Derivation of $T(g)$

Chun Tung Chou

March 22, 2019

The aim of this note is to explain how $T(g)$ can be derived. The derivation can also be found in [1] for a more general case. The aim of this note is to point out some intricacies that were not discussed in [1]. The derivation here is specific to the fork-join system discussed in [2].

Our aim is to find the service time of a fork-join system with N parallel servers as depicted in Figure 1. The servers are labelled as server $1, 2, ..., N$.

When a job arrives at the forking point, it creates N tasks which will be processed by the N servers in parallel. (It is important to point out that we use the term "job" to refer to the work that arrives at the forking point and the term "task" to refer to the work that is done by the parallel servers.) The service demand on server i ($i = 1, 2, .., N - 1$) is exponentially distributed with mean $\frac{1}{\mu}$ while the service demand on server N is exponentially distributed with mean $\frac{1}{\alpha\mu}$. The factor α here is to model the effect of a slower server on the service time of the entire system. We will therefore assume that $\alpha < 1$ and note that α is in fact $\frac{1}{g}$ in [2]. (Note that in [2], $\alpha > 1$ because its aim is to study the effect of the faster server which is α times faster. However, the derivation in [1] and below applies for any positive number α and you can use it to study the effect of both faster and slower servers.)

When a task has been completed by a server, it moves to the joining point. It will wait there until all the other tasks of the same job has been completed. The job is considered to be complete when all the N tasks are done. Thus the service time of a job is the service time of the longest task. There are a number of ways to derive that. We can use probability distribution but this requires you to know integral calculus. Instead we will use Markov Chain to derive the mean service time.

Let us recall that our aim is to find the mean service time of the fork-join system depicted in Figure 1. In order to find the service time of the fork-join system, we will use a closed queueing network with only 1 job as depicted in Figure 2 as the "stepping stone." We know that for a closed queueing network with only one job has the following properties

1. The mean response time of the system = the mean service time because there is no queueing

2. By Little's Law, the mean response time of the system = 1 / mean throughput of the system

1

Thus, when we have a closed queueing network with only 1 job, the mean service time of the system is 1 / the mean throughput of the system. Therefore, in order to find the mean service time, we will find the mean throughput of the system when it is in a closed queueing network with one job.

We will find the mean throughput with Markov chain. We know that a job at the forking point will create N tasks. And, when all these N tasks are completed, then the job is completed. Since the fork-join system is in a closed queueing network, once the job has reached the joining point, it will be re-circulated back to the forking point, thus again N tasks are generated.

The Markov chain for a closed fork-join (with N servers) queueing network with only one job is depicted in Figure 3. The states are of the form (i, j, k). The index i shows the number tasks that are running on the faster servers (i.e. servers 1 to $N - 1$) where $i = 1, .., N - 1$. The index j shows the number of tasks that are running on the slow server where $j = 0, 1$. The index k (which is redundant here) shows the number of tasks that are yet to be completed. The reason why k is included here is to make it consistent with the notation in [1].

The entry point to the state $(N - 1, 1, N)$ (where there are N yet-to-complete tasks) corresponds to the point in time when the job is completed and get re-circulated to the forking point, thus creating N yet to complete tasks. Let us look at some of the transitions in the state transition diagram. The transition from $(N - 1, 1, N)$ to $(N - 2, 1, N)$ corresponds to the completion of a task on one of Servers 1 to $N - 1$. This transition happens at a rate of $(N - 1)\mu$. The transition from $(N - 1, 1, N)$ to $(N - 1, 0, N - 1)$ corresponds to the completion of the task by the slow server etc. You can go through the Markov model to convince you that it does indeed model a closed queueing network of a fork-join system with one job.

Before we go on to calculate the transition probabilities, let us first recall what our aim is. We want to find the mean service time in the fork-join system by first computing what the throughput of the closed queueing network with one job is. The throughput of the closed queueing network is essentially the number of times the job moves from the joining point to the forking point in one second. This means the throughput is the rate at which the state $(N - 1, 1, N)$ is entered (noting again that the entrance to state $(N - 1, 1, N)$ corresponds to the job moving from the joining point to the forking point).

Let

- $P(N)$ = Probability in state $(N - 1, 1, N)$

- $Q(i)$ = Probability in state $(i, 1, i + 1)$ for $i = 0, ..., N - 2$

- $R(i)$ = Probability in state $(i, 0, i)$ for $i = 1, ..., p - 1$

Figure 4 depicts these probabilities. With these definition, we can now expressed the rate at which the state $(N - 1, 1, N)$ is entered as $\alpha\mu Q(0) + \mu R(1)$. By flow balance at state $(N -$

$1, 1, N)$, $\alpha\mu Q(0) + \mu R(1) = (N - 1 + \alpha)\mu P(N)$. Thus, we have

$$\text{mean service time of the fork-join system} \quad = \quad \frac{1}{(N - 1 + \alpha)\mu P(N)} \tag{1}$$

This means that if we can find $P(N)$, then we can find the mean service time. Our goal is therefore to express all the other probabilities (i.e. $Q(i)$ and $R(i)$) in terms of $P(N)$. If we can do that, then we can solve for $P(N)$.

Consider the flow balance for states $(i, 1, i + 1)$ (for $i = 1, ..., N - 2$), we have

$$
\begin{cases}
(N - 2 + \alpha)\mu Q(N - 2) &= (N - 1)\mu P(N) \\
(N - 3 + \alpha)\mu Q(N - 3) &= (N - 2)\mu Q(N - 2) \\
(N - 4 + \alpha)\mu Q(N - 4) &= (N - 3)\mu Q(N - 3) \\
\quad\quad\quad\vdots \\
(1 + \alpha)\mu Q(1) &= 2\mu Q(2)
\end{cases} \tag{2}
$$

We can express $Q(i)$ (for $i = 1, ..., N - 2$) in terms of $P(N)$, as follows,

$$Q(i) \quad = \quad F(i)P(N) \text{ for } i = 1, ..., N - 2 \tag{3}$$

where

$$F(i) \quad = \quad \prod_{j=1}^{N-i-1} \frac{N - j}{N - j - 1 + \alpha} \tag{4}$$

We still need to find $Q(0)$. By using the flow balance at state $(0, 1, 1)$, we have

$$Q(0) \quad = \quad \frac{1}{\alpha}Q(1) = \frac{1}{\alpha}F(1)P(N) \tag{5}$$

We have now expressed all the $Q(i)'s$ in terms of $P(N)$, our next task will be to express $R(i)$ in terms of $P(N)$. Consider the flow balance at state $(N - 1, 0, N - 1)$, we have

$$R(N - 1) \quad = \quad \frac{\alpha}{N - 1}P(N) \tag{6}$$

One way to proceed is to write the flow balance for states $(k, 0, k)$ where $k = 1, .., N - 2$. This will involve some algebraic manipulations. What we will do instead is to use the structure of the system and apply the flow balance to a box instead. Consider Figure 5, if we consider the flow balance of the dashed box, we have

$$(N - 2)\mu R(N - 2) \quad = \quad \alpha\mu P(N) + \alpha\mu Q(N - 2) \tag{7}$$

$$\Rightarrow R(N - 2) \quad = \quad \frac{\alpha(1 + F(N - 2))}{N - 2}P(N) \tag{8}$$

3

Similarly, if we apply the flow balance to the dashed box showed in Figure 6, we have

$$(N-3)\mu R(N-3) = \alpha\mu P(N) + \alpha\mu Q(N-2) + \alpha\mu Q(N-3) \tag{9}$$

$$\Rightarrow R(N-3) = \frac{\alpha(1 + F(N-2) + F(N-3))}{N-3}P(N) \tag{10}$$

Note that we have used the relations $Q(i) = F(i)P(N)$ (which holds for $i = 1, ..., N-2$) in the derivation.

By defining $F(N-1) = 1$, we can express the probabilities $R(i)'s$ as

$$R(i) = \frac{\alpha}{i}(\sum_{j=i}^{N-1} F(j))P(N) \tag{11}$$

We have now found all probabilities $Q(i)'s$ and $R(i)'s$ in terms of $P(N)$, by using sum of probabilities is 1, we arrive at

$$P(N) = (1 + \sum_{i=1}^{N-2} F(i) + \frac{F(1)}{\alpha} + \alpha\sum_{j=1}^{N-1}\frac{1}{j}\sum_{i=j}^{N-1} F(i))^{-1} \tag{12}$$

We can now find the mean service time using equation 1.

References

[1] Mensace et al. Static and Dynamic Processor Scheduling Disciplines in Heterogeneous Parallel Architectures," Journal of Parallel and Distributed Computing, Vol. 28 (1), July 1995, pp. 1-18.

[2] D. Mensace, "Response Time Analysis of Composite Web Services, IEEE Internet Computing, January/February 2004, Vol. 8, No. 1
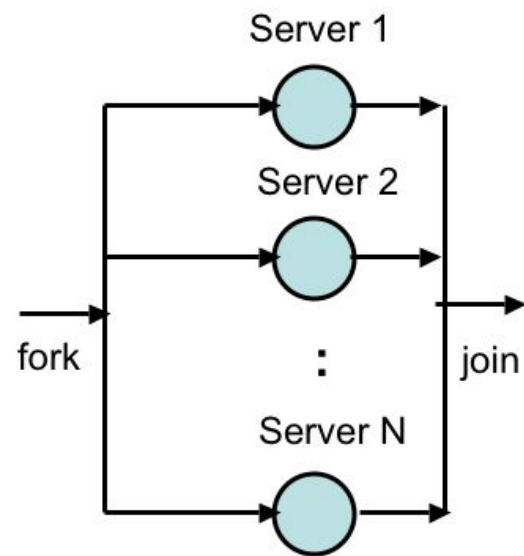
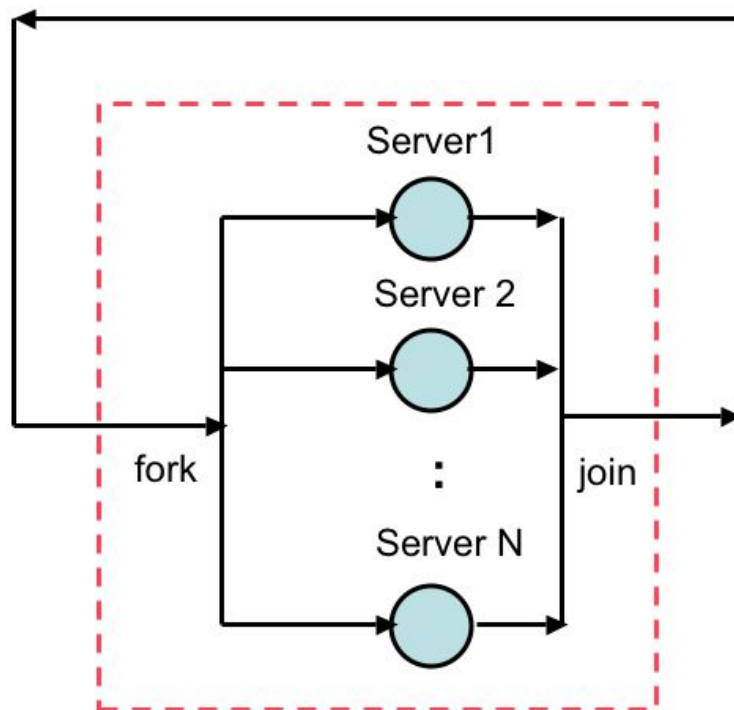Figure 1: Fork-join queue with N servers

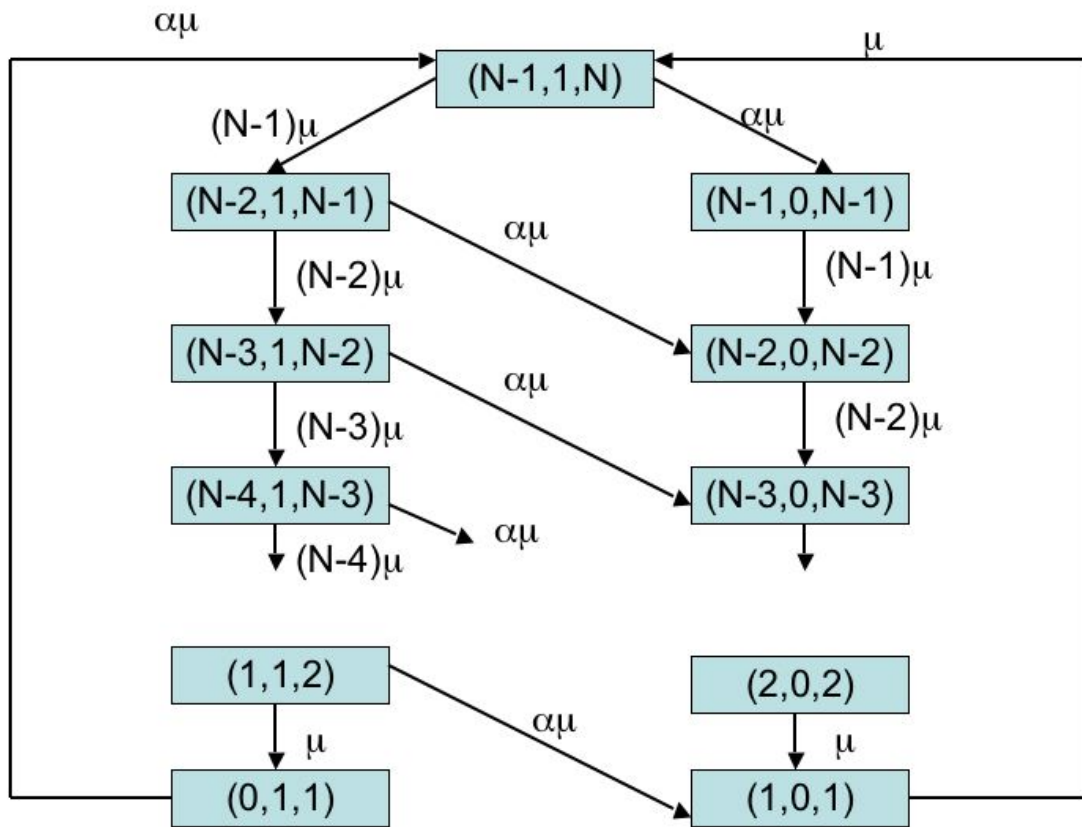Figure 2: Fork-join queue with N servers in a closed queueing network

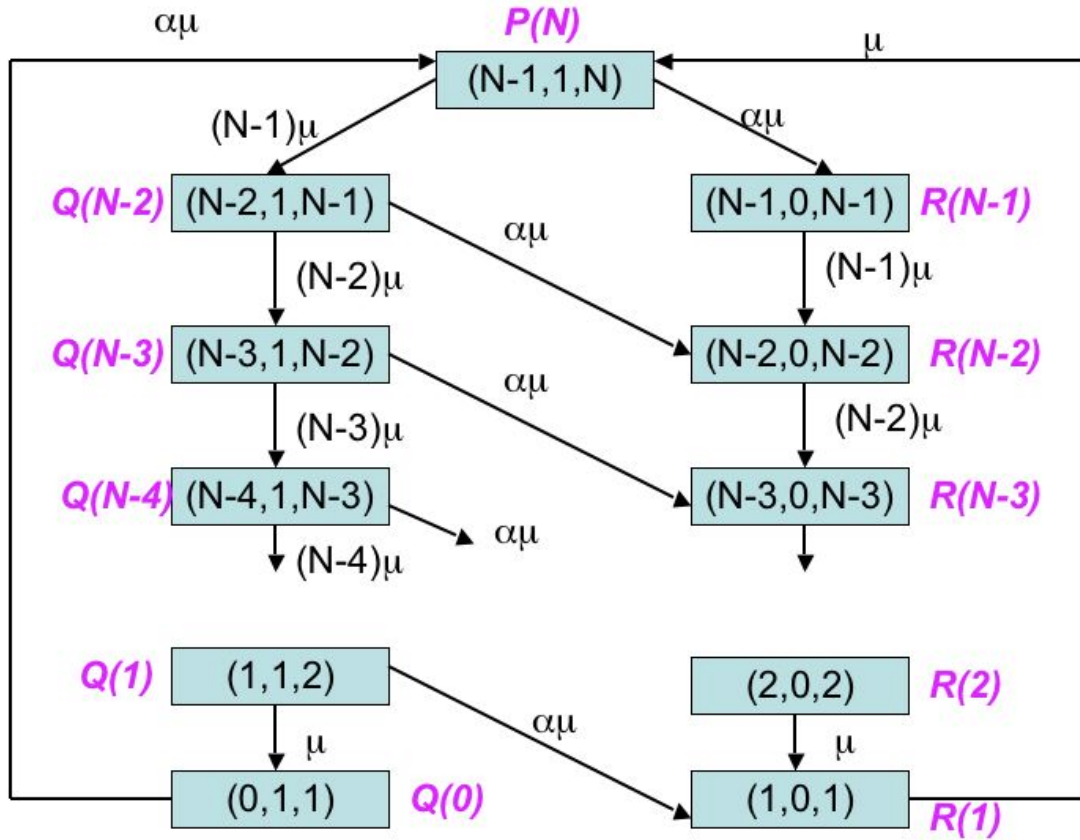Figure 3: Markov model for a closed fork-join queueing network with N servers and 1 job.

Figure 4: Markov model for a closed fork-join queueing network with N servers and 1 job. The definitions of $P(N)$, $Q(i)$ and $R(i)$ are illustrated.
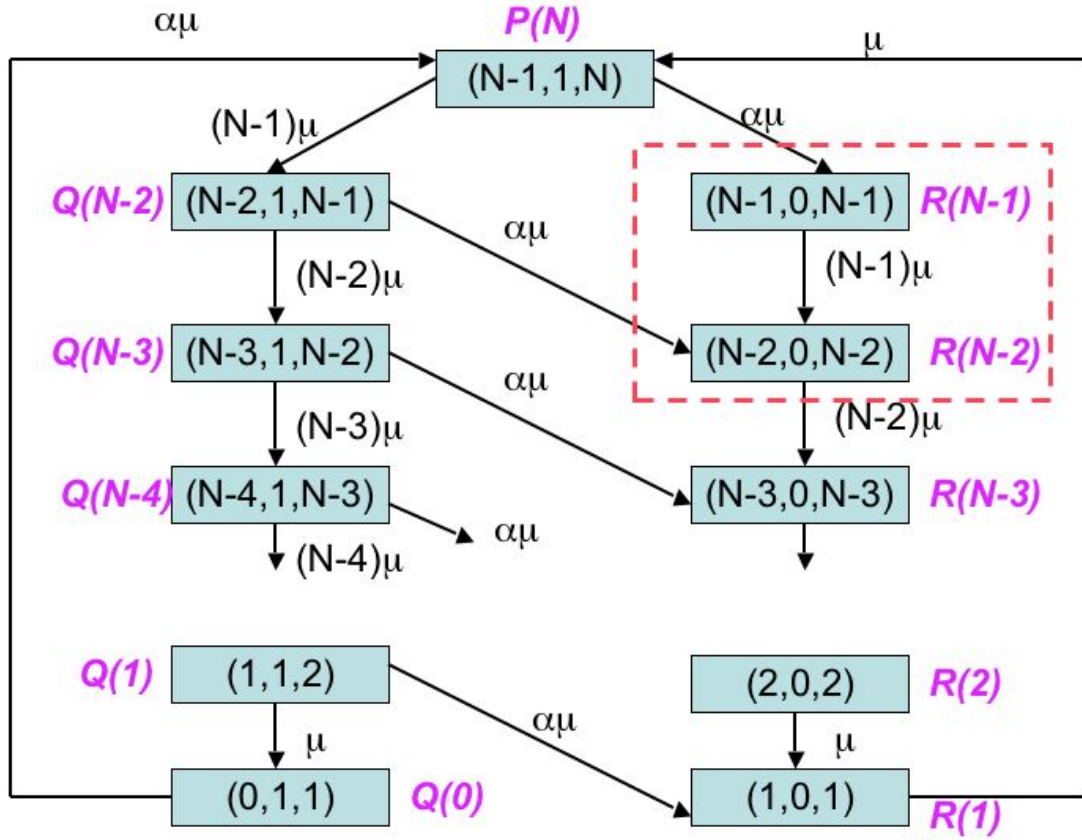
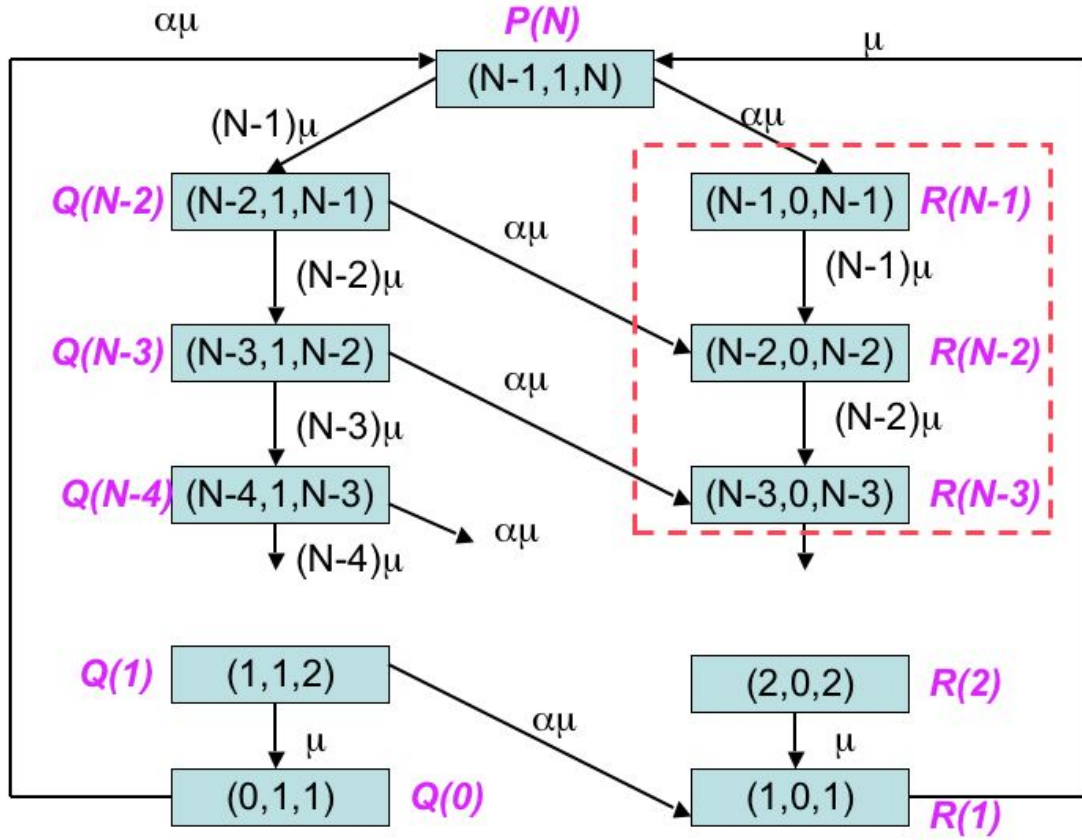Figure 5: This dashed box enables us to quickly find $R(N-2)$ in terms of $P(N)$.

Figure 6: This dashed box enables us to quickly find $R(N-3)$ in terms of $P(N)$.