Wireless Network Calculus - Modeling and Analyzing Retransmission-based Wireless Link using Network Calculus

Supervisor: Prof. Dr.-Ing. Jens B. Schmitt

Author: Hao Wang

Matrikel-Nr.: 359487

Disco-D-016

Distributed Computer Systems Lab
Computer Science Department
University Kaiserslautern

May 12, 2009

Abstract

By dint of advancing network calculus, this thesis analyzes the retransmission-based wireless link and builds a model for it. As known, wireless link is inherently error prone, where data loss often occurs, and according to link layer ARQ protocols the lost data should be retransmitted, so we propose a retransmission-based model for wireless link. First of all, we use network calculus to describe data loss in wireless link, which is a stochastic process. Thus it is not enough to directly apply the deterministic scaling. Extension from the deterministic scaling to stochastic scaling therefore becomes necessary. Particularly, as examples of stochastic scaling, Binary Symmetric Channel, Gilbert-Elliott Channel and Finite-State Markov Channel are represented. Using this new calculus of stochastic scaling, a fluid model of retransmission is subsequently established. Then for some types of given arrival curve and service curve, the arrival curves of retransmission flows are formulated. Furthermore, the performance bounds can be obtained too. A numerical example shows that this model can provide reasonable results. Hence the further application of the model is considerable.

Keywords

Network calculus, stochastic scaling curve, wireless channel, retransmission flow, fixed point

Contents

1 Introduction

During the last decades, network calculus [10] as a new theory of analyzing performance of networked systems has gained more and more concerns. Unlike traditional queuing system, network calculus discarded the very detailed way of analysis as queuing theory did, it introduced a more high-leveled but easily applicable theory viewpoint. Since the QoS guarantee of networked system can tolerate delay and data loss to some degree, it will be hard to dynamically catch the fluctuant network status but generally easy to analyze the worst case, and for doing that, network calculus is a vigorously developed theory tool. Based on min-plus algebra, some important concepts like service curve and arrival curve are introduced. Different performance bounds are calculated. Linear scaling functions are used to abstract some system behaviors around data forwarding. And these theory bases facilitate the efficient end-to-end analysis like e.g. tandem scaled servers or feed-forward network etc, and accordingly, support analyzing and modeling real networked systems. The application of network calculus in wireless sensor network is such an example [18, 19]. At present, there are already rich useful sets of theories or methods though, as a growing theory, network calculus still has some open fundamental problem areas, such as stochastic generalization, data transformations, non-FIFO systems and so on. The research on these issues will make the theory more flexible and more applicable in practice.

Why are we interested in stochastic extension of network calculus in this thesis? As stated above, the demand on transmitting data or other real-time application over network is not so strict that some delay or data loss can still be tolerable. It implies that the research should not only focus on the deterministic network calculus for the worst case but also as a more flexibly applicable extension, develop a stochastic option as the theoretical counterpart. Actually, the stochastic analysis is already identified as a grand challenge for future research of information theory, which is another reason to research it in the domain of network calculus. In addition, because to be modeled wireless link is a stochastic link in the sense of that the capacity of the channel may vary with time randomly due to some possible physical causes like channel impairment or contention, the stochastic network calculus should play an important role when analyzing the link part of the model.

In recent years, there have been many attempts made for stochastic network calculus. They laid emphases on different aspects of network calculus and analyzed the stochastic settings of them. As extensions of deterministic arrival curve and service curves, effective envelope [1] and effective service curve [2] were introduced. And

based on these, as an application, [3] reconciled two principal tools for analyzing network traffics, namely, effective bandwidth and network calculus. There, probabilistic arrival and service bounds are specified and furthermore, effective bandwidth is generally formulated within such framework of stochastic network calculus. After that, [4] has developed stochastic service curve and analyzed end-to-end system and created the model of network service curve. [5] contributed the end-to-end analysis using moment generating function. [6] has generalized stochastic arrival and service curve and various basic results were derived, which included stochastic delay and backlog bounds. The research above focused mainly on arrival data and service. For data transformation process, [7] introduced and defined the concept of scaling function to describe the possible altering of data. As an abstraction of processing data, service curve has its stochastic extension, accordingly there should remain some open tasks to extend data scaling to a stochastic version. This is one original idea to extend data scaling. The other is that data forwarding itself is a stochastic process, which was also a theme of queuing system, so in network calculus area we succeed to this theme.

After we have created the core and basic theory framework of network calculus, we will be eager to apply them to build some models or to obtain some experience when analyzing system with them. If we have tried to survey the way of developing and applying network calculus, we will find that it is planted in fertile soil of information theory, it closely follows the footprints of the development of the theories, experience and models of information theory. Along this routine, i.e. apply network calculus theory to in information theory existed structures, models, protocols or some physical settings, some works have been done and particularly, some models and frameworks have be developed. For example, on the basis of existing catalog of network connectivity models, we can apply scaling function and its inverse to the tandem network analysis, or we can apply multiplexing results of network calculus to feed-forward network analysis, just like what have been done in [7, 8, 18]. There are some other similar examples: [9, 11]. The task in this thesis is no other than such routine: extend network calculus theory and apply it. Merely the object is a different one: data retransmission in wireless link. At the same time, the new extended network calculus would be only a part of theory bases of the model. Nevertheless, although only as a part of the model, the theory research is not easy. So it will cost much effort. After theory extension, we try to characterize some special data scalings and get their stochastic scaling curves.

The main task of this thesis is to analyze and model retransmission-based wireless link with network calculus. As common routine stated above, our modeling begins from an existed protocol - selective ARQ [12]. Before apply stochastic network calculus to selective ARQ protocol, let us know about the enunciation of the protocol. The behavior of selective ARQ is, the sending process continues to send a number of frames specified by a window size even after a frame loss. Unlike Go-Back-N ARQ, the receiving process will continue to accept and acknowledge frames sent after an initial error. This behavior will generate two continuous flows, one goes forward, the other goes back.

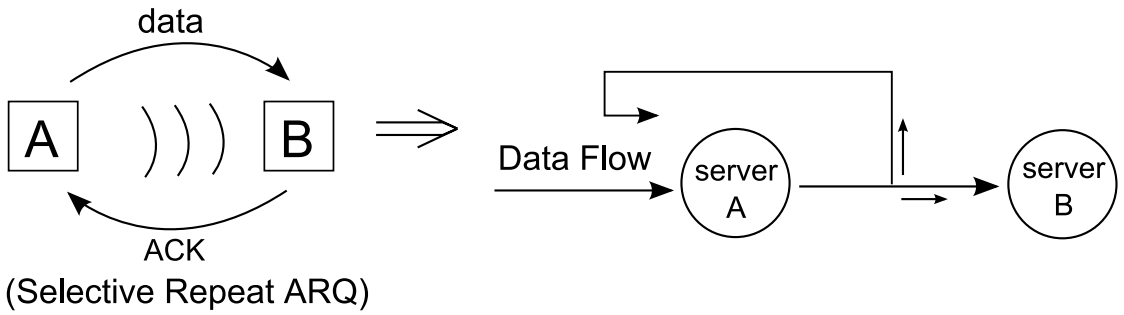

Fig. 1 Rough idea of model

From left to right side, the figure shows an abstract transform. When data arrive at the receiving server i.e. server B in fig. 1, the lost data go back at once and will be retransmitted, that seems like, some in the channel being transmitted data flow back and are retransmitted and the rest part of data go ahead. We can think this model as a fluid model at first. The model will be briefly composed of three parts: the arrival flow, server and divided flows after server. In order to build this model, the first as well as the most important consideration is how to abstract retransmission behavior using network calculus. In this thesis, this behavior is abstractly viewed as being caused by stochastic scaling. A second consideration is how to calculate the retransmission flows. Then the third consideration is how to measure the performance bounds.

From above description, we know that our model focus on data error in wireless link. In fact, there were some related work, which also used network calculus to model wireless link or research into network components with error, however, to the best of our knowledge, few. The earliest research on stochastic service [25] has mentioned ARQ and "channel impairment" process. But there ARQ and "channel impairment" process were just assistant components of providing scheduling. The effort made of that work was mainly to show what are the stochastic service constraints and how service curves can be stochastically delivered over a randomly varying channel. It had not yet known the concept of scaling on the one hand. On the other hand, that work did not consider retransmission. Another work [26] has proposed a model of wireless link using network calculus. It researched data loss in aim of the concept clipper introduced in [10]. Clipper was used in transmitter or receiver. So it is not so general as scaling for abstracting some mid-boxes in the network. At the same time, although retransmission was introduced into the model, it has not yet considered the influence of server on the retransmitted data. But that is logical, because it did not treat data loss as scaling, i.e. as a separated component from the server. Moreover, that research did not definitely present stochastic aspect of wireless link, merely modeled the wireless link as Finite-State Markov Channel. In another study [27], data loss or more generally say data error, was thought and abstracted as a stochastic process. However, this time once again like [25], the introduced error process was composed into server, in order to propose the so called error server model. Although when analyzing the concatenation property of error server model, the error process

was separated from server, it did not yet create a new component of network calculus, hence lost flexibility of application. This research has discussed retransmission. Nevertheless, the retransmission there was not regarded as an object to research but only as a factor influencing the error process to mention.

The remaining chapters are organized as follows. In the next chapter we recall some important concepts and theorems of network calculus. In chapter 3, deterministic scaling is extended to a stochastic version and some examples of stochastic scaling are given. In chapter 4, the retransmission-based model using network calculus is built. In chapter 5, a numerical example is given and the concluded methods in former chapters are tested. We finally discuss some open future work in chapter 6.

2 Network Calculus Basics

Network calculus is a system theory applied to computer networks. It is a theory of deterministic queuing systems. The main difference between network calculus and traditional system theory is that network calculus changes the algebra foundation. Instead of the old algebra basis it uses Min-Plus dioid (also called Min-Plus algebra), where addition becomes minimum and multiplication becomes addition. Based on such an algebra, many concepts can be reformulated, like the important concept of minimum service curve introduced in [20, 21, 22, 23] and the concept of maximum service curve in [24], further the definition of performance bounds and the analysis of tandem queues.

The mostly used set of functions by network calculus is the set of real-valued, non-negative, and wide-sense increasing functions.

$$\mathcal{F} = \{f : \mathbb{R}^+ \to \mathbb{R}^+,\, f(t) \geq f(s)\, \forall t \geq s \geq 0,\, f(0) = 0\}.$$

Because this set of functions can cumulate data along the positive time axis, it is suitable and reasonable to represent some physical circumstance mathematically. It is imaginable that many assumptions are made on this set. It is convenient to generally describe data flows as a function belonging to the space $\mathcal{F}$. How to go along the positive time axis? There are two ways: discrete and continuous. In real systems, because the data granularity can not be infinitely small (packet, word, bit) and a time unit relates to cumulating a data unit, discrete time could always be assumed. However, it is always mathematically simpler to consider continuous time as well as continuous data. We call that *fluid model*. We do assume time and data to be continuous firstly, that is $t \in \mathbb{R}^+ = [0, \infty]$. Then we say that it is always possible to map a continuous time model to a discrete time model with $n \in \mathbb{N}_0$, because we can always choose a time slot δ as a time unit and sample at $t = n\delta$ to the function with continuous time. When we want to build a retransmission-based model for wireless network using network calculus, we also assume the retransmitted data flow to be continuous.

Now, let us bill the mathematical background (Min-Plus calculus respectively Max-Plus calculus) and its applications in network calculus.

Definition 2.1 (Convolution and De-Convolution) Min-Plus convolution and de-convolution of two functions f and g are defined to be

$$h(t) = (f \otimes g)(t) = \inf_{s \in [0,t]} \{f(t-s) + g(s)\}$$
$$h(t) = (f \oslash g)(t) = \sup_{s \in [0,\infty)} \{f(t+s) - g(s)\}.$$

Max-Plus convolution and de-convolution are defined to be

$$h(t) = (f \overline{\otimes} g)(t) = \sup_{s \in [0,t]} \{f(t-s) + g(s)\}$$
$$h(t) = (f \overline{\oslash} g)(t) = \inf_{s \in [0,\infty)} \{f(t+s) - g(s)\}.$$

Because Max-Plus algebra is dual to Min-Plus algebra with similar properties when infimum is replaced by supremum as in definition 2.1 and network calculus use more Min-Plus, the following properties is only listed for Min-Plus convolution.

Theorem 2.1 (Properties of $\otimes$) *Let f, g, $h \in \mathcal{F}$. The properties hold:*

1. *Closure of $\otimes$: $(f \otimes g) \in \mathcal{F}$,*
2. *Commutativity of $\otimes$: $f \otimes g = g \otimes f$,*
3. *Associativity of $\otimes$: $(f \otimes g) \otimes h = f \otimes (g \otimes h)$.*

Note that de-convolution is not closed in $\mathcal{F}$ and not commutative.

Consider now the application of mathematical theory above in network calculus. Assume S to be a black box system, which receives input data and delivers output data after a delay. We can utilize cumulative function of $\mathcal{F}$ to describe arrival traffic flows.

Fig. 2 Simple system illustration

Definition 2.2 (Arrival Function) An arrival function is a cumulative function $F(t) \in \mathcal{F}$ that is defined to be the amount of data of a flow seen in the interval $[0, t]$.

On the basis of arrival function we can describe the input flow and the output flow of system S stated above and further derive the following two performance measures of interest.

Definition 2.3 (Backlog and Delay) Assume a server has input arrival function $F(t)$ and output arrival function $F'(t)$. The backlog at the server for time t is defined to be

$$b(t) = F(t) - F'(t).$$

Assuming first-in first-out data delivery the virtual delay for time t is defined as

$$d(t) = inf\{\tau \geq 0 : \ F(t) \leq F'(t+\tau)\}.$$

The backlog is the amount of bits that are held inside the system S. The virtual delay at time t is the delay that would be experienced by a bit arriving at time t if all bits received before it are served before it. Graphically, the backlog is the vertical deviation between input and output arrival functions; the virtual delay is the horizontal deviation between them.

We can describe traffic flows. If we want to limit them, we use arrival curve to specify upper bound on the amount of traffic. Accordingly, we use service curve to specify the upper bound on the service offered by servers respectively lower bound.

Definition 2.4 (Arrival Curve) Consider an arrival function $F(t)$. Any function $\alpha(t) \in \mathcal{F}$ is said to be an arrival curve of $F(t)$ if for all $t \geq 0$ it holds that

$$\alpha(t) \geq (F \oslash F)(t).$$

Definition 2.5 (Service Curves) Consider a server with input and output arrival function $F(t)$ and $F'(t)$ respectively. Any two functions $\beta(t) \in \mathcal{F}$ and $\gamma(t) \in \mathcal{F}$ are said to be a minimum respectively maximum service curve of the server if for all $t \geq 0$ it holds that

$$\begin{aligned} F'(t) &\geq (F \otimes \beta)(t) \\ F'(t) &\leq (F \otimes \gamma)(t). \end{aligned}$$

Generally the set of arrival curves is limited to sub-additive functions, since there is a tighter upper bound than $\alpha(t)$ which is sub-additive closure of $\alpha(t)$.

Definition 2.6 (Sub-Additive Function) Let f be a function or a sequence of $\mathcal{F}$. Then f is sub-additive if and only if $f(t+s) \leq f(t) + f(s)$ for all $s,\ t \geq 0$.

Definition 2.7 (Sub-Additive Closure) Let f be a function or a sequence of $\mathcal{F}$. Denote $f^{(n)}$ the function obtained by repeating $(n-1)$ convolutions of f with itself. By convention, $f^{(0)} = \delta_0$, so that $f^{(1)} = f$, $f^{(2)} = f \otimes f$, etc. Then the sub-additive closure of f, denoted by $\overline{f}$, is defined by

$$\overline{f} = \delta_0 \wedge f \wedge (f \otimes f) \wedge (f \otimes f \otimes f) \wedge ... = \inf_{n \geq 0}\{f^{(n)}\}.$$

So far, we can characterize the simple system model in fig. 2 using arrival function, arrival and service curve as fig. 3.

Fig. 3 Simple system model using network calculus

Accordingly, three performance bounds can be derived, which are output bound, backlog bound and delay bound.

Theorem 2.2 (Performance Bounds) *Consider a server which offers a minimum service curve $\beta(t)$. Let the input to the server be upper constrained by an arrival curve $\alpha(t)$. An output arrival curve of the server is*

$$\alpha'(t) = (\alpha \oslash \beta)(t).$$

The backlog at the server is upper bounded by

$$b \leq (\alpha \oslash \beta)(0).$$

Assuming first-in first-out order the delay is upper bounded according to

$$d \leq inf\{t \geq 0 : \ (\alpha \oslash \beta)(-t) \leq 0\}.$$

If we look into the graphics, the backlog bound is the maximum vertical deviation of arrival and service curve and the delay bound is the maximum horizontal deviation. With server also offering maximum service curve we can refine the output bound as $\alpha'(t) = ((\alpha \otimes \gamma) \oslash \beta)(t)$.

Theorem 2.3 (Tightness of Bounds) *The backlog and the delay bound in Th. 2.2 are tight, that is there exists a sample path such that the bounds hold with equality, if α, $\beta \in \mathcal{F}$ are tight, which implies that α is sub-additive. If in addition α is left-continuous and $\alpha \overline{\oslash} \alpha$ is not bounded from above then the output bound is also tight.*

Theorem 2.4 (Concatenation) *Consider two servers with minimum service curves $\beta_1(t)$ and $\beta_2(t)$ and maximum service curves $\gamma_1(t)$ and $\gamma_2(t)$ in sequence. There exists an equivalent single server system with minimum and maximum service curve*

$$\begin{aligned} \beta(t) &= (\beta_1 \otimes \beta_2)(t) \\ \gamma(t) &= (\gamma_1 \otimes \gamma_2)(t). \end{aligned}$$

The proof of Th. 2.4 follows from that $((F \otimes \beta_1) \otimes \beta_2)(t) = (F \otimes (\beta_1 \otimes \beta_2))(t)$. The same for maximum service curves. The theorem of concatenation is important for end-to-end analysis. Another issue for end-to-end analysis is multiplexing. Accordingly there are some theories introduced in [10] and [8].

Definition 2.8 (Strict Service Curve) Let $\beta \in \mathcal{F}$. System S offers a strict service curve β to a flow if, during any backlogged period of duration u the output of the flow is at least equal to $\beta(u)$.

Note that any strict service curve is also a service curve.

Theorem 2.5 (Left-over Service Curve under Arbitrary Multiplexing) *Consider a node multiplexing two flows 1 and 2 in arbitrary order. Assume that the node guarantees a strict minimum service curve β to the aggregate of the two flows. Assume that flow 2 has α_2 as an arrival curve. Then*

$$\beta^1 = [\beta - \alpha_2]^+$$

is a service curve for flow 1 if $\beta^1 \in \mathcal{F}$, often also called the left-over service curve for the flow of interest.

If data are processed and altered during forwarding, the amount of data may be changed. To cope with that, [7] has introduced and defined the concepts of scaling function and scaling curves accordingly, which extend the framework of network calculus.

Definition 2.9 (Scaling Function) A scaling function $S \in \mathcal{F}$ assigns an amount of scaled data $S(a)$ to an amount of data a.

Corollary 2.1 (Inverse Scaling Function) *Given a bijective scaling function $S \in \mathcal{F}$ it follows for its inverse S^{-1} that $S^{-1} \in \mathcal{F}$ is a scaling function, too.*

Definition 2.10 (Scaling Curves) Consider a scaling function S. Any two functions $\underline{S} \in \mathcal{F}$ and $\overline{S} \in \mathcal{F}$ are said to be a minimum respectively maximum scaling curve of S if for all $b \geq 0$ it holds that

$$\begin{aligned} \underline{S}(b) &\leq \inf_{a \in [0,\infty)} \{S(b+a) - S(a)\} = (S \overline{\oslash} S)(b) \\ \overline{S}(b) &\geq \sup_{a \in [0,\infty)} \{S(b+a) - S(a)\} = (S \oslash S)(b). \end{aligned}$$

Corollary 2.2 (Inverse Scaling Curves) *Consider a bijective scaling function S* and let $\underline{S}$ and $\overline{S}$ be the respective minimum and maximum scaling curves. If $\underline{S}$ and $\overline{S}$ are bijective, a valid maximum scaling curve of the inverse scaling function $\overline{S^{-1}}$ is $\underline{S}^{-1}$ and a valid minimum scaling curve of the inverse scaling function $\underline{S^{-1}}$ is $\overline{S}^{-1}$.

Now the concatenation of systems becomes more complex, because there exists scaler between two servers. The theorem below about scaled server is of particular importance.

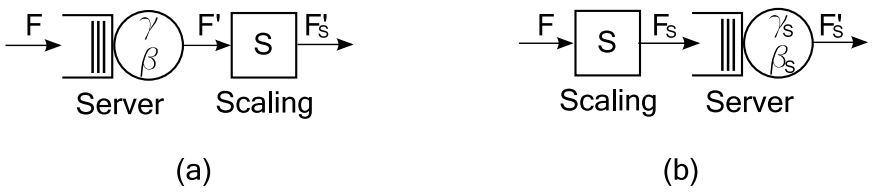

Fig. 4 Scaling of servers

Theorem 2.6 (Scaled Server) *Consider the two systems in Fig. 4 and let $F(t)$ be the input arrival function. System (a) consists of a server with minimum and maximum service curve $\beta(t)$ and $\gamma(t)$ respectively whose output is scaled with scaling function S and system (b) consists of a scaling function S whose output is input to a server with minimum and maximum service curve $\beta_S(t)$ and $\gamma_S(t)$ respectively. Given system (a), the lower and upper bounds of the output arrival function of system (b), that are $(S(F) \otimes \beta_S)(t)$ and $(S(F) \otimes \gamma_S)(t)$ respectively, are also valid lower and upper bounds of the output arrival function of system (a) if*

$$\begin{aligned} \beta_S(t) &= \underline{S}(\beta(t)) \\ \gamma_S(t) &= \overline{S}(\gamma(t)) \end{aligned}$$

where $\underline{S}$ and $\overline{S}$ are the respective scaling curves of S. Given system (b), the lower and upper bounds for the output arrival function of system (a), that are $S((F \otimes \beta)(t))$ and $S((F \otimes \gamma)(t))$ respectively, hold also for system (b) if S is bijective and

$$\begin{aligned} \beta(t) &= \underline{S^{-1}}(\beta_S(t)) \\ \gamma(t) &= \overline{S^{-1}}(\gamma_S(t)) \end{aligned}$$

where $\underline{S^{-1}}$ and $\overline{S^{-1}}$ are the respective scaling curves of S^{-1}.

Based on this theorem, the end-to-end concatenation can be easily done like what analyzed in [7].

Thus far, we recalled some concepts and theories of deterministic network calculus. When we characterize systems or calculate system performance, we can use deterministic description such that e.g. all the bound curves deterministically limit the functions. That is not bad, if we change bound curves to limit the functions and thus

we will get different functions for each change of bound curve. We can do this so many times that the varying system is covered and characterized nearly. That is an idea to describe the varying systems, whereas, not so flexible and not so accurate for some cases. For example, if we already know the whole (100%) appearance region of functions, we can further try to characterize part (e.g. 80%) appearance region of functions, which makes our system analysis more flexible. Hence we need to introduce stochastic issues into our deterministic theory.

Now let us see the stochastic extension of network calculus using Min-Plus algebra. The following definitions directly or indirectly come from [3] and [13]. Note that the terminology is some different. But actually they are stochastic arrival curve and stochastic service curves.

Definition 2.11 (Effective Arrival Envelope) An effective envelope for an arrival process $F(t)$ is defined as a function $\alpha^{\varepsilon}(t) \in \mathcal{F}$ such that for all $t \geq 0$

$$Pr\{\alpha^{\varepsilon}(t) \geq (F \oslash F)(t)\} \geq 1 - \varepsilon,$$

where ε is violation probability.

Definition 2.12 (Effective Service Curves) Given a server with input and output arrival function $F(t)$ and $F'(t)$. Functions $\beta^{\underline{\varepsilon}}(t) \in \mathcal{F}$ and $\gamma^{\overline{\varepsilon}}(t) \in \mathcal{F}$ are said to be a minimum respectively maximum effective service curves if for all $t \geq 0$ it holds that

$$\begin{aligned} Pr\{F'(t) \geq (F \otimes \beta^{\underline{\varepsilon}})(t)\} &\geq 1 - \underline{\varepsilon} \\ Pr\{F'(t) \leq (F \otimes \gamma^{\overline{\varepsilon}})(t)\} &\geq 1 - \overline{\varepsilon}, \end{aligned}$$

where $\underline{\varepsilon}$ and $\overline{\varepsilon}$ are violation probability respectively.

3 Stochastic Scaling Curves

As introduced in chapters above, along the footprints "arrival curve - stochastic arrival curve and service curves - stochastic service curves", we briefly try to extend scaling curves to its stochastic version in this chapter. Two immediate questions come up:

- Can deterministic scaling be directly carried over to the stochastic version only through appending the description about probability? Is there some new consideration?
- Is an efficient end-to-end analysis still possible?

It's not hard to give a negative answer to the first question. Adding description about probability to definition of scaling function and scaling curves cause no problem, but if we are scrupulous to observe the two corollaries about the inversion of scaling function and scaling curves, we will find a questionable presupposition, that is "scaling function $S \in \mathcal{F}$ is bijective". Can stochastic scaling functions be bijective and hence invertible? No. We know that scaling mainly characterizes data forwarding. For a stable link, it could be not a problem that we apply deterministic scaling and assume it bijective, because the data forwarding process is smooth. In other words, there generally exists no data cumulating when forwarding (or say, that could be processed by server), which means for two different input data, scaling will not have the same amount of output data generally. We have introduced that, for an instable link like wireless link, we apply stochastic scaling, because scaling function often changes itself stochastically. Physically explain, that could be caused by data loss when scaling. Data loss implies two stochastic bijective scaling functions, but we can regard it as one scaling function. When something is lost, two values a_1 and a_2 are mapped to the same output $S(a_1) = S(a_2)$. So the stochastic scaling function cannot be bijective generally.

Non-bijectivity of stochastic scaling function is the origin of the discuss in this chapter. Non-bijective scaling function has no inverse function. We try to solve it by using pseudo-inverse function firstly. Under this assumption, the definitions and theorems of stochastic scaling function and curves are given and proven. It is still an open way leading to the solution, although we don't adopt it finally. It could be accurate. Nevertheless, for the sake of easy application in practice, we make a little trick to avoid non-bijectivity mathematically and explain its reasonability at the application level. Remaining part of this chapter is the reorganized definitions of stochastic scaling under the assumption that scaling function is bijective.

3.1 Stochastic Extension with Non-bijective Scaling Function

As a theory tool for the succeeding proof, we recall a concept in advance - pseudo-inverse function.

Definition 3.1 (Pseudo-inverse Function) Let f be a function or a sequence of $\mathcal{F}$. The pseudo-inverse of f is the function

$$f_{pseudo}^{-1}(x) = inf\{t\ such\ that\ f(t) \geq x\},\ f_{pseudo}^{-1} \text{can be written as } f_p^{-1}.$$

Fig. 5 shows the pseudo-inverse function.

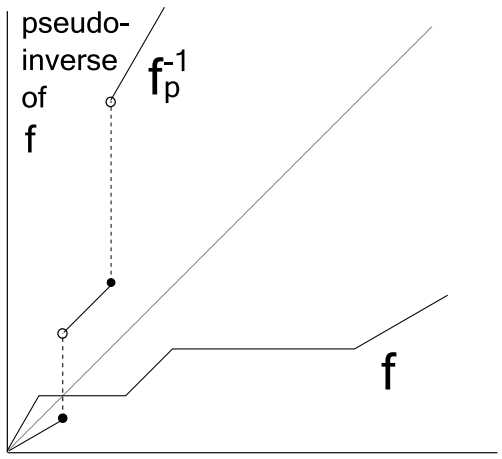

Fig. 5 Pseudo-inverse function

Theorem 3.1 (Properties of Pseudo-Inverse Functions) Let $f \in \mathcal{F}$, $x,\ t \geq 0$.

- (Closure) $f_p^{-1} \in \mathcal{F}$ and $f_p^{-1}(0) = 0$.
- (Pseudo-inversion) We have that

$$\begin{aligned} f(t) \geq x &\Rightarrow f_p^{-1}(x) \leq t \\ f_p^{-1}(x) < t &\Rightarrow f(t) \geq x \end{aligned}$$

- (Equivalent definition)

$$f_p^{-1}(x) = sup\{t\ such\ that\ f(t) < x\}.$$

As an indirect property we introduce a concept: *max deviation* δ, which is of fundamental importance for the latter proofs. We assume f to be a continuous non-bijective function of $\mathcal{F}$, it follows that $f_p^{-1}(f(a))$ may be inequal to a. See fig. 6.

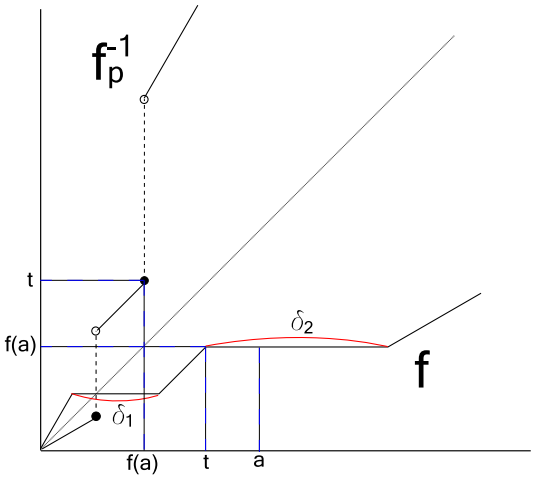

Fig. 6 Deviations and max deviation

It is obvious that $f_p^{-1}(f(a)) = t$. The deviation between $f_p^{-1}(f(a))$ and a is $a - t$. For the first flat segment of f, the deviation could be maximal δ_1, for the second flat segment δ_2. Max deviation $\delta = max(\delta_1, \delta_2, ...)$.

We research the randomicity of the scaling behavior not from the viewpoint of scaling function but from scaling curve. So for each individual scaling function we follow the deterministic definition as definition 2.9. But the proof of the inverse scaling function will be changed as the scaling function is non-bijective.

Corollary 3.1 (Inverse Stochastic Scaling Function) *When scaling function $S \in \mathcal{F}$ is stochastic, it may be not bijective, so S^{-1}doesn't exist but S^{-1}_{pseudo}does. And it follows that $S^{-1}_{pseudo} \in \mathcal{F}$ is also a scaling function.*

Proof: Since $S \in \mathcal{F}$ implies $S: R^+ \to R^+$, the range of S which is the domain of S^{-1}_{pseudo} is R^+. That means S^{-1}_{pseudo} is defined for $R^+$and from the definition of pseudo-inverse, it holds that

$$S^{-1}_{pseudo}(x) = inf\{b\, such\, that\, S(b) \geq x\}.$$

It means that $S^{-1}_{pseudo}(x)$ assigns an amount of scaled data $inf\{b\, such\, that\, S(b) \geq x\}$ to an amount of data x. And because pseudo-inverse function has closure property that $f_p^{-1} \in \mathcal{F}$ and $f_p^{-1}(0) = 0$, S^{-1}_{pseudo} fulfills the definition of a scaling function. □

Definition 3.2 (Stochastic Scaling Curves) Consider a scaling function S. Any two functions $\underline{S}^{\underline{\varepsilon}} \in \mathcal{F}$ and $\overline{S}^{\overline{\varepsilon}} \in \mathcal{F}$ are said to be a minimum resp. maximum stochastic scaling curve of S if for all $b \geq 0$ it holds that

$$Pr(\inf_{a\in[0,\infty)} \{S(b+a) - S(a)\} = (S\overline{\oslash}S)(b) \geq \underline{S}^{\underline{\varepsilon}}(b)) \geq 1 - \underline{\varepsilon}$$

$$Pr(\sup_{a\in[0,\infty)} \{S(b+a) - S(a)\} = (S \oslash S)(b) \leq \overline{S}^{\overline{\varepsilon}}(b)) \geq 1 - \overline{\varepsilon},$$

where $\underline{\varepsilon}$ resp. $\overline{\varepsilon}$ is called s violation probability of scaling function S under lower resp. upper bound.

$\underline{S}^{\underline{\varepsilon}}$ can be seen as a fixed curve. And the randomicity is represented that the probability of "S is under bounded" is greater equal to $1 - \underline{\varepsilon}$, otherwise S is under $\underline{S}$.

Corollary 3.2 (Inverse Stochastic Scaling Curves) *Scaling function $S \in \mathcal{F}$ is no longer bijective but surjective and let $\underline{S}^{\underline{\varepsilon}}$ and $\overline{S}^{\overline{\varepsilon}}$ be the minimum and maximum stochastic scaling curves. $\underline{S}^{\underline{\varepsilon}}$ and $\overline{S}^{\overline{\varepsilon}}$ are not bijective any more. Assume S to be continuous. A maximum stochastic scaling curve of S_p^{-1} is written as $\overline{S_p^{-1}}$ with prob. $\geq 1 - \overline{\varepsilon}_{inv}$, which equals to $\underline{S_p}^{-1} + \delta_{\underline{S}} + \delta_S$ with prob. $\geq 1 - \underline{\varepsilon}$, where δ is the term of max deviation between $S_p^{-1}(S(a))$ and a. Respectively $\underline{S_p^{-1}}$ with prob. $\geq 1 - \underline{\varepsilon}_{inv}$ is $\overline{S}_p^{-1} - \delta_{\overline{S}} - \delta_S$ with prob. $\geq 1 - \overline{\varepsilon}$.*

Proof: Now we prove the first half of the corollary, which is about $\overline{S_p^{-1}}$. If we have the form that

$$\overline{S_p^{-1}}(b-a) \geq S_p^{-1}(b) - S_p^{-1}(a) \text{ with prob. } \geq 1 - \overline{\varepsilon}_{inv}, \tag{3.1}$$

we will get the maximum curve.

For any $b \geq a \geq 0$, $S(b) \geq S(a) \geq 0$. $S(b)$ and $S(a)$ can cover all the range of S, because S is surjective and continuous. If S is not bijective, S^{-1} doesn't exist but S_p^{-1} does. $S(b)$ and $S(a)$ can cover all the domain of S_p^{-1}(although S_p^{-1} have some break points). Then (3.1) is equivalent to the following

$$\overline{S_p^{-1}}(S(b) - S(a)) \geq S_p^{-1}(S(b)) - S_p^{-1}(S(a)). \tag{3.2}$$

As it is easily known that $\underline{S}_p^{-1}$ is wide-sense increasing, we have

$$\underline{S}_p^{-1}(S(b) - S(a)) \geq \underline{S}_p^{-1}(\underline{S}(b-a)) \text{ with prob. } \geq 1 - \underline{\varepsilon}. \tag{3.3}$$

Since $\underline{S}$ is wide-sense increasing and may be not bijective, we can not say that $\underline{S}_p^{-1}(S(b-a)) = b-a$, but we must have

$$\underline{S}_p^{-1}(S(b-a)) \geq b-a-\delta_{\underline{S}}, \textit{ where } \delta_{\underline{S}} \textit{ is max deviation of } \underline{S}_p^{-1}(\underline{S}(x)). \tag{3.4}$$

Then from (3.3) and (3.4) we get

$$\underline{S}_p^{-1}(S(b)-S(a)) + \delta_{\underline{S}} \geq b-a \text{ with prob.} \geq 1-\underline{\varepsilon},$$

and one step forward

$$\underline{S}_p^{-1}(S(b)-S(a)) + \delta_{\underline{S}} + \delta_S \geq b-a+\delta_S, \textit{ where } \delta_S \textit{ is max deviation of } S_p^{-1}(S(x)). \tag{3.5}$$

Because $b \geq S_p^{-1}(S(b))$ and $a \leq S_p^{-1}(S(a)) + \delta_S$, it holds that

$$\begin{aligned} b-a+\delta_S &\geq S_p^{-1}(S(b)) - (S_p^{-1}(S(a)) + \delta_S) + \delta_S \\ &= S_p^{-1}(S(b)) - S_p^{-1}(S(a)). \end{aligned} \tag{3.6}$$

Hence, from (3.5) and (3.6) we get

$$\underline{S}_p^{-1}(S(b)-S(a)) + \delta_{\underline{S}} + \delta_S \geq S_p^{-1}(S(b)) - S_p^{-1}(S(a)).$$

This is just the form of (3.2). Thus consequently $\underline{S}_p^{-1} + \delta_{\underline{S}} + \delta_S$ is $\overline{S_p^{-1}}$ with prob. $\geq 1-\underline{\varepsilon}$.

Then we prove the second half, which is about $\underline{S_p^{-1}}$.

If we have the form that

$$\underline{S_p^{-1}}(S(b)-S(a)) \leq S_p^{-1}(S(b)) - S_p^{-1}(S(a)),$$

we will get the minimum curve.

After using similar reasoning we get

$$\begin{aligned} \overline{S}_p^{-1}(S(b)-S(a)) - \delta_{\overline{S}} - \delta_S &\leq b-a-\delta_S \\ &\leq S_p^{-1}(S(b)) - S_p^{-1}(S(a)). \end{aligned}$$

That means $\underline{S_p^{-1}} = \overline{S}_p^{-1} - \delta_{\overline{S}} - \delta_S$ with prob. $\geq 1-\overline{\varepsilon}$. □

Corollary 3.3 (Sub- and Super-Additive Closure) *Consider a scaling function S with minimum and maximum stochastic scaling curve $\underline{S}^{\underline{\varepsilon}}$ and $\overline{S}^{\overline{\varepsilon}}$. The super-additive closure of $\underline{S}^{\underline{\varepsilon}}$ is a minimum stochastic scaling curve of S and the sub-additive closure of $\overline{S}^{\overline{\varepsilon}}$ is a maximum stochastic scaling curve of S.*

Proof. Def. 3.2 yields: for all $b \geq 0$ and $a \geq 0$, with prob. $\geq 1 - \overline{\varepsilon}$ we have

$$\overline{S}^{\overline{\varepsilon}}(b) \geq S(b+a) - S(a).$$

The probability of positioning $\overline{S}$ doesn't relate to independent variable of S. We have consequently

$$\text{with prob.} \geq 1 - \overline{\varepsilon},\ \overline{S}^{\overline{\varepsilon}}(c-b) \geq S(c+a) - S(b+a) \text{ for all } c \geq b \geq 0,\ a \geq 0.$$

Addition of two inequalities above doesn't affect the probability and yields

$$\text{with prob.} \geq 1 - \overline{\varepsilon},\ \overline{S}^{\overline{\varepsilon}}(c-b) + \overline{S}^{\overline{\varepsilon}}(b) \geq S(c+a) - S(a) \text{ for all } c \geq b \geq 0,\ a \geq 0.$$

Thus

$$\inf_{b \in [0,c]} \{\overline{S}^{\overline{\varepsilon}}(c-b) + \overline{S}^{\overline{\varepsilon}}(b)\} \geq \sup_{a \in [0,\infty)} \{S(c+a) - S(a)\} \text{ with prob.} \geq 1 - \overline{\varepsilon},$$

which can be written as

$$(\overline{S}^{\overline{\varepsilon}} \otimes \overline{S}^{\overline{\varepsilon}})(c) \geq (S \oslash S)(c) \text{ with prob.} \geq 1 - \overline{\varepsilon}.$$

With prob. $\geq 1 - \overline{\varepsilon}$. we get $\overline{S}^{\overline{\varepsilon}}$ and when we get $\overline{S}^{\overline{\varepsilon}}$, we must have $\overline{S}^{\overline{\varepsilon}} \otimes \overline{S}^{\overline{\varepsilon}}$ as a maximum stochastic scaling curve. That means with prob. $\geq 1 - \overline{\varepsilon}$ we have scaling curves $\overline{S}^{\overline{\varepsilon}} \otimes \overline{S}^{\overline{\varepsilon}}$ and $\overline{S}^{\overline{\varepsilon}}$ at the same time. Repeating above step infinitely and taking the infimum of all maximum stochastic scaling curves yields a sub-additive closure. With prob. $\geq 1 - \overline{\varepsilon}$ we can get such sub-additive closure of $\overline{S}^{\overline{\varepsilon}}$ finally. □

Now we extend the theorem of scaled server. It might be the basis analyzing end-to-end system with stochastic scaling.

Theorem 3.2 (Stochastic Scaled Server) *Scaler has stochastic scaling curves. Extend system (a) and (b) in fig. 4 like fig. 7. The crosswise transforming becomes that*

from system (a) to (b)

$$\begin{aligned} \beta_S^{\underline{\varepsilon}}(t) &= \underline{S}(\beta(t)) \text{ with prob. } \geq 1-\underline{\varepsilon} \\ \gamma_S^{\overline{\varepsilon}}(t) &= \overline{S}(\gamma(t)) \text{ with prob. } \geq 1-\overline{\varepsilon}; \end{aligned}$$

from system (b) to (a)

$$\begin{aligned} \beta &= \overline{S}_p^{-1}(\beta_S) - \delta_{\overline{S}} - 2\delta_S \text{ with prob. } \geq 1-\overline{\varepsilon} \\ \gamma &= \underline{S}_p^{-1}(\gamma_S) + \delta_{\underline{S}} + 2\delta_S \text{ with prob. } \geq 1-\underline{\varepsilon}. \end{aligned}$$

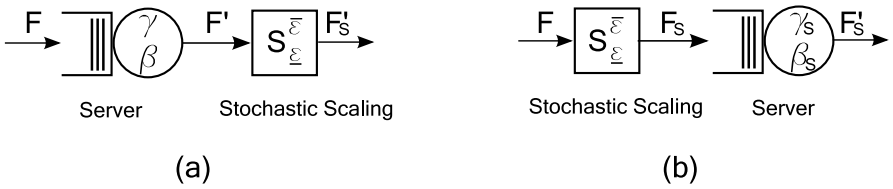

Fig. 7 Stochastic scaling of servers

Proof.

$(a) \rightarrow (b)$

For system (a), we have $F'(t) \geq (F \otimes \beta)(t)$ and it follows that

$$\begin{aligned} S(F')(t) \geq S(F \otimes \beta)(t) &= S(\inf_{0 \leq s \leq t} \{F(t-s) + \beta(s)\}) \\ &= \inf_{0 \leq s \leq t} \{S(F(t-s) + \beta(s))\} \\ &= \inf_{0 \leq s \leq t} \{S(F(t-s)) + S(F(t-s) + \beta(s)) - S(F(t-s))\} \\ \text{with prob. } \geq 1-\underline{\varepsilon} &\geq \inf_{0 \leq s \leq t} \{S(F(t-s)) + \underline{S}(\beta(s))\} \\ &= (S(F) \otimes \underline{S}(\beta))(t). \end{aligned}$$

For system (b), it holds directly that

$$F'_S(t) = (S(F))'(t) \geq (S(F) \otimes \beta_S^{\underline{\varepsilon}})(t) \text{ with prob. } \geq 1-\underline{\varepsilon}.$$

Hence, if $\beta_S^{\varepsilon} = \underline{S}(\beta)$, the lower bound on the output arrival function $F'_S(t)$ of system (b) will be the same as system (a).

As the counterpart, the proof about the upper bound on the output arrival function $F'_S(t)$ is alike. For system (a), we have $F'(t) \leq (F \otimes \gamma)(t)$ and again, it holds that

$$\begin{aligned} S(F')(t) \leq S(F \otimes \gamma)(t) &= S(\inf_{0 \leq s \leq t} \{F(t-s) + \gamma(s)\}) \\ &= \inf_{0 \leq s \leq t} \{S(F(t-s) + \gamma(s))\} \\ &= \inf_{0 \leq s \leq t} \{S(F(t-s)) + S(F(t-s) + \gamma(s)) - S(F(t-s))\} \\ \textit{with prob.} \geq 1 - \overline{\varepsilon} &\leq \inf_{0 \leq s \leq t} \{S(F(t-s)) + \overline{S}(\gamma(s))\} \\ &= (S(F) \otimes \overline{S}(\gamma))(t). \end{aligned}$$

For system (b), it holds as well that

$$S(F')(t) \leq (S(F) \otimes \gamma_S^{\overline{\varepsilon}})(t) \textit{ with prob.} \geq 1 - \overline{\varepsilon}.$$

So once again, if $\gamma_S^{\overline{\varepsilon}} = \overline{S}(\gamma)$, the upper bound on the output arrival function $F'_S(t)$ of system (b) will be the same as system (a).

$(b) \rightarrow (a)$

$\exists S$ not bijective. $S_{pseudo}^{-1} = S_p^{-1}$ exists. We use similar reasoning as $(a) \rightarrow (b)$ now for S_p^{-1}, which is also wide-sense increasing. But we have neither $S_p^{-1}(F'_S(t)) = F'(t)$ nor $S_p^{-1}(F_S(t)) = F(t)$ any more.

Firstly we look at lower bound. On the one hand, for system (b) it holds that $F'_S(t) \geq (F_S \otimes \beta_S)(t)$.

(1) **use max deviation**

$$S_p^{-1}(F'_S(t)) \geq S_p^{-1}(\inf_{s \in [0,t]} \{F_S(s) + \beta_S(t-s)\})$$

According to closure property of pseudo-inverse, which is $F'(t) \geq S_p^{-1}(S(F'(t)))$, we have

$$\begin{aligned} F'(t) &\geq \inf_{s\in[0,t]} \{S_p^{-1}(F_S(s)+\beta_S(t-s))\} \\ &= \inf_{s\in[0,t]} \{S_p^{-1}(F_S(s)) + S_p^{-1}(F_S(s)+\beta_S(t-s)) - S_p^{-1}(F_S(s))\} \\ &\geq \inf_{s\in[0,t]} \{S_p^{-1}(F_S(s)) + \underline{S_p^{-1}}(\beta_S(t-s))\} \textit{ where } \underline{S_p^{-1}} = \overline{S}_p^{-1} - \delta_{\overline{S}} - \delta_S \textit{ with prob.} \geq 1-\bar{\varepsilon} \\ &= \inf_{s\in[0,t]} \{F(s) + \underline{S_p^{-1}}(\beta_S(t-s)) + S_p^{-1}(F_S(s)) - F(s)\}. \end{aligned}$$

If let $\delta(s)$ be the deviation of $S_p^{-1}(S(t))$ from t and δ_S be the worst-case deviation, we get

$$F(s) - S_p^{-1}(F_S(s)) \leq \delta_S.$$

Therefore we have

$$\begin{aligned} F'(t) &\geq \inf_{s\in[0,t]} \{F(s) + \underline{S_p^{-1}}(\beta_S(t-s)) - \delta_S\} \\ &= \inf_{s\in[0,t]} \{F(s) + \overline{S}_p^{-1}(\beta_S(t-s)) - \delta_{\overline{S}} - 2\delta_S\} \end{aligned}$$

and let $G(t-s) = \overline{S}_p^{-1}(\beta_S(t-s)) - \delta_{\overline{S}} - 2\delta_S$, we get so an inequality

$$F'(t) \geq (F \otimes G)(t) \textit{ with prob.} \geq 1-\bar{\varepsilon}.$$

On the other hand, for system (a) we have

$$F'(t) \geq (F \otimes \beta^{\bar{\varepsilon}})(t) \textit{ withprob.} \geq 1-\bar{\varepsilon}.$$

Hence, if $\beta = \overline{S}_p^{-1}(\beta_S) - \delta_{\overline{S}} - 2\delta_S$ with prob. $\geq 1-\bar{\varepsilon}$, system (a) will be able to provide the same lower bound of output as system (b).

For upper bound we use similar reasoning. Since for system (b) we have $F'_S(t) \leq (F_S \otimes \gamma_S)(t)$ and S_p^{-1} is wide-sense increasing function, we get

$$S_p^{-1}(F'_S(t)) \leq S_p^{-1}(F_S \otimes \gamma_S)(t))$$

and because $S_p^{-1}(F_S'(t)) = F'(t) - \delta(t)$, where $\delta(t)$ is the deviation function of $S_p^{-1}(S(x))$, we get such an inequality that

$$\begin{aligned}
F'(t) &\leq \delta(t) + S_p^{-1}(F_S \otimes \gamma_S)(t)) \\
&= \delta(t) + \inf_{s\in[0,t]} \{S_p^{-1}(F_S(s) + \gamma_S(t-s))\} \\
&= \delta(t) + \inf_{s\in[0,t]} \{S_p^{-1}(F_S(s)) + S_p^{-1}(F_S(s) + \gamma_S(t-s)) - S_p^{-1}(F_S(s))\} \\
&\leq \delta(t) + \inf_{s\in[0,t]} \{S_p^{-1}(F_S(s)) + \overline{S_p^{-1}}(\gamma_S(t-s))\} \text{ with prob. } \geq 1-\underline{\varepsilon}, \\
&\quad \text{where } \overline{S_p^{-1}} = \underline{S}_p^{-1} + \delta_{\underline{S}} + \delta_S \text{ according to corollary } 3.2 \\
&= \delta(t) + \inf_{s\in[0,t]} \{F(s) - \delta(s) + \overline{S_p^{-1}}(\gamma_S(t-s))\} \\
&= \inf_{s\in[0,t]} \{F(s) + \overline{S_p^{-1}}(\gamma_S(t-s)) + \delta(t) - \delta(s)\} \\
&\leq \inf_{s\in[0,t]} \{F(s) + \overline{S_p^{-1}}(\gamma_S(t-s)) + \delta\} \\
&\quad \text{where } \delta \text{ is the max deviation of } S_p^{-1}(S(t)) \text{ from } t
\end{aligned}$$

Let $g(t-s) = \overline{S_p^{-1}}(\gamma_S(t-s)) + \delta$, it holds accordingly that

$$F'(t) \leq (F \otimes g)(t) \text{ where } g(t) = \overline{S_p^{-1}}(\gamma_S(t)) + \delta = \underline{S}_p^{-1}(\gamma_S(t)) + \delta_{\underline{S}} + 2\delta_S \text{ with prob. } 1-\underline{\varepsilon}.$$

Hence, if $\gamma = g = \underline{S}_p^{-1}(\gamma_S) + \delta_{\underline{S}} + 2\delta_S$ with prob. $1-\underline{\varepsilon}$, system (a) will be able to provide the same upper bound as system (b). □

With above method, we can see the bound transforming from system (a) to (b) is tightly the same. But conversely it might be not so tight. This will be discussed in the latter parts.

(2) **use optimization**, because method (1) may be not tight. Note that following method only provides a suggestion or a discuss but is not realized.

$F'(t) \geq \inf_{s\in[0,t]} \{S_p^{-1}(F_S(s)) + \underline{S_p^{-1}}(\beta_S(t-s))\}$ with prob. $1-\bar{\varepsilon}$.

$$\begin{cases} H(s,t) = S_p^{-1}(F_S(s)) + \underline{S_p^{-1}}(\beta_S(t-s)) \\ (1,-1)(s,t)^T \leq 0 \\ \text{we want } inf\{H(s,t)\} \end{cases} \Longrightarrow \text{this is nonlinear optimization.}$$

(i) If the relation between s and t is a function, we say $s = f(t)$, and if f is bijective, we have $t = f^{-1}(s) \Rightarrow t - s = f^{-1}(s) - s = g(s) \Rightarrow s = g^{-1}(t-s)$, then it holds that

$$\begin{aligned} F'(t) &\geq \inf_{s\in[0,t]} \{F(s) + \underline{S_p^{-1}}(\beta_S(t-s)) + S_p^{-1}(F_S(s)) - F(s)\} \\ &= \inf_{s\in[0,t]} \{F(s) + \underline{S_p^{-1}}(\beta_S(t-s)) + S_p^{-1}(F_S(g^{-1}(t-s))) - F(g^{-1}(t-s))\} \end{aligned}$$

let $h(t-s) = \underline{S_p^{-1}}(\beta_S(t-s)) + S_p^{-1}(F_S(g^{-1}(t-s))) - F(g^{-1}(t-s))$ we have $F'(t) \geq (F \otimes h)(t)$

(ii) If we can not get $s = f(t)$. That means the bound is not a curve finally.

Since this possibly needs to re-describe the concept of scaling, we don't keep on analyzing it at present.

Now we define backlog and virtual delay of stochastic scaled server. For system (a) we can transform it to system (b) with the same bounds. So we define backlog and virtual delay using system (b).

Definition 3.4 (Backlog and Virtual Delay) Consider a system with input function F and output function F'_S, where the system implements a scaling function S. The backlog of the system is defined as

$$b(t) = F_S(t) - F'_S(t).$$

Assuming FIFO data delivery, the delay of the system is defined as

$$d(t) = inf\{\tau \geq 0 :\ F_S(t) \leq F'_S(t+\tau)\}.$$

In the presence of scaling functions there exists no single definition of backlog. What stated in [7] is not so adaptive any long, because from system (b) to (a) the transforming can not be perfectly turned over, while S^{-1}doesn't exist, S_p^{-1}exists but $S_p^{-1}(F'_S(t)) \neq F'(t)$.

Corollary 3.4 (Bounds for Scaling Functions) *As before but with prob. $\geq 1 - \bar{\varepsilon}$. $F(t)$ is an arrival function which is input to a scaling function S with $\overline{S}^{\bar{\varepsilon}}$ and let $\alpha(t)$ be an arrival curve of $F(t)$. An arrival curve of the stochastic scaled output arrival function $F_S(t)$ is $\alpha_S(t) = \overline{S}^{\bar{\varepsilon}}(\alpha(t))$. And for that S_p^{-1} exists, $\alpha(t) = \overline{S_p^{-1}}(\alpha_S(t)) + \delta$, with prob. $\geq 1 - \underline{\varepsilon}$.*

Proof: For all $t \geq 0$ and $s \geq 0$ we have that

$$\begin{aligned} F_S(s+t) - F_S(s) &= S(F(s+t)) - S(F(s)) \\ &\leq \overline{S}^{\bar{\varepsilon}}(F(s+t) - F(s)) \textit{ with prob. } 1-\bar{\varepsilon} \\ &\leq \overline{S}^{\bar{\varepsilon}}(\alpha(t)) \textit{ with prob. } 1-\bar{\varepsilon}, \end{aligned}$$

which implies that $\overline{S}(\alpha(t))$ is an arrival curve of $F_S(t)$.

And now to prove $\alpha(t) = \overline{S_p^{-1}}(\alpha_S(t)) + \delta$.

$$F(s+t) - \delta \leq S_p^{-1}(F_S(s+t))$$

$$\Longrightarrow$$

$$\begin{aligned} F(s+t) - F(s) - \delta &\leq S_p^{-1}(F_S(s+t)) - F(s) \\ &\leq S_p^{-1}(F_S(s+t)) - S_p^{-1}(F_S(s)) \\ &\leq \overline{S_p^{-1}}(F_S(s+t) - F_S(s)) \textit{ with prob. } \geq 1-\underline{\varepsilon} \\ &\leq \overline{S_p^{-1}}(\alpha_S(t)) \end{aligned}$$

$$\Longrightarrow$$

$$F(s+t) - F(s) \leq \overline{S_p^{-1}}(\alpha_S(t)) + \delta.$$

□

For illustration purpose we show the derivation of backlog and delay bounds for system (b). If we want the bounds of a system with type (a), we'd better firstly transfer it into system type (b) using theorem 3.2. Just like what stated in [7] the derivation of the bound of backlog begins from the definition:

$$b(t) = F_S(t) - F'_S(t) \text{ and } F'_S(t) \geq (F_S \otimes \beta_S)(t).$$

It follows that

$$\begin{aligned} b(t) &\leq F_S(t) - (F_S \otimes \beta_S)(t) \\ &= F_S(t) - \inf_{s\in[0,t]} \{F_S(t-s) + \beta_S(s)\} \\ &= \sup_{s\in[0,t]} \{F_S(t) - F_S(t-s) - \beta_S(s)\}. \end{aligned}$$

Then with $\alpha_S(s) \geq F_S(t) - F_S(t-s)$ it holds that

$$\begin{aligned} b(t) &\leq \sup_{s\in[0,t]} \{\alpha_S(s) - \beta_S(s)\} \\ &= (\alpha_S \oslash \beta_S)(0), \textit{ where } \alpha_S(t) = \overline{S}(\alpha(t)) \textit{ with prob. } \geq 1-\overline{\varepsilon}. \end{aligned}$$

Thus

$$b(t) \leq (\overline{S}(\alpha) \oslash \beta_S)(0) \text{ with prob. } \geq 1-\overline{\varepsilon}.$$

Now regarding the delay definition it follows that

$$\begin{aligned} d(t) &= inf\{\tau \geq 0 : F_S(t) \leq F'_S(t+\tau)\} \\ &= inf\{\tau \geq 0 : S(F(t)) \leq F'_S(t+\tau)\} \\ &\leq inf\{\tau \geq 0 : S(F(t)) \leq (F_S \otimes \beta_S)(t+\tau)\} \\ &= inf\{\tau \geq 0 : S(F(t)) \leq \inf_{s\in[0,t+\tau]} \{F_S(t+\tau-s) + \beta_S(s)\}\} \\ &= inf\{\tau \geq 0 : \sup_{s\in[0,t+\tau]} \{S(F(t)) - F_S(t+\tau-s) - \beta_S(s)\} \leq 0\} \\ &\leq inf\{\tau \geq 0 : \sup_{s\in[0,t+\tau]} \{\alpha_S(s-\tau) - \beta_S(s)\} \leq 0\} \\ &\leq inf\{\tau \geq 0 : (\alpha_S \oslash \beta_S)(-\tau) \leq 0\} \textit{ where } \alpha_S(t) = \overline{S}(\alpha(t)) \textit{ with prob. } \geq 1-\overline{\varepsilon}. \end{aligned}$$

Thus

$$d(t) \leq inf\{\tau \geq 0 : (\overline{S}(\alpha) \oslash \beta_S)(-\tau) \leq 0\} \textit{with prob. } \geq 1-\overline{\varepsilon}.$$

So far, for analyzing end-to-end system some concepts are introduced and some corollaries are derived. Are they really suitable? Will a lot of information be lost during end-to-end analysis using introduced method above? Tightness of bounds plays an important role. We assume tightness is kept, then we get following description. It can be proven wrong.

(Stochastic Tightness of Bounds for Scaling Functions) *The output arrival curve* $\alpha_S(t) = \overline{S}^{\bar{\varepsilon}}(\alpha(t))$ *is tight with prob.* $\geq 1 - \bar{\varepsilon}$*, that is it is attained for a certain sample path, if the input arrival curve* $\alpha(t)$ *and maximum scaling curve* $\overline{S}$ *are sub-additive and simultaneously tight with prob.* $\geq 1 - \bar{\varepsilon}$*. Conversely, although* $\alpha_S(t)$ *and* $\underline{S}$ *are sub-additive and tight,* $\alpha(t)$ *is not tight, except that, for given* $\overline{S}$ *and* $\underline{S}$ *there exists a scaling function* S *such that maximum scaling curve of* S *is* $\overline{S}$ *and at the same time minimum scaling curve of* S *is* $\underline{S}$*. The backlog and delay bounds of scaling functions are zero and thus trivially tight.*

Proof: First half proven as before, greedy source and greedy scaling function S.

But conversely, whether $\alpha(t)$ tight is, is complex. Recall the proof of corollary 3.4:

For all $s \geq 0\, t \geq 0$,

$$\begin{aligned} S_p^{-1}(F_S(s+t)) &= F(s+t) - \delta(s+t) \\ &\geq F(s+t) - \delta_{max} \end{aligned}$$

at some time point t_m(e.g. midpoint $s+t$) we have $\delta(s+t) = \delta_{max}$, which means equality holds.

$F(s+t) - \delta_{max} \leq S_p^{-1}(F_S(s+t)) \Longrightarrow$

$$\begin{aligned} F(s+t) - F(s) - \delta_{max} &\leq S_p^{-1}(F_S(s+t)) - F(s) \\ &\leq S_p^{-1}(F_S(s+t)) - S_p^{-1}(F_S(s)), \textit{ because } F(s) \geq S_p^{-1}(F_S(s)) \end{aligned}$$

Assume that at time point $s+t$ we get δ_{max}, then there must exist a s *and* $s < s+t$, *such that* $F(s) = S_p^{-1}(F_S(s))$. Therefore, the equality above can hold at time point $s+t$. Then if $\overline{S_p^{-1}}$ is sub-additive and tight, we can get $F(s+t) - F(s) - \delta_{max} \leq \overline{S_p^{-1}}(F_S(s+t) - F_S(s))$ with holding equality for any time point. That means, at time point $s+t$, the equality can hold for $F(s+t) - F(s) - \delta_{max} \leq \overline{S_p^{-1}}(F_S(s+t) - F_S(s))$. Furthermore, if $\alpha_S(t)$ is sub-additive and tight, the equality holds for $F(s+t) - F(s) - \delta_{max} \leq \overline{S_p^{-1}}(\alpha_S(t))$ at time point $s+t$. Then the emphasis is $\overline{S_p^{-1}}$. At the fixed time point $s+t$ and s, is $\overline{S_p^{-1}}$ tight? From corollary 3.2 we know that $\overline{S_p^{-1}} = \underline{S}_p^{-1} + \delta_{\underline{S}} + \delta_S$. Now we analyze it like following:

From the proof of corollary 3.2 we know that 3 places decide the equality

(i)from $S(b) - S(a) \geq \underline{S}(b-a)$ we get $\underline{S}(\underline{S}_p^{-1}(S(b) - S(a))) \geq \underline{S}(b-a)$

(ii)from $\underline{S}(\underline{S}_p^{-1}(S(b) - S(a))) \geq \underline{S}(b-a)$ we have $\underline{S}_p^{-1}(S(b) - S(a)) \geq b - a - \delta_{\underline{S}}$, where $\delta_{\underline{S}}$ is max deviation of $\underline{S}_p^{-1}(\underline{S}(x))$

(iii)$b \geq S_p^{-1}(S(b))$ and $a \leq S_p^{-1}(S(a)) + \delta_S$, it holds that $b - a + \delta_S \geq S_p^{-1}(S(b)) - (S_p^{-1}(S(a)) + \delta_S) + \delta_S = S_p^{-1}(S(b)) - S_p^{-1}(S(a))$

for (i), if $\underline{S}$ is sub-additive and tight we get equality

for (ii), when will we use $\delta_{\underline{S}}$? It is clear that the max deviation of $\underline{S}$ can appear when $b - a$ is at the right end of the max flat sequence of $\underline{S}$, which means b or a is just the right end of max flat sequence of S.

for (iii), $\exists b\, b = S_p^{-1}(S(b))$ and when a is just at the right end of the max flat sequence of S, equality of $a \leq S_p^{-1}(S(a)) + \delta_S$ holds. This fulfills (ii).

Continue the analysis above: "at the fixed time point $s + t$ and s, is $\overline{S_p^{-1}}$ tight?". If we let $a = F(s + t)$, we get the max deviation of $S_p^{-1}(S(a))$, then we can get the equality for δ_{max}, at the same time (iii) and (ii) are fulfilled. If $\underline{S}$ is sub-additive and tight, the condition is complete.

Therefore, If $\alpha_S(t)$ and $\underline{S}$ are sub-additive and tight, we can find a sample path such that the tightness of $\alpha(t)$ is attained. But the tightness is attained just at one point, and this point is that some time t makes $S_p^{-1}(S(F(t)))$ reach the max deviation.

But unfortunately $\alpha_S(t)$ is tight only when $\alpha_S(t) = \overline{S}(\alpha(t))$. This results in a critical condition that only when the given $\overline{S}$ and $\underline{S}$ can bound the same S at the same time, the tightness of $\alpha(t)$ can hold. That is impossible for the most cases.

3.2 Stochastic Scaled Server with Bijective Scaling Function

We find that if we use max deviation to solve the proof problem with non-bijective scaling function, tightness will be lost, which implies information loss. This problem exists in a single stochastic scaled server, the rather that among many stochastic scaled servers - it explodes. This way seems harder and harder, although it is still a direct idea to describe the system in a certain sense. To escape from such puzzledom, we might as well find a solution to assume a bijective scaling function. It is different from what we have done before that we don't force a practical stochastic scaling function to be bijective but simulate it. If we get a non-bijective scaling function, what only to

be done is to simulate such a scaling function with a bijective approximation. There are two possible methods.

Assume S_{sm} as an approximation of S and try to make S_{sm} bijective. The process is just like smoothing. One solution looks like following.

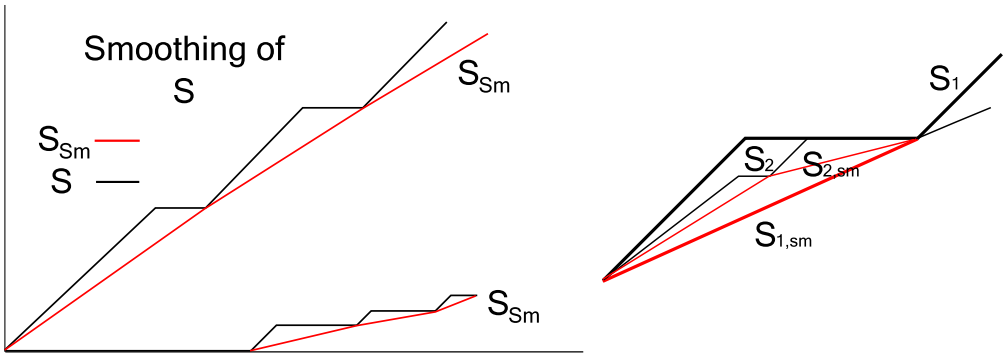

Fig. 8 One way of smoothing scaling function

This method seems to be more easily represented - connect the right inflexions of two neighbor flat segments. But sometimes the order of the original scaling functions will be lost like the right side shown.

The other solution could be:

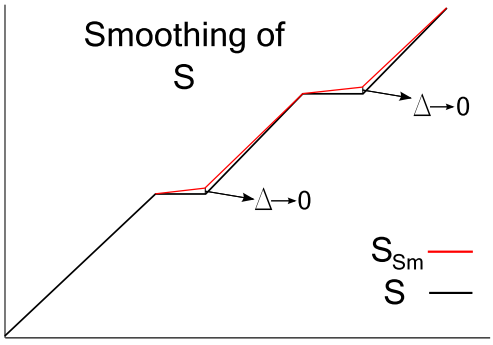

Fig. 9 Another way of smoothing scaling function

This uses Δ, which is a tiny value. We detect each flat segment and add a tiny Δ to the y-axis value of the right end point. It is feasible in practice. It has no changing order problem as above either. We can also adjust the value of Δ to hold a practically enough approximation of the scaling function S.

As we assume the scaling function bijective once again, we need to modify some corollaries and theorems. They are Corollary 3.1, Corollary 3.2, Theorem 3.2, Definition

3.4 and Corollary 3.4. The proof of them become easier - follow the relating proofs in [7] and add probability issue. Moreover, Theorem 3.3 is added to explain the concatenation of scalers.

Corollary 3.1 (Inverse Stochastic Scaling Function) *The same as corollary 3.1 (inverse scaling function) in [7].*

Corollary 3.2 (Inverse Stochastic Scaling Curves) *Consider stochastic scaling function $S \in \mathcal{F}$ and let $\underline{S}^{\underline{\varepsilon}}$ and $\overline{S}^{\overline{\varepsilon}}$ be the minimum and maximum stochastic scaling curves. If $\underline{S}^{\underline{\varepsilon}}$ and $\overline{S}^{\overline{\varepsilon}}$ are bijective, a maximum stochastic scaling curve of S^{-1} is written as $\overline{S^{-1}}$ with prob. $\geq 1 - \overline{\varepsilon}_{inv}$, which equals to $\underline{S}^{-1}$ with prob. $\geq 1 - \underline{\varepsilon}$. Respectively $\underline{S^{-1}}$ with prob. $\geq 1 - \underline{\varepsilon}_{inv}$ is $\overline{S}^{-1}$ with prob. $\geq 1 - \overline{\varepsilon}$.*

Theorem 3.2 (Stochastic Scaled Server) *Consider the two systems in Fig. 4 and let $F(t)$ be the input arrival function. System (a) consists of a server with minimum and maximum service curve $\beta(t)$ and $\gamma(t)$ respectively whose output is scaled with stochastic scaling function S and system (b) consists of a stochastic scaling function S whose output is input to a server with minimum and maximum service curve $\beta_S(t)$ and $\gamma_S(t)$ respectively. Given system (a), the lower and upper bounds of the output arrival function of system (b), that are $(S(F) \otimes \beta_S)(t)$ and $(S(F) \otimes \gamma_S)(t)$ respectively, are also valid lower and upper bounds of the output arrival function of system (a) with prob. $\geq 1 - \underline{\varepsilon}$ respectively $\geq 1 - \overline{\varepsilon}$ if*

$$\begin{aligned} \beta_S(t) &= \underline{S}^{\underline{\varepsilon}}(\beta(t)) \\ \gamma_S(t) &= \overline{S}^{\overline{\varepsilon}}(\gamma(t)) \end{aligned}$$

where $\underline{S}^{\underline{\varepsilon}}$ and $\overline{S}^{\overline{\varepsilon}}$ are the respective minimum and maximum stochastic scaling curves of S. Given system (b), the lower and upper bounds for the output arrival function of system (a), that are $S((F \otimes \beta)(t))$ and $S((F \otimes \gamma)(t))$ respectively, hold also for system (b) with prob. $\geq 1 - \overline{\varepsilon}$ respectively $\geq 1 - \underline{\varepsilon}$, if S is bijective and

$$\begin{aligned} \beta(t) &= \underline{S^{-1}}^{\underline{\varepsilon}_{inv}}(\beta_S(t)) \\ \gamma(t) &= \overline{S^{-1}}^{\overline{\varepsilon}_{inv}}(\gamma_S(t)) \end{aligned}$$

where $\underline{S^{-1}}^{\underline{\varepsilon}_{inv}}$ and $\overline{S^{-1}}^{\overline{\varepsilon}_{inv}}$ are the respective stochastic scaling curves of S^{-1}.

Note, after the stochastic scaler is moved from back to front or from front to back. The probability of the scaler is handed over to the server, which might be deterministic originally. Now the scaler turns to be deterministic and the server becomes stochastic. Because if both the scaler and the server have this probability, the probability in the

system will be falsely less than before. So actually, in graphics like fig. 7, the transforming from system (a) to (b) should be represented as from (a) to (b'), where the scaler is denoted as S but the server is denoted as $\beta_S^{\underline{\varepsilon}}$ and $\gamma_S^{\overline{\varepsilon}}$. And similar is from (b) to (a').

Definition 3.4 (Backlog and Virtual Delay) The same as definition 3.3 in [7]. Backlog and delay are defined as $b(t)$ and $d(t)$, where

$$\begin{aligned} b(t) &= F(t) - S^{-1}(F'_S(t)) \\ d(t) &= inf\{\tau \geq 0 :\ F(t) \leq S^{-1}(F'_S(t+\tau))\}. \end{aligned}$$

Now the backlog and delay bounds of the stochastic scaled server are the same as before

$$\begin{aligned} b(t) &\leq (\alpha \oslash \beta)(0) \\ d(t) &\leq inf\{\tau \geq 0 :\ (\alpha \oslash \beta)(-\tau) \leq 0\}. \end{aligned}$$

Corollary 3.4 (Bounds for Stochastic Scaling Functions) *As before but with prob.* $\geq 1 - \overline{\varepsilon}$. *$F(t)$ is an arrival function which is input to a scaling function S with $\overline{S}^{\overline{\varepsilon}}$ and let $\alpha(t)$ be an arrival curve of $F(t)$. An arrival curve of the stochastic scaled output arrival function $F_S(t)$ is $\alpha_S(t) = \overline{S}^{\overline{\varepsilon}}(\alpha(t))$* with prob. $\geq 1 - \overline{\varepsilon}$. *Backward, $\alpha(t) = \overline{S^{-1}}(\alpha_S(t))$, with prob.* $\geq 1 - \underline{\varepsilon}$.

Theorem 3.3 (Concatenation of scalers) *Consider two scalers S_1 and S_2 in sequence with maximum stochastic scaling curves $\overline{S}_1^{\overline{\varepsilon}_1}$ and $\overline{S}_2^{\overline{\varepsilon}_2}$ as well as minimum stochastic scaling curves $\underline{S}_1^{\underline{\varepsilon}_1}$ and $\underline{S}_2^{\underline{\varepsilon}_2}$. The concatenated scaler $S_2 \circ S_1$ will provide a maximum stochastic scaling curve and a minimum stochastic scaling curve as below*

$$\begin{aligned} \overline{S_2 \circ S_1}^{\overline{\varepsilon}} &= \overline{S}_2 \circ \overline{S}_1 \text{ with prob.} \geq (1-\overline{\varepsilon}_1)(1-\overline{\varepsilon}_2) \\ \underline{S_2 \circ S_1}^{\underline{\varepsilon}} &= \underline{S}_2 \circ \underline{S}_1 \text{ with prob.} \geq (1-\underline{\varepsilon}_1)(1-\underline{\varepsilon}_2). \end{aligned}$$

Proof: with definition 3.2 (stochastic scaling curves), we have

$$Pr((S_1 \oslash S_1)(b) \leq \overline{S}_1^{\overline{\varepsilon}_1}(b)) \geq 1 - \overline{\varepsilon}_1$$

$$Pr((S_1 \overline{\oslash} S_1)(b) \geq \underline{S}_1^{\underline{\varepsilon}_1}(b)) \geq 1 - \underline{\varepsilon}_1$$

$$Pr((S_2 \oslash S_2)(b) \leq \overline{S}_2^{\overline{\varepsilon}_2}(b)) \geq 1 - \overline{\varepsilon}_2$$

$$Pr((S_2 \overline{\oslash} S_2)(b) \geq \underline{S}_2^{\underline{\varepsilon}_2}(b)) \geq 1 - \underline{\varepsilon}_2$$

and apply these to $S_2 \circ S_1$ to achieve

$$Pr(\overline{S}_2^{\overline{\varepsilon}_2}(S_1(F)) \geq S_2(S_1(F)) \oslash S_2(S_1(F))) \geq 1 - \overline{\varepsilon}_2$$

$$Pr(S_2(\overline{S}_1^{\overline{\varepsilon}_1}(F)) \geq S_2(S_1(F) \oslash S_1(F))) \geq 1 - \overline{\varepsilon}_1.$$

Then utilize definition 3.2 to check $S_2 \circ S_1$. Note that S_1, $S_2 \in \mathcal{F}$, so $S_1(0) = 0$ and accordingly $S_1 \overline{\oslash} S_1 \leq S_1 \leq S_1 \oslash S_1$.

$$\begin{array}{rcl} S_2(S_1) \oslash S_2(S_1) & \leq & S_2(S_1 \oslash S_1) \oslash S_2(S_1 \oslash S_1) \\ & \leq & S_2(\overline{S}_1) \oslash S_2(\overline{S}_1) \textit{ with prob.} \geq 1 - \overline{\varepsilon}_1 \\ & \leq & \overline{S}_2 \circ \overline{S}_1 \textit{ with prob.} \geq 1 - \overline{\varepsilon}_2 \\ & \Longrightarrow & \\ \overline{S_2 \circ S_1}^{\overline{\varepsilon}} & = & \overline{S}_2 \circ \overline{S}_1 \textit{ with prob.} \geq (1 - \overline{\varepsilon}_1)(1 - \overline{\varepsilon}_2) \end{array}$$

Now, prove the second half about the minimum stochastic scaling curve of $S_2 \circ S_1$. From definition 3.2 we can as well get

$$Pr(\underline{S}_2^{\underline{\varepsilon}_2}(S_1(F)) \leq S_2(S_1(F)) \overline{\oslash} S_2(S_1(F))) \geq 1 - \underline{\varepsilon}_2$$

$$Pr(S_2(\underline{S}_1^{\underline{\varepsilon}_1}(F)) \leq S_2(S_1(F) \overline{\oslash} S_1(F))) \geq 1 - \underline{\varepsilon}_1.$$

Similarly, we know

$$\begin{array}{rcl} S_2(S_1) \overline{\oslash} S_2(S_1) & \geq & S_2(S_1 \overline{\oslash} S_1) \overline{\oslash} S_2(S_1 \overline{\oslash} S_1) \\ & \geq & S_2(\underline{S}_1) \overline{\oslash} S_2(\underline{S}_1) \textit{ with prob.} \geq 1 - \underline{\varepsilon}_1 \\ & \geq & \underline{S}_2 \circ \underline{S}_1 \textit{ with prob.} \geq 1 - \underline{\varepsilon}_2 \\ & \Longrightarrow & \\ \underline{S_2 \circ S_1}^{\underline{\varepsilon}} & = & \underline{S}_2 \circ \underline{S}_1 \textit{ with prob.} \geq (1 - \underline{\varepsilon}_1)(1 - \underline{\varepsilon}_2). \end{array}$$

□

Now we make a short discuss about stochastic end-to-end analysis. Because of above recasting, the crosswise validity of bounds can hold again. And the end-to-end concatenation could basically refer to the analysis in [7], besides inserting the description about the probabilities derived from stochastic scaling curves. However, this little change may make the stochastic end-to-end analysis hard. We mentioned above after theorem 3.2, when moving stochastic scaling to the front or back of a deterministic server, the stochasticity of the scaling will be brought to the server. And alert reader will note that $\beta_S^{\underline{\varepsilon}}(t)$ and $\gamma_S^{\bar{\varepsilon}}(t)$ in the theorem 3.2 exactly fulfill definition 2.12 (Effective Service Curves), conversely too. In [3] a problem hidden in two concatenated effective services is revealed. However in general, switching the positions of a server and a scaler in an end-to-end system can often result in such situation i.e. two concatenated effective services. In fact, if you observe the output of stochastic scaling like corollary 3.4, you will find that the arrival curve $\alpha_S(t)$ of the stochastic scaled output arrival function $F_S(t)$ even satisfies definition 2.11 (Effective Arrival Envelope). That may cause new problems too. To solve these problems will cost much effort, so in this thesis, we don't discuss them.

3.3 Use Different Channels as Scaler

We have introduced stochastic scaling function and curves. Now let us see some examples of stochastic scaling: Binary Symmetric Channel, Gilbert-Elliott Channel [14, 15] and Finite-State Markov Channel [16]. For short in the latter discuss, we write them as BSC, GEC and FSMC. These channels are not only responsible for data forwarding, but also process or say, scale data in the sense that some being forwarded data can be lost with different probabilities according to different channel states. Hence, such channels can be regarded as scaling of network calculus. At the same time, because the data loss process is stochastic, these scalings are stochastic scalings. How can man formulate the stochastic scaling curves of a BSC? And how for the other two channels? The coming analysis will answer these questions. Actually, the formulation on the one hand can help us to understand stochastic scaling; on the other hand, such kind of channel as one part, is composed into the retransmission-based model discussed in the next chapter. Note that all the following analyses are based on not packet- but bit-scaled model. This could be a problem, because the complexity of algorithms will explode when we use it in practice.

3.3.1 BSC Analysis

We assume the bit loss behavior of a BSC as the scaling. The bit loss process of BSC is i.i.d. Bernoulli with parameter θ. Then θ is the crossover probability of BSC. The probability of k bits loss of n arrival bits can be represented as

$$\theta^k(1-\theta)^{n-k}\left(\begin{array}{c} n \\ k \end{array}\right).$$

The whole region of scaling functions and two bound functions are shown as fig. 10. Note that each segment of scaling function has rate equaling 0 or 1.

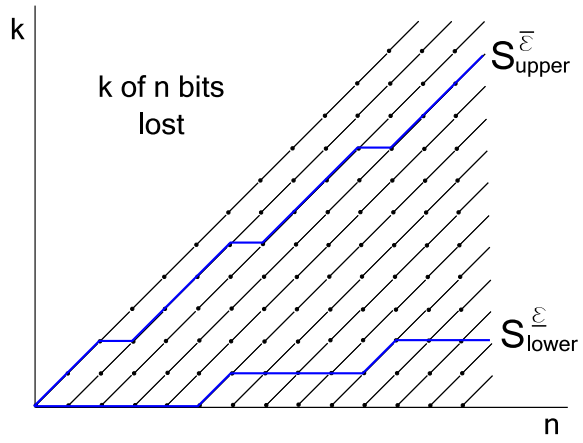

Fig. 10 Stochastic scaling curves of BSC

If the probability of scaling curve is given, which means the appearance region of scaling function is limited, we can write the upper resp. lower bound of scaling functions as following - $S_{upper}(n)$ with probability $\overline{\varepsilon}$ and $S_{lower}(n)$ with probability $\underline{\varepsilon}$.

$$\begin{aligned} S^{\overline{\varepsilon}}_{upper}(n) &= \sum_{k=0}^{n} 1_{\left\{\sum_{i=0}^{k}\theta^i(1-\theta)^{n-i}\left(\begin{array}{c} n \\ i \end{array}\right) < 1-\overline{\varepsilon}\right\}} \\ S^{\underline{\varepsilon}}_{lower}(n) &= [(\sum_{k=0}^{n} 1_{\left\{\sum_{i=0}^{k}\theta^i(1-\theta)^{n-i}\left(\begin{array}{c} n \\ i \end{array}\right) \leq \underline{\varepsilon}\right\}}) - 1]^+ \end{aligned}$$

where θ is bit loss probability of BSC and

$$1_{\{boolean\ expression\}} = \begin{cases} 1, & boolean\ expression\ is\ true; \\ 0, & otherwise. \end{cases}$$

Now let us parse the formulation of $S^{\overline{\varepsilon}}_{upper}(n)$. The formulation in braces means, we cumulate the probability of 0 bits lost, 1 bit lost, 2 bits lost ... up to k bits lost, and then check if the sum is less than $1-\overline{\varepsilon}$. If it is, $1_{\{...\}}$ will be 1. We collect such value "1"

for k from 0 to n, until some $k = m$ makes $\sum_{i=0}^{m} \theta^i (1-\theta)^{n-i} \binom{n}{i} \geq 1 - \bar{\varepsilon}$. Till then, we have already collected m times "1", i.e. $S^{\bar{\varepsilon}}_{upper}(n) = m$. This process illustrates that for n arrival bits, the sum of the probabilities of cases from 0 bits lost to m bits lost happens to be greater than or equal to $1 - \bar{\varepsilon}$. If we connect such m for each n, we get a curve. Or if we connect the points graphically under m for each n, we will get some other curves, which represent the scaling functions. We can find that these curves of scaling functions are under the former curve, we say, "bounded by the former curve". This looks like the behavior of maximum scaling curve. And the amount of these scaling functions is $\geq 1 - \bar{\varepsilon}$ percent of the whole possible scaling functions. On these arguments, the curve $S^{\bar{\varepsilon}}_{upper}(n)$ seems to be a stochastic scaling curve and we can say, for each n, $S^{\bar{\varepsilon}}_{upper}(n)$ bounds some of all the scaling functions with probability $\geq 1 - \bar{\varepsilon}$. And there is a similar analysis for $S^{\underline{\varepsilon}}_{lower}(n)$.

The probability aspect of $S^{\bar{\varepsilon}}_{upper}(n)$ and $S^{\underline{\varepsilon}}_{lower}(n)$ is affirmed. But there is still a doubt: are $S^{\bar{\varepsilon}}_{upper}(n)$ and $S^{\underline{\varepsilon}}_{lower}(n)$ really scaling curves? Now we try to justify them. If we check the definition of scaling curves (definition 2.10), the answer will be blured. Fig. 11 shows what is the puzzle. For convenience, we only analyze maximum scaling curve. Minimum scaling curve can be implied through similar analysis.

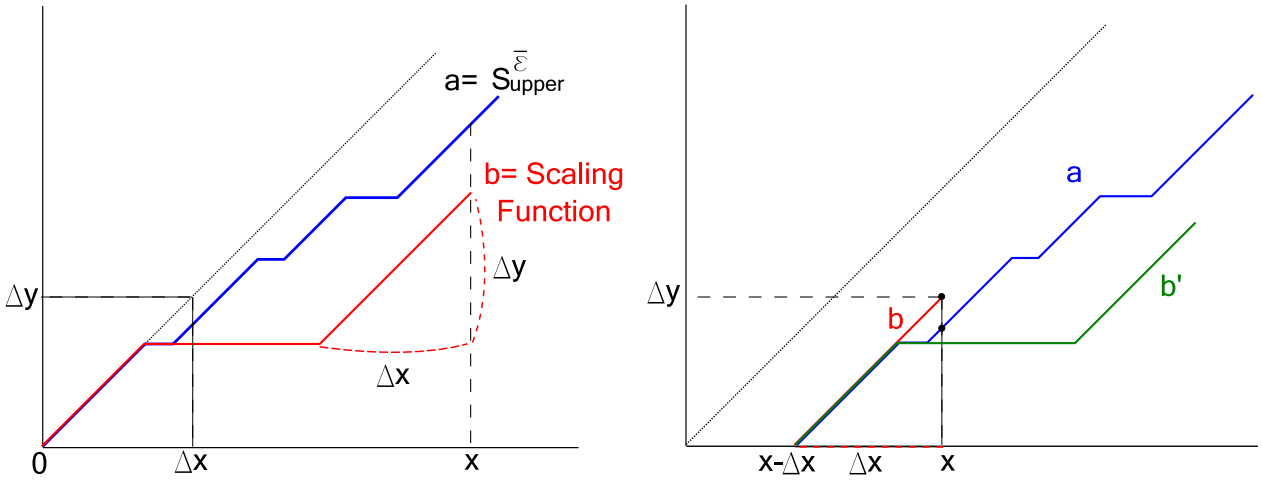

Fig. 11 Why scaling curve

Let curve a represent $S^{\bar{\varepsilon}}_{upper}(n)$ and b represent a scaling function. Observe curve b in the left half of fig. 11 firstly. We can see that for Δx, there is Δy climb. However, $a(\Delta x) < \Delta y$, which means that b can not be bounded by a, i.e. a could not be used as the maximum scaling curve according to definition 2.10. This explain sounds plausible. But in fact, it is not correct. We should always compare the climb of curve a and curve b from the beginning to x. That is, in fig. 11, for each x, we compare $b(x)$ and

$a(x)$ and find that $b(x) < a(x)$ according to the formulation of $a = S^{\overline{\varepsilon}}_{upper}(n)$, which means b is bounded by a. This illustrates, a can be used as the maximum scaling curve according to definition 2.10. Generally, if we want to check whether segment Δx of b is bounded by the maximum scaling curve, we should change our starting point of comparison. Like what is shown in the right half of fig. 11, select $x - \Delta x$ as 0, compare $a(\Delta x)$ and $b(\Delta x)$. It is obvious that b is not bounded by a, but that is logical, because reviewing the formulating of a, b should not be a scaling function bounded by a. So we can use curve a i.e. $S^{\overline{\varepsilon}}_{upper}(n)$ as the maximum scaling curve. Actually, the above analysis relies on an important argument - BSC is memoryless, that can be interpreted as below.

$$Pr\{S(0,\, n) \leq S^{\overline{\varepsilon}}_{upper}(n)\} \geq 1 - \overline{\varepsilon}$$

$$\overset{BSC\ is\ memoryless}{\Longleftrightarrow}$$

$$Pr\{\forall x :\ S(x,\, x+n) \leq S^{\overline{\varepsilon}}_{upper}(n)\} \geq 1 - \overline{\varepsilon}$$

We write $S(x+n) - S(x)$ as $S(x,\, x+n)$. The latter formulation clearly shows a maximum stochastic scaling curve $S^{\overline{\varepsilon}}_{upper}(n)$ referring to definition 3.2.

Finally we use $S^{\overline{\varepsilon}}_{upper}(n)$ and $S^{\underline{\varepsilon}}_{lower}(n)$ as the maximum and minimum stochastic scaling curves of BSC. Then it holds

$$\begin{aligned}
\overline{S}^{\overline{\varepsilon}}(n) = S^{\overline{\varepsilon}}_{upper}(n) &= \sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} \theta^i (1-\theta)^{n-i} \binom{n}{i} < 1-\overline{\varepsilon} \right\}} \\
\underline{S}^{\underline{\varepsilon}}(n) = S^{\underline{\varepsilon}}_{lower}(n) &= [(\sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} \theta^i (1-\theta)^{n-i} \binom{n}{i} \leq \underline{\varepsilon} \right\}}) - 1]^{+}
\end{aligned}$$

In fact, we can assume the bit pass behavior of a BSC as the scaling on the other hand. Such a scaling can be defined as the "complementary scaling" of the bit loss scaling as above. Then the calculation will base on the probability of k bits pass of n arrival bits as below

$$(1-\theta)^k \theta^{n-k} \binom{n}{k}.$$

It follows accordingly the so called "stochastic complementary scaling curves". Note, as parameters, different probabilities of stochastic curves i.e. $\overline{\varepsilon}_{compl}$ and $\underline{\varepsilon}_{compl}$ might be given.

$$\overline{S}^{\bar{\varepsilon}_{compl}}_{compl}(n) = \sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} \theta^{n-i}(1-\theta)^i \binom{n}{i} < 1-\bar{\varepsilon}_{compl} \right\}}$$

$$\underline{S}^{\varepsilon_{compl}}_{compl}(n) = [(\sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} \theta^{n-i}(1-\theta)^i \binom{n}{i} \leq \varepsilon_{compl} \right\}}) - 1]^+$$

Generally, we can introduce a new intuitive concept - scaler. A scaler has two complementary scaling behaviors. For BSC case, one is bit loss scaling, the other is bit pass scaling. Actually, a scaler can also have multiple scaling behaviors not just two complementary scalings. But in this thesis, we don't discuss it in detail.

3.3.2 From BSC to GEC and FSMC

If we use GEC as scaling, will the analysis be different from BSC? It is known that GEC is a channel with two states - good and bad where a BSC is associated with each state. Good state has crossover probability 0 and bad state has 0.5. Channel state can vary between two states in any order. For instance, the sequence of states could be "gbg" or "ggbgbgbbgggbg" ("g": good, "b": bad). In order to investigate the scaling curves of the GEC, it is necessary to denote the GEC firstly. Let $S = \{s_0, s_1\}$ denote the set of states, where s_0 is good state. The above mentioned state sequence can be denoted as a constant Markov process - $\{S_n\}$, $n = 0, 1, 2, ...$ Since constant Markov process has stationary transitions, the transition probability can be written as

$$t_{j,k} = Pr(S_{n+1} = s_k \mid S_n = s_j) \text{ i.e. the } 2 \times 2 \text{ transition matrix}$$

$$\begin{pmatrix} t_{0,0} = Pr(S_{n+1} = s_0 \mid S_n = s_0) & t_{0,1} = Pr(S_{n+1} = s_1 \mid S_n = s_0) \\ t_{1,0} = Pr(S_{n+1} = s_0 \mid S_n = s_1) & t_{1,1} = Pr(S_{n+1} = s_1 \mid S_n = s_1) \end{pmatrix}$$

Note that the sum of the two elements on each row is equal to 1. Moreover, we represent crossover probability of each state as error probability i.e. e_0 and e_1. They are the probability of bit loss. We define steady state probability as $p_0 = Pr(S_n = s_0)$ respectively $p_1 = Pr(S_n = s_1)$. The overall average error probability e of the GEC is then $e = p_0 e_0 + p_1 e_1$. An immediate consideration is to regard a GEC as a BSC with average crossover probability e. Then the average stochastic scaling curves of GEC are

$$\overline{S}_{avg}^{\overline{\varepsilon}}(n) = \sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} e^i(1-e)^{n-i} \binom{n}{i} < 1-\overline{\varepsilon} \right\}}$$

$$\underline{S}_{avg}^{\underline{\varepsilon}}(n) = [\sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} e^i(1-e)^{n-i} \binom{n}{i} \leq \underline{\varepsilon} \right\}} - 1]^{+}.$$

Studying the average situation is not so sound. There are two ways to analyze a GEC in detail: (i) we know the sequence of channel states; (ii) correspond quality level of channel (we can think it as a QoS metric) to different sequences of states and calculate the probability for a given quality level.

(i) If the sequence of channel states is given

The error probability of the bit loss process is no longer Bernoulli, is no longer a static value but dynamic value. It varies with the changing states. We should analyze the states sequence in detail. First to do is to correspond error probability e_0 to state s_0 and e_1 to state s_1. Then the final sequence of error probabilities therefore depends on the states sequence. Moreover, the transition probability between channel states must be considered too. Now, for instance, let the Markov process start with state s_0, in order to list all the possible state transition cases, we create following table as inspiration.

arrival n bits	**1**	**2**		**3**				**...**
lost k bits	s_0	s_0s_0	s_0s_1	$s_0s_0s_0$	$s_0s_0s_1$	$s_0s_1s_0$	$s_0s_1s_1$	
0	$\overline{e_0}$	$\overline{e_0}t_{0,0}\overline{e_0}$	$\overline{e_0}t_{0,1}\overline{e_1}$	$\overline{e_0}t_{0,0}\overline{e_0}t_{0,0}\overline{e_0}$	$\overline{e_0}t_{0,0}\overline{e_0}t_{0,1}\overline{e_1}$	...	...	
1	e_0	$t_{0,0}(\overline{e_0}e_0 + e_0\overline{e_0})$	$t_{0,1}(e_0\overline{e_1} + \overline{e_0}e_1)$	$t_{0,0}t_{0,0}(e_0\overline{e_0}\overline{e_0} + \overline{e_0}e_0\overline{e_0} + \overline{e_0}\overline{e_0}e_0)$	...	...	...	
2	-	$t_{0,0}e_0e_0$	$t_{0,1}e_0e_1$	$t_{0,0}t_{0,0}(e_0\overline{e_0}e_0 + e_0e_0\overline{e_0} + \overline{e_0}e_0e_0)$	...	...	...	
3	-	-	-	$t_{0,0}t_{0,0}e_0e_0e_0$	...	...	...	
...								

Table. 1 Formulation of probability along transition cases

It is noted that each case of state transition can relate to a sequence of transition probabilities. And such a sequence is actually a path in a transition tree. We assume arrival data n to be 3 bits. The transition tree, which is a binary tree, could be

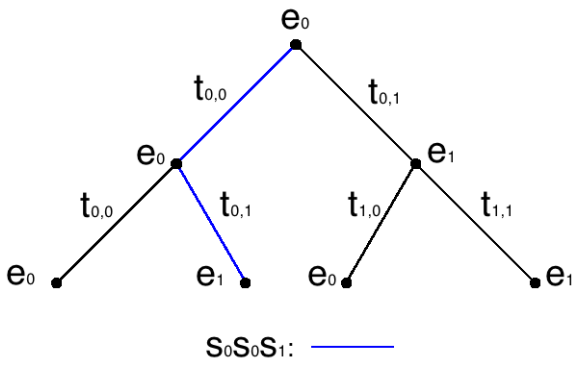

Fig. 12 A simple case of state transition tree of GEC

After the path from the root to a leaf is decided, like the blue path above - $t_{0,0}t_{0,1}$, which means state transition $s_0s_0s_1$, we can get accordingly a sequence of error probabilities - $e_0e_0e_1$. Then we describe the probability of "k bits lost":

Make k complements in the sequence of error probabilities and list all the combinations. Note that complement of x is denoted by $\overline{x} = 1 - x$. Consider $k = 1$ for the blue path in fig. 12, i.e. 1 bit lost, we get 3 combinations $e_0\overline{e_0e_1}$, $\overline{e_0}e_0\overline{e_1}$ and $\overline{e_0e_0}e_1$ in turn. Next to do is to multiply each by state transition probability $t_{0,0}t_{0,1}$ and then sum them up:

$$e_0t_{0,0}\overline{e_0}t_{0,1}\overline{e_1} + \overline{e_0}t_{0,0}e_0t_{0,1}\overline{e_1} + \overline{e_0}t_{0,0}\overline{e_0}t_{0,1}e_1.$$

That is the probability of "1 bit of 3 arrival bits is lost under the condition that state transition is given as $s_0s_0s_1$".

Generally to extend the parameter, we can define above process as $P(n,\ k,\ \{S_n\})$. It means "k bits of n arrival bits are lost under the condition that state transition is given as $\{S_n\}$" . Then

$$P(3,\ 1,\ s_0s_0s_1) = t_{0,0}t_{0,1}(e_0\overline{e_0e_1} + \overline{e_0}e_0\overline{e_1} + \overline{e_0e_0}e_1).$$

Summarizing the former discuss, it follows the general description of algorithm $P(n,\ k,\ \{S_n\})$:

$P(n,\ k,\ \{S_n\})$ ={

1. *For each state of $\{S_n\}$, we know the crossover probability. List them up as a multiplication sequence and set them all as complement (e.g. $\overline{e_0e_0e_0e_1e_1}$);*

2. *List all the possible combinations of k bits loss of n arrival bits. That is, based on the result of step 1, there are* $\begin{pmatrix} n \\ k \end{pmatrix}$ *ways to set complement once again. Do it and sum the results up;*
3. *For each pair of neighbor states in* $\{S_n\}$*, check the state transition matrix for state transition probability, and multiply them all;*
4. *Multiply the result of step 2 and step 4. This result is the value of* $P(n, k, \{S_n\})$.

}

Note: if we want to calculate the bit pass scaling, it is only necessary to modify step 1 - not to set all the elements in the multiplication sequence as complement (e.g. $e_0e_0e_0e_1e_1$).

Imitating the formulation of stochastic scaling curves of BSC, we have

$$\begin{aligned} \overline{S}^{\bar{\varepsilon}}(n) &= \sum_{k=0}^{n} 1_{\left\{\sum_{i=0}^{k} P(n,i,\{S_n\}) < 1-\bar{\varepsilon}\right\}} \\ \underline{S}^{\underline{\varepsilon}}(n) &= [\sum_{k=0}^{n} 1_{\left\{\sum_{i=0}^{k} P(n,i,\{S_n\}) \leq \underline{\varepsilon}\right\}} - 1]^{+}. \end{aligned}$$

Above result is achieved under the condition that the channel state information of GEC is available. That means, e.g. $s_0s_0s_1$ is given, with cumulating the probabilities returned by $P(n, k, \{S_n\})$ and checking the sum we can finally get the scaling curves of a GEC.

(ii) Correspond quality level of channel to different sequences of states

The transition sequence of channel states of a GEC might be not given. How to analyze the channel, when this information is unavailable? We should consider the probability of states transition sequence one by one, that is, the probability of each sample path in the states transition tree. The probabilities of the paths in the transition tree are quite different. Consider a GEC with $p_0 = Pr(S_n = s_0) = 0.8$ and $p_1 = Pr(S_n = s_1) = 0.2$, it is obvious that "$s_0$" has more chances than "s_1" to appear. That means such a state sequence has the greatest probability - "$s_0s_0s_0s_0$...". And sequence with only one "s_1" has the second great probability, sequence with two "s_1" has the third great probability, and so on... We use p_{max}, p_{second} and p_{third} to roughly denote the probabilities of these three cases of possible paths. Then they are depicted as below. Note that the root state i.e. the start state is already set as "s_0".

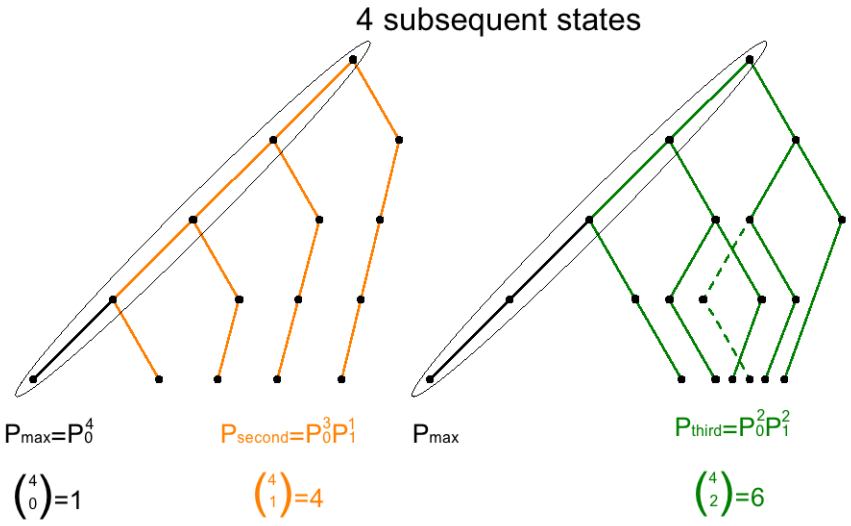

Fig. 13 Probability distribution in state transition tree

Branch towards left demonstrates that channel stays in state s_0 and branch towards right demonstrates state s_1. If we call greater probability as "heavier weight", the left branch will be "heavier" in above example (a GEC with $p_0 = Pr(S_n = s_0) = 0.8$ and $p_1 = Pr(S_n = s_1) = 0.2$). And it could be felt that the "centre of gravity" of the transition tree leans towards left. The most possible states transition of this GEC is the heaviest path, or say, the path with the heaviest weight. Obviously, the path with "0 branch towards right" is the heaviest. Second heavy path then has "1 branch towards right".

Consider a $n+1$ states sequence, that means n next states. Define the **weight level of a path** as

$$weight\ level\ x = n,\ n-1,\ n-2,\ ...,\ 0,$$

where n is states amount as well as arrival bits, if we correspond one state to each arrival bit.

Now we can define what is "weight"?

$$weight\ of\ weight\ level\ x = p_0^x p_1^{n-x}.$$

In fact, "weight" is a probability value. We can then calculate the probability of each weight level x, through counting all the possible paths of weight level x and cumulate their weight $p_0^x p_1^{n-x}$.

$$probability\ of\ weight\ level\ x = \left(\begin{array}{c} n \\ n-x \end{array} \right) p_0^x p_1^{n-x}$$

The above analysis about transition tree and the weight of paths has only one purpose - define the channel quality. In general, the static definition of channel state (s_0: good state, s_1: bad state) is not enough to characterize the channel quality owing to the dynamic changing of channel states. For instance, if the channel state changes dynamically like this case: $s_0 s_0 s_1 s_1 s_0 s_0$, we can not easily judge if it is a good channel quality or not. So with the aid of "the weight level of a path", we assume to define the channel quality as below, i.e. the "very good" channel quality according to weight level n, and "good" channel quality according to weight level $n-1$, $n-2$ and so on.

$$\begin{aligned} very\ good &: x = n \\ good &: x = n-1,\ n-2 \\ normal &: x = n-3,\ n-4,\ n-5 \\ bad &: x = n-6,\ n-7 \\ very\ bad &: x = else \end{aligned}$$

Why can we define channel quality like that? Basically, because $x = n$ means $n - n = 0$ bit lost for n arrival bits, $x = n-1$ means $n - (n-1) = 1$ bit lost for n arrival bits, they can certainly represent "very good" or "good" channel quality. In other words, weight level x connects the amount of bits lost and the quality of channel. Subsequently as an outgrowth, we get the probability of channel in some quality levels e.g. in "good" and "very good" qualities.

$$\begin{aligned} Pr(channel\ quality\ is\ "good") &= \sum_{x=n-2}^{n-1} \left(\begin{array}{c} n \\ n-x \end{array} \right) p_0^x p_1^{n-x} \\ Pr(channel\ quality\ is\ \geq "good") &= \sum_{x=n-2}^{n} \left(\begin{array}{c} n \\ n-x \end{array} \right) p_0^x p_1^{n-x} \end{aligned}$$

Such kind of formulation to calculate probability may be, conversely, a beneficial but not necessary supplement to define the channel quality. Why is it a benefit? Now imagine that at first we do not know how to define channel quality. And we think weight levels $x = n$, $n-1$ and $n-2$ are candidates of very good quality, because they mean 0 bit lost, 1 bit lost and 2 bits lost. Of course, we think $x = n-2$ (2 bits lost) as not so "very good" but also possibly "good" quality. How to decide it? Suppose we have such requirement: "probability of channel in quality 'very good' is greater than 70%". And at the same time we assume the probability of $x = n$, $n-1$ to be

$$\sum_{x=n-1}^{n} \begin{pmatrix} n \\ n-x \end{pmatrix} p_0^x p_1^{n-x} \quad > \quad 80\%,$$

which already can satisfy the requirement that "probability of channel in state 'very good' is greater than 70%", we thus can define "very good" only as $x = n, n-1$ and let $x = n-2$ associate with "good" quality of channel.

So far, we analyzed how to define channel quality of GEC. Next, let us see how to formulate the stochastic scaling curves of a GEC, but this time, associating with a fixed channel quality. We assume channel quality to be $\geq$ "*good*". There are several steps to solve this.

1. Write out every states transition sequence of each required weight level i.e. of weight level $x = n,\ n-1,\ n-2$, because channel quality is $\geq$ "*good*". We define

$$\begin{aligned} \{S_n\}_{x,\#} \quad = \quad & \textit{each possible sequence of weight level } x, \\ & \textit{where \# represents serial number of sequence}; \end{aligned}$$

   e.g. $x = n-2$, the sequence could be the states combination having two s_1 states $s_0s_0s_1s_0s_1s_0s_0$...

2. Then recalling definition of $P(n,\ k,\ \{S_n\})$, sum up $P(n,\ k,\ \{S_n\}_{x,\#})$ for each weight level x, that is

$$sum_x \quad = \quad \sum_{\#=1}^{num} P(n,\ k,\ \{S_n\}_{x,\#}),\ where\ num = \begin{pmatrix} n \\ n-x \end{pmatrix};$$

3. For $x = n,\ n-1,\ n-2$, we want the sum of probabilities, then the formulation will be

$$\sum_{x=n-2}^{n} sum_x \quad = \quad \sum_{x=n-2}^{n} \sum_{\#=1}^{num} P(n,\ k,\ \{S_n\}_{x,\#});$$

4. Finally, the stochastic scaling curves for channel quality$\geq$ "*good*" are

$$\overline{S}^{\bar{\varepsilon}}(n) = \sum_{k=0}^{n} 1 \left\{ \sum_{i=0}^{k} \sum_{x=n-2}^{n} \binom{n}{n-x} \sum_{\#=1} P(n,i,\{S_n\}_{x,\#}) < 1-\bar{\varepsilon} \right\}$$

$$\underline{S}^{\underline{\varepsilon}}(n) = [\sum_{k=0}^{n} 1 \left\{ \sum_{i=0}^{k} \sum_{x=n-2}^{n} \binom{n}{n-x} \sum_{\#=1} P(n,i,\{S_n\}_{x,\#}) \leq \underline{\varepsilon} \right\} - 1]^{+}.$$

So far, the above discuss gives us two ways of calculating the stochastic scaling curves of a GEC: states transition sequence is available or channel quality is given.

Now we consider FSMC as scaling. In fact, a GEC is a special case of FSMC. A GEC is just a two-states Markov Channel. Let the set of states $S = \{s_0, s_1, ..., s_{k-1}\}$ denote a k-states Markov Channel and s_0 is statically the best state of the channel and s_1 worse state and so on. In general, such a static definition is not enough to identify if a dynamic channel is in good state or bad state. The reason is similar to the analysis of dynamic state for GEC but more complex. For instance, on the one hand we must decide if $s_0s_0s_1s_4s_0s_0$ is a good channel state, on the other hand if the channel state changes dynamically like these two cases: $s_0s_0s_0s_2$ and $s_0s_0s_1s_1$, we can not easily judge which one is the better channel state. In order to apply and extend the above analysis of GEC to FSMC, first to do is to extend the binary transition tree to a k-nary transition tree. See fig. 14.

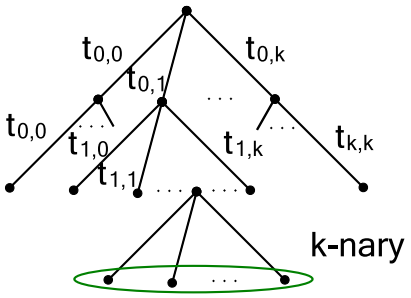

Fig. 14 K-nary state transition tree of FSMC

Based on this k-nary transition tree, the similar algorithm for FSMC as above for GEC can be created. The stochastic scaling functions can be generated too. It should be noted that the decision of weight level becomes complex, because there are not only 2

states. Consider a 4 states Markov Channel where the probability of each state is apart p_0, p_1, p_2, p_3 and $p_0 > p_1 > p_2 > p_3$. An immediate result from above discuss about dynamic state of FSMC comes up: $p_0p_0p_0p_0 > p_0p_0p_0p_1 > (p_0p_0p_0p_2,\ p_0p_0p_1p_1)$,but which one of $p_0p_0p_0p_2$ and $p_0p_0p_1p_1$ is heavier? When we define the weight level of the transition tree of a FSMC, we had better calculate and compare different probability sequences and get the weight order firstly, so that we can clearly characterize the quality of this FSMC varying dynamically. The rest calculation can refer to the algorithm for GEC.

4 Analyzing and Modeling Retransmission-based Wireless Link

We have already introduced our rough idea about the retransmission-based model in chapter 1. In this chapter we particularize the detail of the model.

4.1 Create a Retransmission-based Model using Network Calculus

Recall fig. 1 in chapter 1, we find that the timing of retransmission can be abstractly considered as during transmission in the channel. In this thesis the channel is assumed to be wireless link. Data loss often occurs in wireless link. Hence the wireless channel is a channel with some data loss process. In chapter 3 we considered BSC, GEC, FSMC as stochastic scaling. They have data loss process too. So we regard our wireless channel as a stochastic scaling and describe the retransmission behavior as: during data forwarding, data are stochastically scaled, then some of them are transmitted forward and the rest, i.e. the being recognized as the lost part of data, flows back. Note that we consider the two flows to be continuous because of selective ARQ. Next we turn to server. The same as [7], we characterize a server using the minimum respectively maximum service curve. For the moment we let the server be deterministic. Let $F(t)$ be the input arrival function. Then the retransmission-based model schema is depicted in fig. 15. Note that δ is the retransmission delay.

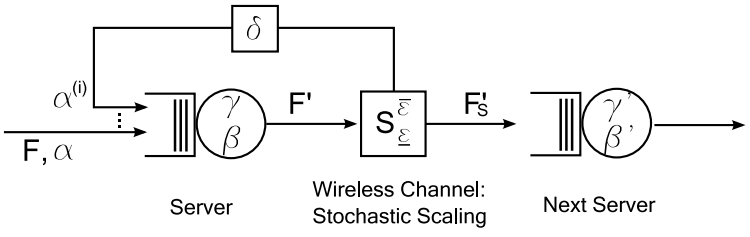

Fig. 15 Retransmission-based model using network calculus

The retransmitted data flow is a new data flow. Therefore, when it goes through the scaling, it has its own sub-flow going for retransmission. After the retransmission

again and again, it seems that flow multiplexing happens at the server node. The in fig. 15 shown flows with arrival curves $\alpha^{(0)} = \alpha$, $\alpha^{(1)}$, ..., $\alpha^{(i)}$ are such multiplexing flows. $\alpha^{(i)}$ relates to the retransmission flow of $\alpha^{(i-1)}$. How to treat these retransmission flows should be carefully thought. If we want to denote the formulation of each $\alpha^{(i)}$, we do certainly use the theory of network calculus multiplexing in chapter 2 to express them. But note that, from the viewpoint of the scaled server, all these retransmission flows come from the original flow (F, α). So no matter how many retransmission flows are generated, there is only one combined flow passing through the scaled server. It is not recognized as multiplexing.

4.2 Solving the Arrival Curves of the Retransmitted Flows

4.2.1 Self-Feedback Problem

Although it is the only combined flow passing by the scaled server, we still need multiplexing theory to formulate each flow. We can easily find that at the same time on the input side of the scaled server there are endless many retransmitted arrival data flows and each of them is bounded by the relating arrival curve $\alpha^{(i)}$. So it follows the first subsequent question: how many retransmitted flows are there? In our model, we consider i as an input parameter. It is feasible to know i with many methods. For instance, given a stochastic scaling i.e. a channel with data loss, we can know about the average data loss percent of each time scaling, e.g. $q = 20\%$. On the other hand, we assume that receiver can recover all the data if e.g. $C = 90\%$ of all data are received and identify this case as "successful transmitted", or, we assume that the data transmission is used for transmitting video data, and e.g. $C = 90\%$ can be user-tolerable. We can calculate i as

$$C = 1 - q^i \Rightarrow i \;\; = \;\; \ulcorner log_q(1-C) \urcorner e.g. = \ulcorner log_{0.2}(1-0.9) \urcorner = 2.$$

A second question is: what kind of multiplexing does the retransmission belong to? From the idea of ARQ, we let a retransmitted flow have a higher priority than its original flow, i.e. flow i has higher priority than flow $i-1$. According to theorem 2.5 (Left-over Service Curve under Arbitrary Multiplexing), each flow i is guaranteed a partial capacity of the service curve β. The higher priority a flow has, the more capacity of β it can achieve. The service curves for each flow from 0 to i are listed below:

$$\begin{aligned}
\beta^{(0)} &= [\beta - \sum_{k=1}^{i} \alpha^{(k)}]^{+} \\
\beta^{(1)} &= [\beta - \sum_{k=2}^{i} \alpha^{(k)}]^{+} \\
\beta^{(2)} &= [\beta - \sum_{k=3}^{i} \alpha^{(k)}]^{+} \\
&\ldots \\
\beta^{(i)} &= [\beta - \sum_{k=i+1}^{i} \alpha^{(k)}]^{+}.
\end{aligned}$$

At the same time, from theorem 2.2 (Performance Bounds) it is derived that $\alpha^{(k)} \oslash \beta^{(k)}$ is the output arrival curve. Then from corollary 3.4 (Bounds for Stochastic Scaling Functions), we can easily write the formulation of the arrival curve of each retransmission flow.

$$\begin{aligned}
\alpha^{(0)} &= \alpha \\
\alpha^{(1)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(0)} \oslash \beta^{(0)}) \oslash \delta_T \\
\alpha^{(2)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(1)} \oslash \beta^{(1)}) \oslash \delta_T \\
&\ldots \\
\alpha^{(i)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(i-1)} \oslash \beta^{(i-1)}) \oslash \delta_T,
\end{aligned}$$

where δ_T is retransmission delay. If we ignore delay, i.e. $\delta_T = \delta_0$, the above formulations will be simplified as below:

$$\begin{aligned}
\alpha^{(0)} &= \alpha \\
\alpha^{(1)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(0)} \oslash \beta^{(0)}) \\
\alpha^{(2)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(1)} \oslash \beta^{(1)}) \\
&\ldots \\
\alpha^{(i)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(i-1)} \oslash \beta^{(i-1)}).
\end{aligned}$$

Our goal is to solve each $\alpha^{(i)}$ and subsequently calculate the performance bounds associated with these flows or the combined flow. But, observing the above formulations, the alert reader will find that if for example, here $\beta^{(1)}$ in $\alpha^{(2)}$ is substituted with $\beta^{(1)} = [\beta - \sum_{k=2}^{i} \alpha^{(k)}]^{+}$, we will get

$$\alpha^{(2)} = \overline{S}^{\overline{\varepsilon}}(\alpha^{(1)} \oslash [\beta - \alpha^{(2)} - \alpha^{(3)} - ... - \alpha^{(i)}]^{+}).$$

This formulation implies a self-feedback of $\alpha^{(2)}$. And the same situation exists for each retransmission flow. They are deadlock formulations, therefore we can not directly use i equations to solve i variables.

4.2.2 A Solution

In order to solve the so called self-feedback problem of retransmission flows, first we write out a clear form of the equation system as

$$\begin{aligned} for\, \alpha^{(0)} &= \alpha, \\ \alpha^{(1)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(0)} \oslash [\beta - \alpha^{(1)} - \alpha^{(2)} - ... - \alpha^{(i)}]^{+}) \\ \alpha^{(2)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(1)} \oslash [\beta - \alpha^{(2)} - \alpha^{(3)} - ... - \alpha^{(i)}]^{+}) \\ &... \\ \alpha^{(i)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(i-1)} \oslash [\beta - \alpha^{(i)}]^{+}). \end{aligned}$$

Note that, i, α, β *and* $\overline{S}^{\overline{\varepsilon}}$ are given. How can we solve the deadlock problem generally? Let us first abstract the right side of the i equations as a black box system T as below.

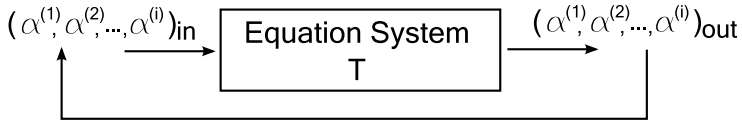

Fig. 16 Self-feedback equation system

For the first set of inputs $(\alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(i)})_1$, there is an output $(\alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(i)})_2$. Next step we use $(\alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(i)})_2$ as the next time input. And so on, until

$$(\alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(i)})_{n+1} = (\alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(i)})_n.$$

This process looks like a "training". It's the better the process is convergent. But it may be not. Actually, such a process can be described using fixed point theory [17]. As the definition of fixed point described: "for a given mapping, $T: A \rightarrow B$, every solution x of the equation $Tx = x$ is called a *fixed point* of T", our calculation of retransmission can be abstracted as

$$\alpha_{n+1} \quad = \quad T\alpha_n, \textit{ where } \alpha_n \textit{ denotes } (\alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(i)})_n.$$

Now we must face the challenges that are being always discussed in mathematical area of fixed point theory.

- The brief challenge: is T convergent? i.e. is our equation system convergent?
- An accessory challenge: each element of α_i is 2-dimensional curve, whose analysis is more complex than just 1-dimensional variable.

It's obvious that we can benefit directly from the solution of the first question. If we can judge that the equation system is convergent, we can then try to find a method to calculate what are the convergent values of $(\alpha^{(1)}, \alpha^{(2)}, ..., \alpha^{(i)})$, which are just the solutions of the arrival curves of all the retransmission flows.The judgment of convergence is not easy, because the objects of the mapping T are 2-dimensional curves. We can make use of the concept Banach Space. First, abstract each sequence of curves $\{\alpha_1^{(1)}, \alpha_2^{(1)}, ..., \alpha_\infty^{(1)}\}$, $\{\alpha_1^{(2)}, \alpha_2^{(2)}, ..., \alpha_\infty^{(2)}\}$, ..., $\{\alpha_1^{(i)}, \alpha_2^{(i)}, ..., \alpha_\infty^{(i)}\}$ as a Banach Space. And then we try to define the "norm ($\| \cdot \|$)" for the curve, for example, we define a supremum norm ($\| \alpha_n^{(i)} \| = sup\{| \alpha_n^{(i)}(x) | : x \in [a, b]\}$). Next to do is to use the norm to check whether the Cauchy Sequence (with respect to distance $d(\alpha_n^{(i)}, \alpha_{n-1}^{(i)}) = \| \alpha_n^{(i)} - \alpha_{n-1}^{(i)} \|$) for each Banach Space above has a limit. If there is a limit, it holds that the sequence of curves is convergent. However, it is hard to define a norm, because each $\alpha^{(i)}$ as input is arbitrarily free curves of $\mathcal{F}$. At the same time, the relation between $\alpha_n^{(i)}$ and $\alpha_{n-1}^{(i)}$ is not simple. Let us review our equations. Pick one of them: $\alpha_n^{(1)} = \overline{S}^{\bar{\varepsilon}}(\alpha^{(0)} \oslash [\beta - \alpha_{n-1}^{(1)} - \alpha^{(2)} - ... - \alpha^{(i)}]^+)$ and observe $\alpha^{(1)}$. The relation between $\alpha_n^{(1)}$ and $\alpha_{n-1}^{(1)}$ depends on such a complex calculation, which makes calculating $d(\alpha_n^{(1)}, \alpha_{n-1}^{(1)})$ more difficult even impossible. That means, it is therefore blured, hard even impossible to judge if the sequence is convergent. So the condition $\alpha^{(i)} \in \mathcal{F}$ is too rough for analyzing convergence. A complete research is not a mission for this thesis but a task of mathematical personnel. In this thesis, we try to simplify the calculation of each equation via specializing $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(i)}$ and β.

We can specialize $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(i)}$ and β as straight line. Then such calculation $\alpha_{out}^{(1)} = \overline{S}^{\bar{\varepsilon}}(\alpha^{(0)} \oslash [\beta - \alpha^{(1)} - \alpha^{(2)} - ... - \alpha^{(i)}]^+)$ will be easy and accordingly, if the generated $\alpha_{out}^{(1)}$ is used as next input $\alpha^{(1)}$, the next step of calculation will not turn more complex. Hence, the definition of norm will be not blured and it's easy to calculate distance $d(\alpha_n^{(i)}, \alpha_{n-1}^{(i)})$ and further judge the convergence.

Why do we say the calculation becomes easy with straight line? Let us analyze the formulation of $\alpha_{out}^{(1)}$. Begin with the inner part of the formulation, we know the form

of "$\alpha^{(0)} \oslash [...]^{+}$" about that, the rate of the result from such form will be decided by $\alpha^{(0)}$. Then observe the outer part of $\alpha_{out}^{(1)} = \overline{S}^{\bar{\varepsilon}}(...)$, we will find that the result of "$\alpha^{(0)} \oslash [...]^{+}$" will be still then modified by $\overline{S}^{\bar{\varepsilon}}$, if the deconvolution exists in $\mathbb{R}^{+}$. So, if $\overline{S}^{\bar{\varepsilon}}(x)$ is assumed to be a constant influence, like Cx, the result of $\alpha_{out}^{(1)}$ is finally expectable and the rate will be Cr_0, where r_0 is the rate of "$\alpha^{(0)} \oslash [...]^{+}$" i.e. rate of $\alpha^{(0)}$(note: because the curve itself of $\overline{S}^{\bar{\varepsilon}}(x)$ is deterministic and the stochastic behavior comes from scaling functions, $\bar{\varepsilon}$ has then no influence on the calculation).

Hence, since each round of calculating $\alpha_{out}^{(1)}$ will generate a straight line with rate Cr_0 as the next $\alpha_{in}^{(1)}$, we can collect them into a set of parallel lines. So far, let us recall fixed point theory. If the mapping T i.e. the equation system can make it possible to define norm of curve and calculate the distance between two curves, we can then decide whether the distance is limited to 0 (that means, Cauchy Sequence has a limit) and further conclude whether the mapping T is convergent or not. If T is convergent, we may finally calculate the formulations of all arrival curves of retransmission flows. Now for parallel lines, define the norm of a line as the $y-axis$ value of the cross point of this line and $y-axis$, then the calculation of distance becomes clear $\mid b_n - b_{n-1} \mid$, and as a subsequent result, the judgment if the mapping T is convergent will be easy to make - if the distance between two neighbor parallel lines approaches 0, the mapping is convergent and it will convergently generate a final line as the limit of this set of parallel lines. This is shown in fig. 17.

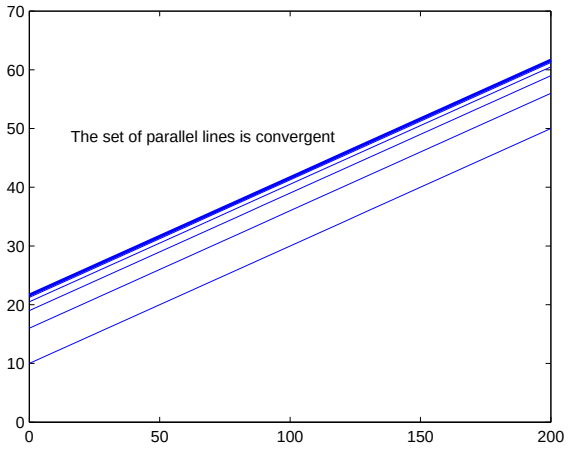

Fig. 17 Idea of convergence

So far, the idea of specializing $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(i)}$ and β turns the 2-dimensional problem to 1-dimensional and gives a principle to judge the convergence of equation system. Finally we may get the formulation of all the arrival curves of retransmission flows.

Since this is a doable method, we assume $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(i)}$ as Token Bucket $\gamma_{r,b}$ and β as Rate-Latency service $\beta_{R,T}$. In practice, such an assumption can cover many cases of a system.

Please follow the steps below to know about how to calculate the arrival curves of all the retransmitted flows - $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(i)}$. The steps are classified into three:

(1) The simplest situation: $i = 1$ - there is only one retransmission flow. It is shown that there is a fixed point i.e. the calculation of $\alpha^{(1)}$ will be convergent.

(2) The extended situation: $i = 2$ - there are two retransmission flows. Based on (1), it is shown how $\alpha^{(1)}$ and $\alpha^{(2)}$ influence each other. There may be no fixed point. But mostly there is.

(3) The general situation: $i = k$ - there are k retransmission flows. General form for calculating $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(i)}$ is given.

For all three cases, it is given $\alpha = \gamma_{r,b}$, $\beta_{R,T}$ and $\overline{S}^{\overline{\varepsilon}}(x) = Cx + B$, where $0 \leq C \leq 1$, $B \geq 0$, C and B are constant. Why do we set $\overline{S}^{\overline{\varepsilon}}(x) = Cx + B$? In fact, $\overline{S}^{\overline{\varepsilon}}(x)$ could be a set of affine functions like what in [7] introduced. In our discuss, we start with a simple situation that $\overline{S}^{\overline{\varepsilon}}(x)$ is only a single affine function. So $\overline{S}^{\overline{\varepsilon}}(x)$ looks like a "Contractor", which is defined as C and $0 \leq C \leq 1$. Without too much loss of generality, we define $\overline{S}^{\overline{\varepsilon}}(x)$ as $Cx + B$. Regarding $\overline{S}^{\overline{\varepsilon}}(x)$ as a set of affine functions is future work not discussed here.

(1) i = 1 i.e. 1 retransmission flow

We know $\alpha^{(0)} = \alpha$, $\alpha^{(1)} = \overline{S}^{\overline{\varepsilon}}(\alpha^{(0)} \oslash \beta^{(0)}) = \overline{S}^{\overline{\varepsilon}}(\alpha \oslash [\beta - \alpha^{(1)}]^{+})$. It is to be checked whether the mapping for $\alpha^{(1)}$ is convergent and $\alpha_{\infty}^{(1)} =$? The brief task is to check if there is a convergent limit b_{∞} for b_1 and what is the value, if let the initial input of $\alpha^{(1)}$ as $\alpha_1^{(1)} = \gamma_{Cr,b_1}$. Note, why we set the rate of $\alpha^{(1)}$ as Cr? Because whatever rate of $\alpha^{(1)}$ we set, after the deconvolution $\alpha \oslash [\beta - \alpha^{(1)}]^{+}$, the next rate will still be limited to the rate of α i.e. r. And after the function $\overline{S}^{\overline{\varepsilon}}(x) = Cx + B$, the rate of $\alpha^{(1)}$ will finally always be Cr. What we do is step by step calculating the formulation $\alpha^{(1)} = \overline{S}^{\overline{\varepsilon}}(\alpha \oslash [\beta - \alpha^{(1)}]^{+})$ until we achieve enough information to judge and calculate the convergent value of b_1, which is depicted in fig. 18 and explained in the following steps.

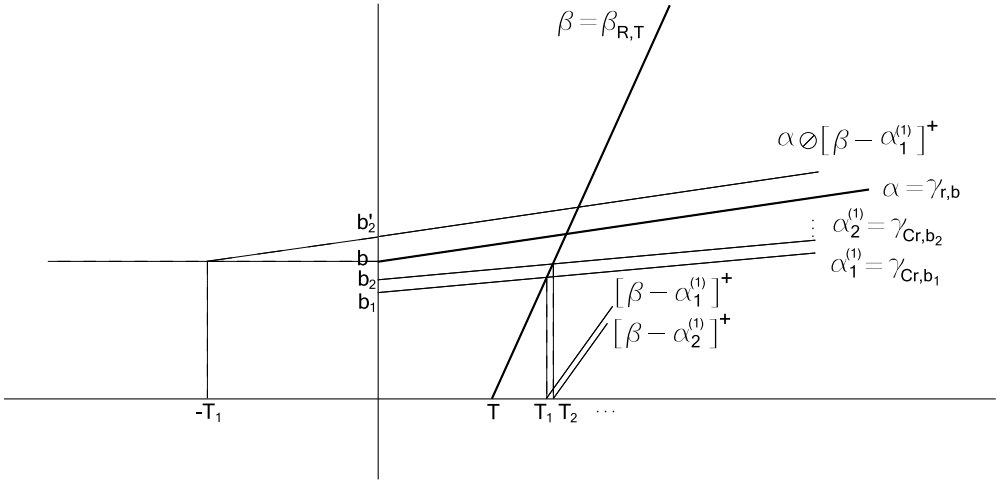

Fig. 18 Illustration of Calculation for One Retransmission Flow

(a) Get curve $[\beta - \alpha_1^{(1)}]^+ = \beta_{R-Cr,T_1}$ (shown in fig. 18) and relatively T_1;

$$\frac{CrT_1 + b_1}{T_1 - T} = R \implies T_1 = \frac{RT + b_1}{R - Cr}$$

(b) Calculate $\alpha \oslash [\beta - \alpha_1^{(1)}]^+$;

Get point $(-T_1, b)$ and the line from $(-T_1, b)$ stretches with $rate = rate\,of\,\alpha = r$ to $\mathbb{R}^+$. And calculate b_2' as

$$\frac{b_2' - b}{T_1} = r \implies b_2' = rT_1 + b.$$

(c) At last calculate $\alpha_2^{(1)} = \overline{S}^{\bar{\varepsilon}}(\alpha \oslash [\beta - \alpha_1^{(1)}]^+)$;

$$\alpha_2^{(1)} = \gamma_{Cr,b_2},\ where\ b_2 = Cb_2' + B = C(rT_1 + b) + B.$$

Repeat (a), (b) and (c) for $\alpha_2^{(1)}$, we get

$$\begin{aligned} b_3 = Cb_3' + B\ and\ b_3' = rT_2 + b &\implies b_3 = C(rT_2 + b) + B \\ \frac{CrT_2 + b_2}{T_2 - T} = R &\implies T_2 = \frac{RT + b_2}{R - Cr}. \end{aligned}$$

And $b_4 = C(rT_3 + b) + B$ and so on ... That means the convergence of $\alpha^{(1)}$ depends on the sequence of T_1, T_2, T_3, ... Now we bill this sequence as below.

$$\begin{aligned} T_1 &= \frac{RT + b_1}{R - Cr} \\ T_2 &= \frac{RT + b_2}{R - Cr} = \frac{RT + C(rT_1 + b) + B}{R - Cr} \\ T_3 &= \frac{RT + b_3}{R - Cr} = \frac{RT + C(rT_2 + b) + B}{R - Cr} \\ &\dots \\ T_i &= \frac{RT + b_i}{R - Cr} = \frac{RT + C(rT_{i-1} + b) + B}{R - Cr} \\ &= \frac{Cr}{R - Cr} T_{i-1} + \frac{RT + Cb + B}{R - Cr}. \end{aligned}$$

And conversely

$$b_i = (R - Cr)T_i - RT.$$

And because the deconvolution $\alpha \oslash [\beta - \alpha^{(1)}]^+$ exists, it holds that

$$\begin{aligned} rate\,of\,[\beta - \alpha^{(1)}]^+ > rate\,of\,\alpha &\implies R - Cr > r \\ &\implies R > r + Cr \\ &\implies \frac{Cr}{R - Cr} < C \leq 1. \end{aligned}$$

This result is feasible. We consider $R > r + Cr$, it means the capacity of server can satisfy the need of the two flows: the original flow and the first retransmitted flow. Because when we research network calculus, we very often assume a server to have enough capacity. Apply condition $\frac{Cr}{R-Cr} < 1$ to $T_i = \frac{Cr}{R-Cr} T_{i-1} + \frac{RT+Cb+B}{R-Cr}$, we can conclude that the sequence of T_i is convergent i.e. there is a fixed point, where

$$T_\infty = \frac{RT + Cb + B}{R - 2Cr}.$$

Finally we get the arrival curve of the retransmission flow

$$\begin{aligned} \alpha^{(1)} &= \gamma_{Cr,b_\infty}, \textit{ where} \\ b_\infty &= (R-Cr)T_\infty - RT = (R-Cr)\frac{RT+Cb+B}{R-2Cr} - RT. \end{aligned}$$

(2) i = 2 i.e. 2 retransmission flows

We know

$$\begin{aligned} \alpha^{(0)} &= \alpha \\ \alpha^{(1)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(0)} \oslash \beta^{(0)}) = \overline{S}^{\overline{\varepsilon}}(\alpha \oslash [\beta - \alpha^{(1)} - \alpha^{(2)}]^+) \\ \alpha^{(2)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(1)} \oslash \beta^{(1)}) = \overline{S}^{\overline{\varepsilon}}(\alpha^{(1)} \oslash [\beta - \alpha^{(2)}]^+) \end{aligned}$$

and let initial input $\alpha_1^{(1)} = \gamma_{Cr,b_{11}}$ and $\alpha_1^{(2)} = \gamma_{C^2r,b_{21}}$. It is to be checked whether the mapping for $(\alpha^{(1)}, \alpha^{(2)})$ is convergent and $\alpha_\infty^{(1)}, \alpha_\infty^{(2)}$ =? What we do are described in fig. 19 and explained below. Although the calculation process of this case looks similar to *(1)* $i = 1$, actually there are many different places. The same is that we do still parse the formulation of $\alpha^{(1)}$ and $\alpha^{(2)}$ from inner to outer. But the difference is that the variants $\alpha_1^{(1)}$ and $\alpha_1^{(2)}$ influence each other, which makes the calculation be more complex and deliver no absolutely convergent results. As *(1)*, the steps are reflected into fig. 19.

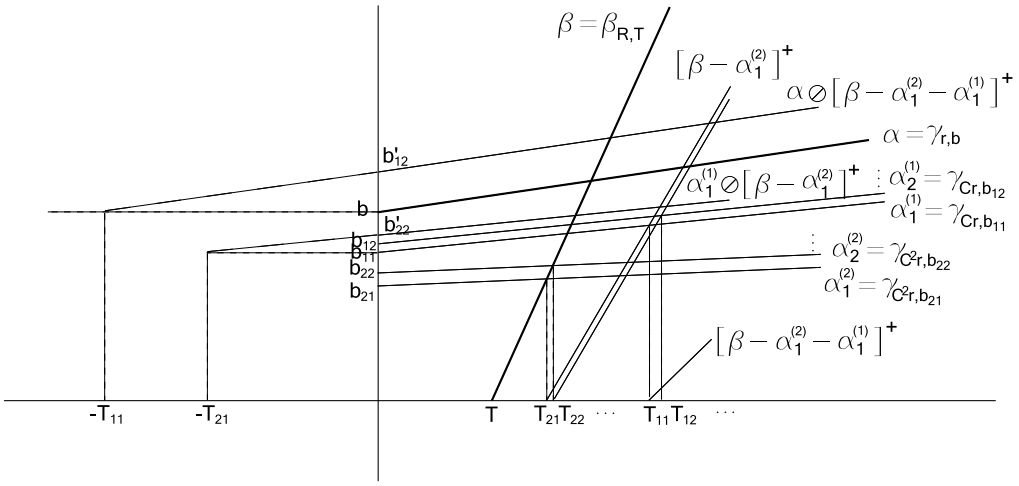

Fig. 19 Illustration of Calculation for Two Retransmission Flow

(a) Get curve $[\beta - \alpha_1^{(2)}]^+$ and $[\beta - \alpha_1^{(1)} - \alpha_1^{(2)}]^+$ then T_{21} and T_{11}, which in turn denotes

the shifting of $[\beta - \alpha_1^{(2)}]^+$ respectively $[\beta - \alpha_1^{(1)} - \alpha_1^{(2)}]^+$;

$$\frac{C^2 r T_{21} + b_{21}}{T_{21} - T} = R \implies T_{21} = \frac{RT + b_{21}}{R - C^2 r}$$
$$\frac{C r T_{11} + b_{11}}{T_{11} - T_{21}} = R - C^2 r \implies T_{11} = \frac{(R - C^2 r) T_{21} + b_{11}}{R - C^2 r - Cr} = \frac{RT + b_{21} + b_{11}}{R - C^2 r - Cr}$$

(b) Calculate the deconvolution curves $\alpha_1^{(1)} \oslash [\beta - \alpha_1^{(2)}]^+$ and $\alpha \oslash [\beta - \alpha_1^{(2)} - \alpha_1^{(1)}]^+$;

Get point $(-T_{21}, b_{11})$ and the line from $(-T_{21}, b_{11})$ stretches with $rate = rate\, of\, \alpha^{(1)} = Cr$ to $\mathbb{R}^+$. And calculate b'_{22} as

$$\frac{b'_{22} - b_{11}}{T_{21}} = Cr \implies b'_{22} = CrT_{21} + b_{11}.$$

Then $\alpha_1^{(1)} \oslash [\beta - \alpha_1^{(2)}]^+$ is $\gamma_{Cr, b'_{22}}$.

Get point $(-T_{11}, b)$ and the line through $(-T_{11}, b)$ stretches with $rate = rate\, of\, \alpha = r$ to $\mathbb{R}^+$. Next calculate b'_{12} as

$$\frac{b'_{12} - b}{T_{11}} = r \implies b'_{12} = rT_{11} + b.$$

Then $\alpha \oslash [\beta - \alpha_1^{(2)} - \alpha_1^{(1)}]^+$ is $\gamma_{r, b'_{12}}$.

(c) Calculate $\alpha_2^{(2)} = \overline{S}^{\overline{\varepsilon}}(\alpha_1^{(1)} \oslash [\beta - \alpha_1^{(2)}]^+)$ and $\alpha_2^{(1)} = \overline{S}^{\overline{\varepsilon}}(\alpha \oslash [\beta - \alpha_1^{(1)} - \alpha_1^{(2)}]^+)$;

$$\begin{aligned} \alpha_2^{(2)} &= \gamma_{C^2 r, b_{22}},\ where\ b_{22} = Cb'_{22} + B = C(CrT_{21} + b_{11}) + B \\ \alpha_2^{(1)} &= \gamma_{Cr, b_{12}},\ where\ b_{12} = Cb'_{12} + B = C(rT_{11} + b) + B \end{aligned}$$

Repeat (a), (b) and (c) for $\alpha_2^{(1)}$ and $\alpha_2^{(2)}$, we get

$$\begin{aligned} b_{23} = Cb'_{23} + B\ and\ b'_{23} = CrT_{22} + b_{12} &\implies b_{23} = C(CrT_{22} + b_{12}) + B \\ \frac{C^2 r T_{22} + b_{22}}{T_{22} - T} = R &\implies T_{22} = \frac{RT + b_{22}}{R - C^2 r} \\ b_{13} = Cb'_{13} + B\ and\ b'_{13} = rT_{12} + b &\implies b_{13} = C(rT_{12} + b) + B \\ \frac{CrT_{12} + b_{12}}{T_{12} - T_{22}} = R - C^2 r &\implies T_{12} = \frac{(R - C^2 r) T_{22} + b_{12}}{R - C^2 r - Cr}. \end{aligned}$$

And so on ... Then the general form of $T_{1,n}$, $T_{2,n}$, $b_{1,n}$ and $b_{2,n}$ are derived

$$\begin{aligned} T_{1,n} &= \frac{RT + b_{1,n} + b_{2,n}}{R - C^2r - Cr} \\ T_{2,n} &= \frac{RT + b_{2,n}}{R - C^2r} \\ b_{1,n} &= C(rT_{1,n-1} + b) + B = CrT_{1,n-1} + Cb + B \\ b_{2,n} &= C(CrT_{2,n-1} + b_{1,n-1}) + B = C^2rT_{2,n-1} + C^2rT_{1,n-2} + C^2b + CB + B. \end{aligned}$$

Let $n \to \infty$, we have such a mapping that

$$\begin{pmatrix} R^2 - 2(C^2 + C)r & -C^2r \\ -C^2r & R - 2C^2r \end{pmatrix} \begin{pmatrix} T_{1,\infty} \\ T_{2,\infty} \end{pmatrix} = \begin{pmatrix} RT + Cb + C^2b + B + CB + B \\ RT + C^2b + CB + B \end{pmatrix}.$$

Then Cramer's Rule is used to judge if there are roots in $\mathbb{R}^+$ for both $T_{1,\infty}$ and $T_{2,\infty}$. Let

$$\begin{aligned} D &= det \begin{pmatrix} R^2 - 2(C^2 + C)r & -C^2r \\ -C^2r & R - 2C^2r \end{pmatrix} \\ D_1 &= det \begin{pmatrix} RT + Cb + C^2b + B + CB + B & -C^2r \\ RT + C^2b + CB + B & R - 2C^2r \end{pmatrix} \\ D_2 &= det \begin{pmatrix} R^2 - 2(C^2 + C)r & RT + Cb + C^2b + B + CB + B \\ -C^2r & RT + C^2b + CB + B \end{pmatrix} \end{aligned}$$

If there are positive roots, it means that the mapping is convergent and the fixed point exists.

$$\begin{aligned} T_{1,\infty} &= \frac{D_1}{D} \stackrel{?}{>} 0 \\ T_{2,\infty} &= \frac{D_2}{D} \stackrel{?}{>} 0. \end{aligned}$$

When we justify $T_{1,\infty}$ and $T_{2,\infty}$, there is a condition of importance that because $\alpha^{(1)} \oslash [\beta - \alpha^{(2)}]^+$ and $\alpha \oslash [\beta - \alpha^{(2)} - \alpha^{(1)}]^+$ exist, we have

$$R - C^2r > Cr \; and \; R - C^2r - Cr > r \implies R > (1 + C + C^2)r.$$

Then we write D, D_1 *and* D_2 as below

$$\begin{aligned} D &= (R^2 - 2(C^2 + C)r)(R - 2C^2r) - C^4r^2 \\ D_1 &= (RT + Cb + C^2b + B + CB + B)(R - 2C^2r) + (RT + C^2b + CB + B)C^2r \\ D_2 &= (R^2 - 2(C^2 + C)r)(RT + C^2b + CB + B) + C^2r(RT + Cb + C^2b + B + CB + B). \end{aligned}$$

If we view R as the only variable and others are given value, the question is transformed into solving the polynomial equations of R and answer the question: which intervals of R can satisfy $DD_1 > 0$ and $DD_2 > 0$? For that, there is Newton's method to be used. And then together with $R > (1 + C + C^2)r$, we can know about all the possible intervals of R and know how to adjust the server capacity to satisfy the requirement of arrival flows. On the contrary, if we view r as the only variable, we can as well adjust the arrival flows to adapt the server capacity. Of course, we can also already know all the parameters in D, D_1 *and* D_2 and calculate the results.

From the calculation above we can imagine, there might be no fixed point. But anyway, it is mostly possible that the fixed point exist. Finally, if the fixed point exist, we get the arrival curves of $\alpha^{(1)}$ and $\alpha^{(2)}$

$$\begin{aligned} \alpha^{(1)} &= \gamma_{Cr,b_{1,\infty}} \; where \; b_{1,\infty} = CrT_{1,\infty} + Cb + B \\ \alpha^{(2)} &= \gamma_{C^2r,b_{2,\infty}} \; where \; b_{2,\infty} = C^2rT_{2,\infty} + C^2rT_{1,\infty} + C^2b + CB + B. \end{aligned}$$

(3) i = k i.e. k retransmission flows

Because the process of $i > 2$ is the same as $i = 2$, we ignore the analysis process here and directly give the general form of some necessary results. First, given as below

$$\begin{aligned} \alpha^{(1)} &= \overline{S}^{\overline{\varepsilon}}(\alpha \oslash [\beta - \sum_{i=1}^{k} \alpha^{(i)}]^+) \\ \alpha^{(2)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(1)} \oslash [\beta - \sum_{i=2}^{k} \alpha^{(i)}]^+) \\ &\ldots \\ \alpha^{(j)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(j-1)} \oslash [\beta - \sum_{i=j}^{k} \alpha^{(i)}]^+) \\ &\ldots \\ \alpha^{(k)} &= \overline{S}^{\overline{\varepsilon}}(\alpha^{(k-1)} \oslash [\beta - \sum_{i=k}^{k} \alpha^{(i)}]^+). \end{aligned}$$

Then it is concluded

$$R > (1 + C + C^2 + ... + C^k)r = \frac{1 - C^{k+1}}{1 - C}r.$$

It should be noted that this inequality may be used as a tool to adjust the server or the input flow.

The sequence of $T_{j,n}$ and $b_{j,n}$ will respectively be

$$\begin{aligned}
T_{1,n} &= \frac{RT + b_{k,n} + b_{k-1,n} + ... + b_{1,n}}{R - C^k r - C^{k-1} r - ... - Cr} \\
T_{2,n} &= \frac{RT + b_{k,n} + b_{k-1,n} + ... + b_{2,n}}{R - C^k r - C^{k-1} r - ... - C^2 r} \\
&... \\
T_{j,n} &= \frac{RT + b_{k,n} + b_{k-1,n} + ... + b_{j,n}}{R - C^k r - C^{k-1} r - ... - C^j r} \\
&... \\
T_{k,n} &= \frac{RT + b_{k,n}}{R - C^k r}
\end{aligned}$$

and

$$\begin{aligned}
b_{1,n} &= C(rT_{1,n-1} + b) + B \\
b_{2,n} &= C(CrT_{2,n-1} + b_{1,n-1}) + B \\
&... \\
b_{j,n} &= C(C^{j-1} r T_{j,n-1} + b_{j-1,n-1}) + B \\
&= C^j r(T_{j,n-1} + T_{j-1,n-2} + ... + T_{1,n-j}) + C^j b + (C^{j-1} + ... + C + 1)B \\
&... \\
b_{k,n} &= C(C^{k-1} r T_{k,n-1} + b_{k-1,n-1}) + B \\
&= C^k r(T_{k,n-1} + T_{k-2,n-2} + ... + T_{1,n-k}) + C^k b + (C^k + ... + C + 1)B.
\end{aligned}$$

Let $n \to \infty$, the equation system will be

$$A \begin{pmatrix} T_{1,\infty} \\ T_{2,\infty} \\ ... \\ T_{j,\infty} \\ ... \\ T_{k,\infty} \end{pmatrix} = \phi, \text{ where } A =$$

$$\begin{pmatrix} R-2r\sum_{i=1}^{k}C^i & -r\sum_{i=2}^{k}C^i & -r\sum_{i=3}^{k}C^i & ... & ... & -r\sum_{i=m}^{k}C^i & ... & -r\sum_{i=k}^{k}C^i \\ -r\sum_{i=2}^{k}C^i & R-2r\sum_{i=2}^{k}C^i & -r\sum_{i=3}^{k}C^i & ... & ... & -r\sum_{i=m}^{k}C^i & ... & -r\sum_{i=k}^{k}C^i \\ ... & ... & ... & ... & ... & ... & ... & ... \\ -r\sum_{i=j}^{k}C^i & -r\sum_{i=j}^{k}C^i & ... & R-2r\sum_{i=j}^{k}C^i & ... & -r\sum_{i=m}^{k}C^i & ... & -r\sum_{i=k}^{k}C^i \\ ... & ... & ... & ... & ... & ... & ... & ... \\ -r\sum_{i=k}^{k}C^i & -r\sum_{i=k}^{k}C^i & ... & ... & ... & ... & -r\sum_{i=k}^{k}C^i & R-2r\sum_{i=k}^{k}C^i \end{pmatrix}$$

$$\phi = \begin{pmatrix} RT+b\sum_{i=1}^{k}C^i+B\cdot\sum^{rows}\begin{pmatrix} 1 \\ 1+C \\ ... \\ 1+C+...+C^{k-1} \end{pmatrix} \\ RT+b\sum_{i=2}^{k}C^i+B\cdot\sum^{rows}\begin{pmatrix} 1\,empty\,line \\ 1+C \\ ... \\ 1+C+...+C^{k-1} \end{pmatrix} \\ ... \\ RT+b\sum_{i=j}^{k}C^i+B\cdot\sum^{rows}\begin{pmatrix} j-1\,empty\,lines \\ 1+C+...+C^{j-1} \\ ... \\ 1+C+...+C^{k-1} \end{pmatrix} \\ ... \\ RT+b\sum_{i=k}^{k}C^i+B\cdot\sum^{rows}\begin{pmatrix} k-1\,empty\,lines \\ \\ ... \\ 1+C+...+C^{k-1} \end{pmatrix} \end{pmatrix}.$$

Hence, we can use Cramer's Rule to calculate D, D_1, D_2, ..., D_k and get the roots as

$$T_{1,\infty} = \frac{D_1}{D},\ T_{2,\infty} = \frac{D_2}{D},\ ...,\ T_{k,\infty} = \frac{D_k}{D}.$$

If all these roots are justified to be positive, there exists fixed point. We use them then to calculate the sequence of $b_{1,\infty}$, $b_{1,\infty}$, ..., $b_{1,\infty}$. And finally the arrival curves of all the k retransmission flows are listed below

$$
\begin{aligned}
\alpha^{(1)} &= \gamma_{Cr,b_{1,\infty}},\ where\ b_{1,\infty} = Cr\frac{D_1}{D} + Cb + B \\
\alpha^{(2)} &= \gamma_{C^2r,b_{2,\infty}},\ where\ b_{2,\infty} = C^2r(\frac{D_2}{D} + \frac{D_1}{D}) + C^2b + (C+1)B \\
&\dots \\
\alpha^{(j)} &= \gamma_{C^jr,b_{j,\infty}}\ where\ b_{j,\infty} = C^jr(\frac{D_j}{D} + \frac{D_{j-1}}{D} + \dots + \frac{D_1}{D}) + C^jb + (C^{j-1} + \dots + C + 1)B \\
&\dots \\
\alpha^{(k)} &= \gamma_{C^kr,b_{k,\infty}}\ where\ b_{k,\infty} = C^kr(\frac{D_k}{D} + \frac{D_{k-1}}{D} + \dots + \frac{D_1}{D}) + C^kb + (C^{k-1} + \dots + C + 1)B.
\end{aligned}
$$

4.3 Performance Measures

Review fig. 15, so far, we have already abstracted and formulated almost every part of the model. Let us call the model as "a retransmission-based stochastically scaled server". For such a scaled server, there are input flows with arrival curves α, $\alpha^{(1)}$, $\alpha^{(2)}$, ..., $\alpha^{(i)}$, but what are the output flows? The output flows should be the scaled data flows traversing the scaled server and towards the next server. That means, the above in the calculation used scaling, which is defined as the bit loss behavior of the channel, can not satisfy such need. Actually, the above used scaling is however an inner scaling for the scaled server. The data are scaled per this inner scaling and then go back to the input side. We should use the bit pass behavior of the channel i.e. complement scaling. As an example, at the end of 3.3.1, how to understand and calculate the stochastic complementary scaling curves of BSC is given. The kernel idea is to cumulate the probability of bit pass from 0 bit to k bits. Therefore, our model in fig. 15 should be transformed to fig. 20 as a more accurate schema.

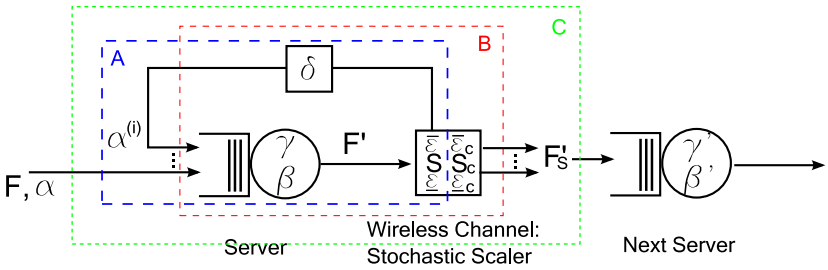

Fig. 20 Retransmission-based model using network calculus with 3 viewpoints

The channel is thought as a scaler with two complementary scaling behaviors. Three viewpoints that are possibly helpful to understand the model and calculate delay bound or backlog bound or output bound, are pointed out.

View point A: bit loss scaling $S_{\underline{\varepsilon}}^{\overline{\varepsilon}}$ is used. F, α can be seen as input. This part of model focus on providing retransmission flows.

View point B: bit pass scaling $Sc_{\underline{\varepsilon}_c}^{\overline{\varepsilon}_c}$ is used. F, α together with all the retransmission flows $\alpha^{(1)}...\alpha^{(i)}$ are viewed as input. The output are accordingly multiple flows α^*, $\alpha^{(1)*}$, ..., $\alpha^{(i)*}$.

View point C: stochastically scaled server is considered. Input is F, α. Output is a sum of the output flows from viewpoint B.

Delay bound of the flow with arrival curve $\alpha^{(i)}$ is the max horizontal distance between $\alpha^{(i)}$ and $\beta^{(i)}$. In last section, we have discussed how to formulate $\alpha^{(i)}$ and $\beta^{(i)}$. The results can be applied here for the calculation of $h(\alpha^{(i)}, \beta^{(i)})$. Nevertheless, considering retransmission, a data unit (e.g. a packet) in the original flow may traverse $i+1$ times server. In other words, a data unit may experience $i+1$ max delay caused by server and original or retransmitted flow. So for calculating the max delay of a data unit in a given flow $\alpha^{(k)}$, we should sum up all the delays achieved from flow k to i.

$$Delay\ bound\ for\ flow\ \alpha^{(k)} = \sum_{j=k}^{i} h(\alpha^{(j)}, \beta^{(j)}).$$

Backlog bound of the flow with arrival curve $\alpha^{(i)}$ is the max vertical distance between $\alpha^{(i)}$ and $\beta^{(i)}$, i.e. $v(\alpha^{(i)}, \beta^{(i)})$.

When calculate output bound, bit pass behavior should be used as scaling then we have the bound $\overline{S}_{compl}^{\overline{\varepsilon}_{compl}}(\alpha^{(i)} \oslash \beta^{(i)})$.

5 An Integrated Numerical Example

This coming chapter will focus on the application of the above model and theories. It aims at giving a clear calculation pattern with some given parameters. Review fig. 20, let us denote the system as a sequence of $(F, \alpha;\ \beta, \gamma;\ S_{\underline{\varepsilon}}^{\overline{\varepsilon}}, Sc_{\underline{\varepsilon}_c}^{\overline{\varepsilon}_c})$. Now the following input parameters are given. The output contents are listed. The calculation is summarized.

Input:

- Consider the scaler to be a BSC with bit loss probability $\theta = 0.1$. The scaler has two complementary scaling behaviors: S and S_{compl}, which are bit loss scaling and bit pass scaling respectively.

- The arrival curve of input flow is $\alpha = \gamma_{r,b} = \gamma_{0.5,3}$.

- The minimum service curve is assumed to be $\beta = \beta_{R,T} = \beta_{0.8,3}$.

- Let the probability of scaling curves be respectively $\overline{\varepsilon} = 0.1$ and $\underline{\varepsilon} = 0.1$.

Output:

- The arrival curves of all the retransmission flows.

- Performance measures.

Calculation:

Basically, the calculation is divided into two parts: (I) calculate the arrival curves of all the retransmission flows and (II) calculate the performance bounds.

5.1 Calculate Arrival Curves of the Retransmission Flows

This part is further divided into two steps: (i) analyze the channel and get the stochastic scaling curves; (ii) utilize the results in last chapter to get the arrival curves of retransmission flows.

(i) For a BSC with bit loss probability $\theta = 0.1$ and probability for stochastic scaling $\overline{\varepsilon} = 0.1$, we recall the formulation in chapter 3 section 3.1 that

$$\overline{S}^{\overline{\varepsilon}}(n) \;=\; \sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} \theta^i (1-\theta)^{n-i} \binom{n}{i} < 1-\overline{\varepsilon} \right\}}.$$

Note, in order to calculate the arrival curve of retransmission flow, we only need the maximum scaling curve. It is not so hard to get this curve (for example, we can program or use some tools like Matlab to calculate above formulation). But along with the increasing of arrival bits n, the above calculation may last very long (at least the calculation becomes very stagnant per Matlab). Hence we try to do sampling and draw this function roughly. Let us select the sampling n as below firstly and calculate relating k bits loss as well as the rate k/n.

$$\begin{pmatrix} n \\ k \\ k/n \end{pmatrix} = \begin{pmatrix} 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\ 2 & 4 & 5 & 6 & 8 & 9 & 10 & 12 & 13 & 14 \\ 0.2 & 0.2 & 0.167 & 0.15 & 0.16 & 0.15 & 0.143 & 0.15 & 0.144 & 0.14 \end{pmatrix}$$
$$\begin{pmatrix} 110 & 120 & 130 & 140 & 150 & 160 & 170 & 180 & 190 & 200 \\ 15 & 16 & 17 & 19 & 20 & 21 & 22 & 23 & 24 & 26 \\ 0.136 & 0.133 & 0.131 & 0.136 & 0.133 & 0.131 & 0.129 & 0.128 & 0.126 & 0.13 \end{pmatrix}$$

The results of k are logical. Why do we conclude that? We know the one bit loss probability $\theta = 0.1$. It generally follows that, for n arrival bits, around $n/10$ bits will be lost and the bigger n is, the more prominent this character will be. Now we observe the above results for n from 10 to 200, although sometimes k/n fluctuates appreciably, it still decreases in principle and closes to $\theta = 0.1$. Let us check this rule in following results.

$$\begin{pmatrix} n \\ k \\ k/n \end{pmatrix} = \begin{pmatrix} 1000 & 2000 & 3000 \\ 112 & 217 & 321 \\ 0.112 & 0.1085 & 0.107 \end{pmatrix}$$

We can see k/n turns smaller and closer to 0.1. Now, can all the points of (n, k) be connected to approximately simulate the maximum stochastic scaling curve directly? Actually, the above calculation is not enough. Such a curve may be a violated curve, which means, it may be not the most upper bound. The reason is that the sampling n is most possibly not the very left point of a flat segment of the truly original maximum stochastic scaling curve. See fig. 21. Note that a "flat" segment means a curve segment with the same k for an interval of n. Hence, the right thing we should do is to search the very left points of the flat segments and connect these left points to approximate the real scaling curve. With the similar interval of n as above, we can get a sequence of left points for some n.

$$\begin{pmatrix} n \\ k \\ left\ point\ of\ n \end{pmatrix} = \begin{pmatrix} 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\ 2 & 4 & 5 & 6 & 8 & 9 & 10 & 12 & 13 & 14 \\ 6 & 19 & 26 & 33 & 48 & 56 & 64 & 80 & 89 & 97 \end{pmatrix}$$

$$\begin{pmatrix} 110 & 120 & 130 & 140 & 150 & 160 & 170 & 180 & 190 & 200 \\ 15 & 16 & 17 & 19 & 20 & 21 & 22 & 23 & 24 & 26 \\ 105 & 114 & 122 & 139 & 148 & 157 & 165 & 174 & 183 & 200 \end{pmatrix}$$

$$for\ some\ other\ n \quad : \quad \begin{pmatrix} 1980 & 1990 & 2000 & 2980 & 2990 & 3000 \\ 215 & 216 & 217 & 319 & 320 & 321 \\ 1974 & 1983 & 1993 & 2975 & 2985 & 2994 \end{pmatrix}$$

The connection of the left points i.e. the non-violated curve for n from 10 to 200 is shown in fig. 21 too.

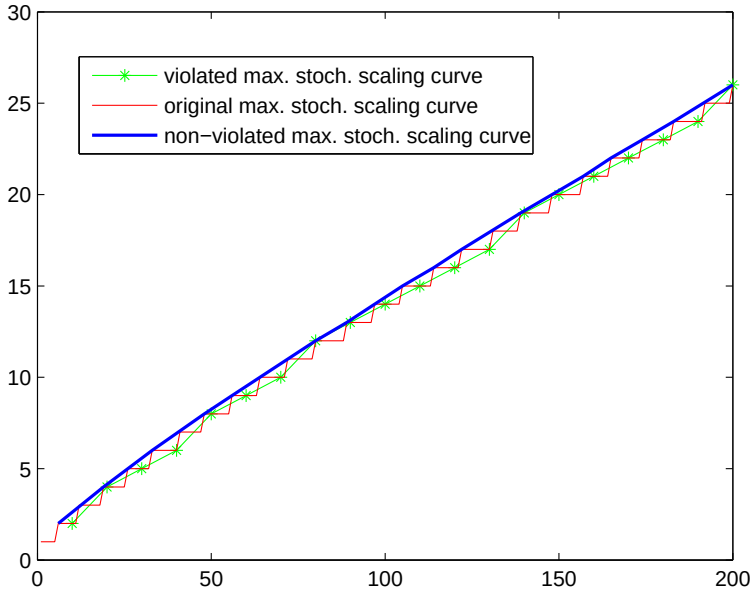

Fig. 21 Getting non-violated maximum stochastic scaling curve

Now let us recall what is discussed in chapter 4 section 2.2 i.e. the maximum scaling curve is assumed to be a "contractor - $Cx + B$" during the calculation. How to select C and B is the next question. From fig. 21, we can imagine that the trend of the non-violated scaling curve goes close to the nearly straight line with rate $r \approx 0.1$ for increasing n, except that at the beginning, the curve is not so straight. Consider a big $n = n_m$, the curve phase around $x = n_m$ approaches to the normal situation, i.e. this phase of curve fluctuates almost not. So, drawing a line at point (n_m, k_m), we

can simulate the real maximum stochastic scaling curve, and as the immediate result, we get C and B. Actually, that is not the end of creating a "contractor". How to select (n_m, k_m) is not only to select a big n but we should also check the neighbor segments around our selected n_m. Now see fig. 22, select the left points of three flat segments $n = 1974, 1983, 1993$ to do a further analysis. For $n = [1974, 1983]$, $\Delta k/\Delta n = 1/9$; for $n = [1983, 1993]$, $\Delta k/\Delta n = 1/10$. The rate decreases. Let us connect the two endpoints of the second interval between $(1983, 216)$ and $(1993, 217)$ as our "contractor". We get $C = 1/10 = 0.1$ and $B = 17.7$. Such $C = 0.1$ is too perfect. Next segment after $(1993, 217)$ may have a rate greater than 0.1 most possibly. If that happens, we will see a violation, what is not we want. So we can set an parameter a to adjust the "contractor" to avoid violation - let the second point be $(1993 - a, 217)$. Selecting a in this example as 0.5, we will finally get $C = 0.1053$ and $B = 7.2632$. Fig. 22 shows the result from the hairlike as well as the macro viewpoint.

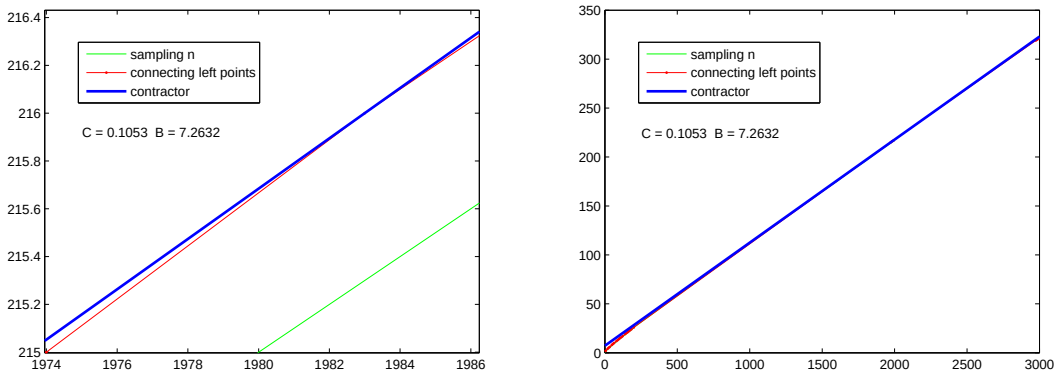

Fig. 22 Getting contractor from the scaling curve

If we select other segments of the scaling curve to calculate the "contractor" line, what does the result look like? Consider $n = 2975, 2985, 2994$, for $n = [2975, 2985]$, $\Delta k/\Delta n = 1/10$; for $n = [2985, 2994]$, $\Delta k/\Delta n = 1/9$, the rate increases. That is so called "fluctuation" of the scaling curve. Respectively selecting any segment of them to be the "contractor" will still cause violation. So we connect these two end points $(2975, 319)$ and $(2994, 321)$ as our "contractor". Therefore, $C = 2/19 \approx 0.1053$ and $B \approx 5.8421$. We can see, this is a parallel line of the above calculated line but with less B. That means violation. So we don't use it. Although this straight line may cause violation, such method is still a good way to avoid violation step by step when calculating "contractor".

Finally, we decide $C = 0.1053$ and $B = 7.2632$. Next we calculate the arrival curves of retransmission flows.

(ii) The first thing to do is to know the number of retransmission flows i.e. i. There should be many ways to know i. But let us assume that is not a task of our model.

Think i as given. However, as an example, let us in several sentences guess a possible method to calculate i. Now, let us presume (note: here is only a presume and might be a wrong reference, but anyway, i should be a known value) the method like what discussed in 4.3.2 i.e.

$$\begin{aligned} Completeness\ of\ data\ transmission\ C &= 1 - q^i \Longrightarrow i = \ulcorner log_q(1-C) \urcorner \\ where\ q - average\ data\ loss\ probability &= a\ function\ of\ \theta,\ let's\ assume\ q\ to\ be\ \theta \\ \Longrightarrow i &= \ulcorner log_{0.1}(1-92\%) \urcorner = 2. \end{aligned}$$

So let us select $i = 2$ as the number of retransmission flows. Then the goal is to calculate $\alpha^{(1)}$ and $\alpha^{(2)}$. Recall the retransmission-based model for case $i = k$ in chapter 4 section 2.2, we have $A(T_{1,\infty},\ T_{2,\infty},\ ...T_{k,\infty})^T = \phi$. Now $k = 2$, that is $A(T_{1,\infty},\ T_{2,\infty},)^T = \phi$.

We have already known the maximum stochastic scaling curve of BSC i.e. $(C,\ B) = (0.1053,\ 7.2632)$ as the given parameters of the polynomial equation system. We have also known $\alpha = \gamma_{r,b} = \gamma_{0.5,3}$ and $\beta = \beta_{R,T} = \beta_{0.8,3}$. Then let us clearly calculate A and θ and solve the equation system.

$$A = \begin{pmatrix} R^2 - 2(C^2 + C)r & -C^2 r \\ -C^2 r & R - 2C^2 r \end{pmatrix} = \begin{pmatrix} 0.5236 & -0.0055 \\ -0.0055 & 0.4889 \end{pmatrix}$$

$$\phi = \begin{pmatrix} RT + Cb + C^2 b + B + CB + B \\ RT + C^2 b + CB + B \end{pmatrix} = \begin{pmatrix} 18.0404 \\ 10.4613 \end{pmatrix}.$$

Then apply Cramer's rule: $D = det(A) = 0.2560$, $D_1 = 14.2903$, $D_2 = 5.5777$ to solve the equation system and get

$$T_{1,\infty} = \frac{D_1}{D} = 55.8280,\ T_{2,\infty} = \frac{D_2}{D} = 21.7904$$

$$b_{1,\infty} = CrT_{1,\infty} + Cb + B = 10.5184,\ b_{2,\infty} = C^2 r(T_{2,\infty} + T_{1,\infty}) + C^2 b + (C+1)B = 8.4916.$$

Finally we get following functions as $\alpha^{(1)}$ and $\alpha^{(2)}$:

$$\begin{aligned} \alpha^{(1)} &= \gamma_{Cr,b_{1,\infty}} = \gamma_{0.0527,\,10.5184} \\ \alpha^{(2)} &= \gamma_{C^2 r,b_{2,\infty}} = \gamma_{0.0055,\,8.4916}. \end{aligned}$$

Is the result reasonable in practice? Yes. Compare the rate of α and $\alpha^{(1)}$: the rate of $\alpha^{(1)}$ Cr is much smaller than rate of α, which is feasible, because $\alpha^{(1)}$ is the retransmission

flow caused by data loss and the probability of data loss is very small, that means the retransmission flow will be very weak. The result above reflects this reality. As the retransmission flow of $\alpha^{(1)}$, $\alpha^{(2)}$ is even much weaker. Because the retransmission flows are very weak, it is concluded that the channel is really in a good state. Then compare b with b_1 and b_2: $b_1 = 10.5184 > b_2 = 8.4916 > b = 3$. Essentially b as the original flow's parameter had better be greater than b_1 and b_2. However, the decision of C and B will cause the tightness lost at the beginning of the maximum stochastic scaling curve (fig. 22). This influence is brought to b_1 and b_2. So at the beginning, $\alpha^{(1)}$ and $\alpha^{(2)}$ will be not so tight. But this untight phase will last not so long, because the rates of $\alpha^{(1)}$ and $\alpha^{(2)}$ are much less than α and therefore they extend shortly lower than α. See fig. 23.

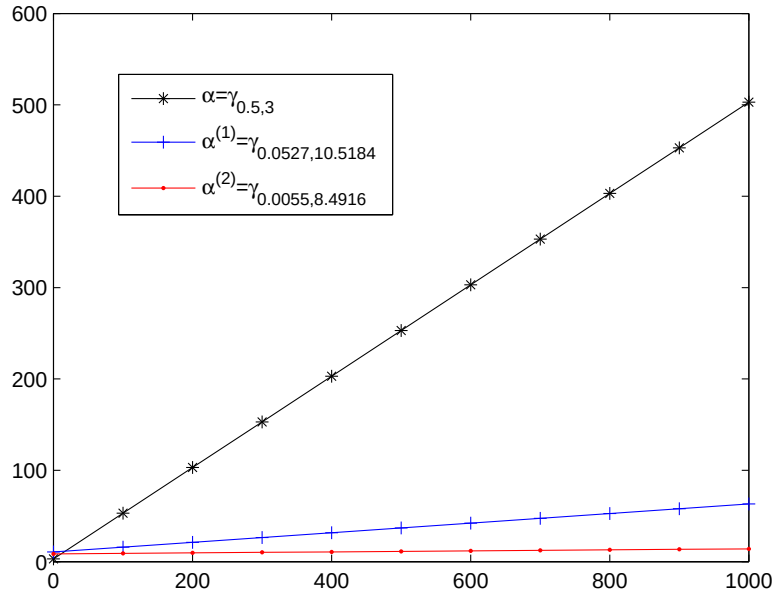

Fig. 23 Arrival curves of retransmission flows

So far, we calculate the arrival curves of all the retransmission flows of a BSC. If the channel belongs to another model like GEC, the calculation will be more complex. The former discuss in chapter 3 section 3.2 shows us a pattern to calculate the stochastic scaling curves of a GEC. There we discussed two different ways of given condition: one is, the sequence of states is given, then cumulate the probability to satisfy the requirement e.g. $prob. \geq 1 - \bar{\varepsilon}$; the other way is that we have known or defined different quality levels of a GEC (e.g. "best", "better", "good", "normal", "bad", "worse", "worst"), and each level relates to some combinations of states, then we cumulate the probability of those state combinations for a given quality level to satisfy the requirement $prob. \geq 1 - \bar{\varepsilon}$. For the former case, we finally achieve the stochastic scaling

curves of some state sequence (e.g. $s_0s_0s_0s_1s_1s_0s_0s_1s_0$...); for the latter case, we finally achieve the stochastic scaling curves for a given channel's quality (e.g. "best") or for some quality levels (e.g. $quality \geq normal$"). Although the calculation pattern changes a little bit, the cumulating of the probability is quite the same as calculating BSC. And since we can get the stochastic scaling curves of a GEC, it is similar to abstract it as a "contractor - $Cx + B$". The latter calculation about retransmission flows is the same as the BSC.

5.2 Calculate Performance Measures

Continue with the same example as above. Consider $h(curve1, curve2)$ as the max horizontal distance between curve1 and curve2 and $v(curve1, curve2)$ as the max vertical distance between curve1 and curve2. For $\alpha = \gamma_{0.5,3}$, $\beta = \beta_{0.8,3}$, $\alpha^{(1)} = \gamma_{0.0527,10.5184}$ and $\alpha^{(2)} = \gamma_{0.0055,8.4916}$, we use the results from chapter 3 to calculate performance bounds, i.e. delay bound, backlog bound and output bound. Basically along two ways to go: we can focus on each flow or the combined flow.

Delay Bounds:

$$\begin{aligned} d_0 &\leq h(\alpha, [\beta - \alpha^{(1)} - \alpha^{(2)}]^+) = (RT + b_1 + b_2 + b)/(R - r_1 - r_2) = 32.9064 \\ d_1 &\leq h(\alpha^{(1)}, [\beta - \alpha^{(2)}]^+) = (RT + b_2 + b_1)/(R - r_2) = 26.9478 \\ d_2 &\leq h(\alpha^{(2)}, \beta) = T + b_2/R = 13.6145 \end{aligned}$$

and the delay bound of the original flow will be the sum of above three, which is

$$d_{0,bound} = d_{0max} + d_{1max} + d_{2max} = 73.4687;$$

The worst case of forwarding data from original flow should be that the data is after twice retransmission transmitted towards the next node. That means, the three possible max delays are all experienced by it. With the same reason as the original flow, the first as well as the second retransmission flow will have the delay bound

$$\begin{aligned} d_{1,bound} &= d_{1max} + d_{2max} = 40.5623 \\ d_{2,bound} &= d_{2max} = 13.6145. \end{aligned}$$

Backlog Bounds:

$$\begin{aligned} bl_0 &\leq v(\alpha, [\beta - \alpha^{(1)} - \alpha^{(2)}]^+) = b + r(RT + b_1 + b_2)/(R - r_1 - r_2) = 17.4311 \\ bl_1 &\leq v(\alpha^{(1)}, [\beta - \alpha^{(2)}]^+) = b_1 + r_1(RT + b_2)/(R - r_2) = 11.2409 \\ bl_2 &\leq v(\alpha^{(2)}, \beta) = b_2 + r_2 T = 8.5081 \end{aligned}$$

and calculate the cumulative backlog bound as

$$bl \leq v((\alpha + \alpha^{(1)} + \alpha^{(2)}), \beta) = 23.6846;$$

Output Bounds:

In order to calculate output bound of a stochastic scaled server, we should know what is the output. From the viewpoint of the whole stochastic scaled server, the output are all the traversed flows generated by all the three flows α, $\alpha^{(1)}$ and $\alpha^{(2)}$. And they traverse not the above calculated scaling i.e. bit loss scaling, but the bit pass scaling. This scaling is so called "complementary scaling", which is already stated in chapter 3 section 3.1. And this BSC is viewed as a "scaler". The calculation process is the same as the bit loss scaling except replacing the bit loss probability θ with $1 - \theta = 1 - 0.1 = 0.9$.

$$\overline{S}^{\bar{\varepsilon}_{compl}}_{compl}(n) = \sum_{k=0}^{n} 1_{\left\{ \sum_{i=0}^{k} (1-\theta)^i \theta^{n-i} \binom{n}{i} < 1 - \bar{\varepsilon}_{compl} \right\}}.$$

Now sample n. Similarly, we check $n = 10\ to\ 200$ firstly.

$$\begin{pmatrix} n \\ k \\ k/n \\ left\ point\ of\ n \end{pmatrix} = \begin{pmatrix} 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\ 10 & 20 & 29 & 38 & 48 & 57 & 66 & 75 & 85 & 94 \\ 1 & 1 & 0.9667 & 0.95 & 0.96 & 0.95 & 0.9429 & 0.9375 & 0.9444 & 0.94 \\ 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \end{pmatrix}$$

$$\begin{pmatrix} 110 & 120 & 130 & 140 & 150 & 160 & 170 & 180 & 190 & 200 \\ 103 & 112 & 121 & 130 & 140 & 149 & 158 & 167 & 176 & 185 \\ 0.9364 & 0.9333 & 0.9308 & 0.9286 & 0.9333 & 0.9313 & 0.9294 & 0.9278 & 0.9263 & 0.9250 \\ 110 & 120 & 130 & 139 & 150 & 160 & 170 & 180 & 190 & 200 \end{pmatrix}$$

Note that, k/n decreases and closes to $1 - \theta = 0.9$, which looks like the former analysis for the collected numerical results above. This time, we need not (of course, can also) simulate the maximum stochastic scaling curve using a straight line anymore. We can connect all the left points of each flat segment and get a set of affine functions like below.

$$\overline{S}_{compl}^{\bar{\varepsilon}_{compl}}(n) \quad = \quad \min_{i\in[1,m]}\{C_i n + B_i\}$$

The graphic looks as below. Because $1-\theta = 0.9$, for 1 bit arrival increment the required sum of probabilities will basically absorb the probability caused by one more bit. This is the reason why the line connecting left points seems nearly the same as sampling n, except of the segment around the point $n = 140$.

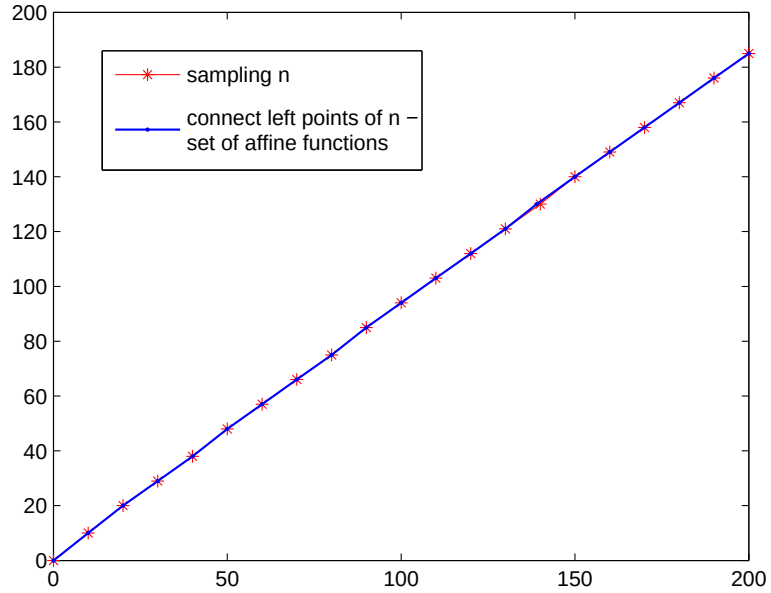

Fig. 24 Maximum complement stochastic scaling curve

Then the output bound for each traversing flow will be the arrival curve of output.

$$\begin{aligned}
\alpha^* &= \overline{S}_{compl}^{\bar{\varepsilon}_{compl}}(\alpha \oslash [\beta - \alpha^{(1)} - \alpha^{(2)}]^+) \\
\alpha^{(1)*} &= \overline{S}_{compl}^{\bar{\varepsilon}_{compl}}(\alpha^{(1)} \oslash [\beta - \alpha^{(2)}]^+) \\
\alpha^{(2)*} &= \overline{S}_{compl}^{\bar{\varepsilon}_{compl}}(\alpha^{(2)} \oslash \beta)
\end{aligned}$$

From the point of view of the next server, the arrival curve of the input flow will be then $\alpha^* + \alpha^{(1)*} + \alpha^{(2)*}$.

6 Conclusion and Outlook

6.1 Conclusion

As other network calculus extensions from deterministic to stochastic, we have extended the deterministic scaling introduced in [7] to a stochastic version, in this thesis. And based on these stochastic scaling theories, the data forwarding behavior of wireless link is analyzed as stochastic scaling and accordingly a retransmission-based model has been built up.

We have shown the difficulty extending deterministic scaling to stochastic version, that is the so called non-bijectivity of scaling function. For non-bijective scaling function, inverse function may not exist. That causes, we could not transform the theories in [7] directly to stochastic ones only via adding some descriptions about stochasticity. Two attempts were made to solve this problem: one is to use pseudo-inverse scaling function and define a further concept max deviation; the other is to try to smooth the non-bijective scaling function to be bijective - we add a tiny increment to the right end of each flat segment in original scaling function. After this smoothing, we defined the stochasticity into scaling function, scaling curves and further theorems. To further understand the concept of stochastic scaling, we have introduced the formulating of the BSC as well as the extension from BSC to GEC and FSMC as examples. The key idea of calculation is to cumulate the probability from 0 bit loss (or pass) to k bits loss (or pass).

We have composed arrival flow, server and stochastic scaler into one model to represent retransmission-based wireless link. The wireless channel is abstracted as a stochastic scaler and this stochastic scaler holds two complementary stochastic scaling behaviors. Multiple retransmission flows are generated traversing through the bit loss scaling. And without considering all the possible situation of arrival curves and service curves but considering a most possible situation i.e. token bucket as arrival curve and rate-latency as service curve, a general form of calculating the arrival curves of $i = k$ retransmission flows is given (in chapter 4 section 2). One conclusion should be noted that calculation process may not always be able to convergently generate a reasonable set of arrival curves of retransmission flows. Of course, as the most possible result, we do get the formulations of these arrival curves. With these arrival curves, we recalled the theory results (in chapter 3) to calculate the performance bounds. And in order to understand the model better, a numerical example is shown

in the sequel of this thesis. The numerical example shows a possible way of application of the model, which will or should be applied in a practical wireless networked system in the future, although at the moment it looks incomplete and idealized more or less.

6.2 Outlook

As an attempt of retransmission-based modeling using network calculus, this thesis only focuses on the most basic issues - extend the basic network calculus theory and construct the retransmission-based scaled server. In practice, a real application of this model in network system bases on the end-to-end analysis.

One of future work is generalizing the retransmission-based model. This thesis only processed the situation that arrival curve is token bucket and service curve is rate-latency. We should however, extend the analysis to more general curves, which will of course, bring more complex application of fixed point theory. At the same time, when abstracting stochastic scaling curve of the channel as a "contractor", it is the better to formulate it as a set of affine functions. And retransmission delay depicted in the model is to be handled too. Moreover, it is also necessary to completely provide a more general wireless channel model like FSMC.

A second interesting work could be creating a packet-scaled retransmission-based model. This is in fact a task about analyzing concatenation of both packetizer and stochastic scaler. This work will make the model more applicable. For example, when we give a numerical example again, the calculation for getting stochastic scaling curves of channel like BSC will be scaled larger and could be accordingly quick.

The very important to do is obviously carrying out the end-to-end analysis, for which, in fact, the theory part of this thesis (chapter 3) has already provided some useful bases (theorem 3.2 3.3). These bases can represent reality of system, but can not solve the problems yet. As what we mentioned after theorem 3.2, the stochasticity could transfer from the stochastic scaler to the (deterministic) server, that will generate the concatenation of multiple stochastic servers, which is proven to be problematic when calculating the output bound in [3]. How to solve such problem? We may find answer by utilizing our scaler. It will be interesting to transfer stochasticity among scalers and servers. Moreover, this end-to-end analysis with stochastic issue is basic, thus we can at first exclude the retransmission consideration.

References

[1] R. Boorstyn, A. Burchard, J. Liebeherr, C. Oottamakorn. Statistical service assurances for traffic scheduling algorithms. *IEEE Journal. Selected Areas in Communications*, 18(12):2651-2664, Dec 2000.

[2] A. Burchard, J. Liebeherr, S. D. Patek. A Min-Plus Calculus for End-to-End Statistical Service Guarantees. *IEEE Transactions. Information Theory*, 52(9):4105 - 4114, Sept 2006.

[3] C. Li, A. Burchard, J. Liebeherr. A Network Calculus With Effective Bandwidth. *IEEE/ACM Transactions. Networking*, 15(6):1442-1453, Dec 2007.

[4] F. Ciucu, A. Burchard, J. Liebeherr. A Network Service Curve Approach for the Stochastic Analysis of Networks. In *Proc. ACM SIGMETRICS*, pages 279-290, Jun 2005.

[5] M. Fidler. An End-to-End Probabilistic Network Calculus with Moment Generating Functions. In *14th IEEE International Workshop. Quality of Service*, pages 261-270, Jun 2006.

[6] Y. Jiang. A Basic Stochastic Network Calculus. *ACM SIGCOMM Computer Communication Review*. In *Proc. of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*. 36(4):123-134, Oct 2006.

[7] M. Fidler, J. B. Schmitt. On the Way to a Distributed Systems Calculus: An End-to-End Network Calculus with Data Scaling. In *Proc. ACM SIGMETRICS*, pages 287-298, 2006.

[8] J. B. Schmitt, F. A. Zdarsky, M. Fidler. Delay Bounds under Arbitrary Multiplexing: When Network Calculus Leaves You in the Lurch... In *Proc. IEEE INFOCOM*, pages 1669-1677, Apr 2008.

[9] J. B. Schmitt, N. Gollan, I. Martinovic. End-to-End Worst-Case Analysis of Non-FIFO Systems. *Technical Report 370/08*. University of Kaiserslautern, Germany, Aug 2008.

[10] J.-Y. Le Boudec and P. Thiran. *Network Calculus A Theory of Deterministic Queuing Systems for the Internet*. Number 2050 in Lecture Notes in Computer Science. Springer-Verlag, Berlin, Germany, 2001.

[11] R. Agrawal, R. L. Cruz, C. Okino, and R. Rajan. Performance Bounds for Flow Control Protocols. *IEEE/ACM Transactions on Networking*, 7(3):310-323, Jun 1999.

[12] RFC 3366: Advice to Link Designers on Link ARQ, Aug 2002.

[13] A. Burchard, J. Liebeherr, S. Patek. A Calculus for End-to-end Statistical Service Guaran-

tees. *Technical Report*: CS-2001-19, University of Virginia, Charlottesville, VA, USA, 2001.

[14] E. N. Gilbert. Capacity of a burst-noise channel. *Bell System Technical Journal*, vol. 39, pp. 1253-1265, Sept 1960.

[15] E. O. Elliott. Estimates of Error Rates for Codes on Burst-Noise Channels. *Bell System Technical Journal*, vol. 42, pp. 1977-1997, Sept 1963.

[16] H. S. Wang, N. Moayeri. Finite-State Markov Channel - A Useful Model for Radio Communication Channels. *IEEE TRANSACTIONS. Vehicular technology*, 44(1):163-171, Feb 1995.

[17] E. Zeidler. *Nonlinear Functional Analysis and its Applications I, Fixed-Point Theorems*. Springer-Verlag, 1986.

[18] J. B. Schmitt, U. Roedig. Sensor Network Calculus - A Framework for Worst Case Analysis. In *Proc. IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS'05)*, Marina del Rey, USA, pages 141-154, Jun 2005.

[19] J. B. Schmitt, F. A. Zdarsky, L. Thiele. A Comprehensive Worst-Case Calculus for Wireless Sensor Networks with In-Network Processing. In *IEEE Real-Time Systems Symposium (RTSS'07)*, Tucson, AZ, USA, Dec 2007.

[20] R. L. Cruz. Quality of service guarantees in virtual circuit switched networks. *IEEE J. Select. Areas Commun.*, 13(6):1048-1056, Aug 1995.

[21] H. Sariowan, R. L. Cruz, and G. C. Polyzos. Scheduling for quality of service guarantees via service curves. In *Proc. IEEE ICCCN*, pages 512–520, Sept 1995.

[22] C.-S. Chang. On deterministic traffic regulation and service guarantees: A systematic approach by filtering. *IEEE Trans. Inform. Theory*, 44(3):1097–1110, May 1998.

[23] J.-Y. Le Boudec. Application of network calculus to guaranteed service networks. *IEEE Trans. Inform. Theory*, 44(3):1087-1096, May 1998.

[24] R. L. Cruz. SCED+: Efficient management of quality of service guarantees. In *Proc. IEEE INFOCOM*, volume 2, pages 625-634, Mar 1998.

[25] R. L. Cruz. Quality of Service Management in Integrated Services Networks. In *Proc. of the 1st Semi-Annual Research Review*, CWC, UCSD, Jun 1996.

[26] F. Agharebparast, V. C. M. Leung. Modeling wireless link layer by network for efficient evaluations of multimedia QoS. *IEEE communications*, volume 2, pages 1256-1260, May 2005.

[27] J. Xie, Y. Jiang. An analysis on error servers for stochastic network calculus. *IEEE Local Computer Networks*, pages 184-191, Oct 2008.