvement involves two aspects: efficiency and precision. For efficiency, [9] proved that the output curves can be calculated with a finite length of input curves, thus greatly shortening the computation time for output curves. This work was further generalized in [10], which eliminates the dependency of Finitary RTC on the semantics of GPC and applied Finitary RTC to the level of RTC-operators. For precision, [8] proposed to reduce the unused resource from total resource before calculating the output arrival curves, which improves the precision of output arrival curves.

Contributions. In this work, we improve the analysis of GPC in two aspects. First, we revise the proof of output curves in GPC and complement the missing deduction steps, which further clarifies the correctness of the original results. Second, we propose a new method for calculating output arrival curves, which generates more precise results based on the connection between output arrival curves and the number of accumulated events, rather than focusing on the interaction of between arrival and service curves as in [8].

2 RTC Basics

2.1 Arrival and Service Curves

RTC uses arrival curves and service curves to describe timing properties of event streams and available resource.

Definition 1 (Arrival Curve). *Let $R[s, t)$ denote the total number of arrived events in time interval $[s, t)$[3], where s and t are two arbitrary non-negative real numbers. Then, the corresponding upper and lower arrival curves are denoted as α^u and α^l, respectively, and satisfy:*

$$\forall s < t, \quad \alpha^l(t - s) \leq R[s, t) \leq \alpha^u(t - s),$$

where $\alpha^u(0) = \alpha^l(0) = 0$.

Generally we assume that α^u is concave satisfying $\alpha^u(t) + \alpha^u(s) \geq \alpha^u(t+s)$.

Definition 2 (Service Curve). *Let $C[s, t)$ denote the total available resource in time interval $[s, t)$[4]. Then, the corresponding upper and lower service curves are denoted as β^u and β^l, respectively, and satisfy:*

$$\forall s < t, \quad \beta^l(t - s) \leq C[s, t) \leq \beta^u(t - s),$$

where $\beta^u(0) = \beta^l(0) = 0$.

In our work, we adopt the PJD workload model for arrival curve and TDMA model for service curve in [4] for easy understanding. In detail, the input arrival

[3] $R[s, t) = -R[t, s), R[s, s) = 0$
[4] $C[s, t) = -C[t, s), C[s, s) = 0$

curves are characterized by (p, j, d), where p denotes the period, j the jitter, and d the minimum inter-arrival distance of events in the modeled stream:

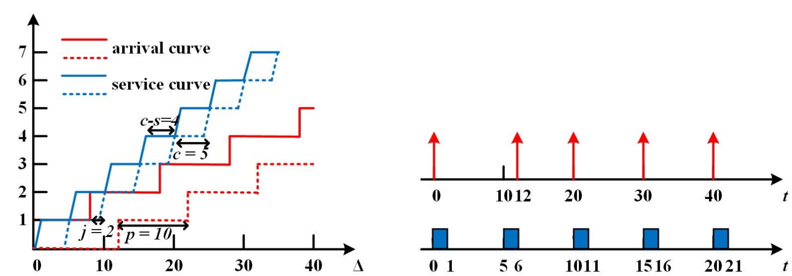
$$\alpha^u(\Delta) = \min\left(\left\lceil \frac{\Delta + j}{p} \right\rceil, \left\lceil \frac{\Delta}{d} \right\rceil\right), \quad \alpha^l(\Delta) = \left\lfloor \frac{\Delta - j}{p} \right\rfloor.$$

The service curves are characterized by (s, c, b):

$$\beta^u(\Delta) = \left(\left\lfloor \frac{\Delta}{c} \right\rfloor \cdot s + \min(\Delta \bmod c, s)\right) \cdot b, \beta^l(\Delta) = \left(\left\lfloor \frac{\Delta'}{c} \right\rfloor \cdot s + \min(\Delta' \bmod c, s)\right) \cdot b$$

, where $\Delta' = \max(\Delta - c + s, 0)$.

Fig. 1(a) shows an example of arrival curve with $p = 10, j = 2, d = 0$ and service curve $s = 1, c = 5, b = 1$, and Fig. 1(b) shows a possible sequence of workload and TDMA resource corresponding to Fig. 1(a).



(a) Input arrival/service curve.

(b) A possible event stream and TDMA resource

Fig. 1. An example for inputs

2.2 Greedy Processing Component (GPC)

In this paper, we focus on one of the widely used abstract components in RTC called Greedy Processing Component (GPC). A GPC processes events from the input event stream in a greedy fashion, as long as it complies with the availability of resource. If we use $R'[s, t)$ and $C'[s, t)$ to describe the accumulated number of processed events and remaining resources during the time interval $[s, t)$, then it satisfies

$$R'[s, t) = C[s, t) - C'[s, t)$$
$$C'[s, t) = \sup_{s \le u \le t} \{C[s, u) - R[s, u) - B(s), 0\}$$

The output event stream produced by GPC is described by arrival curve α'^u, α'^l, and the remaining resource is described by service curve β'^u, β'^l.

$$\alpha'^u = \min((\alpha^u \otimes \beta^u) \oslash \beta^l, \beta^u),$$

$$\alpha'^l = \min((\alpha^l \oslash \beta^u) \otimes \beta^l, \beta^l),$$
$$\beta'^u = (\beta^u - \alpha^l)\overline{\oslash}0,$$
$$\beta'^l = (\beta^l - \alpha^u)\overline{\otimes}0,$$

where

$$(f \otimes g)(\Delta) = \inf_{0\leq\lambda\leq\Delta}\{f(\Delta-\lambda)+g(\lambda)\},$$
$$(f \overline{\otimes} g)(\Delta) = \sup_{0\leq\lambda\leq\Delta}\{f(\Delta-\lambda)+g(\lambda)\},$$
$$(f \oslash g)(\Delta) = \sup_{\lambda\geq 0}\{f(\Delta+\lambda)-g(\lambda)\},$$
$$(f \overline{\oslash} g)(\Delta) = \inf_{\lambda\geq 0}\{f(\Delta+\lambda)-g(\lambda)\}.$$

The maximum delay d_{max} experienced by any event on the event stream and the amount of events in the input buffer b_{max}, i.e., the backlog, are respectively bounded by

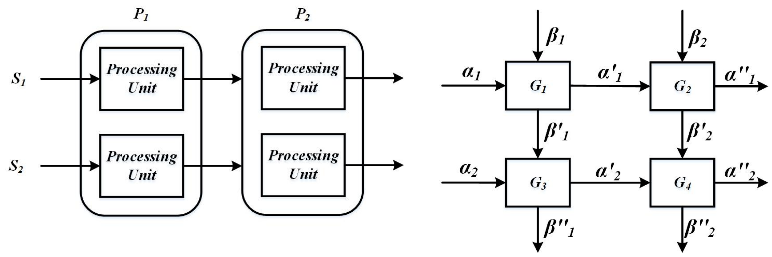
$$d_{max} \leq \sup_{\lambda\geq 0}\left\{\inf\{\delta \geq 0 : \alpha^u(\lambda) \leq \beta^l(\lambda+\delta)\}\right\} = Del(\alpha^u, \beta^l)$$
$$b_{max} \leq \sup_{\lambda\geq 0}\{\alpha^u(\lambda) - \beta^l(\lambda)\} = Buf(\alpha^u, \beta^l)$$

GPCs can be connected together to model systems with various scheduling and arbitration policies. One of the typical scenarios is to model fixed-priority scheduling, where the resource allocation for tasks with different priorities is described by the resource stream direction.

An example. Suppose two event streams S_1, S_2 are processed on two successive processors. S_1 has higher priority than S_2 and the execution of S_2 will be preempted by S_1 as long as an event of S_1 arrives on each processor. Such a system (Fig. 2 (a)) can be modeled by a GPC network (Fig. 2 (b)). The processing of each stream on each processor is modeled as a GPC component. The output arrival curves of G_1 are the input arrival curves of G_2, corresponding to the system behavior that the events of S_1 completing execution on P_1 are further processed on P_2. And the remaining service curve of $G_1(G_2)$ is the input service curve $G_3(G_4)$, since S_1 has higher priority on both processors.

3 Revised Proof of Output Curves

Although how to calculate bounds for output curves in GPC has been proved in [4], some important deduction steps are missing. In detail, the process of specifying the numerical relations between some critical parameters is too simple to be convincing. In this section, we add these missing deduction parts and give revised proof of calculating output curves in GPC.



(a) A networked system. (b) A GPC network

Fig. 2. An example for GPC network.

Theorem 1. *Given a GPC with arrival curve α^u, α^l and service curve β^u, β^l, then its remaining service curves are bounded by*

$$\beta'^u = ((\beta^u - \alpha^l)\overline{\oslash}0)^+$$
$$\beta'^l = (\beta^l - \alpha^u)\overline{\otimes}0$$

where for a function $f(\Delta)$, $(f(\Delta))^+ = max(f(\Delta), 0)$.

Proof. (1) We first prove β'^l.

Suppose p is an arbitrarily small time such that the backlog satisfies $B(p) = 0$.

Then for all $p \le s \le t$,

$$\begin{aligned}
&C'[s,t) \\
&= C'[p,t) - C'[p,s) \\
&= \sup_{p \le a \le t} \{C[p,a) - R[p,a)\}^+ - \sup_{p \le b \le s} \{C[p,b) - R[p,b)\}^+
\end{aligned}$$

Since $C[p,p) = R[p,p) = C[p,p) - R[p,p) = 0$, the suprema are nonnegative and we have

$$\begin{aligned}
&C'[s,t) \\
&= \sup_{p \le a \le t} \{C[p,a) - R[p,a)\} - \sup_{p \le b \le s} \{C[p,b) - R[p,b)\} \\
&= \inf_{p \le b \le s} \{ \sup_{p \le a \le t} \{C[b,a) - R[b,a)\}\} \\
&= \inf_{p \le b \le s} \{max\{ \sup_{b \le a \le t} \{C[b,a) - R[b,a)\}, \sup_{p \le a \le b} \{C[b,a) - R[b,a)\}\}\}
\end{aligned}$$

Let $\chi_1(b) = \sup_{b \le a \le t} \{C[b,a) - R[b,a)\} = max\{C[b,b) - R[b,b), C[b,b+1) - R[b,b+1), ..., C[b,t) - R[b,t)\} \ge 0^5$,

[5] For ease of presentation we assume s, t, a, b, p to be integer in following proofs.

$\chi_2(b) = \sup_{p \le a \le b}\{C[b,a) - R[b,a)\} = max\{C[b,p) - R[b,p), C[b,p+1) - R[b,p+1),...,C[b,b-1) - R[b,b-1), C[b,b) - R[b,b)\} \ge 0$.

Next we prove that[6]

$C'[s,t) = \inf_{p \le b \le s}\{max\{\chi_1(b), \chi_2(b)\}\} = \inf_{p \le b \le s}\{\chi_1(b)\}$.

We consider two cases:

1) For any $i \in [p,s]$, $\chi_1(i) \ge \chi_2(i)$, then

$C'[s,t) = \inf_{p \le b \le s}\{max\{\chi_1(b), \chi_2(b)\} = \inf_{p \le b \le s}\{\chi_1(b)\}$,

then $C'[s,t) = \inf_{p \le b \le s}\{\chi_1(b)\}$.

2) There exists at least one $i \in [p,s]$ that $\chi_1(i) < \chi_2(i)$, that is, there exists one $x \in [p,i]$ that

$C[b,a) - R[b,a) = C[i,x) - R[i,x) = max\{\chi_1(i), \chi_2(i)\} = max\{C[i,p) - R[i,p),...,C[i,x-1) - R[i,x-1),...,C[i,x+1) - R[i,x+1),...,C[i,t) - R[i,t)\}$

Then we have

$$R[x,i) - C[x,i) \ge R[p,i) - C[p,i) \Rightarrow C[p,x) \ge R[p,x)$$

$$R[x,i) - C[x,i) \ge R[p+1,i) - C[p+1,i) \Rightarrow C[p+1,x) \ge R[p+1,x)$$

$$...$$

$$R[x,i) - C[x,i) \ge R[x-1,i) - C[x-1,i) \Rightarrow C[x-1,x) \ge R[x-1,x)$$

$$R[x,i) - C[x,i) \ge R[x+1,i) - C[x+1,i) \Rightarrow R[x,x+1) \ge C[x,x+1)$$

$$R[x,i) - C[x,i) \ge C[i,i+1) - R[i,i+1) \Rightarrow R[x,i+1) \ge C[x,i+1)$$

$$R[x,i) - C[x,i) \ge C[i,t) - R[i,t) \Rightarrow R[x,t) \ge C[x,t)$$

Then when $b = x$, we have

$\chi_1(x) = max\{C[x,x) - R[x,x), C[x,x+1) - R[x,x+1),...,C[x,t) - R[x,t)\} = 0$,

$\chi_2(x) = max\{C[x,p) - R[x,p), C[x,p+1) - R[x,p+1),...,C[x,b-1) - R[x,b-1), C[x,x) - R[x,x)\} = 0$.

So $max\{\chi_1(x), \chi_2(x)\} = 0$.

Then

$C'[s,t) = \inf_{p \le b \le s}\{max\{\chi_1(b), \chi_2(b)\}\} = max\{\chi_1(x), \chi_2(x)\} = 0$ (since $C'[s,t) \ge 0$).

On the other hand,

$\inf_{p \le b \le s}\{\chi_1(b)\} = min\{\chi_1(p), \chi_1(p+1),...,\chi_1(x),...,\chi_1(s)\} = 0$, then we have $C'[s,t) = \inf_{p \le b \le s}\{\chi_1(b)\} = 0$.

So in both cases, $C'[s,t) = \inf_{p \le b \le s}\{\chi_1(b)\}$, which implies that removing the cases when $a < b$ does not influence the final result. Note that $a \ge b$ is a consequence of above deduction, but not simply a direct result of $s \le t$ as implied in [4].

[6] This part is just briefly explained as $'a \ge b$ since $t \ge s'$ in the existing proof [4].

Then we continue to lower bound $C'[s,t)$.

$$
\begin{aligned}
&C'[s,t) \\
&= \inf_{p\le b\le s}\{\sup_{b\le a\le t}\{C[b,a)-R[b,a)\} \\
&= \inf_{p\le b\le s}\{\sup_{0\le a-b\le t-b}\{C[b,a)-R[b,a)\} \\
&\ge \inf_{p\le b\le s}\{\sup_{0\le \lambda\le t-b}\{\beta^l(\lambda)-\alpha^u(\lambda)\} \\
&\ge \sup_{0\le \lambda\le t-s}\{\beta^l(\lambda)-\alpha^u(\lambda)\} \\
&= (\beta^l-\alpha^u)\overline{\otimes}0
\end{aligned}
$$

(2) Next we prove β'^u.

$$
\begin{aligned}
&C'[s,t) \\
&= C'[p,t)-C'[p,s) \\
&= \sup_{p\le a\le t}\{\inf_{p\le b\le s}\{C[b,a)-R[b,a)\}\} \\
&= max\{\sup_{p\le a\le s}\{\inf_{p\le b\le s}\{C[b,a)-R[b,a)\}\}, \sup_{s\le a\le t}\{\inf_{p\le b\le s}\{C[b,a)-R[b,a)\}\}\} \\
&= max\{\sup_{p\le a\le s}\{min\{\inf_{a\le b\le s}\{C[b,a)-R[b,a)\}, \inf_{b\le a\le s}\{C[b,a)-R[b,a)\}\}\}, \sup_{s\le a\le t}\{\inf_{p\le b\le s}\{C[b,a)-R[b,a)\}\}\}
\end{aligned}
$$

Similar as above, we define $\psi = \sup_{p\le a\le s}\{\inf_{p\le b\le s}\{C[b,a)-R[b,a)\}\}$, $\psi_1(a) = \inf_{a\le b\le s}\{C[b,a)-R[b,a)\}$, and $\psi_2(a) = \inf_{b\le a\le s}\{C[b,a)-R[b,a)\}$. Next we prove $\psi \le 0$.

For any $a \in [p,s]$, we consider three cases:

1) $\psi_1(a) = \psi_2(a)$. Then there must exist $b_{inf} = a$ such that $C[b_{inf},a) - R[b_{inf},a) = 0 = min\{\psi_1(a),\psi_2(a)\}$.

2) $\psi_1(a) > \psi_2(a)$. Then there must exist $b_{inf} < a$ such that
$C[b_{inf},a) - R[b_{inf},a) = min\{\psi_1(a),\psi_2(a)\} < \psi_1(a) \le 0$.

3) $\psi_1(a) < \psi_2(a)$. Then there must exist $b_{inf} > a$ such that
$C[b_{inf},a) - R[b_{inf},a) = min\{\psi_1(a),\psi_2(a)\} < \psi_2(a) \le 0$.

Then combining the above three cases, for each $a \in [p,s]$, $min\{\psi_1(a),\psi_2(a)\} \le 0$, so $\psi \le 0$.

By now we have

$$
\begin{aligned}
&C'[s,t) \\
&= \sup_{p\le a\le t}\{\inf_{p\le b\le s}\{C[b,a)-R[b,a)\}\} \\
&= \sup_{s\le a\le t}\{\inf_{p\le b\le s}\{C[b,a)-R[b,a)\}\}^+ \\
&= \sup_{s\le a\le t}\{\inf_{a-s\le a-b\le a-p}\{C[b,a)-R[b,a)\}\}^+
\end{aligned}
$$

Note that $a \ge b$ is a consequence of $a \ge s$, but not a direct result of $s \le t$.

Then with substitution λ we have

$$\begin{aligned} C'[s,t] &\le \sup_{s\le a\le t}\{\inf_{a-s\le a-b\le a-p}\{\beta^u(\lambda)-\alpha^l(\lambda)\}\}^+ \\ &\le \sup_{s\le a\le t}\{\inf_{t-s\le a-b\le p}\{\beta^u(\lambda)-\alpha^l(\lambda)\}\}^+ \\ &= \inf_{t-s\le\lambda\le p}\{\beta^u(\lambda)-\alpha^l(\lambda)\}\}^+ \\ &= \inf_{t-s\le\lambda}\{\beta^u(\lambda)-\alpha^l(\lambda)\}\}^+ \\ &= ((\beta^u-\alpha^l)\overline{\oslash}0)^+ \end{aligned}$$

□

Theorem 2. *Given a GPC with arrival curve α^u, α^l and service curve β^u, β^l, then its output arrival curves are bounded by*

$$\alpha'^u = \min((\alpha^u \otimes \beta^u) \oslash \beta^l, \beta^u)$$
$$\alpha'^l = \min((\alpha^l \oslash \beta^u) \otimes \beta^l, \beta^l)$$

Proof. The proofs are presented in the Appendix. □

An example. We take the calculation of β'^l as an example to show $C'[s,t] = \inf_{p\le b\le s}\{\chi_1(b)\}$. Let $p=0$,$s=3$, $t=5$. Then the value of $C'[s,t]$ with different values of a, b is shown in Table 1.

	$b=0$	$b=1$	$b=2$	$b=3$
$a=0$	0	$R[0,1)-C[0,1)$	$R[0,2)-C[0,2)$	$R[0,3)-C[0,3)$
$a=1$	$C[0,1)-R[0,1)$	0	$R[1,2)-C[1,2)$	$R[1,3)-C[1,3)$
$a=2$	$C[0,2)-R[0,2)$	$C[1,2)-R[1,2)$	0	$R[2,3)-C[2,3)$
$a=3$	$C[0,3)-R[0,3)$	$C[1,3)-R[1,3)$	$C[2,3)-R[2,3)$	0
$a=4$	$C[0,4)-C[0,4)$	$C[1,4)-R[1,4)$	$C[2,4)-R[2,4)$	$C[3,4)-R[3,4)$
$a=5$	$C[0,5)-R[0,5)$	$C[1,5)-R[1,5)$	$C[2,5)-R[2,5)$	$C[3,5)-R[3,5)$

Table 1. An example for part of Theorem 1.

Assume that when $b=2$, $\chi_1(2) < \chi_2(2)$ and $\chi_2(2) = R[1,2)-C[1,2)$. Then we have

$$R[1,2)-C[1,2) \ge R[0,2)-C[0,2) \Rightarrow C[0,1) \ge R[0,1)$$

$$R[1,2)-C[1,2) \ge C[2,3)-R[2,3) \Rightarrow R[1,3) \ge C[1,3)$$

$$R[1,2)-C[1,2) \ge C[2,4)-R[2,4) \Rightarrow R[1,4) \ge C[1,4)$$

$$R[1,2) - C[1,2) \geq C[2,5) - R[2,5) \Rightarrow R[1,5) \geq C[1,5)$$

Then when $b = 1$, $max\{\chi_1(1), \chi_2(1)\} = \chi_1(1) = 0$, and
$C'[0,3) = \inf_{0 \leq b \leq 3} \{max\{\chi_1(b), \chi_2(b)\}\} = 0 = min\{\chi_1(0), \chi_1(1), \chi_1(2), \chi_1(3)\} = \inf_{0 \leq b \leq 3} \{\chi_1(b)\}$.

4 Improving Output Arrival Curves

In this section, we adopt the idea in [3] and derive a new upper bound for output arrival curves, which combined with the original result in [4] generates a tighter upper bound, as shown in Section 5.

Lemma 1. *Given an event stream with input function $R(t)$, output function $R'(t)$, arrival curve α^u, α^l, service curve β^u, β^l, the output arrival curve of a GPC is upper bounded by $\alpha'^u(\Delta) = \alpha^u(\Delta) + Buf(\alpha^u, \beta^l) - \alpha^u(0^+)$.*

Proof. We use $B(s)$ to denote the backlog at time s. For all $s \leq t$, we have

$$B(t) - B(s) = R[s,t) - R'[s,t)$$

Suppose p is an arbitrarily small time satisfying $B(p) = 0$, then substituting t with p gives

$$B(s) = R[p,s) - R'[p,s)$$

Based on the behaviors of GPC, it holds that

$$C'[p,s) = \sup_{p \leq u \leq s} \{C[p,u) - R[p,u) - B(p), 0\}$$

Then

$$R'[p,s) = C[p,s) - C'[p,s) = C[p,s) - \sup_{p \leq u \leq s} \{C[p,u) - R[p,u) - B(p), 0\}$$

Since $B(p) = 0$ and $C[p,p) - R[p,p) = 0$, we have

$$R'[p,s) = C[p,s) - C'[p,s) = C[p,s) - \sup_{p \leq u \leq s} \{C[p,u) - R[p,u)\}$$

Since $\alpha^u(\Delta)$ is concave, there exists $\widetilde{\alpha^u}(\Delta)$ satisfying that for any $\Delta > 0$, $\alpha^u(\Delta) = \widetilde{\alpha^u}(\Delta) + \alpha^u(0^+)$. Then

$$
\begin{aligned}
R'[s,t) &= R[s,t) + B(s) - B(t) \\
&\leq R[s,t) + B(s) \\
&= R[p,t) - R[p,s) + R[p,s) - \{C[p,s) - \sup_{p \leq u \leq s}\{C[p,u) - R[p,u)\}\} \\
&= R[p,t) - C[p,s) + \sup_{p \leq u \leq s}\{C[p,u) - R[p,u)\} \\
&= \sup_{p \leq u \leq s}\{R[u,t) - C[u,s)\} \\
&\leq \sup_{p \leq u \leq s}\{\alpha^u(t-u) - \beta^l(s-u)\} \\
&= \sup_{p \leq u \leq s}\{\widetilde{\alpha^u}(t-u) + \alpha^u(0^+) - \beta^l(s-u)\} \\
&\leq \sup_{p \leq u \leq s}\{\widetilde{\alpha^u}(s-u) + \widetilde{\alpha^u}(t-s) + \alpha^u(0^+) - \beta^l(s-u)\} \\
&= \widetilde{\alpha^u}(t-s) + \sup_{p \leq u \leq s}\{\alpha^u(s-u) - \beta^l(s-u)\} \\
&\leq \widetilde{\alpha^u}(t-s) + Buf(\alpha^u, \beta^l) \\
&= \alpha^u(t-s) + Buf(\alpha^u, \beta^l) - \alpha^u(0^+)
\end{aligned}
$$

Then the lemma is proved. □

Theorem 3. *Given a GPC with input arrival curves α^u, α^l and service curves β^u, β^l, the output events can be upper bounded by:*
$\alpha'^u = ((\alpha^u \otimes \beta^u) \oslash \beta^l) \wedge \beta^u \wedge (\alpha^u + Buf(\alpha^u, \beta^l) - \alpha^u(0^+))$,
where $f \wedge g = min(f, g)$.

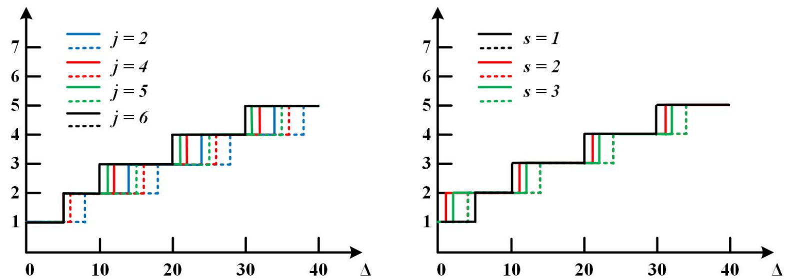
An example. Suppose a GPC has arrival curves and service curves as in Fig. 1, then the output arrival curves calculated by Theorem 3 (blue dashed lines) and the original result (blue full lines) are shown in Fig. 3(a). Next we show the influence of j, s to the results calculated with these two methods, where the result of original method is denoted with full lines, and that of our new method (Theorem 3) is denoted with dashed lines.

Fig. 3(a) shows the influence of j with 4 different sets of inputs. All these 4 inputs are generated with $p = 10, d = 0, s = 1, c = 5, b = 1$, and they differ in j which equals $2, 4, 5, 6$ respectively. Comparing the cases when $j = 2, 4, 5$ and that when $j = 6$, it implies when $p - j > c - s$, the new method is more possible to outperform the original one[7]. Focusing on the cases when $j = 2, 4, 5$, it is observed that the new method performs better when j is smaller.

Fig. 3(b) shows the influence of s with 3 different sets of inputs. All these 4 inputs are generated with $p = 10, j = 6, d = 0, c = 5, b = 1$, and they differ in s which equals $1, 2, 3$ respectively. Our method outperforms the original one

[7] Note that this is not applicable to all task sets.

when $s = 2, 3$ with $p - j > c - s$, which is consistent with the trend in Fig. 3(a). Considering the inputs with $s = 2, 3$, the difference between our method and the original one grows larger when s is smaller.



(a) Different j. (b) Different s.

Fig. 3. Intuitive observations about the influence of j and s.

5 Experiments

We implement our new theoretical results in RTC Toolbox [11] and conduct experiments to evaluate their performance. The new proposed method (denoted by *new*) is compared with the original GPC (denoted by *org*), and the existing work in [8](denoted by *ext*). Task sets of two different parameter settings are generated, under which both single GPC and GPC network are considered . For single GPC, the comparison is conducted with regards to two aspects:

(1) Percentage, denoted by $p(method_1, method_2)$, describes the ratio between the number of task sets where $method_1$ generates more precise upper output arrival curves than $method_2$ and the total number of generated task sets.

(2) Average distance, denoted by $d(method_1, method_2)$, shows the average numerical distance between upper output arrival curves calculated with $method_1$ and $method_2$ in each setting. The distance of two curves f and g is defined as follows, and n is fixed to be 500.

$$dist(f, g) = \frac{\sum_{\Delta=1}^{n} |(f(\Delta) - g(\Delta))|}{n}.$$

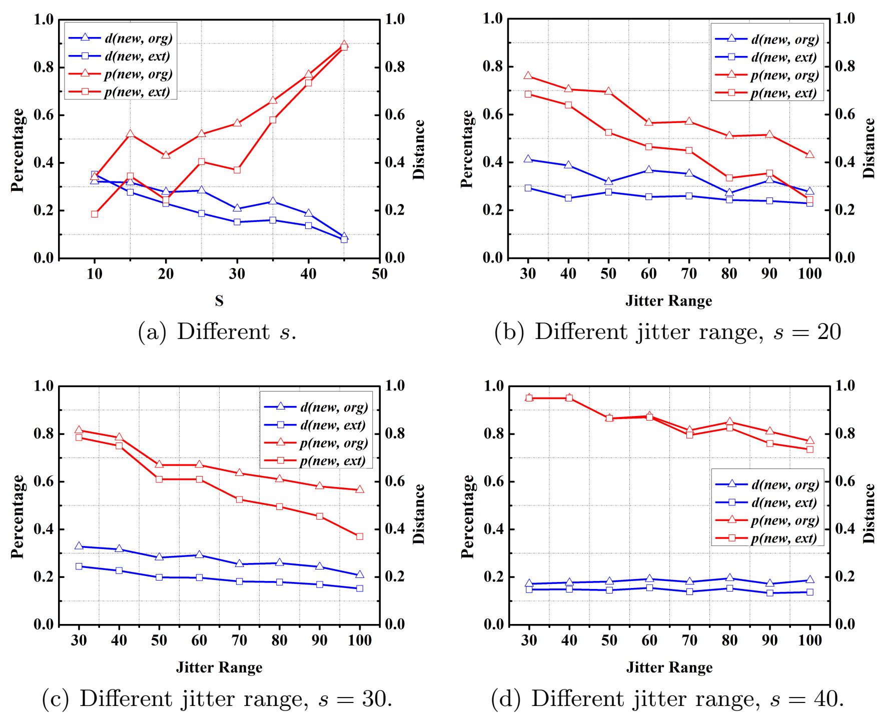
For GPC network, we compare the delay bound calculated by different methods.

5.1 Parameter setting I

Under this parameter setting, arrival curves and service curves are generated as: $p \in [20, 100]$, $j \in [1, x]$, $d \in [1, 20]$, $s = y$, $c = 50$, $b = 1$ (x and y to be specified in the following)[8]. For each setting, we generate 200 task sets.

Single GPC. Fig. 4 (a) shows the results with different s (X-axis) with $x = 100$. Fig. 4 (b)(c)(d) shows the results with different jitter range (X-axis) with $y = 20, 30, 40$ respectively. The number of task sets where our new method generates more precise results than the original calculation and existing work increases with larger s and smaller jitter range.

GPC network. In Fig. 5, we generate $3 \times 3, 4 \times 4, 5 \times 5$ GPC networks and evaluate the normalized quality, which is the ratio between delay bound calculated with two different methods. The parameters are the same as Fig.4 (b)(c)(d). The improvement of our method is more obvious when the network scale is larger.



(a) Different s.

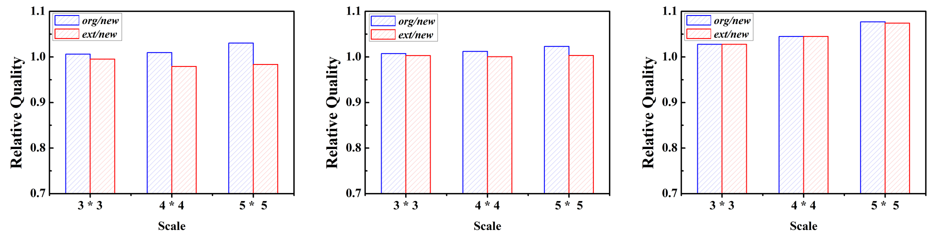
(b) Different jitter range, $s = 20$

(c) Different jitter range, $s = 30$.

(d) Different jitter range, $s = 40$.

Fig. 4. Experiment results for single GPC under parameter setting I.

[8] Note that p and c are relatively small since larger values will cause computation exception with larger-scale GPC networks.



(a) $s = 20$. (b) $s = 30$. (c) $s = 40$.

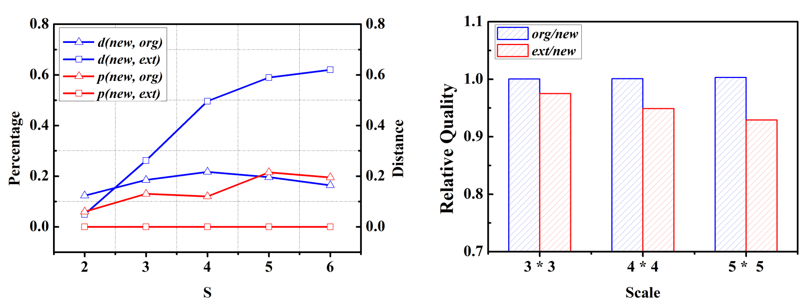
Fig. 5. Experiment results for GPC network under parameter setting I.

5.2 Parameter setting II

Under this parameter setting, arrival curves and service curves are generated as: $p \in [20, 50]$, $j \in [10, 100]$, $d \in [1, 10]$, $c = 60$, $b = 1$, and s varies for different groups of experiments (corresponding to the X-axis). With each s value, we generate 200 task sets.

Single GPC. Fig. 6(a) shows the results with s from 2 to 6. Under this parameter setting, the percentage of task sets where the new proposed method outperforms the original calculation grows with increasing s, while it does not perform better than the method in existing work [8].

GPC Network. As shown in Fig. 6(b), we generate $3 \times 3, 4 \times 4, 5 \times 5$ GPC networks and evaluate the normalized quality. s is fixed to be 20. The method in existing work [8] has better performance than the new proposed method with regards to delay bound.



(a) Different s. (b) $s = 20$.

Fig. 6. Experiment results under parameter setting II.

6 Conclusion

In this work, the improvement of GPC is conducted in two aspects. First, we revise the existing proof of output curves in GPC. Specially, we add the missing deduction parts about the numerical relations between critical parameters. Second, we propose a new method to calculate output arrival curves, which generates more precise results than original methods. In future work, we tend to explore other fundamental abstract components in RTC and improve the precision of related calculation.

References

1. Thiele, L., Chakraborty, S., Gries, M., Kunzli, S.: Design space exploration of network processor architectures. Network Processor Design : Issues and Practices **1**, 1–12 (2002).
2. Fersman, E., Mokrushin, L. , Pettersson, P., Yi, W.: Schedulability analysis of fixed-priority systems using timed automata.Theoretical Computer Science - Tools and algorithms for the construction and analysis of systems **354**(2), 301–317 (2006).
3. Bondorf, S., Schmitt, J.: Improving cross-traffic bounds in feed-forward networks there is a job for everyone. In: Remke A., Haverkort B.R. (eds) Measurement, Modeling and Evaluation of Dependable Computer and Communication Systems, Lecture Notes in Computer Science, Springer, Cham, **9629**, 9–24 (2016).
4. Wandeler, E: Modular performance analysis and interface-based design for embedded real-time systems. Ph.D. Thesis, Publisher,Swiss federal institute of technology Zurich (2006).
5. Chakraborty, S., Knzli, S. , Thiele, L.: A general framework for analysing system properties in platform-based embedded system designs. In: DATE, pp. 1–6. IEEE, Munich, Germany, Germany (2003)
6. Thiele, L., Chakraborty, S., Gries, M. ,Kunzli, S. : A framework for evaluating design tradeoffs in packet processing architectures. In: DAC, pp. 880–885. IEEE, New Orleans, Louisiana, USA (2002)
7. Chakraborty, S., Phan, L. T. X., Thiagarajan, P. S.: Event count automata: a state-based model for stream processing systems. In: RTSS, pp. 87–98. IEEE, Miami, FL, USA (2005)
8. Tang, Y., Guan, N. , Liu, W. C. , Phan, L. T. X. , Yi, W.: Revisiting GPC and AND connector in real-time calculus. In: RTSS, pp. 1–10. IEEE, Paris, France (2017)
9. Guan. N. , Yi, W: Finitary real-time calculus: efficient performance analysis of distributed embedded systems. In: RTSS, pp. 1–10. Vancouver, BC, Canada (2013)
10. Lampka, K., Bondorf S., Schmitt, J.B., Guan N., Yi, W.: Generalized finitary real-time calculus. In: INFOCOM, pp. 1–9. Atlanta, GA, USA (2017)
11. RTC Toolbox Homepage, https://www.mpa.ethz.ch/static/html/Navigation.html

Appendix : Proof of Theorem 2

(1) We first prove α'^{u}. Suppose p is an arbitrarily small time such that the backlog satisfies $B(p) = 0$.

Then for all $p \leq s \leq t$,

$$
\begin{aligned}
&R'[s,t) \\
&= R'[p,t) - R'[p,s) \\
&= \sup_{p\leq b\leq s}\{C[p,b) - R[p,b)\}^+ - \sup_{p\leq a\leq t}\{C[p,a) - R[p,a)\}^+ + C[p,t) - C[p,s)
\end{aligned}
$$

Since $C[p,p) = R[p,p) = C[p,p) - R[p,p) = 0$, the suprema are nonnegative and we have

$$
\begin{aligned}
&R'[s,t) \\
&= \sup_{p\leq b\leq s}\{C[p,b) - R[p,b)\} - \sup_{p\leq a\leq t}\{C[p,a) - R[p,a)\} + C[p,t) - C[p,s) \\
&= \sup_{p\leq b\leq s}\{\inf_{p\leq a\leq t}\{R[b,a) + C[a,t) - C[b,s)\}\} \\
&= \sup_{p\leq b\leq s}\{\inf_{p\leq a\leq t}\{C[s,t) + C[a,b) - R[a,b)\}\} \\
&= C[s,t) + \sup_{p\leq b\leq s}\{\inf_{p\leq a\leq t}\{C[a,b) - R[a,b)\}\} \\
&= C[s,t) + \sup_{p\leq b\leq s}\{min\{\inf_{b\leq a\leq t}\{C[a,b) - R[a,b)\}, \inf_{p\leq a\leq b}\{C[a,b) - R[a,b)\}\}\}
\end{aligned}
$$

Let $\chi = \sup_{p\leq b\leq s}\{min\{\inf_{b\leq a\leq t}\{C[a,b)-R[a,b)\}, \inf_{p\leq a\leq b}\{C[a,b)-R[a,b)\}\}\}$,

$\chi_1(b) = \inf_{b\leq a\leq t}\{C[a,b) - R[a,b)\} = min\{C[b,b) - R[b,b), C[b+1,b) - R[b+1,b), ..., C[t,b) - R[t,b)\} \leq 0$,

$\chi_2(b) = \inf_{p\leq a\leq b}\{C[a,b) - R[a,b)\} = min\{C[p,b) - R[p,b), C[p+1,b) - R[p+1,b), ..., C[b-1,b) - R[b-1,b), C[b,b) - R[b,b)\} \leq 0$.

Next we prove $\chi = \sup_{p\leq b\leq s}\{\chi_1(b)\}$[9].

We consider two cases:

1) For any $i \in [p,s]$, $\chi_1(i) \leq \chi_2(i)$, then $\chi = \sup_{p\leq b\leq s}\{\chi_1(b)\}$.

2) There exists at least one $i \in [p,s]$ satisfying $\chi_1(i) > \chi_2(i)$, then $min\{\chi_1(i), \chi_2(i)\} = \chi_2(i) < 0$,

Similar as the proof for β''^l, there exists $x \in [p,s]$ such that $\min\{\chi_1(x), \chi_2(x)\} = \chi_1(x) = \chi_2(x) = 0$.

Then

$$\chi = \sup_{p\leq b\leq s}\{min\{\chi_1(b), \chi_2(b)\}\}$$

[9] This is not detailed in [4].

$$= max\{min\{\chi_1(p), \chi_2(p)\}, ..., min\{\chi_1(x), \chi_2(x)\}, ..., min\{\chi_1(s), \chi_2(s)\}\}$$
$$= \sup_{b \in \phi}\{min\{\chi_1(b), \chi_2(b)\}\} = \sup_{b \in \phi}\{\chi_1(b)\}$$

where ψ is the set of values in $[p, s]$ which satisfy for any $b \in \phi$, $\chi_1(b) \leq \chi_2(b)$.

On the other hand, $\sup_{p \leq b \leq s}\{\chi_1(b)\} = \sup_{b \in \psi}\{\chi_1(b)\}$, since when $b \in ([p, s] - \psi)$, $\chi_1(b) < 0$. Then we have $\chi = \sup_{p \leq b \leq s}\{\chi_1(b)\}$.

So in both two cases we have $\chi = \sup_{p \leq b \leq s}\{\chi_1(b)\}$.

Then

$$\begin{aligned}
&R'[s, t) \\
&= C[s, t) + \sup_{p \leq b \leq s}\{\inf_{p \leq a \leq t}\{C[a, b) - R[a, b)\}\} \\
&= C[s, t) + \sup_{p \leq b \leq s}\{\inf_{b \leq a \leq t}\{C[a, b) - R[a, b)\}\} \\
&= \sup_{p \leq b \leq s}\{\inf_{b \leq a \leq t}\{C[s, t) + C[a, b) - R[a, b)\}\} \\
&= \sup_{p \leq b \leq s}\{\inf_{b \leq a \leq t}\{R[b, a) + C[a, t) - C[b, s)\}\}
\end{aligned}$$

Then with $\lambda = s - b$ and $\mu = a + \lambda - s$, we have

$$\begin{aligned}
&R'[s, t) \\
&= \sup_{p \leq b \leq s}\{\inf_{b \leq a \leq t}\{R[b, a) + C[a, t) - C[b, s)\}\} \\
&= \sup_{0 \leq \lambda \leq s - p}\{\inf_{0 \leq \mu \leq \lambda + (t - s)}\{R[s - \lambda, \mu - \lambda + s) + C[\mu - \lambda + s, t) - C[s - \lambda, s)\}\} \\
&\leq \sup_{0 \leq \lambda \leq s - p}\{\inf_{0 \leq \mu \leq \lambda + (t - s)}\{\alpha^u(\mu) + \beta^u(\lambda + (t - s) - \mu) - \beta^l(\lambda)\}\} \\
&\leq \sup_{0 \leq \lambda}\{\inf_{0 \leq \mu \leq \lambda + (t - s)}\{\alpha^u(\mu) + \beta^u(\lambda + (t - s) - \mu) - \beta^l(\lambda)\}\} \\
&= (\alpha^u \otimes \beta^u) \oslash \beta^l
\end{aligned}$$

Since the number of processed events can not be larger than the available resource, $R'[s, t) \leq \beta^u(t - s)$, then we have $R'[s, t) \leq \min((\alpha^u \otimes \beta^u) \oslash \beta^l, \beta^u)$.

(2) The results for α'^l can be proved as with a combination of β'^u and α'^u .