E-Companion — "A Geometrically Convergent Solution to Spatial Hypercube Queueing Models"

Cheng Hua*, Jun Luo*‡, Arthur J. Swersey†, Yixing Wen*

*Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai 200030

‡Data-Driven Management Decision Making Lab, Shanghai Jiao Tong University, Shanghai 200030

†Yale School of Management, Yale University, New Haven, CT 06511

{cheng.hua, jluo_ms}@sjtu.edu.cn, arthur.swersey@yale.edu, star-winmex@sjtu.edu.cn

EC.1. Miscellaneous Proofs

EC.1.1. Proof of Proposition 1

Proof. We show that the conditional probability formulation is a transformation from the two-dimensional formulation. First, we express the joint probability $P\{n, B_m\}$ as

$$P\{n, B_m\} = p(n) \cdot p_n(B_m). \tag{EC.1}$$

Next, the balance equations (4) become, for $m \in \mathscr{C}_n$ and $n = 0, 1 \ldots, N$,

$$p(n)p_n(B_m)(\lambda_m + \mu_m) = \sum_{l \in \mathscr{C}_{n-1}} p(n-1)p_{n-1}(B_l)\lambda_{lm} + \sum_{l \in \mathscr{C}^m_{n+1}} p(n+1)p_{n+1}(B_l)\mu_{lm}. \tag{EC.2}$$

We note that $p(n)$ represents the steady-state probability in the birth-death process, and is determined by

$$p(n) = \frac{1}{G}\prod_{s=1}^{n}\frac{\lambda(s-1)}{\mu(s)}, \quad \text{where} \quad G = 1 + \sum_{n=1}^{N}\prod_{s=1}^{n}\frac{\lambda(s-1)}{\mu(s)}.$$

Here, G serves as a normalization factor for the birth-death process. Applying the formula of $p(n)$ to the above equation (EC.2), we obtain

$$\begin{aligned}\frac{1}{G}\prod_{s=1}^{n}\frac{\lambda(s-1)}{\mu(s)}p_n(B_m)(\lambda_m + \mu_m) = &\sum_{l \in \mathscr{C}_{n-1}} \frac{1}{G}\prod_{s=1}^{n-1}\frac{\lambda(s-1)}{\mu(s)}p_{n-1}(B_l)\lambda_{lm} \\ &+ \sum_{l \in \mathscr{C}^m_{n+1}} \frac{1}{G}\prod_{s=1}^{n+1}\frac{\lambda(s-1)}{\mu(s)}p_{n+1}(B_l)\mu_{lm}.\end{aligned}$$

After canceling G and terms in the product, the equation simplifies to

$$p_n(B_m)(\lambda_m + \mu_m) = \sum_{l \in \mathscr{C}_{n-1}} p_{n-1}(B_l)\frac{\mu(n)}{\lambda(n-1)}\lambda_{lm} + \sum_{l \in \mathscr{C}^m_{n+1}} p_{n+1}(B_l)\frac{\lambda(n)}{\mu(n+1)}\mu_{lm}.$$

By dividing both sides of this equation by $\lambda_m + \mu_m$, we obtain

$$p_n(B_m) = \sum_{l \in \mathscr{C}_{n-1}} p_{n-1}(B_l)\frac{\mu(n)}{\lambda(n-1)}\frac{\lambda_{lm}}{\lambda_m + \mu_m} + \frac{\mu_{lm}}{\lambda_m + \mu_m}\sum_{l \in \mathscr{C}^m_{n+1}} p_{n+1}(B_l)\frac{\lambda(n)}{\mu(n+1)}.$$

We also require that the conditional probabilities sum to 1 for each layer $\mathscr{C}_n$, for $n = 0, 1, \ldots, N$,

$$\sum_{m \in \mathscr{C}_n} p_n(B_m) = 1.$$

In the concluding step, we determine the probability distribution of steady states using the conditional probabilities $p_n(B_m)$. The relationship is expressed as

$$P\{B_m\} = P\{w(B_m), B_m\} = p(w(B_m)) \cdot p_{w(B_m)}(B_m).$$

where we replace $n = w(B_m)$ in Equation (EC.1) in the above equation. □

EC.1.2. Proof of Lemma 1

Proof. We prove this lemma by induction. For $k = 1$, we initialize $p_n^1(B_m) = p_n^{1,1}(B_m) = \frac{1}{\binom{N}{n}}$. This ensures $\sum_{m \in \mathscr{C}_n} p_n^{1,1}(B_m) = 1$ for all $n = 0, 1, \ldots, N$.

Assuming the statement holds for $k-1$, we proceed with the inductive hypothesis to verify the properties for k. Note that $\lambda_m = \sum_l \lambda_{ml} = \lambda$, which is valid for every state except when all units are busy. This is attributed to the sum of transition rates out of state B_m upon arrival being equal to the total system arrival rate λ. Subsequently, we derive the following results:

(i) For $n = 0$, we have only one state, B_0. Using Equation (12), we can simplify this as:

$$p_0^{k,r}(B_0)\lambda = \sum_{l \in \mathscr{C}_1} p_1^{k-1}(B_l) \frac{\lambda^{k-1}(0)}{\mu^{k-1}(1)} \mu_{l0} = \frac{\lambda^{k-1}(0)}{\mu^{k-1}(1)} \sum_{l \in \mathscr{C}_1} p_1^{k-1}(B_l) \mu_{l0}.$$

By utilizing Equations (13) and (14), we have:

$$p_0^{k,r}(B_0)\lambda = \frac{\lambda^{k-1}(0)}{\mu^{k-1}(1)} \mu^{k-1}(1) = \lambda^{k-1}(0) = p_0^{k-1}(B_0)\,\lambda.$$

Thus, $p_0^{k,r}(B_0) = p_0^{k-1}(B_0)$ for all $r = 1, 2, \ldots$. Given our assumption that the two statements are true for $k-1$, they are also valid for k when $n = 0$.

(ii) Assume the statements hold for $n-1$, with $p_{n-1}^{k,r}(B_m)$, $\lambda^{k,r}(n)$ converging to $p_{n-1}^k(B_m)$ and $\lambda^k(n)$ respectively. Then, using Equation (12) (i.e., subtracting it by replacing r with $r-1$) in Equation (12), we have:

$$\left(p_n^{k,r}(B_m) - p_n^{k,r-1}(B_m)\right)(\lambda + \mu_m) = \sum_{l \in \mathscr{C}_{n-1}} p_{n-1}^k(B_l) \frac{\mu^{k,r-1}(n)}{\lambda^k(n-1)} \lambda_{lm} - \sum_{l \in \mathscr{C}_{n-1}} p_{n-1}^k(B_l) \frac{\mu^{k,r-2}(n)}{\lambda^k(n-1)} \lambda_{lm}.$$

By the induction hypothesis, $\lambda^k(n-1) = \lambda$ because of the normalization property. We define

$$\Delta_{n,m}^{k,r} = p_n^{k,r}(B_m) - p_n^{k,r-1}(B_m),$$

and rewrite the above equation as:

$$\begin{aligned}\Delta_{n,m}^{k,r}\left(\lambda+\mu_m\right) &= \left(\frac{1}{\lambda}\sum_{l\in\mathscr{C}_{n-1}} p_{n-1}^k(B_l)\lambda_{lm}\right)\left(\mu^{k,r-1}(n)-\mu^{k,r-2}(n)\right)\\ &= \left(\frac{1}{\lambda}\sum_{l\in\mathscr{C}_{n-1}} p_{n-1}^k(B_l)\lambda_{lm}\right)\left(\sum_{m\in\mathscr{C}_n}\mu_m\Delta_{n,m}^{k,r-1}\right),\end{aligned}$$

where the second equality arises from Equation (14). By applying the *Triangle Inequality*, and taking absolute values on both sides, we obtain

$$\left(\lambda+\mu_m\right)\left|\Delta_{n,m}^{k,r}\right| \le \left(\frac{1}{\lambda}\sum_{l\in\mathscr{C}_{n-1}} p_{n-1}^k(B_l)\lambda_{lm}\right)\left(\sum_{m\in\mathscr{C}_n}\mu_m\left|\Delta_{n,m}^{k,r-1}\right|\right).$$

Summing both sides over all $m\in\mathscr{C}_n$, we obtain

$$\begin{aligned}\sum_{m\in\mathscr{C}_n}\left(\lambda+\mu_m\right)\left|\Delta_{n,m}^{k,r}\right| &\le \left(\sum_{m\in\mathscr{C}_n}\mu_m\left|\Delta_{n,m}^{k,r-1}\right|\right)\sum_{m\in\mathscr{C}_n}\left(\frac{1}{\lambda}\sum_{l\in\mathscr{C}_{n-1}} p_{n-1}^k(B_l)\lambda_{lm}\right)\\ &= \left(\sum_{m\in\mathscr{C}_n}\mu_m\left|\Delta_{n,m}^{k,r-1}\right|\right)\left(\frac{1}{\lambda}\sum_{l\in\mathscr{C}_{n-1}} p_{n-1}^k(B_l)\underbrace{\sum_{m\in\mathscr{C}_n}\lambda_{lm}}_{=\lambda}\right)\\ &= \left(\sum_{m\in\mathscr{C}_n}\mu_m\left|\Delta_{n,m}^{k,r-1}\right|\right)\left(\frac{1}{\lambda}\lambda^k(n-1)\right)\\ &= \sum_{m\in\mathscr{C}_n}\left(\lambda+\mu_m\right)\left|\Delta_{n,m}^{k,r-1}\right| - \lambda\sum_{m\in\mathscr{C}_n}\left|\Delta_{n,m}^{k,r-1}\right|.\end{aligned}$$

Define $A_r=\sum_{m\in\mathscr{C}_n}\left|\Delta_{n,m}^{k,r}\right|$. Then, for all r, we have

$$\begin{aligned}A_r &< \sum_{m\in\mathscr{C}_n}\left(\lambda+\mu_m\right)\left|\Delta_{n,m}^{k,r}\right|\\ &\le \sum_{m\in\mathscr{C}_n}\left(\lambda+\mu_m\right)\left|\Delta_{n,m}^{k,r-1}\right| - \lambda A_{r-1}\\ &\le \cdots\\ &\le \sum_{m\in\mathscr{C}_n}\left(\lambda+\mu_m\right)\left|\Delta_{n,m}^{k,1}\right| - \lambda\sum_{s=1}^{r-1}A_s.\end{aligned}$$

Using the induction hypothesis $\sum_{m\in\mathscr{C}_n}p_n^{k-1}(B_m)=1$, we have

$$\sum_{m\in\mathscr{C}_n}\left(\lambda+\mu_m\right)\left|\Delta_{n,m}^{k,1}\right| \le \sum_{m\in\mathscr{C}_n}\left(\lambda+\mu_m\right)\left(p_n^{k,1}(B_m)+p_n^{k-1}(B_m)\right)=D_n^{k,1}.$$

At iteration k, $p_n^{k-1}(B_m)$ is fixed, and given all probabilities and transition rates before layer n at step k, $p_n^{k,1}(B_m)$ is known, so $D_n^{k,1}$ is fixed. Because $A_s\ge 0$ for any $s\ge 1$, the sum $\sum_{s=1}^{r-1}A_s$ is nondecreasing and bounded above by $D_n^{k,1}/\lambda$. Therefore,

$$\lim_{r\to\infty}\left|\Delta_{n,m}^{k,r}\right| \le \lim_{r\to\infty}A_r=0,$$

and $\left|p_n^{k,r}(B_m)-p_n^{k,r-1}(B_m)\right|$ converges to 0 as r approaches infinity.

To prove the normalization property, we revisit Equation (12) and take the sum of both sides:

$$\begin{aligned}
&\sum_{m\in\mathscr{C}_n}(\lambda+\mu_m)\,p_n^{k,r}(B_m)\\
&=\sum_{m\in\mathscr{C}_n}\sum_{l\in\mathscr{C}_{n-1}}p_{n-1}^{k}(B_l)\frac{\mu^{k,r-1}(n)}{\lambda^k(n-1)}\lambda_{lm}+\sum_{m\in\mathscr{C}_n}\sum_{l\in\mathscr{C}_{n+1}^m}p_{n+1}^{k-1}(B_l)\frac{\lambda^{k-1}(n)}{\mu^{k-1}(n+1)}\mu_{lm}\\
&=\left(\frac{\mu^{k,r-1}(n)}{\lambda^k(n-1)}\underbrace{\sum_{l\in\mathscr{C}_{n-1}}p_{n-1}^{k}(B_l)\sum_{m\in\mathscr{C}_n}\lambda_{lm}}_{=\lambda^k(n-1)}\right)+\left(\frac{\lambda^{k-1}(n)}{\mu^{k-1}(n+1)}\underbrace{\sum_{l\in\mathscr{C}_{n+1}}p_{n+1}^{k-1}(B_l)\sum_{m\in\mathscr{C}_n}\mu_{lm}}_{=\mu^{k-1}(n+1)}\right)\\
&=\mu^{k,r-1}(n)+\lambda\\
&=\sum_{m\in\mathscr{C}_n}\mu_m p_n^{k,r-1}(B_m)+\lambda.
\end{aligned}$$

Taking the limit of both sides of the equation, we have:

$$\lim_{r\to\infty}\sum_{m\in\mathscr{C}_n}\mu_m p_n^{k,r}(B_m)+\lim_{r\to\infty}\sum_{m\in\mathscr{C}_n}\lambda p_n^{k,r}(B_m)=\lim_{r\to\infty}\sum_{m\in\mathscr{C}_n}\mu_m p_n^{k,r-1}(B_m)+\lambda.$$

Given our earlier proof that $p_n^{k,r}(B_m)$ converges, the first terms on both sides of the above equation are equal. Therefore,

$$\lambda\lim_{r\to\infty}\sum_{m\in\mathscr{C}_n}p_n^{k,r}(B_m)=\lambda.$$

This implies

$$\lim_{r\to\infty}\sum_{m\in\mathscr{C}_n}p_n^{k,r}(B_m)=1.$$

We have thus demonstrated that these properties are valid for layer n at step k, which concludes our proof. □

EC.1.3. Proof of Theorem 1

Proof. By Lemma 1, we can write Equation (12) as

$$p_n^k(B_m)=\sum_{l\in\mathscr{C}_{n-1}}p_{n-1}^{k}(B_l)\frac{\mu^k(n)}{\lambda^k(n-1)}\frac{\lambda_{lm}}{\lambda_m+\mu_m}+\sum_{l\in\mathscr{C}_{n+1}^m}p_{n+1}^{k-1}(B_l)\frac{\lambda^{k-1}(n)}{\mu^{k-1}(n+1)}\frac{\mu_{lm}}{\lambda_m+\mu_m}. \quad \text{(EC.3)}$$

By subtracting the value at the k-th iteration from the value at the $(k+1)$-th iteration, we obtain

$$\begin{aligned}
p_n^{k+1}(B_m)-p_n^k(B_m)=&\sum_{l\in\mathscr{C}_{n-1}}\left(\frac{\mu^{k+1}(n)}{\lambda^{k+1}(n-1)}p_{n-1}^{k+1}(B_l)-\frac{\mu^k(n)}{\lambda^k(n-1)}p_{n-1}^{k}(B_l)\right)\frac{\lambda_{lm}}{\lambda+\mu_m}\\
&+\sum_{l\in\mathscr{C}_{n+1}^m}\left(\frac{\lambda^k(n)}{\mu^k(n+1)}p_{n+1}^{k}(B_l)-\frac{\lambda^{k-1}(n)}{\mu^{k-1}(n+1)}p_{n+1}^{k-1}(B_l)\right)\frac{\mu_{lm}}{\lambda+\mu_m}.
\end{aligned}$$

By taking the absolute values on both sides and applying the *Triangle Inequality*, we obtain

$$\begin{aligned}
\left|p_n^{k+1}(B_m)-p_n^k(B_m)\right| \le & \left|\sum_{l\in\mathscr{C}_{n-1}}\left(\frac{\mu^{k+1}(n)}{\lambda^{k+1}(n-1)}p_{n-1}^{k+1}(B_l)-\frac{\mu^k(n)}{\lambda^k(n-1)}p_{n-1}^k(B_l)\right)\frac{\lambda_{lm}}{\lambda+\mu_m}\right| \\
&+\left|\sum_{l\in\mathscr{C}_{n+1}^m}\left(\frac{\lambda^k(n)}{\mu^k(n+1)}p_{n+1}^k(B_l)-\frac{\lambda^{k-1}(n)}{\mu^{k-1}(n+1)}p_{n+1}^{k-1}(B_l)\right)\frac{\mu_{lm}}{\lambda+\mu_m}\right| \\
\le & \frac{1}{\lambda+\underline{\mu}_n}\left|\sum_{l\in\mathscr{C}_{n-1}}\left(\frac{\mu^{k+1}(n)}{\lambda^{k+1}(n-1)}p_{n-1}^{k+1}(B_l)-\frac{\mu^k(n)}{\lambda^k(n-1)}p_{n-1}^k(B_l)\right)\lambda_{lm}\right| \\
&+\frac{1}{\lambda+\underline{\mu}_n}\left|\sum_{l\in\mathscr{C}_{n+1}^m}\left(\frac{\lambda^k(n)}{\mu^k(n+1)}p_{n+1}^k(B_l)-\frac{\lambda^{k-1}(n)}{\mu^{k-1}(n+1)}p_{n+1}^{k-1}(B_l)\right)\mu_{lm}\right|.
\end{aligned}$$

Taking the sum on both sides, we obtain

$$\begin{aligned}
\sum_{m\in\mathscr{C}_n}\left|p_n^{k+1}(B_m)-p_n^k(B_m)\right| \le & \frac{1}{\lambda+\underline{\mu}_n}\sum_{m\in\mathscr{C}_n}\left|\sum_{l\in\mathscr{C}_{n-1}}\left(\frac{\mu^{k+1}(n)}{\lambda^{k+1}(n-1)}p_{n-1}^{k+1}(B_l)-\frac{\mu^k(n)}{\lambda^k(n-1)}p_{n-1}^k(B_l)\right)\lambda_{lm}\right| \\
&+\frac{1}{\lambda+\underline{\mu}_n}\sum_{m\in\mathscr{C}_n}\left|\sum_{l\in\mathscr{C}_{n+1}^m}\left(\frac{\lambda^k(n)}{\mu^k(n+1)}p_{n+1}^k(B_l)-\frac{\lambda^{k-1}(n)}{\mu^{k-1}(n+1)}p_{n+1}^{k-1}(B_l)\right)\mu_{lm}\right|.
\end{aligned}$$

According to Lemma 1, we have:

$$\begin{aligned}
&\lambda^{k+1}(n-1)=\lambda^k(n-1)=\lambda^{k-1}(n)=\lambda^k(n)=\lambda; \\
&\mu^{k+1}(n)\le\overline{\mu}_n,\ \mu^k(n)\le\overline{\mu}_n,\ \mu^k(n+1)\ge\underline{\mu}_{n+1},\ \mu^{k-1}(n+1)\ge\underline{\mu}_{n+1}.
\end{aligned}$$

We also define:

$$\Delta_{n,m}^k=p_n^k(B_m)-p_n^{k-1}(B_m).$$

Therefore, we have

$$\begin{aligned}
\sum_{m\in\mathscr{C}_n}\left|\Delta_{n,m}^{k+1}\right| \le & \frac{1}{\lambda+\underline{\mu}_n}\left(\sum_{m\in\mathscr{C}_n}\sum_{l\in\mathscr{C}_{n-1}}\frac{\overline{\mu}_n}{\lambda}\lambda_{lm}\left|\Delta_{n-1,l}^{k+1}\right|+\sum_{m\in\mathscr{C}_n}\sum_{l\in\mathscr{C}_{n+1}^m}\frac{\lambda}{\underline{\mu}_{n+1}}\mu_{lm}\left|\Delta_{n+1,l}^k\right|\right) \\
= & \frac{1}{\lambda+\underline{\mu}_n}\left(\sum_{l\in\mathscr{C}_{n-1}}\frac{\overline{\mu}_n}{\lambda}\left|\Delta_{n-1,l}^{k+1}\right|\underbrace{\sum_{m\in\mathscr{C}_n}\lambda_{lm}}_{=\lambda}+\sum_{l\in\mathscr{C}_{n+1}}\frac{\lambda}{\underline{\mu}_{n+1}}\left|\Delta_{n+1,l}^k\right|\underbrace{\sum_{m\in\mathscr{C}_n}\mu_{lm}}_{=\mu_l}\right) \\
= & \frac{1}{\lambda+\underline{\mu}_n}\left(\sum_{l\in\mathscr{C}_{n-1}}\overline{\mu}_n\left|\Delta_{n-1,l}^{k+1}\right|+\sum_{l\in\mathscr{C}_{n+1}}\frac{\lambda\mu_l}{\underline{\mu}_{n+1}}\left|\Delta_{n+1,l}^k\right|\right) \\
\le & \frac{1}{\lambda+\underline{\mu}_n}\left(\overline{\mu}_n\sum_{l\in\mathscr{C}_{n-1}}\left|\Delta_{n-1,l}^{k+1}\right|+\frac{\lambda\overline{\mu}_{n+1}}{\underline{\mu}_{n+1}}\sum_{l\in\mathscr{C}_{n+1}}\left|\Delta_{n+1,l}^k\right|\right).
\end{aligned}$$

Let $M_{k,n} = \sum_{l \in \mathscr{C}_n} \left|\Delta^k_{n,l}\right|$ and $\gamma_n = \frac{\overline{\mu}_n}{\underline{\mu}_n}$. Then, we have

$$M_{k+1,n} \le \frac{1}{\lambda + \underline{\mu}_n} \left(\overline{\mu}_n M_{k+1,n-1} + \gamma_{n+1} \lambda M_{k,n+1}\right).$$

For every step k, we have:

$$\begin{aligned} M_{k,0} &= \sum_{l \in \mathscr{C}_0} \left|p^k_0(B_l) - p^{k-1}_0(B_l)\right| = \left|p^k_0(B_0) - p^{k-1}_0(B_0)\right| = |1-1| = 0, \\ M_{k,N} &= \sum_{l \in \mathscr{C}_N} \left|p^k_N(B_l) - p^{k-1}_N(B_l)\right| = \left|p^k_N(B_{2^N-1}) - p^{k-1}_N(B_{2^N-1})\right| = |1-1| = 0. \end{aligned}$$

Let $M_k = \sup_n M_{k,n}$,

$$\begin{aligned} M_{k+1,1} &\le \frac{1}{\lambda + \underline{\mu}_1}\left(\overline{\mu}_1 M_{k+1,0} + \gamma_2 \lambda M_{k,2}\right) = \frac{1}{\lambda + \underline{\mu}_1} \gamma_2 \lambda M_{k,2} \le \frac{\lambda}{\lambda + \underline{\mu}_1} \gamma_2 M_k \\ M_{k+1,2} &\le \frac{1}{\lambda + \underline{\mu}_2}\left(\overline{\mu}_2 M_{k+1,1} + \gamma_3 \lambda M_{k,3}\right) \le \frac{\lambda}{\lambda + \underline{\mu}_2}\left(\frac{\overline{\mu}_2}{\lambda + \underline{\mu}_1}\gamma_2 + \gamma_3\right) M_k \\ M_{k+1,3} &\le \frac{1}{\lambda + \underline{\mu}_3}\left(\overline{\mu}_3 M_{k+1,2} + \gamma_4 \lambda M_{k,4}\right) \le \frac{\lambda}{\lambda + \underline{\mu}_3}\left(\frac{\overline{\mu}_3}{\lambda + \underline{\mu}_2}\left(\frac{\overline{\mu}_2}{\lambda + \underline{\mu}_1}\gamma_2 + \gamma_3\right) + \gamma_4\right) M_k \\ &= \frac{\lambda}{\lambda + \underline{\mu}_3}\left(\frac{\overline{\mu}_3}{\lambda + \underline{\mu}_2}\frac{\overline{\mu}_2}{\lambda + \underline{\mu}_1}\gamma_2 + \frac{\overline{\mu}_3}{\lambda + \underline{\mu}_2}\gamma_3 + \gamma_4\right) M_k \\ M_{k+1,4} &\le \frac{1}{\lambda + \underline{\mu}_4}\left(\overline{\mu}_4 M_{k+1,3} + \gamma_5 \lambda M_{k,5}\right) \\ &\le \frac{\lambda}{\lambda + \underline{\mu}_4}\left(\frac{\overline{\mu}_4}{\lambda + \underline{\mu}_3}\left(\frac{\overline{\mu}_3}{\lambda + \underline{\mu}_2}\frac{\overline{\mu}_2}{\lambda + \underline{\mu}_1}\gamma_2 + \frac{\overline{\mu}_3}{\lambda + \underline{\mu}_2}\gamma_3 + \gamma_4\right) + \gamma_5\right) M_k \\ &= \frac{\lambda}{\lambda + \underline{\mu}_4}\left(\frac{\overline{\mu}_4}{\lambda + \underline{\mu}_3}\frac{\overline{\mu}_3}{\lambda + \underline{\mu}_2}\frac{\overline{\mu}_2}{\lambda + \underline{\mu}_1}\gamma_2 + \frac{\overline{\mu}_4}{\lambda + \underline{\mu}_3}\frac{\overline{\mu}_3}{\lambda + \underline{\mu}_2}\gamma_3 + \frac{\overline{\mu}_4}{\lambda + \underline{\mu}_3}\gamma_4 + \gamma_5\right) M_k. \end{aligned}$$

For notational convenience, we define:

$$\begin{aligned} \phi_{n,q} &= \gamma_q \prod_{j=q}^{n} \frac{\overline{\mu}_j}{\lambda + \underline{\mu}_{j-1}} \ (2 \le q \le n) \quad \text{and} \quad \phi_{n,n+1} = \gamma_{n+1}, \\ \Phi_n &= \frac{\lambda}{\lambda + \underline{\mu}_n} \sum_{q=2}^{n+1} \phi_{n,q}, \ (n \ge 2). \end{aligned}$$

We proceed by induction to show that for any $n \ge 2$,

$$M_{k+1,n} \le \Phi_n M_k. \tag{EC.4}$$

We have previously shown the case for $n = 2$. Assuming the formula holds for $n = s$, for the case $n = s+1$, we have

$$M_{k+1,s+1} \le \frac{1}{\lambda + \underline{\mu}_{s+1}}\left(\overline{\mu}_{s+1} \underbrace{M_{k+1,s}}_{\le \Phi_s M_k} + \gamma_{s+2} \lambda \underbrace{M_{k,s+2}}_{\le M_k}\right)$$

$$
\begin{aligned}
&\leq \frac{\lambda}{\lambda+\underline{\mu}_{s+1}}\left(\overline{\mu}_{s+1}\left(\frac{1}{\lambda+\underline{\mu}_s}\sum_{q=2}^{s}\phi_{s,q}+\frac{1}{\lambda+\underline{\mu}_s}\gamma_{s+1}\right)+\gamma_{s+2}\right)M_k\\
&=\frac{\lambda}{\lambda+\underline{\mu}_{s+1}}\left(\frac{\overline{\mu}_{s+1}}{\lambda+\underline{\mu}_s}\sum_{q=2}^{s}\gamma_q\prod_{j=q}^{s}\frac{\overline{\mu}_j}{\lambda+\underline{\mu}_{j-1}}+\frac{\overline{\mu}_{s+1}}{\lambda+\underline{\mu}_s}\gamma_{s+1}+\gamma_{s+2}\right)M_k\\
&=\frac{\lambda}{\lambda+\underline{\mu}_{s+1}}\left(\underbrace{\sum_{q=2}^{s}\gamma_q\prod_{j=q}^{s+1}\frac{\overline{\mu}_j}{\lambda+\underline{\mu}_{j-1}}+\frac{\overline{\mu}_{s+1}}{\lambda+\underline{\mu}_s}\gamma_{s+1}}_{=\sum_{q=2}^{s+1}\phi_{s+1,q}}+\gamma_{s+2}\right)M_k\\
&=\Phi_{s+1}M_k.
\end{aligned}
$$

Thus, we confirm that (EC.4) holds and is independent of k. Notice that $\Phi_N=\max_n\Phi_n<1$, where this inequality follows from Assumption 3.1. Therefore, $M_{k+1}\leq\Phi_N M_k$ for each k because

$$M_{k+1}=\sup_n M_{k+1,n}\leq\Phi_N M_k. \tag{EC.5}$$

Moreover, based on (EC.5),

$$M_{k+1}\leq\Phi_N M_k\leq\Phi_N^2 M_{k-1}\leq\Phi_N^k M_1. \tag{EC.6}$$

Given that $\Phi_N<1$ and is independent of k,

$$\lim_{k\to\infty}M_{k+1}\leq\lim_{k\to\infty}\Phi_N^k M_1=0.$$

Hence, for all n and k,

$$\left|p_n^{k+1}(B_l)-p_n^k(B_l)\right|\leq\sum_{l\in\mathscr{C}_n}\left|p_n^{k+1}(B_l)-p_n^k(B_l)\right|=M_{k+1,n}\leq\sup_n M_{k+1,n}=M_{k+1}.$$

In conclusion,

$$\lim_{k\to\infty}\left|p_n^{k+1}(B_l)-p_n^k(B_l)\right|\leq\lim_{k\to\infty}M_{k+1}\leq\lim_{k\to\infty}\Phi_N^k M_1=0.$$

Thus, for each n, $\left|p_n^{k+1}(B_l)-p_n^k(B_l)\right|$ converges to 0 as k approaches infinity at a geometric rate.

Next, we show that it converges to the original hypercube solution. Suppose the algorithm converges to $\bar{p}_n(B_m)$, with $\lim_{k\to\infty}p_n^k(B_m)=\bar{p}_n(B_m)\neq p_n(B_m)$. If the algorithm converges, there exists an $M>0$ and an infinitesimally small $\epsilon>0$ such that for all $m>M$,

$$|p_n^m(B_m)-\bar{p}_n(B_m)|\leq\epsilon,\quad\text{and}\quad|p_n^{m-1}(B_m)-\bar{p}_n(B_m)|\leq\epsilon,$$

for all $n=0,1,\ldots,N$ and all m. By proving this and noting that $p_n^m(B_m)$ and $p_n^{m-1}(B_m)$ follow Equation (EC.3), we can infer that $\bar{p}_n(B_m)$ also satisfies Equation (EC.3) when $\epsilon\to 0$. We then have both $\bar{p}_n(B_m)$ and $p_n(B_m)$ as solutions.

Given that all states in our model are positive recurrent and the continuous Markov process for the spatial queueing system is irreducible, there is a guarantee of a unique positive stationary distribution. Thus, the steady state distribution $P\{n, B_m\}$ corresponding to (4) and (5) is unique. We have $p_n(B_m) = P\{n, B_m\}/p(n)$, which is a one-to-one mapping from state joint probability $P\{n, B_m\}$ to conditional probability $p_n(B_m)$. Consequently, we have $\bar{p}_n(B_m) = p_n(B_m)$. □

EC.2. Extended Numerical Analysis

This section provides detailed times, including both coefficient generation and computation, for St. Paul, MN, and Greenville County, SC.

EC.2.1. Heterogeneous Case

In this section, we present the time for heterogeneous service rates for different cases, detailed in Table EC.1. In these instances, our CPU algorithm (i.e., Algorithm 1) is compared with a method that solves the problem using a sparse solver, given that both the original hypercube model and the modified model employing alternating hyperplane methods are infeasible for heterogeneous service rates.

For each scenario, we detail the computation times for both the sparse solver and our algorithm, with the latter emphasized in boldface. Note that NA indicates that results are omitted because the original hypercube solution becomes computationally prohibitive for systems with more than 20 units. From the tables, we note that the computation time for the sparse solver increases significantly as N grows. For instance, the sparse solver requires more than 9,000 seconds to tackle a 15-unit problem and becomes impractical for scenarios with over 15 units. Conversely, our method is able to solve considerably larger problems efficiently. Specifically, for the 15-unit case, our method is 2,000 times faster compared to the sparse solver.

EC.2.2. Homogeneous Case

This section evaluates the total time for cases with homogeneous service rates, using the CPU algorithm and the alternating hyperplane method by Larson (1974). We find that our coefficient generation operates almost

Table EC.1 Detailed Time (s) for the Heterogeneous Case.

	ρ	# Units (N)																			
		11		12		13		14		15		16		17		18		19		20	
St. Paul	0.1	2.5	**1.3**	23.3	**2.0**	198.5	**3.2**	1120.1	**6.1**	9042.9	**9.7**	NA	**35.2**	NA	**108.8**	NA	**265.1**	NA	**606.7**	NA	**1764.4**
	0.2	2.5	**0.7**	22.9	**0.9**	194.8	**1.6**	1086.0	**3.2**	9226.3	**3.5**	NA	**10.6**	NA	**53.0**	NA	**123.9**	NA	**291.9**	NA	**645.2**
	0.3	2.4	**0.7**	22.9	**1.0**	194.4	**1.9**	1097.1	**2.6**	9012.0	**4.1**	NA	**13.0**	NA	**63.8**	NA	**148.9**	NA	**342.1**	NA	**715.9**
	0.4	2.4	**0.7**	23.0	**1.0**	193.4	**2.0**	1088.6	**2.5**	9227.1	**4.3**	NA	**14.0**	NA	**68.2**	NA	**155.5**	NA	**356.5**	NA	**718.0**
	0.5	2.5	**0.8**	23.0	**1.0**	193.9	**1.9**	1075.5	**2.5**	9310.8	**4.2**	NA	**13.8**	NA	**69.7**	NA	**156.6**	NA	**360.4**	NA	**702.6**
	0.6	2.5	**0.8**	22.9	**1.0**	195.0	**1.8**	1086.7	**2.4**	9276.2	**4.1**	NA	**13.4**	NA	**65.8**	NA	**150.1**	NA	**344.5**	NA	**674.6**
	0.7	2.5	**0.8**	22.8	**0.9**	194.1	**1.7**	1113.1	**2.2**	9185.2	**3.9**	NA	**12.8**	NA	**63.7**	NA	**139.6**	NA	**327.2**	NA	**645.6**
	0.8	2.5	**0.8**	22.8	**0.9**	194.1	**1.6**	1087.1	**2.1**	9226.3	**3.6**	NA	**11.9**	NA	**59.4**	NA	**133.1**	NA	**310.2**	NA	**601.8**
	0.9	2.5	**0.8**	22.8	**0.9**	193.6	**1.4**	1097.0	**2.0**	9246.6	**3.4**	NA	**11.0**	NA	**53.9**	NA	**123.5**	NA	**283.4**	NA	**575.3**
Greenville	0.1	2.5	**1.5**	24.5	**1.7**	177.3	**3.2**	1006.2	**4.4**	8713.4	**8.7**	NA	**25.4**	NA	**76.4**	NA	**188.7**	NA	**575.8**	NA	**1152.4**
	0.2	2.5	**0.7**	23.5	**0.8**	172.4	**1.1**	1035.4	**1.7**	9165.5	**5.3**	NA	**15.4**	NA	**40.4**	NA	**100.9**	NA	**253.0**	NA	**475.4**
	0.3	2.4	**0.8**	23.6	**1.0**	172.3	**1.4**	1024.3	**2.0**	8581.8	**6.4**	NA	**18.8**	NA	**48.1**	NA	**121.5**	NA	**301.8**	NA	**622.0**
	0.4	2.4	**0.9**	23.9	**1.1**	174.3	**1.5**	1192.7	**2.1**	8883.9	**6.9**	NA	**20.4**	NA	**52.6**	NA	**127.7**	NA	**338.0**	NA	**674.6**
	0.5	2.4	**0.9**	24.2	**1.2**	175.6	**1.6**	1019.8	**2.2**	8813.1	**7.1**	NA	**21.2**	NA	**54.2**	NA	**130.8**	NA	**345.4**	NA	**690.6**
	0.6	2.5	**0.9**	23.5	**1.1**	175.9	**1.6**	1022.8	**2.2**	9184.3	**6.9**	NA	**21.1**	NA	**54.3**	NA	**127.7**	NA	**327.9**	NA	**676.0**
	0.7	2.4	**0.9**	23.5	**1.1**	175.7	**1.5**	1021.3	**2.1**	8521.4	**6.9**	NA	**20.8**	NA	**52.7**	NA	**124.7**	NA	**308.4**	NA	**632.3**
	0.8	2.4	**0.8**	24.7	**1.1**	172.3	**1.4**	1132.7	**2.0**	8577.4	**6.7**	NA	**20.0**	NA	**51.1**	NA	**121.4**	NA	**293.0**	NA	**603.9**
	0.9	2.4	**0.8**	23.6	**1.0**	181.7	**1.4**	1341.3	**1.9**	8619.7	**6.2**	NA	**19.5**	NA	**49.6**	NA	**116.2**	NA	**278.0**	NA	**560.1**

Note. Regular: computation time of the sparse solver. Bold: computation time of CPU algorithm. NA: results not included.

twice as fast as the original TOUR method, and the computation time using our algorithm is more than 97% faster than the computation time using the alternating hyperplane method. To make a fairer comparison, we improved the alternating hyperplane method by using current computer methodologies. We implemented sparse matrix manipulation while adhering to Larson's method in the other portions of his procedure. We then compared our algorithm with the enhanced version of Larson's method. Our method reduces the total time by about 50% compared to the enhanced Larson method.

In the Tables EC.2 and EC.3, CPU represents our method, and AH denotes the original alternating hyperplane solution. The original version used point updating with for-loops. We also modified the original code to enhance its speed, labeling it as MAH, which stands for the modified solution of the original hypercube method. In this modified solution, we first convert the storage vector into a sparse matrix and then replace the for-loops with sparse matrix multiplications. For each case, we report the times for generating the transition rates (highlighted in boldface) and for computation. The original alternating hyperplane solution requires a significant amount of time for systems larger than 20 units, taking over 20 minutes for the 21-unit case and more than four hours for the 25-unit case. Therefore, we did not include these results in the table.

References

Larson RC (1974) A hypercube queuing model for facility location and redistricting in urban emergency services. *Computers & Operations Research* 1(1):67–95.

Table EC.2 Detailed Time (s) for the Homogeneous Case, St. Paul.

	ρ	# Units (N) 11		12		13		14		15		16		17		18		19		20		21		22		23		24		25	
CPU	0.1	**0.2**	0.0	**0.4**	0.1	**0.9**	0.1	**1.8**	0.2	**3.6**	0.3	**7.4**	0.5	**15.1**	0.9	**30.4**	2.2	**62.1**	4.6	**126.6**	11.8	**257.0**	29.5	**522.2**	63.7	**1064.0**	143.4	**2111.5**	324.3	**4374.1**	692.6
	0.2	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.8**	0.2	**3.7**	0.4	**7.4**	0.7	**15.1**	1.4	**31.0**	3.1	**62.5**	6.7	**126.3**	16.2	**254.3**	40.8	**519.0**	87.4	**1060.8**	198.9	**2155.2**	456.2	**4326.9**	997.2
	0.3	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.8**	0.3	**3.6**	0.4	**7.4**	0.8	**15.3**	1.7	**30.4**	3.6	**62.1**	7.9	**126.3**	18.1	**255.6**	45.1	**520.4**	98.2	**1052.7**	228.9	**2117.4**	525.4	**4383.1**	1166.9
	0.4	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.7**	0.3	**3.6**	0.5	**7.4**	0.9	**15.3**	1.8	**30.8**	3.9	**62.1**	8.3	**126.4**	18.4	**255.9**	45.1	**519.7**	98.3	**1060.7**	233.1	**2143.8**	539.3	**4366.4**	1215.6
	0.5	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.7**	0.3	**3.6**	0.5	**7.5**	0.9	**15.2**	1.9	**30.5**	4.0	**62.1**	8.4	**126.4**	18.1	**255.4**	44.2	**525.1**	98.4	**1061.1**	226.1	**2130.7**	523.8	**4354.4**	1187.6
	0.6	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.7**	0.3	**3.6**	0.4	**7.4**	0.9	**15.1**	1.8	**30.6**	3.8	**61.9**	8.1	**126.7**	17.3	**255.7**	42.5	**521.9**	94.7	**1079.2**	214.9	**2133.5**	499.4	**4370.3**	1113.7
	0.7	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.8**	0.2	**3.6**	0.4	**7.3**	0.8	**15.2**	1.7	**30.4**	3.6	**62.9**	7.7	**126.4**	16.6	**254.9**	40.8	**513.9**	88.1	**1065.7**	203.0	**2132.3**	465.5	**4260.4**	1040.3
	0.8	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.7**	0.2	**3.6**	0.4	**7.4**	0.7	**15.3**	1.6	**31.0**	3.3	**62.0**	7.2	**125.5**	15.4	**257.3**	38.2	**503.4**	83.4	**1065.8**	191.5	**2086.9**	431.5	**4386.2**	941.7
	0.9	**0.2**	0.1	**0.4**	0.1	**0.9**	0.2	**1.7**	0.2	**3.7**	0.4	**7.4**	0.7	**15.1**	1.5	**30.6**	3.2	**62.0**	6.8	**126.4**	14.7	**256.8**	35.3	**521.8**	78.3	**1067.9**	179.4	**2112.3**	406.9	**4251.2**	905.5
AH	0.1	**0.5**	0.2	**1.0**	0.7	**2.1**	1.7	**4.2**	3.9	**8.4**	8.6	**16.6**	20.0	**32.5**	44.0	**65.7**	93.6	**133.6**	215.2	**267.6**	510.9	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.2	**0.5**	0.3	**1.0**	0.8	**2.1**	2.3	**4.2**	5.1	**8.2**	11.3	**16.6**	26.9	**32.6**	59.4	**65.8**	123.4	**133.7**	281.7	**269.2**	666.1	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.3	**0.5**	0.3	**1.0**	0.9	**2.1**	2.5	**4.2**	5.6	**8.3**	13.0	**16.2**	30.7	**32.5**	69.1	**65.9**	143.6	**133.1**	319.1	**269.2**	726.3	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.4	**0.5**	0.3	**1.0**	0.9	**2.1**	2.6	**4.2**	5.6	**8.4**	12.9	**16.3**	32.4	**32.6**	72.8	**65.8**	148.1	**134.6**	337.6	**268.5**	727.0	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.5	**0.5**	0.4	**1.0**	0.9	**2.1**	2.6	**4.2**	5.4	**8.4**	12.8	**16.3**	31.7	**32.6**	70.8	**65.7**	148.1	**132.8**	327.0	**269.4**	712.1	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.6	**0.5**	0.3	**1.0**	0.8	**2.1**	2.5	**4.3**	5.3	**8.4**	12.2	**16.4**	29.4	**32.5**	66.4	**65.9**	137.2	**133.3**	316.1	**268.1**	663.7	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.7	**0.5**	0.3	**1.0**	0.8	**2.1**	2.3	**4.2**	5.0	**8.4**	11.4	**16.3**	27.9	**32.1**	62.3	**66.0**	130.6	**133.7**	288.9	**269.2**	609.4	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.8	**0.5**	0.3	**1.0**	0.7	**2.1**	2.1	**4.2**	4.6	**8.5**	10.4	**16.5**	25.4	**32.5**	56.9	**66.1**	121.4	**133.7**	269.6	**268.8**	572.6	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.9	**0.5**	0.3	**1.0**	0.7	**2.1**	2.0	**4.3**	4.2	**8.4**	9.7	**16.5**	23.1	**32.4**	52.0	**65.9**	110.6	**132.5**	245.5	**263.7**	536.2	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
MAH	0.1	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.3**	0.1	**8.4**	0.3	**16.6**	0.7	**32.4**	1.4	**66.9**	2.9	**133.8**	6.3	**262.3**	14.0	**535.5**	30.2	**1075.0**	60.2	**2107.0**	146.5	**4271.5**	300.8	**8456.6**	725.9
	0.2	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.2**	0.2	**8.4**	0.3	**16.5**	0.7	**32.6**	1.5	**66.6**	3.6	**133.6**	7.5	**263.3**	17.0	**533.0**	32.7	**1070.6**	74.7	**2105.0**	177.2	**4246.7**	373.6	**8554.4**	857.0
	0.3	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.2**	0.2	**8.3**	0.3	**16.5**	0.7	**32.2**	1.6	**66.5**	4.0	**133.7**	8.1	**262.0**	18.0	**531.8**	36.5	**1072.3**	83.1	**2111.5**	188.4	**4233.8**	415.8	**8460.6**	931.1
	0.4	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.2**	0.2	**8.4**	0.3	**16.5**	0.8	**32.5**	1.7	**66.7**	4.2	**133.8**	8.4	**264.2**	18.1	**534.0**	37.5	**1070.5**	85.3	**2116.7**	189.6	**4245.9**	426.4	**8477.9**	967.6
	0.5	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.2**	0.2	**8.4**	0.3	**16.5**	0.8	**32.7**	1.7	**66.3**	4.1	**134.7**	8.3	**264.9**	17.9	**530.6**	37.4	**1080.4**	84.6	**2121.2**	187.1	**4242.7**	423.2	**8451.6**	957.9
	0.6	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.2**	0.2	**8.3**	0.3	**16.4**	0.7	**32.9**	1.6	**66.4**	3.9	**134.3**	8.1	**263.1**	16.9	**527.1**	36.1	**1077.2**	79.8	**2101.1**	181.1	**4200.3**	403.1	**8520.9**	903.6
	0.7	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.3**	0.2	**8.3**	0.3	**16.4**	0.7	**32.8**	1.5	**67.1**	3.8	**134.0**	7.6	**262.9**	16.1	**526.1**	33.9	**1075.1**	76.3	**2118.9**	172.6	**4216.5**	389.7	**8497.2**	863.9
	0.8	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.3**	0.1	**8.3**	0.3	**16.4**	0.7	**32.8**	1.4	**66.4**	3.6	**134.4**	7.2	**263.7**	15.3	**526.7**	32.4	**1077.9**	72.3	**2092.2**	161.6	**4196.4**	367.1	**8490.5**	819.0
	0.9	**0.5**	0.0	**1.0**	0.0	**2.1**	0.1	**4.2**	0.1	**8.4**	0.3	**16.4**	0.6	**32.8**	1.4	**66.5**	3.4	**134.7**	7.0	**264.3**	14.7	**527.1**	31.2	**1080.7**	69.6	**2112.5**	153.1	**4230.6**	344.3	**8481.0**	788.0

Note. Bold: coefficient generation time. Regular: computation time. NA: results not included. CPU: our algorithm. AH: alternating hyperplane algorithm. MAH: modified alternating hyperplane algorithm.

Table EC.3 Detailed Time (s) for the Homogeneous Case, Greenville County.

	ρ	# Units (N) 11		12		13		14		15		16		17		18		19		20		21		22		23		24		25	
	0.1	**0.3**	0.1	**0.6**	0.1	**1.2**	0.1	**2.4**	0.1	**4.8**	0.2	**9.7**	0.3	**20.1**	0.7	**40.5**	1.5	**82.7**	3.6	**166.4**	8.9	**339.4**	19.9	**687.5**	43.7	**1405.0**	108.7	**2859.1**	246.1	**5690.8**	526.1
	0.2	**0.3**	0.1	**0.6**	0.1	**1.2**	0.1	**2.4**	0.2	**4.9**	0.3	**9.9**	0.5	**20.0**	1.0	**40.5**	2.2	**82.4**	5.4	**166.8**	13.4	**336.4**	28.4	**689.3**	63.5	**1382.9**	156.3	**2802.2**	370.9	**5687.1**	792.2
	0.3	**0.3**	0.1	**0.6**	0.1	**1.2**	0.1	**2.4**	0.2	**4.8**	0.3	**9.9**	0.6	**20.1**	1.2	**40.7**	2.7	**82.0**	6.6	**166.3**	16.2	**335.1**	35.3	**688.0**	78.0	**1406.6**	183.0	**2823.4**	463.5	**5725.9**	982.8
	0.4	**0.3**	0.1	**0.6**	0.1	**1.2**	0.2	**2.4**	0.2	**4.9**	0.4	**9.9**	0.6	**20.1**	1.3	**40.8**	3.0	**82.5**	7.2	**164.8**	18.3	**336.4**	39.5	**683.2**	85.3	**1400.9**	202.8	**2789.1**	522.5	**5723.8**	1105.4
CPU	0.5	**0.3**	0.1	**0.6**	0.1	**1.2**	0.2	**2.4**	0.2	**4.9**	0.4	**9.9**	0.7	**20.1**	1.4	**40.6**	3.1	**82.6**	7.6	**166.7**	19.1	**336.3**	41.2	**686.3**	90.7	**1397.4**	206.8	**2825.7**	541.9	**5733.1**	1165.2
	0.6	**0.3**	0.1	**0.6**	0.1	**1.2**	0.2	**2.4**	0.2	**4.9**	0.4	**9.9**	0.7	**20.0**	1.4	**40.5**	3.1	**82.2**	7.4	**166.2**	18.7	**338.0**	41.2	**683.4**	89.0	**1412.1**	211.2	**2812.2**	533.9	**5673.4**	1149.3
	0.7	**0.3**	0.1	**0.6**	0.1	**1.2**	0.2	**2.4**	0.2	**4.8**	0.4	**10.0**	0.6	**20.0**	1.3	**40.4**	3.0	**82.1**	7.1	**166.6**	17.9	**334.7**	39.5	**686.8**	87.2	**1407.4**	202.8	**2840.0**	499.8	**5650.6**	1080.4
	0.8	**0.3**	0.1	**0.6**	0.1	**1.2**	0.1	**2.4**	0.2	**4.9**	0.4	**9.9**	0.6	**20.1**	1.3	**40.3**	2.9	**82.0**	6.9	**166.5**	17.1	**336.9**	36.9	**692.2**	81.7	**1408.4**	191.1	**2784.5**	463.6	**5712.9**	1009.8
	0.9	**0.3**	0.1	**0.6**	0.1	**1.2**	0.1	**2.4**	0.2	**4.8**	0.3	**9.8**	0.6	**20.0**	1.2	**40.7**	2.8	**82.6**	6.6	**166.8**	16.3	**335.5**	35.2	**679.1**	78.1	**1403.4**	180.5	**2814.8**	430.5	**5831.0**	937.5
	0.1	**0.6**	0.3	**1.3**	0.7	**2.5**	1.6	**5.3**	3.8	**10.7**	8.5	**21.5**	18.9	**42.9**	40.8	**86.5**	90.0	**171.1**	217.2	**354.5**	483.2	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.2	**0.6**	0.4	**1.3**	0.8	**2.6**	2.1	**5.4**	4.9	**10.3**	10.9	**21.4**	23.0	**43.3**	52.7	**86.6**	119.1	**173.0**	291.8	**353.4**	658.1	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.3	**0.7**	0.5	**1.3**	1.0	**2.5**	2.4	**5.1**	5.1	**10.5**	11.6	**21.2**	25.8	**42.8**	60.9	**88.6**	138.0	**172.6**	337.7	**356.0**	748.1	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.4	**0.6**	0.5	**1.3**	1.0	**2.6**	2.4	**5.5**	5.9	**10.6**	13.2	**21.7**	28.5	**43.1**	65.3	**87.6**	146.8	**172.8**	357.3	**350.8**	788.3	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
AH	0.5	**0.6**	0.5	**1.3**	1.0	**2.6**	2.6	**5.2**	5.6	**10.2**	12.5	**21.5**	28.3	**43.3**	65.5	**86.9**	141.0	**172.7**	361.8	**350.3**	814.0	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.6	**0.6**	0.5	**1.2**	1.0	**2.6**	2.5	**5.3**	5.4	**10.5**	11.6	**21.0**	26.9	**43.8**	62.9	**85.5**	113.2	**178.7**	356.8	**350.4**	767.2	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.7	**0.6**	0.4	**1.3**	0.9	**2.6**	2.0	**5.2**	5.2	**10.8**	12.0	**21.5**	25.7	**42.9**	59.1	**75.5**	114.9	**175.3**	332.2	**351.6**	730.6	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.8	**0.7**	0.4	**1.2**	0.8	**2.5**	2.2	**5.4**	5.0	**10.1**	10.8	**21.6**	25.5	**43.2**	55.5	**83.1**	121.7	**172.5**	296.1	**355.7**	686.8	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.9	**0.6**	0.4	**1.1**	0.9	**2.6**	2.1	**4.9**	4.7	**10.6**	10.9	**21.0**	23.7	**42.7**	53.7	**85.4**	114.1	**172.9**	278.3	**350.9**	570.4	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	0.1	**0.6**	0.0	**1.3**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**20.6**	0.6	**41.7**	1.2	**84.1**	2.6	**168.4**	5.8	**341.7**	12.2	**684.7**	27.3	**1409.3**	58.4	**2648.4**	126.6	**5243.0**	317.3	**12649.3**	778.0
	0.2	**0.6**	0.0	**1.2**	0.0	**2.5**	0.1	**5.1**	0.1	**10.3**	0.3	**20.7**	0.6	**41.8**	1.2	**84.5**	3.1	**168.1**	6.7	**341.0**	14.8	**688.3**	32.5	**1411.9**	71.2	**2651.7**	153.9	**5264.2**	386.7	**12561.3**	903.0
	0.3	**0.6**	0.0	**1.2**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**20.5**	0.6	**41.8**	1.4	**84.3**	3.4	**168.1**	7.4	**340.8**	16.3	**686.4**	35.5	**1418.6**	77.7	**2649.1**	169.2	**5225.6**	421.6	**12571.4**	993.3
	0.4	**0.6**	0.0	**1.3**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**21.0**	0.6	**41.7**	1.4	**84.1**	3.5	**167.9**	7.8	**340.8**	17.1	**685.1**	36.7	**1418.5**	80.8	**2656.6**	178.2	**5339.2**	438.6	**12494.7**	1047.8
MAH	0.5	**0.6**	0.0	**1.3**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**20.7**	0.6	**41.6**	1.4	**84.3**	3.5	**168.2**	7.8	**340.9**	17.4	**686.8**	36.7	**1417.5**	80.8	**2636.6**	181.1	**5260.8**	438.5	**12527.6**	1047.6
	0.6	**0.6**	0.0	**1.3**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**20.5**	0.6	**42.0**	1.4	**84.5**	3.4	**168.0**	7.6	**342.5**	16.9	**687.4**	35.5	**1415.7**	78.5	**2643.4**	177.9	**5272.8**	423.4	**12547.9**	1017.7
	0.7	**0.6**	0.0	**1.3**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**20.6**	0.6	**41.7**	1.3	**84.5**	3.3	**168.3**	7.3	**342.9**	16.3	**686.3**	34.2	**1418.4**	75.7	**2647.9**	170.4	**5262.0**	408.2	**12553.3**	982.0
	0.8	**0.6**	0.0	**1.3**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**20.7**	0.6	**41.8**	1.3	**85.4**	3.2	**167.9**	6.9	**342.6**	15.4	**686.3**	32.3	**1414.9**	71.7	**2647.9**	163.8	**5258.7**	385.0	**12681.9**	929.6
	0.9	**0.6**	0.0	**1.3**	0.0	**2.5**	0.1	**5.1**	0.1	**10.2**	0.3	**20.7**	0.6	**41.5**	1.2	**84.3**	3.1	**168.5**	6.6	**343.5**	14.6	**685.3**	31.0	**1419.6**	67.6	**2643.7**	157.1	**5258.4**	369.5	**12516.2**	877.3

Note. Bold: coefficient generation time. Regular: computation time. NA: results not included. CPU: our algorithm. AH: alternating hyperplane algorithm. MAH: modified alternating hyperplane algorithm.