Modelling and dynamic behaviour analysis of the software rejuvenation system with periodic impulse

Huixia Huo, Houbao Xu and Zhuoqian Chen

Department of Mathematics and Statistics, Beijing Institute of Technology, Beijing China

ABSTRACT
Software rejuvenation is a policy to counter the phenomenon of software ageing. However, how to implement software rejuvenation is still an important issue. In this paper, periodic impulse control is proposed as an effective tool to perform software rejuvenation and improve the availability of software system. First, we formulate the software rejuvenation system with periodic impulse by a group of coupled differential equations with impulsive action. Then the well-posedness of the system is demonstrated by using operator semigroup theory. At the end of the paper, numerical examples are shown to illustrate the dynamic behaviour of the system and the relationship between the system instantaneous availability and the impulsive indexes, including impulse interval and impulse strength, which yields that the system availability can be improved by adjusting the impulsive indexes.

1. Introduction

The failure of computers is mainly due to the failure of the software rather than the hardware [1]. Nowadays a software comes in multiple releases to keep it updated, relevant in the market, and patch/fix the vulnerabilities. Common experience suggests that the most software failures, called Mandelbugs, are transient in nature [2–4]. Since all ageing-related bugs are Mandelbugs, it must be tolerated in the operational phase of software system regardless of the software version [5,6]. Software ageing refers to the phenomenon that the failure rate and/or performance decline of a long-running software system is caused by the activation and spread of ageing-related bugs [7], such as Apache web server [1], telecommunication switching and billing system [8], cloud computing infrastructure [9] and Android operating systems [10]. The fault-error-failure chains shown in [11] suggested that the activation of ageing-related bugs will lead to the system shift from a correct state to a failure-prone one. After a long period of execution or a large number of accumulated errors, such bugs can lead to ageing-related failures that can result in economic loss or endanger human life [12].

The research on the software rejuvenation system is already well developed. Huang et al. [13] proposed a methodology of software rejuvenation to counter software ageing, as shown in Figure 1, and modelled the four-state process as a continuous time Markov
Figure 1. State transition diagram of software rejuvenation system.

chain, and then derived the steady-state system availability with the state transition probability following negative exponential distribution. Figure 1 describes the state transition diagram of the software rejuvenation system, where O is normal operation state, P stands for the failure probable state, F refers to the failure state and R is the software rejuvenation state. Dohi et al. [14] presented the system availability with the state transition probability following general distribution. Xu et al. [15] analysed the asymptotic stability system with $C_0$-semigroup. In the context of Huang et al.’s model [13], Garg et al. [16] introduced the concept of periodic rejuvenation to deal with the deterministic interval between continuous rejuvenation. Dohi et al. [17] and Suzuki et al. [18] showed that Huang et al.’s model [13] and Garg et al.’s model [16] can be extended to the semi-Markov process and the Markov regenerative process, respectively, and then estimate the optimal failure time of the software rejuvenation. Xu [19] modelled the behaviour of software system by a group of ordinary and partial equations and proved that the system is well posed with the theory of strong continuous semigroup. Dohi et al. [20] considered the optimal software rejuvenation policies maximizing the interval reliability using the Markov regenerative process formalism and deduced the optimal software rejuvenation timing. Koutras and Platis [21] developed a multi-objective optimization strategy to optimize the system overall performance capability.

Impulse systems arising from the real world describes the dynamic processes of sudden discontinuous jumps, which can be modelled by impulse differential equations of [22]. Impulse control is a newly developed control theory based on impulse differential equations, in which a continuously evolving state is modified by discrete control action [23]. As we all know, periodic motion is a very important phenomenon, not only in natural science, but also in climate, food supplement, population, sustainable development and other social sciences [24–27]. The existence of almost periodic solutions for impulse systems has been considered in [28,29]. Wang et al. [30] studied the stability and periodic optimal control of a class of impulsive periodic system in the Banach space. The existence of periodic solutions of impulsive differential equations is obtained by proving the boundedness of solutions in [31–33]. By using continuous time-varying and discontinuous Lyapunov functions, Briat [34] researched the stability and control of linear periodic impulse systems. By fixed point theorem and constructing a Lyapunov function,
Zhou et al. [35] established some sufficient conditions to ensure the permanence and uniformly asymptotic stability of positive almost periodic solutions for a class of impulsive prey-predator model.

Discontinuous periodic rejuvenation is a discrete control action for a continuously evolving state, which can be considered as an impulse control problem. Analysing dynamic behaviour of systems is an important topic for studying dynamical system [36], and requires solving the well-posed problem, including the existence and uniqueness of system solutions and the stability of the system [37,38]. As one of the important branches of functional analysis, operator semigroup theory is a useful tool to study the well-posedness of the abstract equation [39]. For the software system whose sojourn time in each state follows an exponential distribution [13,34,40], modelled it as a Markov chain and governed by a group of ordinary differential equations, so that the dynamical solution exists and tends to the steady-state solution. However, for the sojourn time in at least one state of the system does not follow an exponential distribution, it is debatable whether the system is still well-posed. In the real-world, the repair time usually follows a general distribution rather than an exponential distribution [18,41,42], in which case the system loses its Markovian character. Supplementary variable method can change the non-Markovian nature of the system to Markovian by introducing a new variable [43]. Moreover, [43–45] developed the Chapman-Kolmogorov differential equation for queueing system, Polytube industry and evaporation system, respectively, where the boundary equation is given directly. To make our model more practical, in the same context of [40], this paper will model the system and improve the system availability by means of supplementary variable method and operator semigroup theory, respectively, based on the assumption that the repair time follows a general distribution.

The rest part of this paper is organized as follows. Section 2 formulates the software rejuvenation system with periodic impulse by a group of coupled differential equations with impulsive action and transforms the system into an abstract Cauchy problem in Banach space. The existence and uniqueness of system solutions and stability of the system are investigated in section 3. Section 4 gives numerical examples to illustrate the dynamic behaviour of the system and the relationship between the system availability and impulsive indexes, which yields that the system availability can be improved by adjusting the impulsive indexes. Section 5 concludes the paper.

2. System formulation

In this section, we will establish the mathematical model of the software rejuvenation system with periodic impulse and transform the system into an abstract Cauchy problem.

The software rejuvenation system with periodic impulse discussed in this paper has three states: good, failure probable and failure, as shown in Figure 2. For simplicity, we define \( X(t), t \geq 0 \) as the state of the system at time \( t \):

\[
X(t) = \begin{cases} 
0, & \text{normal operation state} \\
1, & \text{failure probable state} \\
2, & \text{failure state}
\end{cases}
\]
The system from state 0 to state 1, from state 0 to state 2, and from state 1 to state 2 are exponentially distributed with rates $\beta$, $\alpha_0$, $\alpha_1$, respectively. $\mu(x)$ stands for the time-dependent repair rate with an elapsed repair time $x$ when the device in state 2. $\eta(0 < \eta \leq 1)$ and $1 - \eta$ mean the rejuvenation rate and the non-rejuvenation rate in failure probable state, respectively.

In this model, the system in state 0 can either enter the state 1 with rate $\beta$ due to software ageing or go into the state 2 with rate $\alpha_0$ due to failure. The system in state 1 can fail with rate $\alpha_1$ and go to state 2, or rejuvenate the device with rate $\eta$ and go to state 0, or non-rejuvenate the device with rate $1 - \eta$ and still stay at state 1. In state 2, the system will be repaired and returned to state 0 with rate $\mu(x)$.

We make the following assumptions:

$$
\mu(x) \geq 0; \sup_{x \in [0, \infty)} \mu(x) < \infty; \int_0^T \mu(x)dx < \infty, \forall T \in [0, \infty); \int_0^\infty \mu(x)dx = \infty. \quad (1)
$$

In order to modelling the software rejuvenation system with periodic impulse, we first modelled the software system without impulse rejuvenation, namely $\eta = 0$, as shown in Figure 3.

Let $p_0(t)$ stand for the probability that the device is in good state at time $t$; $p_1(t)$ stand for the probability that the device is in failure probable state at time $t$; $p_2(t, x)$ stand for the probability density (with respect to the repair time) that the failed device is in failure state and has an elapsed repair time of $x$ at time $t$. Then, for $p_0(t + h)$, we have

$$
p_0(t + h) = p_0(t) + \int_t^{t+h} \int_0^\infty \mu(x)p_2(s, x)dxds - \int_t^{t+h} (\alpha_0 + \beta)p_0(s)ds. \quad (2)
$$

Differentiate (2) with respect to $h$,

$$
\frac{dp_0(t)}{dt} = \int_0^\infty \mu(x)p_2(t, x)dx - (\alpha_0 + \beta)p_0(t). \quad (3)
$$

**Figure 2.** State transition diagram of the software rejuvenation system with periodic impulse.
For $p_1(t + h)$, we have

$$p_1(t + h) = p_1(t) + \int_t^{t+h} \beta p_0(s) \, ds - \int_t^{t+h} \alpha_1 p_1(s) \, ds.$$  \(4\)

Differentiate (4) with respect to $h$,

$$\frac{dp_1(t)}{dt} = \beta p_0(t) - \alpha_1 p_1(t).$$  \(5\)

For $p_2(t, x)$, consider the function

$$N(t, x) = \int_0^x p_2(t, \sigma) \, d\sigma,$$

representing the probability of failure that, at time $t$, have elapsed time less than or equal to $x$. Then, we have, for $h > 0$,

$$N(t + h, x + h) = N(t, x) + \int_t^{t+h} (\alpha_0 p_0(s) + \alpha_1 p_1(s)) \, ds - \int_t^{t+h} \int_0^{x+s-t} \mu(\sigma) p_2(t + s, \sigma) \, d\sigma \, ds.$$  \(6\)

The second term on the right in (6) gives the input of all new failure devices in the time interval $[t, t + h]$. The third term on the right of (6) gives the loss from the initial group of $N(t, x)$ and from the new failure devices, through the time interval $[t, t + h]$. By using the substitution integral method for the third term in (6), we obtain

$$N(t + h, x + h) = N(t, x) + \int_t^{t+h} (\alpha_0 p_0(s) + \alpha_1 p_1(s)) \, ds - \int_0^h \int_0^{x+s} \mu(\sigma) p_2(t + s, \sigma) \, d\sigma \, ds.$$  \(7\)
Differentiate (7) with respect to $h$, and set $h = 0$,

$$N_t(t, x) + N_x(t, x) = \alpha_0 p_0(t) + \alpha_1 p_1(t) - \int_0^x \mu(\sigma)p_2(t, \sigma)d\sigma,$$

that is,

$$\int_0^x \frac{\partial p_2(t, \sigma)}{\partial t}d\sigma + p_2(t, x) = \alpha_0 p_0(t) + \alpha_1 p_1(t) - \int_0^x \mu(\sigma)p_2(t, \sigma)d\sigma. \tag{9}$$

From (9), setting $x = 0$, then

$$p_2(t, 0) = \alpha_0 p_0(t) + \alpha_1 p_1(t) \tag{10}$$

and differentiating (9) with respect to $x$,

$$\frac{\partial p_2(t, x)}{\partial x} + \frac{\partial p_2(t, x)}{\partial t} = -\mu(x)p_2(t, x). \tag{11}$$

So the system can be formulated as

$$\begin{cases}
\frac{dp_0(t)}{dt} = -(\alpha_0 + \beta) p_0(t) + \int_0^\infty \mu(x)p_2(t, x)dx \\
\frac{dp_1(t)}{dt} = \beta p_0(t) - \alpha_1 p_1(t) \\
\frac{\partial p_2(t, x)}{\partial t} + \frac{\partial p_2(t, x)}{\partial x} = -\mu(x)p_2(t, x)
\end{cases} \tag{12}$$

with the boundary condition

$$p_2(t, 0) = \alpha_0 p_0(t) + \alpha_1 p_1(t). \tag{13}$$

The initial values satisfy

$$p_0(0) + p_1(0) + p_2(0, x) = 1. \tag{14}$$

Define the state space $X = R^2 \times L^1[0, \infty)$ with $\| \cdot \|_X = | \cdot | + | \cdot | + \| \cdot \|_{L^1[0, \infty)} < \infty$, which is obviously the Banach space, and the system operators in $X$ and its domain as follows,

$$A = \begin{pmatrix}
-(\alpha_0 + \beta) & 0 & 0 \\
0 & -\alpha_1 & 0 \\
0 & 0 & -\frac{d}{dx} - \mu(x)
\end{pmatrix}, B\begin{pmatrix}
p_0 \\
p_1 \\
p_2(x)
\end{pmatrix} = \begin{pmatrix}
\int_0^\infty \mu(x)p_2(x)dx \\
\beta p_0 \\
0
\end{pmatrix}.$$

$$D(A + B) = \left\{ (p_0, p_1, p_2(x))^T \in X | \frac{dp_2(x)}{dx} \text{ is absolutely continuous function} \right\}.$$

Let $\bar{P}(t) = ((p_0(t), p_1(t), p_2(t, \cdot))^T$, the system (12) can be described as an abstract Cauchy problem in $X$:

$$\begin{cases}
\frac{d\bar{P}(t)}{dt} = (A + B)\bar{P}(t) \\
\bar{P}(0) = \bar{P}_0
\end{cases} \tag{15}$$

On this basis, we consider a discrete time set $\{\tau_n\}, n = 1, 2, \cdots$ that is equidistant with impulse interval $\delta$ of impulsive jump instants, which satisfy
\[ \tau_0^+ = \tau_0 = 0, \tau_n - \tau_{n-1} = \delta, n = 1, 2, \ldots, \]

\[ 0 = \tau_0 < \tau_1 < \tau_2 < \cdots, \]

\[ \lim_{n \to \infty} \tau_n = \infty, \tau_n = n\delta, n = 1, 2, \ldots. \]

Without loss of generality, we assume that \( \forall \epsilon > 0, \bar{P}(t) \) is left continuous for \( t \in (n\delta - \epsilon, n\delta], n = 1, 2, \ldots. \) Since \( \eta \) represents the impulse ratio in failure probable state, we have

\[ p_0((n\delta)^+) = p_0(n\delta) + \eta p_1(n\delta), \]

\[ p_1((n\delta)^+) = (1 - \eta)p_1(n\delta), \]

\[ p_2((n\delta)^+, x) = p_2(n\delta, x). \]

Then,

\[ \Delta p_0(n\delta) = p_0((n\delta)^+) - p_0(n\delta) = \eta p_1(n\delta), \]

\[ \Delta p_1(n\delta) = p_1((n\delta)^+) - p_1(n\delta) = -\eta p_1(n\delta), \]

\[ \Delta p_2(n\delta, x) = p_2((n\delta)^+, x) - p_2(n\delta, x) = 0. \]

Let impulse operator

\[ C = \begin{pmatrix} 0 & \eta & 0 \\ 0 & -\eta & 0 \\ 0 & 0 & 0 \end{pmatrix}, \]

which is obviously \( D(C) = X. \)

Thus, the software rejuvenation system with periodic impulse can be expressed as,

\[ \begin{cases} \frac{d\bar{P}(t)}{dt} = (A + B)\bar{P}(t), t \neq n\delta, n = 1, 2, \ldots \\ \Delta \bar{P}(t) = C\bar{P}(t), t = n\delta, n = 1, 2, \ldots \\ \bar{P}(0) = \bar{P}_0 \end{cases} \]

(17)

3. Dynamic behaviour analysis

3.1. Well-posedness of system (17)

In this subsection, we will prove the well-posedness of the software rejuvenation system with periodic impulse (17) by using operator semigroup theory.

To be self-contained, we introduce the following Theorem 3.1.
Theorem 3.1 [46]. Let $A$ be a densely defined resolvent positive operator. If $D(A)_+$ is cofinal in $X_+$ or if $D(A^+_*)$ is cofinal in $X^+_*$, where $D(A)_+ = X_+ \cap D(A)$ and $D(A^+_*) = X^*_+ \cap D(A^*)$, then $A$ is the generator of a positive $C_0$-semigroup. Moreover, $s(A) = \omega(A)$.

Theorem 3.2. The software rejuvenation system with periodic impulse (17) has a unique nonnegative dynamical solution $\tilde{P}(t, \cdot)$ which satisfies,

$$\tilde{P}(t, \cdot) = T(t - n\delta)((I + C)T(\delta))^n\tilde{P}_0, t \in (n\delta, (n + 1)\delta], n = 0, 1, 2, \cdots$$

and

$$\| \tilde{P}(t, \cdot) \| = 1, \forall t \in [0, +\infty).$$

Proof. By proving that the system operator $A + B$ is the generator of a positive $C_0$-semigroup $T(t)$ (see Appendix A), it can be obtained that there exists a unique nonnegative dynamical solution

$$\tilde{P}(t, \cdot) = T(t)\tilde{P}_0$$

of system (15) that satisfies $\| \tilde{P}(t, \cdot) \| = 1, \forall t \in [0, +\infty)$ [47]. For the software system with impulse rejuvenation (17), when $t \in (0, \delta]$, the dynamical solution is the same as the system without impulsive rejuvenation, that is,

$$\tilde{P}(t, \cdot) = T(t)\tilde{P}_0,$$

which yields as follows:

$$\tilde{P}(\delta, \cdot) = T(\delta)\tilde{P}_0.$$

But when $t \in (\delta, 2\delta]$, we have

$$\tilde{P}(\delta^+, \cdot) = (I + C)T(\delta)\tilde{P}_0$$

and

$$\tilde{P}(t, \cdot) = T(t - \delta)\tilde{P}(\delta^+, \cdot) = T(t - \delta)(I + C)T(\delta)\tilde{P}_0,$$

$$\tilde{P}(2\delta, \cdot) = T(\delta)(I + C)T(\delta)\tilde{P}_0.$$

When $t \in (2\delta, 3\delta]$, we obtain

$$\tilde{P}(2\delta^+, \cdot) = (I + C)T(\delta)(I + C)T(\delta)\tilde{P}_0 = ((I + C)T(\delta))^2\tilde{P}_0$$

and

$$\tilde{P}(t, \cdot) = T(t - 2\delta)\tilde{P}(2\delta^+, \cdot) = T(t - 2\delta)((I + C)T(\delta))^2\tilde{P}_0,$$

$$\tilde{P}(3\delta, \cdot) = T(\delta)((I + C)T(\delta))^2\tilde{P}_0.$$

By repeating the above steps, we can derive the dynamical solution of the software rejuvenation system with periodic impulse (17) as
\[ \tilde{P}(t, \cdot) = T(t - n\delta)((I + C)T(\delta))^n\tilde{P}_0, \quad t \in (n\delta, (n + 1)\delta], \quad n = 0, 1, 2, \cdots \]

In addition, it is obvious that \( \| \tilde{P}(t, \cdot) \| = 1, \quad \forall t \in [0, \infty) \).

**Theorem 3.3.** The interval dynamical solution of the system (17)

\[ \tilde{P}(t, \cdot) = T(t - n\delta)((I + C)T(\delta))^n\tilde{P}_0, \quad t \in (n\delta, (n + 1)\delta], \quad n = 0, 1, 2, \cdots \quad (21) \]

strongly converges to its steady-state solution \( T(u)\tilde{P}^{**}, \quad u = t - n\delta \in (0, \delta] \), where \( \tilde{P}^{**} \) is the eigenfunction corresponding to eigenvalue 1 of the system operator \( (I + C)T(\delta) \) satisfying \( \| \tilde{P}^{**} \| = 1 \).

**Proof.** By proving \( \{ y \in C | \text{Re} y > 0 \text{ or } y = ia, a \in R, a \neq 0 \} \subset \rho(A + B) \) (see Appendix A) and \( 0 \in \sigma_p(A + B) \) (see Appendix B), and applying Spectral Mapping Theorem [47], we deduce that \( \{ y \in C | \text{Re}y > 1 \text{ or } y = 1 + ia, a \in R, a \neq 0 \} \subset \rho(T(t)) \) and \( 1 \in \sigma_p(T(t)) \) for \( t \geq 0 \). Moreover, \( 1 \in \sigma_p((I + C)T(\delta)) \) and \( \{ y \in C | \text{Re} y > 1 \text{ or } y = 1 + ia, a \in R, a \neq 0 \} \subset \rho((I + C)T(\delta)) \) can be derived by contradiction, which yields the impulsive system (17) has a steady state initial solution \( \tilde{P}^{**} \neq 0 \), satisfying

\[ ((I + C)T(\delta))\tilde{P}^{**} = \tilde{P}^{**}, \]

that is,

\[ \lim_{n \to \infty} ((I + C)T(\delta))^n\tilde{P}_0 = \tilde{P}^{**}. \]

In addition, \( T(t - n\delta) = T(t) \) based on the property of the semigroup [47]. This completes the proof.

### 3.2. System instantaneous availability

Since state 0 and state 1 of the system (17) are the available states as described in Section 2, the system instantaneous availability can be described as

\[ A(t) = p_0(t) + p_1(t) = (1, 1, 0)T(t - n\delta)((I + C)T(\delta))^n\tilde{P}_0, \quad t \in (n\delta, (n + 1)\delta], \quad (22) \]

by equation (21), in which \( n = 0, 1, 2, \cdots \).

Based on the expression (16) of impulse operator \( C \), the system instantaneous availability (22) is related to the impulse interval \( \delta \) and impulse strength \( \eta \).

### 4. Numerical examples

In this section, based on the Theorem 3.3, numerical examples will be presented to illustrate the dynamic behaviour of the system (17) and the relationship between the system availability and impulsive indexes, including impulse interval \( \delta \) and impulse strength \( \eta \).
To reproduce the real-world behaviour, we take the virtual machine as an example and set \( \alpha_0 = \frac{1}{10} \), \( \alpha_1 = \frac{1}{2} \), \( \beta = \frac{1}{2} \), and the repair time follows Weibull distribution \( \mu(x) = \tau h(x)^{h-1} \) with \( \tau = 5 \) and \( h = 2 \). The default parameter values \( \alpha_0, \alpha_1, \beta \) used are based on the experimental studies [48–51], \( \mu(x) \) without references are set by ourselves according to the values with references [50,51].

### 4.1. Dynamic behaviour of the software rejuvenation system with periodic impulse

To simulate the dynamic behaviour of the software rejuvenation system with periodic impulse (17), we first analyse the dynamic behaviour of the software system without impulse rejuvenation (15) based on discrete method.

Since the solution \( p_2(t,x) \in L^1[0,\infty) \) for any given time \( t \), then \( \forall \varepsilon > 0 \) there exists an \( l < \infty \) such that \( \int_0^\infty |p_2(t,x)|dx < \varepsilon \). So we just have to consider the interval \([0,l]\) for \( x \). We can divide the interval \([0,l]\) into \( m \) equal subintervals \([x_k,x_{k+1}]\), \( x_k = k\Delta x, k = 0,1,2,\cdots,m-1 \) and \( \Delta x = \frac{l}{m} \). Let \( p_{2,k}(t) = p_2(t,x_k), \mu_k = \mu(x_k) \). We apply the difference scheme to discretize the first-order partial derivative term in \( p_2(t,x) \) and use the (right) rectangular rule to approximate the integral term in the first equation of equations (12), which lead to the following approximating model

\[
\begin{align*}
\frac{dp_0(t)}{dt} &= -(\alpha_0 + \beta)p_0(t) + \Delta x \sum_{k=1}^{m} \mu_k p_{2,k}(t) \\
\frac{dp_1(t)}{dt} &= \beta p_0(t) - \alpha_1 p_1(t) \\
\frac{dp_{2,k}(t)}{dt} &= \frac{p_{2,k-1}(t) - p_{2,k}(t)}{\Delta x} - \mu_k p_{2,k}(t), k = 1,2,\cdots,m
\end{align*}
\]  

(23)

marching forward with the boundary condition
\[
\begin{align*}
p_{2,0}(t) &= \alpha_0 p_0(t) + \alpha_1 p_1(t) \\
p_0(0) &= 1, p_1(0) = p_{2,k}(0) = 0, k = 1,2,\cdots,m.
\end{align*}
\]

(24)

(25)

By defining
\[
\bar{P}_m(t) = (p_0(t),p_1(t),p_{2,1}(t),p_{2,2}(t),\cdots,p_{2,m}(t))^T,
\]

(26)

and the asymptotic generator
\[
(A + B)_m = \begin{pmatrix}
-(\alpha_0 + \beta) & 0 & \Delta x \mu_1 & \Delta x \mu_2 & \cdots & \Delta x \mu_m \\
\beta & -\alpha_1 & 0 & 0 & \cdots & 0 \\
\frac{\alpha_0}{\Delta x} & \frac{\alpha_1}{\Delta x} & \Delta_1 & 0 & \cdots & 0 \\
0 & 0 & \frac{1}{\Delta x} \Delta_2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \frac{1}{\Delta x} \Delta_m & \cdots & 0
\end{pmatrix}_{(m+2)\times(m+2)}
\]

on \( X_m = R^{m+2} \), where \( \Delta_j = -(\frac{1}{\Delta x} + \mu_j), j = 1,2,\cdots,m \), the above asymptotic system (23)-(25) can be transformed into an abstract Cauchy problem of ODE system
\[
\begin{aligned}
\frac{d\tilde{P}_m(t)}{dt} &= (A + B)_m \tilde{P}_m(t), \quad t \geq 0 \\
\tilde{P}_m(0) &= (1, 0, 0, \ldots, 0)^T_{1 \times (m+2)}.
\end{aligned}
\]

(27)

It can be proved that the solution \( \tilde{P}_m(t) = e^{(A+B)_m t} \tilde{P}_0 \) of (27) strongly converges to (15) as \( m \to \infty \) on bounded \( t \) – intervals by applying Trotter-Kato Theorem [52]. In addition, the solution \( \tilde{P}_m(t) = e^{(A+B)_m t} \tilde{P}_0 \) can be obtained by Matlab.

By solving the equations (23)-(25), we can derive the approximating solution of model (12)-(14) directly as \( m \to 1 \). When \( m \to 1 \), Figure 4 shows that \( p_2(t,x) \to 0 \) as \( x \) grows. This situation agrees with \( p_2(t,x) \in L^1[0,\infty) \) for any given time \( t \) and also conforms to reality. Correspondingly, the probability distributions \( p_0(t), p_1(t) \) and \( \tilde{p}_2(t) = \int_0^1 p_2(t,x)dx \) of the software system without impulse rejuvenation are shown in Figure 5.

Then define impulse operator of the asymptotic system (27) as

\[
C_m = \begin{pmatrix}
0 & \eta & 0 & \cdots & 0 \\
0 & -\eta & 0 & \cdots & 0 \\
0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & 0
\end{pmatrix}_{(m+2) \times (m+2)}
\]

and let impulse strength \( \eta = 0.8 \) and impulse interval \( \delta = 1 \), the dynamic behaviours of the software rejuvenation system with periodic impulse are depicted in Figure 6.

Figure 6 clearly presents the interval dynamical solutions \( p_0(t), p_1(t) \) and \( \tilde{p}_2(t) = \int_0^1 p_2(t,x)dx \) of the software rejuvenation system with periodic impulse based on equation (21). From Figure 6, we find that once the interval initial values of the system are determined, the interval dynamical solutions can be obtained according to the given semigroup \( T(t) \).

Figure 4. Behaviour of solution \( p_2(t,x) \).
Figure 5. Probability distributions of the software system without impulse rejuvenation.

Figure 6. Probability distributions of the software rejuvenation system with periodic impulse.
4.2. **The relationship between the system instantaneous availability and impulsive indexes**

In this subsection, we give some numerical examples to illustrate the relationship between the instantaneous availability of the system (17) and the impulsive indexes based on equation (22).

Let impulse interval δ = 1, Figures 7 and 8 describe the instantaneous availability of the system (17) with different impulse strength η at \( t \in [0, 6] \) and at \( t \in (n, n+1], n \to \infty \), respectively. In Figure 7, we find that there exists an optimal value for time \( t \) to maximize the system instantaneous availability at a given time interval \( (n, n+1], n = 1, 2, 3, 4, 5 \). As can be seen from the Figure 8, when \( n \to \infty \), the instantaneous availability is directly proportional to the impulse strength \( η \). In addition, it is obvious that the instantaneous availability of the software rejuvenation system with periodic impulse \( (0 < η \leq 1) \) is significantly higher than that of the system without periodic impulse \( (η = 0) \).

Let impulse strength \( η = 0.8 \), Figures 9 and 10 illustrate the instantaneous availability of the system (17) with different impulse interval δ. In Figure 9, we find that there exists an optimal value for time \( t \) to maximize the system instantaneous availability at a given time interval \( (nδ, (n+1)δ], n = 1, 2, 3, \cdots \). From Figure 10, we can see that when \( n \to \infty \), the instantaneous availability is inversely proportional to the impulse interval \( δ \).

Based on the relationship between the availability of the system (17) and impulsive indexes, that is, the instantaneous availability is directly proportional to the impulse strength \( η \) and inversely proportional to the impulse interval \( δ \) when \( n \to \infty \), we can improve the system instantaneous availability by adjusting the impulsive indexes, such as increasing the system impulse strength or decreasing the system impulse interval.

In addition, a brief description of the algorithm in this section is given as follows:
- Step 1: Input \( m, α_0, α_1, β, t, l \).
- Step 2: Set \( Δx = \frac{l}{m} \). For \( i = 1 : m \), input \( μ_k = μ(x_k), \tilde{P}_0 \) and \( (A + B)_m \).

![Figure 7. Instantaneous availability with different impulse strength η.](image-url)
Step 3: The dynamical behaviour of the software system without impulse rejuvenation can be depicted by calculating $P_m(t) = e^{(A+B)t}P_0$.

Step 4: Input impulse operator $C_m$, then the dynamical behaviour of the software rejuvenation system with periodic impulse can be depicted based on equation (21).

Step 5: Instantaneous availability with different impulse strength and impulse interval of the software rejuvenation system with periodic impulse can be depicted according to equation (22).

Figure 8. Instantaneous availability at different impulse strength $\eta$ as $n \to \infty$.

Figure 9. Instantaneous availability with different impulse interval $\delta$. 

5. Conclusion

This paper discussed the modelling and dynamic behaviour of the software rejuvenation system with periodic impulse under the assumption that the repair time follows a general distribution. In the system modelling, the elapsed repair time $x$ as a parameter was introduced into the software system without impulse rejuvenation, and the system was then modelled as a set of coupled differential equations by using the supplementary variable method, where the boundary condition of the equations can be derived. On this basis, a group of coupled differential equations with impulsive action can be used to model the software rejuvenation system with periodic impulse after considering the equidistant impulses. Moreover, the equations of the software rejuvenation system without or with periodic impulse were transformed into an abstract Cauchy problem.

It is well known that operator semigroup theory is a useful tool to study the well-posedness of abstract equation. In the theoretical part, the well-posedness of the software rejuvenation system with periodic impulse, as an important theoretical basis for studying the impulse system behaviour, is analysed using operator semigroup theory, in which the existence and uniqueness of the system solution was proved by the theory of cofinal and resolvent positive operator, and the interval asymptotic stability of the system was obtained by analysing the spectral distribution of the operator $(I + C)T(\delta)$. In addition, the expression of the system instantaneous availability is given.

In the numerical examples, we took the virtual machine as an example to depict the dynamic behaviour of the software rejuvenation system with periodic impulse and the relationship between the system availability and impulsive indexes. It is shown that the instantaneous availability is directly proportional to the impulse strength and inversely proportional to the impulse interval when $n \rightarrow \infty$. Therefore, the impulse strength and impulse interval can be used as effective tools to improve the system availability.
Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This research was supported by the NSAF under Grant No. U1430125.

ORCID

Huixia Huo http://orcid.org/0000-0002-3999-4386
Houbao Xu http://orcid.org/0000-0001-7238-5956

References

[1] M. Grottker, L. Li, K. Vaidyanathan, and K.S. Trivedi, Analysis of software aging in a web server, IEEE. T. Reliab. 55 (2006), pp. 411–420. doi:10.1109/TR.2006.879609.
[2] J. Gray, Why do computers stop and what can be done about it? ISRDSDS. Los Angeles, CA, USA, (1998), pp. 3–12.
[3] M. Grottker and K.S. Trivedi, Fighting bugs: Remove, retry, replicate, and rejuvenate, IEEE. Comput. 40 (2007), pp. 107–109. doi:10.1109/MC.2007.55.
[4] J. Gray and D.P. Siewiorek, High-availability computer systems, IEEE. Comput. 9 (1991), pp. 39–48. doi:10.1109/2.84898.
[5] M. Grottker and K.S. Trivedi, A classification of software fault. ISRE. Chicago, IL, USA, (2005), pp. 19–20.
[6] J.J. Zheng, H. Okamura, and T. Dohi, A transient interval reliability analysis for software rejuvenation models with phase expansion, Software. Qual. J. 28 (2020), pp. 173–194. doi:10.1007/s11219-019-09458-1.
[7] D. Cotroneo, R. Natella, R. Pietrantuono, and S. Russo, A survey of software aging and rejuvenation studies, ACM. J. Emerg. Tech. Com. 10 (2014), pp. 1–34. doi:10.1145/2539117.
[8] K. Iwamoto, T. Dohi, H. Okamura, and N. Kaio, Discrete-time cost analysis for a telecommunication billing application with rejuvenation, Computers & Mathematics with Applications 51 (2) (2006), pp. 335–344. doi:10.1016/j.camwa.2005.11.007.
[9] H. Sukhwani, R. Matias, and K.S. Trivedi, Monitoring and mitigating software aging on IBM cloud controller system. ISREW. Toulouse, France, 2017, pp. 266–272.
[10] D. Cotroneo, A.K. Iannillo, R. Natella, and R. Pietrantuono, A comprehensive study on software aging across android versions and vendors, Empir. Softw. Eng. 25 (2020), pp. 3357–3395. doi:10.1007/s10664-020-09838-3.
[11] A. Avrizer and E.J. Weyuker, The role of modeling in the performance testing of e-commerce applications, IEEE. T. Software. Eng. 30 (2004), pp. 1072–1083. doi:10.1109/TSE.2004.107.
[12] E. Marshall, Fatal error: How patriot overlooked a scud, Science 255 (1992), pp. 1347. doi:10.1126/science.255.5050.1347.
[13] Y. Huang, C. Kintala, N. Kolettis, and N.D. Fulton, Software rejuvenation: Analysis, module and applications, FTCS. Pasadena, CA, USA, 1 (1995), pp. 381.
[14] T. Dohi, Estimating software rejuvenation schedules in high-assurance systems, Comput. J. 44 (2001), pp. 473–485. doi:10.1093/comjnl/44.6.473.
[15] H.B. Xu, W.B. Xu, J.Y. Yu, and G.T. Zhu, The asymptotic stability of the solution of software rejuvenation system, Math. Prac. Theory. 34 (2004), pp. 112–118.
[16] S. Garg, A. Puliafito, M. Telek, and K.S. Trivedi, Analysis of software rejuvenation using Markov regenerative stochastic petri net, ISSRE. Toulouse, France, (1995), pp. 24–27.
[17] T. Dohi, K. Goševa-Popstojanova, and K.S. Trivedi, Statistical non-parametric algorithms to estimate the optimal software rejuvenation schedule, PRDC. Los Angeles, CA, USA, (2000), pp. 77–84.

[18] H. Suzuki, T. Dohi, and H. Okamura, Cost-effective analysis of software systems with periodic rejuvenation, IEICE. T. Fund. Electr. 85 (2002), pp. 2923–2932.

[19] H.B. Xu, Modelling and well-posed analysis for software system with rejuvenation, Math. Comp. Model. Dyn. 17 (2011), pp. 583–600. doi:10.1080/13873954.2011.588607.

[20] T. Dohi, J.J. Zheng, H. Okamura, and K.S. Trivedi, Optimal periodic software rejuvenation policies based on interval reliability criteria, Reliab. Eng. Syst. Safe. 180 (2018), pp. 463–475. doi:10.1016/j.ress.2018.08.009.

[21] V.P. Koutras and A.N. Platis, On the performance of software rejuvenation models with multiple degradation levels, Software Qual. J. 28 (2020), pp. 135–171. doi:10.1007/s11219-019-09491-0.

[22] A.M. Samoilenko and N.A. Perestyuk, Impulsive Differential Equations, World Scientific, Singapore, 1995.

[23] R. Leander, S. Lenhart, and V. Protopopescu, Optimal control of continuous systems with impulse controls, Optim. Contr. Appl. Met. 36 (2015), pp. 535–549. doi:10.1002/oca.2128.

[24] Y. Dong, Sublinear impulse effects and solvability of boundary value problems for differential equations with impulses, J. Math. Anal. Appl. 264 (2001), pp. 32–48. doi:10.1006/jmaa.2001.7548.

[25] P.S. Rivadeneira and C.H. Moog, Impulsive control of single-input nonlinear systems with application to HIV dynamics, Appl. Math. Comput. 218 (2012), pp. 8462–8474.

[26] A.R. Teel, A. Subbaraman, and A. Sferlazza, Stability analysis for stochastic hybrid systems: A survey, Automatica 50 (2014), pp. 2435–2456. doi:10.1016/j.automatica.2014.08.006.

[27] J.J. Jiao, Z.Z. Liu, L.M. Li, and X.Y. Nie, Threshold dynamics of a stage-structured single population model with non-transient and transient impulse effects, Appl. Math. Lett. 97 (2019), pp. 88–92. doi:10.1016/j.aml.2019.05.024.

[28] X. Xiang and N.U. Ahmed, Existence of periodic solutions of semilinear evolution equations with time lags, Nonlinear. Anal. 18 (1992), pp. 1063–1070. doi:10.1016/0362-546X(92)90195-K.

[29] Y.K. Li and P. Wang, Asymptotical stability of almost periodic solution for an impulsive multispecies competition-predation system with time delays on time scales, Math. Method. Appl. Sci. 40 (2017), pp. 6007–6034. doi:10.1002/mma.4450.

[30] J.R. Wang, X.L. Xiang, and W. Wei, Stability analysis and periodic optimal control problems for impulsive periodic system, Math.Commun. 16 (2011), pp. 235–250.

[31] J. Shen, J. Li, and Q. Wang, Boundedness and periodicity in impulsive ordinary and functional differential equations, Nonlinear. Anal. 65 (2006), pp. 1986–2002. doi:10.1016/j.na.2005.11.006.

[32] J.K. Hale, Theory of Functional Differential Equations, Springer-Verlag, New York, 1977.

[33] J.L. Massera, The existence of periodic solutions of systems of differential equations, Duke Math. J. 17 (1950), pp. 457–475. doi:10.1215/S0012-7094-50-01741-8.

[34] C. Briat, Convex conditions for robust stability analysis and stabilization of linear aperiodic impulsive and sampled-data systems under dwell-time Constraints, Automatica. 49 (2013), pp. 3449–3457. doi:10.1016/j.automatica.2013.08.022.

[35] H. Zhou, W. Wang, and L. Yang, Permanence and stability of solutions for almost periodic prey-predator model with impulsive effects, Qual. Theor. Dyn. Syst. 17 (2018), pp. 463–474. doi:10.1007/s12346-017-0247-y.

[36] P. Mellodge, A Practical Approach to Dynamical Systems for Engineers, Woodhead Publishing, Cambridge, 2016.

[37] H.B. Xu, J.Y. Yu, and G.T. Zhu, Asymptotic property of a repairable multi-state device, Q, Appl. Math. 63 (2005), pp. 779–789.

[38] I.V. Mel'nikova and M.A. Al'shansky, Well-posedness of the Cauchy problem in a Banach space: Regular and degenerate case, J. Math. Sci. 87 (1997), pp. 3732–3780. doi:10.1007/BF02355451.
[39] E. Klausjochen and N. Rainer, *One-Parameter Semigroups for Linear Evolution Equations*, Springer, New York, 2000.

[40] H.X. Huo, T.T. Win, and H.B. Xu, *Availability analysis of the software rejuvenation system with impulse control*, CCDC. Nanchang, China, (2019), pp. 825–829. doi: 10.1109/CCDC.2019.8833116.

[41] W. Xie, Y.G. Hong, and K. Trivedi, *Analysis of a two-level software rejuvenation policy*, Reliab. Eng. Syst. Safe. 87 (2005), pp. 13–22. doi:10.1016/j.ress.2004.02.011.

[42] B.N. Khoury, *Optimal control of production rate in a manufacturing system prone to failure with general distribution for repair time*, IEEE. T. Automat. Contr. 61 (2016), pp. 4112–4117. doi:10.1109/TAC.2016.2547978.

[43] K. Ramesh and M.S. Kadyan, *Performance analysis and maintenance planning of evaporation system in the sugar industry by using supplementary variable technique*, J. Ind. Integr. Manag 3 (2018), pp. 1–22-1850004. doi:10.1142/S2424862218500045.

[44] S.A. Attahiru and T.S.S. Srinivasa Rao, *Supplementary variable technique in stochastic models*, Probab. Eng. Inform. Sc. 14 (2000), pp. 203–218. doi:10.1017/S0269964800142068.

[45] S. Shakuntla, A.K. Lal, S.S. Bhatia, and J. Singh, *Reliability analysis of Polytube industry using supplementary variable technique*, App. Math. Comp. 218 (2011), pp. 3981–3992. doi:10.1016/j.amc.2011.10.016.

[46] A. Wolfgang, *Resolvent positive operators*, Proc, Landon. Math. Soc. 54 (1987), pp. 321–349.

[47] A. Pazy, *Semigroups of Linear Operators and Applications to Partial Differential Equations*, Springer, New York, 1983.

[48] K. Kourai and S. Chiba, *A fast rejuvenation technique for server consolidation with virtual machines*, DSN. Edinburgh, UK, (2007), pp. 245–255.

[49] D. Kim, F. Machida, and K.S. Trivedi, *Availability modeling and analysis of a virtualized system*, PRDC. Shanghai, China, 2009, 365–371. doi: 10.1109/PRDC.2009.64.

[50] X. Chang, T. Wang, R.J. Rodriguez, and Z. Zhang, *Modeling and analysis of high availability techniques in a virtualized system*, Comput. J 61 (2018), pp. 180–198. doi:10.1093/comjnl/bxx049.

[51] M. Torquato, P. Maciel, and M. Vieira, *A model for availability and security risk evaluation for systems with VMM rejuvenation enabled by VM migration scheduling*, ACCESS (2019), pp. 138315–138326. doi:10.1109/ACCESS.2019.2943273.

[52] K. Ito and F. Kappel, *The Trotter-Kato theorem and approximation of PDEs*, Math. Computer 67 (1998), pp. 21–44.

[53] J.H. Cao and K. Cheng, *Introduction to Reliability Mathematics*, Higher Education Press, Beijing, 1986.

[54] B.R. Fang, J.D. Zhou, and Y.M. Li, *Matrix Theory*, Tsinghua University Press, Beijing, 2004.

[55] Y.B. Li, G.Q. Qin, and Z.H. Wang, *The Foundation of the Bounded Linear Operators Semigroup with Application*, Liaoning Science and Technology Press, Liaoning, 1992.
Appendix A: Prove that the system operator $A+B$ is the generator of a positive $C_0$-semigroup $T(t)$

Firstly, we prove that $D(A+B)$ is dense in $X$. To this end, we prove that \( \{ y \in C \mid \text{Re} y > 0 \} \) or $y = ia, a \in \mathbb{R} \setminus \{ 0 \}$ belongs to the resolvent set of the system operator $A+B$.

For any $\text{Re} y > 0$ or $y = ia, a \in \mathbb{R} \setminus \{ 0 \}$, $G = (g_0, g_1, g_2(x))^T \in X$, consider the resolvent equation $(yI - A - B)\tilde{P} = G$, that is

\[
(y + \alpha_0 + \beta)p_0 = g_0 + \int_0^\infty \mu(x)p_2(x)dx, \quad (A1)
\]
\[
(y + \alpha_1)p_1 - \beta p_0 = g_1, \quad (A2)
\]
\[
\frac{dp_2(x)}{dx} + (y + \mu(x))p_2(x) = g_2(x), \quad (A3)
\]
\[
p_2(0) = \alpha_0p_0 + \alpha_1p_1. \quad (A4)
\]

Calculating (A3), we obtain

\[
p_2(x) = p_2(0)e^{-\int_0^x (y + \mu(t))dt} + \int_0^x g_2(\tau)e^{-\int_\tau^x (y + \mu(t))dt}d\tau, \quad (A5)
\]

which yields $p_2(x) \in L^1[0, \infty)$ [53]. Integrate the equation (A3) from $x$ equals $0$ to $\infty$ on both sides obtain $\int_0^\infty \mu(x)p_2(x)dx = \int_0^\infty g_2(x)dx + p_2(0) - y\int_0^\infty p_2(x)dx$, which is a finite value. Substitute equation (A5) into equation (A1), we have

\[
(y + \alpha_0 + \beta - \alpha_0Q_1)p_0 - \alpha_1Q_1p_1 = g_0 + Q_2, \quad (A6)
\]

where $Q_1 = \int_0^\infty \mu(x)e^{-\int_0^x (y + \mu(t))dt}dx, Q_2 = \int_0^\infty \mu(x)\int_0^x g_2(\tau)e^{-\int_\tau^x (y + \mu(t))dt}d\tau dx.$

By combing with equations (A6) and (A2), we can derive the following matrix equation

\[
\begin{bmatrix}
\alpha_0 + \beta - \alpha_0 & -\alpha_1Q_1 \\
-\beta & \gamma + \alpha_1
\end{bmatrix}
\begin{bmatrix}
p_0 \\
p_1
\end{bmatrix}
= \begin{bmatrix}
g_0 + Q_2 \\
g_1
\end{bmatrix}. \quad (A7)
\]

Obviously, the matrix equation has a unique solution if and only if the coefficient determinant is nonzero. When $\text{Re} y > 0$, the coefficient determinant $D(y) \neq 0$.

Let $p_0, p_1$ be the solution of the matrix equation (A7), then $\tilde{P} = (p_0, p_1, p_2(x))^T \in D(A + B)$ is the solution of the resolvent equations (A1)-(A3). Therefore, \( \{ y \in C \mid \text{Re} y > 0 \} \) or $y = ia, a \in \mathbb{R} \setminus \{ 0 \}$ belongs to the resolvent set of the system operator $A+B$.

Defining $\tilde{P}_y = y(yI - A - B)^{-1}G$ for all $G = (g_0, g_1, g_2(x))^T \in X$ and $y > 0$. Since $\tilde{P}_y \in D(A + B)$, it is sufficient to show that $\tilde{P}_y \rightarrow G(y \rightarrow \infty)$ in $X$. From the resolvent equations (A1)-(A3), we have $\tilde{P}_y = y(p_0, p_1, p_2(x))^T$, here

\[
p_0 = \frac{g_0 + \int_0^\infty \mu(x)p_2(x)dx}{y + \alpha_0 + \beta}, p_1 = \frac{\beta(g_0 + \int_0^\infty \mu(x)p_2(x)dx)}{(y + \alpha_1)(y + \alpha_0 + \beta)} + \frac{g_1}{y + \alpha_1},
\]
\[
p_2(x) = (\alpha_0p_0 + \alpha_1p_1)e^{-\int_0^x (y + \mu(t))dt} + \int_0^x g_2(\tau)e^{-\int_\tau^x (y + \mu(t))dt}d\tau,
\]

\[
p_0(x) = \frac{g_0 + \int_0^\infty \mu(x)p_2(x)dx}{y + \alpha_0 + \beta}, p_1 = \frac{\beta(g_0 + \int_0^\infty \mu(x)p_2(x)dx)}{(y + \alpha_1)(y + \alpha_0 + \beta)} + \frac{g_1}{y + \alpha_1},
\]
\[
p_2(x) = (\alpha_0p_0 + \alpha_1p_1)e^{-\int_0^x (y + \mu(t))dt} + \int_0^x g_2(\tau)e^{-\int_\tau^x (y + \mu(t))dt}d\tau,
\]
\[
\int_0^\infty \mu(x)p_2(x)dx = \frac{M_1 + (y + \alpha_1)(\gamma + \alpha_0 + \beta) \int_0^\infty g_2(\tau)e^{-\int_\tau^\infty (\gamma + \mu(\xi))d\xi}d\tau dx}{y^2 + (\beta + \alpha_1)y + [(y + \alpha_1)\alpha_1 + \alpha_1 \beta]y\int_0^\infty e^{-\int_0^\infty (\gamma + \mu(\xi))d\xi}dx},
\]

\[M_1 = [(\alpha_0(y + \alpha_1) + \alpha_1 \beta)g_0 + \alpha_1(y + \alpha_0 + \beta)g_1]\int_0^\infty \mu(x)e^{-\int_0^\infty (\gamma + \mu(\xi))d\xi}dx.
\]

Then we can write \(\|\tilde{P}_y - \tilde{G}\| \leq U_1 + U_2\), where

\[
U_1 = \frac{(-\alpha_0 + \beta)g_0 + p_3}{y^2 + \alpha_1 y + \alpha_1 \beta + \alpha_0 \beta}\left|\frac{\beta y + \int_0^\infty \mu(x)p_2(x)dx}{(y + \alpha_1)(\gamma + \alpha_0 + \beta)}\right| + \left|\frac{\alpha_1 g_1}{y^2 + \alpha_1 y + \alpha_1 \beta + \alpha_0 \beta}\right| + \left|\frac{\alpha_0 g_0 + \int_0^\infty \mu(x)p_2(x)dx}{(y + \alpha_1)(\gamma + \alpha_0 + \beta)}\right| + \left|\frac{\alpha_1 g_1}{y^2 + \alpha_1 y + \alpha_1 \beta + \alpha_0 \beta}\right|,
\]

\[
U_2 = \int_0^\infty |\int_0^\infty yg_2(\tau)e^{-\int_\tau^\infty (\gamma + \mu(\xi))d\xi}d\tau - g_2(x)|dx.
\]

It is easy to obtain that \(\lim_{y \to 0} J_1 = 0\). On the other hand, \(U_2 \leq \int_0^\infty |\int_0^\infty ye^{-\gamma g_2(x-s)}ds - g_2(x)|dx\). Let \(s > 0\) and define \(g_2^s \in L^1\) by

\[
g_2^s(x) = \begin{cases} 0, & 0 < x < s \\ g_2(x-s), & \text{a.e. } x \geq s \end{cases}
\]

Since \(\|g_2^s - g_2\|_{L^1} = \int_0^s |g_2(x)|dx + \int_0^\infty |g_2(x) - g_2(x+t)|dx\), we have \(\lim_{s \to 0^+} \|g_2^s - g_2\|_{L^1} = 0\) and

\[
\int_0^\infty |\int_0^\infty ye^{-\gamma g_2(x-s)}ds - g_2(x)|dx \to 0, \; (y \to \infty),
\]

which yields \(\lim_{y \to \infty} U_2 = 0\). Thus, we come to the conclusion that \(\lim_{y \to \infty} \|\tilde{P}_y - \tilde{G}\| = 0\). This indicates that \(D(A + B)\) is dense in \(X\).

Secondly, we prove that the system operator \(A + B\) is a resolvent positive operator. For any given \(\bar{Y} = (y_0, y_1, y_2(x))^T \in X\), consider the equation \((yI - A)\bar{P} = \bar{Y}\), that is,

\[
\begin{cases}
(y + \alpha_0 + \beta)p_0 = y_0 \\
(y + \alpha_1)p_1 = y_1 \\
\left(\frac{d}{dx}g_2(x) + (y + \mu(x))p_2(x)\right) = y_2(x)
\end{cases}
\]

(A8)

We can derive that \(\|\gammaI - A\|^{-1} \leq \frac{1}{\gamma} \) for \(\gamma > 0\) by solving equation (A8). When \(\bar{Y}\) is nonnegative, \(\bar{P} = (yI - A)^{-1}\) is also nonnegative. This implies that \(A\) is a resolvent positive operator. From the expression of operator \(B\), it is known that \(B\) is a positive operator, and \(\|B\| \leq \sup\{\int_0^\infty \mu(x)dx, \beta\} = M\). Therefore, when \(\gamma > M\), \(\|(yI - A)^{-1}B\| \leq \|(yI - A)^{-1}\| \cdot \|B\| \leq \frac{M}{\gamma} < 1\). According to the theorem of Reference [54], we have \([I - (yI - A)^{-1}B]^{-1}\) is also a positive operator. Therefore, \(A + B\) is a resolvent positive operator.

Finally, we prove that the system operator \(A + B\) generates a positive \(C_0\)-semigroup \(T(t)\). For any \(\bar{Q} \in D(A + B)\), \(\bar{Q} = (q_0, q_1, q_2(x))^T \in D(A + B)^* = \{(q_0, q_1, q_2(x))^T \in X^*| q_2(x) \text{ is absolutely continuous fuction}, \frac{dq_2(x)}{dx} \in L^\infty[0, \infty]\} \) and \(\forall \bar{Q} \in X^*\), by calculating \((A + B)\bar{P}, \bar{Q})\), we have

\[
(A + B)^*\bar{Q} = \begin{pmatrix}
-\alpha_0 \beta + \alpha_0 q_0 + \beta q_1 + \alpha_0 q_2(0) \\
\alpha_1 \beta q_1 + \alpha_1 q_2(0) \\
\left(\frac{d}{dx} - \mu(x)\right) q_2(x) + \mu(x)q_0
\end{pmatrix}
\]

(A9)
Defining the positive cone in the system as \( X_+ = \{(p_0, p_1, p_2(x))^T \in X | p_0 \geq 0, p_1 \geq 0, p_2(x) \geq 0, \forall x \in [0, \infty)\} \) and positive cone of the adjoint space \( X^* \) of \( X \) as \( X^*_+ = \{(q_0, q_1, q_2(x))^T \in X^* | q_0 \geq 0, q_1 \geq 0, q_2(x) \geq 0, \forall x \in [0, \infty)\} \) [55].

According to Theorem 3.1, we have \( D((A + B)^*)_+ = X^*_+ \cap D((A + B)^*) \). For any given \( F = (F_0, F_1, F_2(x))^T \in X^*_+ \), we denoted \( M_1 = \max(|F_0|, |F_1|, \|F_2\|_{L^\infty[0, \infty)}) \), since \( 1(x) \in L^\infty[0, \infty) \) is absolutely continuous and \( \frac{d}{dx}1(x) \in L^\infty[0, \infty) \), then

\[
G = (M_1, M_1, M_1 \cdot 1(x))^T \in D((A + B)^*)_+, \quad G \geq F,
\]

which implies that \( D((A + B)^*)_+ \) is cofinal in \( X^*_+ \). Theorem 3.1 implies that \( A + B \) generates a positive \( C_0 \)-semigroup \( T(t) \).

**Appendix B: Prove that 0 is the simple eigenvalue of the system operator \( A + B \)**

Consider the equation \((A + B)\vec{p} = 0\). That is

\[
-(\alpha_0 + \beta)p_0 + \int_0^\infty \mu(x)p_2(t, x)dx = 0 \tag{B1}
\]

\[
\alpha_1p_1 = \beta p_0 \tag{B2}
\]

\[
\frac{dp_2(x)}{dx} + \mu(x)p_2(x) = 0 \tag{B3}
\]

Solving the equations (B1)-(B3) with boundary condition (13) yields

\[
p_1 = \frac{\beta}{\alpha_1}p_0, \quad p_2(x) = \left(\alpha_0 p_0 + \alpha_1 p_1\right) e^{-\int_0^x \mu(\xi)d\xi} = \left(\alpha_0 + \alpha_1\right)p_0 e^{-\int_0^x \mu(\xi)d\xi}. \tag{B4}
\]

Substitute (B4) into (B1), we obtain \(-\left(\alpha_0 + \beta\right) + \left(\alpha_0 + \beta\right)p_0 = 0\). Obviously, \( p_0 \) is an arbitrary real number. Then \( \vec{p} \in D(A) \) and suppose \( p_0 > 0 \). So the vector \( \vec{P}^* = (p_0, p_1, p_2(x))^T \) is the positive eigenfunction of the system operator \( A + B \) corresponding to the eigenvalue 0. Let \( \vec{Q} = (1, 1, 1(x))^T \), then \( \langle \vec{P}^*, \vec{Q} \rangle = p_0 + p_1 + \int_0^\infty p_2(x)dx > 0 \) and for any \( \vec{P} \in D(A + B) \), \( \langle (A + B)\vec{P}, \vec{Q} \rangle = 0 \), that is, \( (A + B)^*\vec{Q} = 0 \). So 0 is the simple eigenvalue of the system operator \( A + B \).