SIAM REVIEW
Vol. 61, No. 2, pp. 347–360

Case for First Courses on Finite Markov Chain Modeling to Include Sojourn Time Cycle Chart*

Samuel Awoniyi†
Ira Wheaton†

Abstract. This education article presents a case for first courses on Markov chain modeling to include the topic "sojourn time cycle chart" (STC chart). This article does so through a teaching module that consists of (i) hypothetical examples illustrating the application-motivated concept of STC chart, and (ii) a network-based procedure for computing sojourn times of finite Markov chains inside subsets of their states. The sojourn time computation procedure is a simplified version of computations that already exist in several journal articles, and it applies to discrete-time Markov chain (DTMC), continuous-time Markov chain (CTMC) in rate form, and CTMC in time form. The only requirement for success in the computations is that Markov chain balance equations have a unique solution.

Key words. sojourn times cycles, Markov chains, DTMC, CTMC

AMS subject classifications. 60J10, 60G

DOI. 10.1137/16M1082147

1. Introduction.

The aim of this education article is to present a case for first courses on Markov chain modeling to include the topic of "sojourn time cycle chart" (STC chart). The case is made through a teaching module on the topic. This teaching module consists of "Examples of STC Chart Applications" in section 2 of this article and "Computation of Sojourn Times" in section 3.

The concept of Markov chain STC chart is based on "sojourn times of finite Markov chains inside subsets of their states." Such a Markov chain is required to attain some long-run pattern, but it does not have to be ergodic. For any given subset of states, say G, of such a Markov chain, the corresponding STC is defined as $s(G) + s(G^c)$, with $s(G)$ denoting the Markov chain's sojourn time inside G (that is, an average amount of time for which the Markov chain will stay or sojourn inside G before going out to G^c, each time that it does so), and $s(G^c)$ denoting the Markov chain's sojourn time inside the complement G^c. An STC chart is a chart displaying a finite number of such STCs.

As demonstrated in this article, STC charts can provide useful information in planning ahead for special resources needed in complex systems modeled as Markov chains. STCs may also be used to compare alternative system designs, yet the topic of the computation of sojourn times inside subsets of states is not covered in cur-

*Received by the editors June 28, 2016; accepted for publication (in revised form) June 8, 2018; published electronically May 8, 2019.
http://www.siam.org/journals/sirev/61-2/M108214.html

†Department of Industrial and Manufacturing Engineering, FAMU-FSU College of Engineering, Tallahassee, FL 32310 (awoniyi@eng.fsu.edu, imw11@my.fsu.edu).

rent textbooks (see, for example, [1, 3, 4, 6, 9, 11]). Several journal articles have presented procedures for computing sojourn times of Markov chains inside subsets of their states [5, 7]. However, because those journal articles have broad objectives other than Markov chain modeling, they concern rigorous analysis of random variable distribution, and their computational results are presented as complicated-looking formulas and operations.

Perhaps that explains why current textbooks only compute sojourn times of Markov chains at each state considered alone. Even Solberg's relatively recent textbook of 2009 [9], with its explicit emphasis on modeling, and the 2010 teaching-oriented textbook of Kulkarni [4] are no exception. In this regard, current textbooks and journal articles do not fully take advantage of the fact that a finite Markov chain is completely represented by a finite network (see an illustration in [8] or Chapter 8 of [9]).

As it is limited to finite Markov chains, the education module presented in this article utilizes finite networks to simplify the explanation of the procedure for computing sojourn times, and thereby avoids the kind of probability theory explanations contained in [5, 7]. The only requirement for success is that the Markov chain balance equations have a unique solution. Our computation procedure applies to discrete-time Markov chain (DTMC), continuous-time Markov chain (CTMC) in rate form, and CTMC in time form.

In particular, we want to emphasize here at the outset that Markov chain ergodicity is not required in this article, even though STC is a steady state concept. In place of ergodicity, the balance equations of each Markov chain are assumed to have a unique solution, and an algorithm of Grassman, Taskar, and Heyman [2, 10] provides a numerically stable method of computing such a unique solution of balance equations.

The remainder of this article presents the teaching module in two sections. Section 2 describes illustrative examples regarding STC chart applications, whereas section 3 describes sojourn time computation procedures for DTMC and CTMC.

2. Examples on STC Chart Applications.

We begin here in Example 1 by utilizing a machine maintenance instance to illustrate the concept of STC. Thereafter, Example 2 illustrates how STCs are combined to form STC charts. In Example 3, we state a practical application of STC chart in health care systems.

Example 1 (illustrating the concept of STC). We will utilize the following machine maintenance instance to clarify the concept of sojourn times of a Markov chain inside subsets of its states. A certain machine, M, is inspected once every 30 minutes for record purposes. The following are the possible conditions (states) in which machine M can be found upon inspection:

state 1 $\leftrightarrow$ machine M is doing self-repair;
state 2 $\leftrightarrow$ machine M is 98% efficient;
state 3 $\leftrightarrow$ machine M is 96% efficient;
state 4 $\leftrightarrow$ machine M is 94% efficient;
state 5 $\leftrightarrow$ machine M is 90% efficient.

Historical data have been generated over time regarding how machine M can go from state to state, and the following is a reliable summary of that information:

- Whenever machine M is found to be doing self-repair, the next inspection will find: machine M is at 98% efficiency about 1 out of 4 times; machine M is at 96% efficiency about 1 out of 2 times; or machine M is at 94% efficiency about 1 out of 4 times.

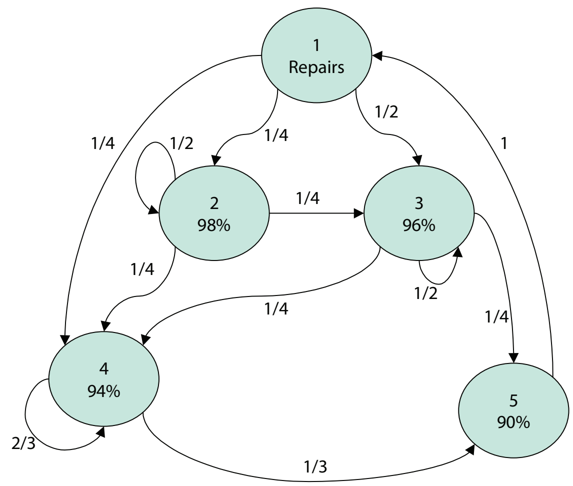

Fig. 1 *Maintenance Markov chain.*

- Whenever machine M is found to be at 98% efficiency, the next inspection will find: machine M is still at 98% efficiency about 1 out of 2 times; machine M is at 96% efficiency about 1 out of 4 times; or machine M is at 94% efficiency about 1 out of 4 times.
- Whenever machine M is found to be at 96% efficiency, the next inspection will find: machine M is still at 96% efficiency about 1 out of 2 times; machine M is at 94% efficiency about 1 out of 4 times; or machine M is at 90% efficiency 1 out of 4 times.
- Whenever machine M is found to be at 94% efficiency, the next inspection will find: machine M is still at 94% efficiency about 2 out of 3 times; or machine M is at 90% efficiency about 1 out of 3 times.
- Whenever machine M is found to be at 90% efficiency, the next inspection will certainly find machine M doing automated self-repair.

Figure 1 displays a Markov chain that models this maintenance instance. Now, for illustration, let G denote the subset of states representing 98% and 96% levels of efficiency. That is, in terms of the enumeration of states shown in Figure 1, $G = \{2, 3\}$, so that the complement $G^c = \{1, 4, 5\}$.

One can show that, in the long run, this Markov chain will settle into a repetitive pattern (steady state) going back-and-forth between G and G^c. The kind of sojourn time question that is fundamental in this article is as follows: for how many units of time (on the average) can one expect this Markov chain to remain inside G or inside G^c, each time it does so in its back-and-forth between G and G^c? One may be interested in looking into this kind of question, for example, because this G consists of states where machine M may be considered to be “not close to needing maintenance for now.”

Using the computation procedure described in section 3 of this article, one can show that this Markov chain, in steady state, will alternately spend an average of 1 hour 10 minutes inside G and an average of 2 hours 43 minutes inside G^c. Thus, the cycle of this Markov chain relative to subset G, $s(G) + s(G^c)$, is of length 3 hours 53 minutes in the long run (steady state). Such STCs (for various subsets such as G) are used to define the concept of STC chart in the next example.

G_1	■				■				■				--→
G_2	■	■		■	■		■	■		■	■		--→
G_3	■	■			■	■			■	■			--→
	1	2	3	4	5	6	7	8	9	10	11	12	--→

Fig. 2 *Example of an STC chart.*

G_1	■				■				■				--→
G_2	■	■		■	■		■	■		■	■		--→
G_3		■	■			■	■			■	■		--→
	1	2	3	4	5	6	7	8	9	10	11	12	--→

Fig. 3 *A section of another STC chart.*

Example 2 (illustrating the concept of STC chart). We consider an n-state DTMC, say M, that has the ability to settle into a pattern after a while. This means, in formal terms, that steady state is attainable for M and its transition balance equations have a unique solution.

Suppose there are three types of highly expensive resource/material that M consumes as it transitions/moves from state to state. For $j = 1, 2, 3$, let G_j denote a subset of states where resource type j is needed as M transitions about in its states, and suppose average sojourn times of M inside G_j and G_j^c have been computed (using the procedure to be described later in section 3) with results as shown in the table below.

Subset G	Sojourn time inside G, $s(G)$	Sojourn time inside G^c, $s(G^c)$
G_1	1	3
G_2	2	1
G_3	2	2

From this table, we obtain three distinct STCs, $s(G) + s(G^c)$, and combine them into the following chart (see Figure 2) which we shall refer to as an "STC chart."

In Figure 2, the numbers $1, \ldots, 12, \ldots$ enumerate transitions of M as a DTMC. For $j = 1, 2, 3$, the dark squares indicate computed sojourn time inside G_j, and the blanks between them indicate computed sojourn time inside G_j^c. The STC chart then becomes a maintenance readiness planning guide in this instance, because it can be used to plan ahead for the three high-maintenance resources corresponding to G_1, G_2, and G_3.

Note that in the particular STC chart displayed in Figure 2, it is assumed that the intersection $G_1 \cap G_2 \cap G_3$ is not empty, and then Markov chain transition #1 occurs inside $G_1 \cap G_2 \cap G_3$. However, in the case where some G_j does not intersect some other G_j's, one can readily produce corresponding STC charts. For instance, if we assume that $G_1 \cap G_2$ and $G_2 \cap G_3$ are not empty whereas $G_1 \cap G_3$ is empty, then the resultant STC chart may be as shown in Figure 3, where we are assuming that the Markov chain transition #1 occurs inside $G_1 \cap G_2$.

STC charts are intended to be utilized as a dynamic monitoring or planning tool in a manner similar to how project control Gantt charts may be used. The $s(G_i)$'s that comprise STC charts are statistical averages or expected values during steady state (that is, after the Markov chain has attained a single long-run pattern). Prior to steady state, each new observation of an ambient real-world system is a realization that may warrant an update of the associated Markov chain, and that may cause a corresponding STC chart to restart. Thus, at any time point prior to steady state, the STC chart offers only a short-term forecast to be continually updated as new real-world data become available.

Example 3 (an STC chart application in health care systems). As an illustrative example in the area of health care, consider a finite DTMC, say M, whose states represent all possible "condition observations/indicators" of a typical patient having *one* specific disorder. Suppose that this disorder is considered to be "incurable" but sustainable or "manageable," so that the system for managing the disorder may be regarded as settling into a pattern after a while.

The data modeled by this type of Markov chain might be the result of some "big data" analysis of global, historical statistics involving thousands of patients all over a large geographical region or a whole country such as the U.S.

For each $i = 1, 2$, suppose that there is also a very special, expensive health care resource with G_i denoting the subset of states of M that require special resource i. An STC chart displaying STCs for G_1 and G_2 would be a useful tool in planning ahead for the availability of resources 1 and 2. For this type of health care application of STC chart, the underlying Markov chain may have to be updated at regular time intervals. Accordingly, associated STC charts must be recomputed and restarted regularly in order for the short-term forecasts of resource needs to be realistic and useful.

A particular instance of this health care system application of STC chart is given in Example 5 after computational formulas are stated in section 3.

3. Computation of Sojourn Times.

3.1. Computing Sojourn Times of DTMC.

We begin with a statement of some notation. We suppose that we are given an n-state DTMC, which may be in terms of its network (as in Example 1 above) or equivalently its one-step transition probability matrix $P \equiv (p_{ij})$. We also suppose that we are given a subset G of the n states, and that we want to compute the average number of time units for which the given DTMC is expected to sojourn inside G before going out to sojourn inside the complement G^c.

This sojourn time computation requires that the DTMC's transition balance equations have a unique solution which we will denote by column n-vector π having components $\pi_1, \dots, \pi_n$. The components of π are nonnegative numbers, and they sum up to 1. For all DTMC's, $\pi_1, \dots, \pi_n$ will be interpreted as "fractions of total time" (which is also "fraction of total number of transitions").

We will compute $s(G)$ as

$$s(G) = t(G)/r(G),$$

where $t(G)$ indicates "fraction of time inside G" and is computed with

$$t(G) = \sum_{i \in G} \pi_i,$$

and $r(G)$ indicates "fraction of all one-step transitions that are from G to G^c" and is computed with

$$r(G) = \sum_{i \in G} \left(\sum_{j \in G^c} \pi_i p_{ij} \right).$$

In matrix form, $r(G) = \pi.P^G\mathbf{1}$, where $\mathbf{1}$ is the column n-vector of 1's and P^G is the matrix obtained when 0 replaces every element of rows of P corresponding to G^c and every element of columns of P corresponding to G.

3.1.1. A Validation of the Formulas for DTMC. First, we note that, for $i = 1,\dots,n$, the fraction π_i is the fraction of one-step transitions (that is, single inspections) that end up in state i in the long run; for DTMC, π_i is also the fraction of total time that the Markov chain may be found in state i in the long run, because each one-step transition takes one time unit for DTMC.

The computation formulas are validated using the network representation of finite DTMC, as follows. Suppose that one has observed the DTMC over a finite time period, say T, after it has attained long-run pattern or steady state. Then the portion of total time T spent inside subset G of states is

$$\sum_{i\in G} T.\pi_i, \quad \text{i.e., } T.t(G).$$

The expected total number of one-step transitions from state i ($i \in G$) to state j ($j \in G^c$) is $T.\pi_i p_{ij}$. Accordingly, the total number of G-to-G^c transitions is

$$\sum_{i\in G}\left(\sum_{j\in G^c} T.\pi_i p_{ij}\right), \quad \text{i.e., } T.r(G).$$

Therefore, the average amount of time per G-to-G^c transition is

$$\sum_{i\in G} T.\pi_i \Big/ \sum_{i\in G}\left(\sum_{j\in G^c} T.\pi_i p_{ij}\right), \quad \text{i.e., } t(G)/r(G).$$

It is straightforward to check that this result agrees with the well-known sojourn time at each state considered alone. We next give a numerical illustration.

Example 4 (some STC's for the machine maintenance instance of Example 1). From Figure 1 in Example 1, we have

$$P = \begin{pmatrix} 0 & 1/4 & 1/2 & 1/4 & 0 \\ 0 & 1/2 & 1/4 & 1/4 & 0 \\ 0 & 0 & 1/2 & 1/4 & 1/4 \\ 0 & 0 & 0 & 2/3 & 1/3 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Solving the transition balance equations, we have

$$\pi = (0.1720, 0.0860, 0.2151, 0.3548, 0.1720).$$

Now recall that we set $G = \{2,3\}$ and $G^c = \{1,4,5\}$ in Example 1. Using the procedure stated above, we then have

$$t(G) = 0.3011,\ r(G) = 0.1290, \ \text{ and } s(G) = 2.3333; \ \text{ also, } s(G^c) = 5.4167.$$

Since one time unit in this instance is equal to 30 minutes, one can see that $s(G)$ is about 1 hour 10 minutes, and $s(G^c)$ is about 2 hours 43 minutes.

The next two tables display selected STCs in this instance (Example 1) so as to possibly encourage instructive queries from students about STC length (and connectedness of Markov chain states, in general) beyond the scope of this short article. As an example of such a query, how does one explain the regularity of STC = 5.8125 for each one of $\{1\}$, $\{1,2\}$, $\{1,2,3\}$, $\{1,2,3,4\}$ on the one hand, and the apparent wildness of STC for $\{2\}$ on the other hand? This is an example of the kind of excitement that students may experience in giving practical meaning to STCs.

Subset G	$s(G)$	$s(G^c)$	STC
$\{1\}$	1	4.8125	5.8125
$\{1,2\}$	1.5	4.3125	5.8125
$\{1,2,3\}$	2.75	3.0625	5.8125
$\{1,2,3,4\}$	4.8125	1	5.8125

Subset G	$s(G)$	$s(G^c)$	STC
$\{1\}$	1	4.8125	5.8125
$\{2\}$	2	21.25	23.25
$\{3\}$	2	7.3	9.3
$\{4\}$	3	5.4545	8.4545
$\{5\}$	1	4.8125	5.8125

Example 5 (an STC chart for a particular instance of health care system of Example 3). Suppose that a patient has a disorder that cannot be cured but may be managed fairly well with monthly observations. Each monthly observation can find the patient in one of five conditions: C1, C2, C3, C4, and C5. Condition C1 is an "almost normal" condition, whereas conditions C2, C3, C4, and C5 are "slightly ill" conditions requiring different restoration treatments. Each restoration treatment takes about one month to complete, and then the patient returns to condition C1 with probability 1.

Historical data from many hundreds of other patients translate into the following health care current operational assumptions. Whenever the patient is found to be in condition C1, the next monthly observation will find that the patient is:

still in condition C1 with probability 2/3;
in condition C2 with probability 1/6;
in condition C3 with probability 1/12;
in condition C4 with probability 1/24;
in condition C5 with probability 1/24.

The treatment for condition C2 requires special planning and other special resources. One may assume that the conditions C1, C2, C3, C4, and C5 are statistically independent of one another.

This instance may be modeled as a DTMC with state labels C1, C2, C3, C4, and C5, as in the Figure 4 network below. We want to display an STC chart for $G_1 = \{C1\}$ and $G_2 = \{C2\}$ because that would be a desirable tool for planning ahead for condition C2 and also for knowing how long one might expect the desirable condition, almost normal condition C1, to last each time that it rolls around.

The corresponding one-step transition probability matrix, P, is given by

$$P = \begin{pmatrix} 2/3 & 1/6 & 1/12 & 1/24 & 1/24 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

The needed STCs are computed with the formulas given above and are displayed in the following table.

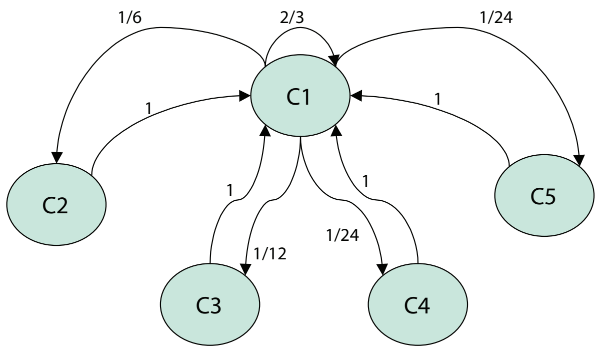

Fig. 4 *Health care example DTMC.*

{C1}	■	■	■		■	■	■		■	■	■		- - →
{C2}				■								■	- - →
	1	2	3	4	5	6	7	8	9	10	11	12	- - →

Fig. 5 *An STC chart for a health care example.*

Subset G	$s(G)$	$s(G^c)$	STC
{C1}	3	1	4
{C2}	1	7	8

Figure 5 displays a corresponding initial STC chart. It should be an instructive exercise for students to expand the STC chart in Figure 5 to include the STC for condition C3 too.

3.2. Computing Sojourn Times of CTMC. We describe here a procedure for computing sojourn times of CTMC in rate form. This procedure is similar to the procedure described above for computing sojourn times of DTMC. The main difference is that the CTMC's rate matrix, say $R \equiv (r_{ij})$, plays the role played by DTMC's matrix $P \equiv (p_{ij})$, and the unique solution of CTMC's rate balance equations, say ω, plays the role of DTMC's π. Accordingly, for CTMC in rate form, the computation of $s(G)$ is as follows:

$$s(G) = t(G)/r(G),$$

where $t(G)$ indicates "fraction of time inside G" and is computed with

$$t(G) = \sum_{i \in G} \omega_i$$

and $r(G)$ indicates "fraction of all transitions that are from G to G^c" and is computed with

$$r(G) = \sum_{i \in G} \left(\sum_{j \in G^c} \omega_i r_{ij} \right).$$

Analogous to what we stated above for DTMC, in matrix form, $r(G) = \omega.R^G\mathbf{1}$, where $\mathbf{1}$ is the column n-vector of 1's and R^G is the matrix obtained when 0 replaces every element of rows of R corresponding to G^c and every element of columns of R corresponding to G.

3.2.1. A Validation of the Formulas for CTMC in Rate Form. First, we note that for $i = 1, \dots, n$, the fraction ω_i is the fraction of total time that the Markov chain may be found in state i in the long run. The computation formulas for CTMC in rate form can be validated using a network-based reasoning in the same way as was done for DTMC above.

To do that, let us suppose that one has observed the CTMC over a finite time period, say T, after it has attained long-run pattern or steady state. Then the portion of time T spent by the CTMC inside subset G of states is

$$\sum_{i \in G} T.\omega_i, \quad \text{i.e., } T.t(G).$$

Since r_{ij} is the rate of going from state i to state j, the total number of transitions in time period $T.\omega_i$ from state i ($i \in G$) to state j ($j \in G^c$) is $T.\omega_i r_{ij}$. Accordingly, the total number of G-to-G^c transitions is

$$\sum_{i \in G} \left(\sum_{j \in G^c} T.\omega_i r_{ij} \right), \quad \text{i.e., } T.r(G).$$

Then the average amount of time per G-to-G^c transition is

$$\sum_{i \in G} T.\omega_i / \sum_{i \in G} \left(\sum_{j \in G^c} T.\omega_i r_{ij} \right), \quad \text{i.e., } t(G)/r(G).$$

The sojourn time computation procedure described above for CTMC in rate form translates into an analogous procedure for CTMC in time form, since a CTMC in time form can be readily converted into a CTMC in rate form, and vice versa.

Example 6 below gives a numerical illustration of the computation procedure described above for CTMC in rate form.

Example 6 (STC chart for an airport shuttle bus station instance). Potential passengers arrive at shuttle bus Station A of a busy airport at the rate of 20 per hour, and interarrival times have a negative exponential distribution. A four-passenger shuttle bus goes from Station A to destination point B and back to Station A, and that to-and-fro trip takes an average of 10 minutes in an exponentially distributed fashion.

At Station A, there is a passenger waiting room that can hold no more than four passengers. Potential passengers are promptly directed away to another nearby shuttle bus station whenever the waiting room in Station A is full.

Each time it returns to Station A, the shuttle bus will only take three or more passengers from Station A. The bus will wait at Station A for more passengers to arrive if there are fewer than three passengers in the waiting room.

Whenever this shuttle bus is busy, a nearby airport office will assign an attendant to temporarily staff the waiting room. If only for personnel scheduling purposes, it would be desirable to have information that helps to plan ahead in assigning attendants to staff the waiting room.

Figure 6 displays a CTMC that models this shuttle bus station. It is in rate form, with each state denoting the "number of passengers in the waiting room" along with information on whether the shuttle bus is "busy" or "idle."

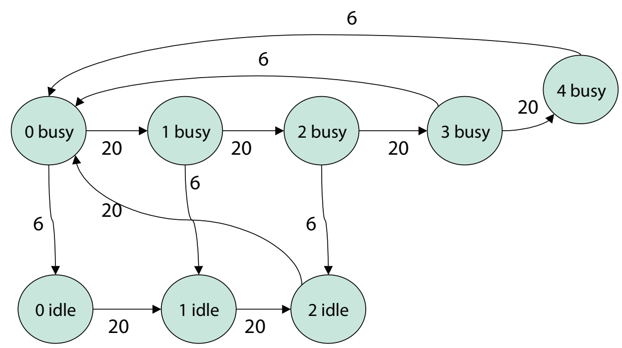

Fig. 6 *A CTMC in rate form.*

$G = \{$busy states$\}$	■	■	■		■	■	■		■	■	■		- - →
	1	2	3	4	5	6	7	8	9	10	11	12	- - →

Fig. 7 *STC chart for the airport shuttle example*

To illustrate the STC chart in this instance, let us compute the STC for $G = \{0 \text{ busy}, 1 \text{ busy}, 2 \text{ busy}, 3 \text{ busy}, 4 \text{ busy}\}$, on account of associated extra costs. From Figure 6, we have

$$R = \begin{pmatrix} 0 & 20 & 0 & 0 & 0 & 6 & 0 & 0 \\ 0 & 0 & 20 & 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 20 & 0 & 0 & 0 & 6 \\ 6 & 0 & 0 & 0 & 20 & 0 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 20 \\ 20 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Using the computation procedure stated above, we obtain

$$\omega = (0.1703, 0.1310, 0.1008, 0.0775, 0.2584, 0.0511, 0.0904, 0.1206)^T$$

and a corresponding STC is given in the following table:

Subset G	$s(G)$	$s(G^c)$	STC
$\{$all busy states$\}$	0.3059	0.1086	0.4145

A corresponding STC chart is given in Figure 7 where we approximate 0.3059 with 0.3 and 0.1086 with 0.1, and we also assume that an observation is recorded every 6 minutes (so that each unit of the enumeration $1, \ldots, 12, \ldots$ represents 6 minutes).

Example 7 (using STC information to compare inventory policies). The inventory department of a manufacturing shop floor maintains inventory of a major part under the following operational procedure. This part is issued out to other departments (in the manufacturing shop floor) upon demand one unit at a time. Time between consecutive demands for this part has an exponential distribution with mean

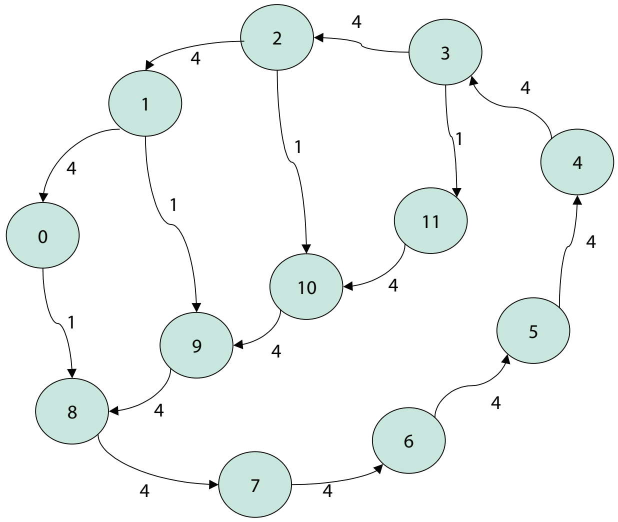

Fig. 8 *Inventory policy option* 1.

1/4 time units. Whenever the stock level of this major part is low, the department will order some restock. The time period between ordering restock and receiving it has an exponential distribution with mean of 1 time unit.

This inventory department will try to avoid having more than nine items in stock, because having more than nine items requires renting additional storage space from another department in the manufacturing shop. On the other hand, that additional storage cost must be lined up against the substantial cost of having shortages, which may result from stocking too few items of the part.

In view of those cost considerations, the department is considering two alternative restocking policies as follows. For the first alternative restocking policy, whenever the total number (of this major part) left in inventory is three, a restocking order will be placed immediately for a batch of eight additional units of the part. For the second alternative policy, whenever the total number of this part left in inventory is two, a restocking order will be placed immediately for a batch of nine additional units of the part.

We want to represent this inventory policy instance as a CTMC and use its STC information to compare the two options. To model the first policy alternative as CTMC, we let the states be $0, 1, \ldots, 11$, with state i meaning that there are i units of the part remaining in inventory. Then the desired CTMC in rate form is as shown in Figure 8.

In view of the cost considerations stated above, we are interested in obtaining STC information for $\mathrm{G}_1 = \{0,1,2\}$, the states that are considered to be close to part shortage, and $\mathrm{G}_2 = \{10,11\}$, the states where additional storage costs are incurred. Using the computational procedure described above for CTMC in rate form, we obtain the following table of STCs, which may be displayed in an STC chart similar to those presented in other examples above.

Subset G	$s(G)$	$s(G^c)$	STC
$G_1 \equiv \{0, 1, 2\}$	1	2.14	3.14
$G_2 \equiv \{10, 11\}$	0.3889	6.5889	6.9778

To model the second inventory policy alternative as a CTMC, again let the states be $0, 1, \ldots, 11$. Then the desired CTMC in rate form is almost as shown in Figure 8 for the first policy alternative, the only differences being that the four arcs (0,8), (1,9), (2,10), and (3,11) in Figure 8 will now be replaced by the three arcs (0,9), (1,10), and (2,11) each with rate 1. The corresponding STCs are given in the following table.

Subset G	$s(G)$	$s(G^c)$	STC
G_1	1	1.89	2.89
G_2	0.3889	7.6389	8.0278

Using STC information from the two tables above, it is clear that the second policy option would result in *more frequent* part shortages (because $2.89 < 3.14$) and *less frequent* additional storage cost (because $8.0278 > 6.9778$) than the first policy option. This is a sensible trade-off quantified by STC information. It should be instructive to watch students design what they consider to be additional policy options for this inventory instance, and thereafter use STC information to compare the options.

4. Concluding Remarks.

4.1. About Solving Balance Equations. We begin here by recalling what Markov chain balance equations really represent. Reasoning from Markov chain network representation, one can see that DTMC's transition balance equations are the equations obtained by setting "transition in" equal to "transition out" in the following table, along with normalization $\pi_1 + \cdots + \pi_n = 1$ and $\pi_i \geq 0$ for all i.

State	Transition in	Transition out
1	$\sum_{i=1}^{i=n} \pi_i p_{i,1}$	π_1
$\vdots$	$\vdots$	$\vdots$
n	$\sum_{i=1}^{i=n} \pi_i p_{i,n}$	π_n

In the table, P is the DTMC's one-time transition matrix. Similarly, with $R \equiv (r_{i,j})$ denoting the CTMC's rate matrix, the rate balance equations of CTMC are the equations obtained by setting "rate in" equal to "rate out" in the following table, along with $\omega_1 + \cdots + \omega_n = 1$ and $\omega_i \geq 0$ for all i.

State	Rate in	Rate out
1	$\sum_{i=1}^{i=n} \omega_i r_{i,1}$	$\omega_1 \sum_{i=1}^{i=n} r_{1,i}$
$\vdots$	$\vdots$	$\vdots$
n	$\sum_{i=1}^{i=n} \omega_i r_{i,n}$	$\omega_n \sum_{i=1}^{i=n} r_{n,i}$

In matrix form, the DTMC's transition balance equations are

$$\left\{ \begin{array}{c} \pi = P^T \pi \\ \pi_1 + \cdots + \pi_n = 1 \\ \pi \geq 0 \end{array} \right\} \cdots (1)$$

Usually, students like to solve system (1) with a fixed-point iteration (that is, P is raised to a sufficiently large power until convergence is observed). Such a fixed-point method works if DTMC is ergodic, but it may fail when DTMC is not ergodic, as one can see by considering the DTMC represented by

$$P = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

The DTMC represented by this matrix P is not ergodic because it is cyclic. Accordingly, a fixed-point iteration method does not solve its transition balance equations, even though the equations have a unique solution, namely, $\pi^T = (0,\ 1/3,\ 1/3,\ 1/3)$.

An algorithm of Grassman, Taskar, and Heyman (GTH) [2, 11] always solves DTMC transition balance equations that have unique solutions. The GTH algorithm is regarded as "stable" because it does not involve doing certain types of subtraction operations.

Regarding solving CTMC rate balance equations, one method consists of (i) first applying operations known in the literature as "uniformization" to transform the rate balance equations into equivalent DTMC transition balance equations, and (ii) thereafter applying the GTH algorithm. It is a straightforward, indirect procedure.

4.2. Other. For reasons outlined above, the STC chart, along with necessary computation of sojourn times of finite Markov chains in subsets of their states (as explained in this teaching note), should be covered in every first course on Markov chain modeling. As one can see from Examples 5, 6, and 7 of this article, the exciting nature of STC computation and interpretation can encourage students, especially students of engineering and management science, to further explore Markov chain modeling and computation. Also, the simplicity of network-based computation procedures is instructive as a testament to the efficacy of using networks to express finite stochastic processes whenever possible.

For a completeness, we briefly mention here the result of our efforts to extend our computation of STC to the case where one partitions Markov chain states into a finite number of disjoint subsets, $G_1, \ldots, G_m$, instead of just G, G^c. We have found that our computational method does not cover this extension in general because the "rate of going from G_i to G_j" may not be equal to the "rate of going from G_j to G_i," unlike $r(G) = r(G^c)$. However, as demonstrated in Example 2, an STC chart displaying STC's for "G_i, G_j" pairs can give useful information for monitoring purposes.

REFERENCES

[1] R. Durrett, *Essentials of Stochastic Processes*, Springer, New York, 1999. (Cited on p. 348)

[2] W. K. Grassman, M. I. Taskar, and D. P. Heyman, *Regenerative analysis and steady state distributions for Markov chains*, Oper. Res., 33 (1985), pp. 1107–1116. (Cited on pp. 348, 359)

[3] S. Karlin and H. M. Taylor, *A First Course in Stochastic Processes*, Academic Press, New York, 1975. (Cited on p. 348)

[4] V. G. Kulkarni, *Introduction to Modeling and Analysis of Stochastic Systems*, Springer, New York, 2010. (Cited on p. 348)

[5] C. Nunes and A. Pacheco, *Sojourn and passage times in Markov chains*, in Matrix-Analytic Methods, Proceedings of the Fourth International Conference, World Scientific, 2002, pp. 311–332. (Cited on p. 348)

[6] S. M. Ross, *Introduction to Probability Models*, 10th ed., Academic Press, Boston, 2010. (Cited on p. 348)

[7] G. Rubino and B. Sericola, *Sojourn times in finite Markov processes*, J. Appl. Prob., 27 (1989), pp. 744–756. (Cited on p. 348)

[8] J. Solberg, *A graph theoretic formula for the steady-state distribution of finite Markov processes*, Management Sci., 21 (1974/1975), pp. 1040–1048. (Cited on p. 348)

[9] J. J. Solberg, *Modeling Random Processes for Engineers and Managers*, John Wiley & Sons, 2009. (Cited on p. 348)

[10] W. J. Stewart, *Introduction to Numerical Solution of Markov Chains*, Princeton University Press, Princeton, NJ, 1994. (Cited on p. 348)

[11] W. J. Stewart, *Probability, Markov Chain, Queues and Simulation: The Mathematical Basis of Performance Modeling*, Princeton University Press, Princeton, NJ, 2009. (Cited on pp. 348, 359)