IISE Transactions

Taylor & Francis
Taylor & Francis Group

IISE Transactions

ISSN: 2472-5854 (Print) 2472-5862 (Online) Journal homepage: https://www.tandfonline.com/loi/uiie21

State space modeling of autocorrelated multivariate Poisson counts

Chen Zhang, Nan Chen & Zhiguo Li

To cite this article: Chen Zhang, Nan Chen & Zhiguo Li (2017) State space modeling of autocorrelated multivariate Poisson counts, IISE Transactions, 49:5, 518-531, DOI: 10.1080/24725854.2016.1251665

To link to this article: https://doi.org/10.1080/24725854.2016.1251665

IISE TRANSACTIONS
2017, VOL. 49, NO. 5, 518–531
http://dx.doi.org/10.1080/24725854.2016.1251665

Taylor & Francis
Taylor & Francis Group

State space modeling of autocorrelated multivariate Poisson counts

Chen Zhang[a], Nan Chen[a] and Zhiguo Li[b]

[a]Department of Industrial and Systems Engineering, National University of Singapore, Singapore; [b]IBM T.J. Watson Research Center, Rochester, NY, USA

ABSTRACT

Although many applications involve autocorrelated multivariate counts, there is a scarcity of research on their statistical modeling. To fill this research gap, this article proposes a state space model to describe autocorrelated multivariate counts. The model builds upon the multivariate log-normal mixture Poisson distribution and allows for serial correlations by considering the Poisson mean vector as a latent process driven by a nonlinear autoregressive model. In this way, the model allows for flexible cross-correlation and autocorrelation structures of count data and can also capture overdispersion. The Monte Carlo Expectation Maximization algorithm, together with particle filtering and smoothing methods, provides satisfactory estimators for the model parameters and the latent process variables. Numerical studies show that, compared with other state-of-the-art models, the proposed model has superiority and more generality with respect to describing count data generated from different mechanisms of the process of counts. Finally, we use this model to analyze counts of different types of damage collected from a power utility system as a case study. Supplementary materials are available for this article. Go to the publisher's online edition of *IISE Transactions* for additional tables and figures.

ARTICLE HISTORY
Received 27 October 2015
Accepted 20 September 2016

KEYWORDS
Multivariate time series; multivariate Poisson distribution; overdispersion; state space model; particle filtering and smoothing; Monte Carlo expectation maximization

1. Introduction

Count data arise in many areas, including ecology, epidemiology, economics, manufacturing, etc., where usually multiple counts are observed together. As they usually exhibit certain correlations with each other—i.e., cross-correlations—we call them multivariate counts and want to analyze them together. For example, in ecology, counts of different species interact due to the common environmental features (Billheimer *et al.*, 2001). In epidemiology, counts of patients with related diseases may be correlated with each other (Paul *et al.*, 2008). In marketing, sales volumes of different products or brands may influence each other (Chen *et al.*, 2015). In quality control, counts of different types of defects may be caused or influenced by some common factors.

In addition to cross-correlation, in many applications multivariate counts evolve over time and have serial correlations with their previous observations; i.e., autocorrelations. For example, the count of persons with an infectious disease in this month is influenced by that in the previous month. The weekly sales volume of a product often fluctuates with seasonal or economic variations. The number of defects in neighbor samples in manufacturing may be driven by certain common inertial elements when the sampling interval is small. Usually, the serial dependence can be either positive or negative.

On top of these correlation features, another common feature of autocorrelated multivariate counts is overdispersion, which means, with respect to a model, that the variance of the count data is greater than the expectation. Overdispersion is often an affiliation property of cross-correlations and autocorrelations and is intensively investigated in Poisson regression models and time series models. Usually, overdispersion is caused by some unobserved heterogeneities across the count data. Suppose that the multivariate counts depend on an unobserved or omitted covariate z_t. Then the change of z_t over time will introduce additional variance into the count data. A detailed discussion can be found in Cox and Isham (1980).

Although autocorrelated multivariate counts are quite common in our daily lives, to the best of our knowledge, a general model to describe them is yet to be established. A reasonable model should be able to not only describe cross-correlation and autocorrelation structures of count data flexibly, but also accommodate overdispersion. Although statistical models of univariate time series of count data are thoroughly explored in previous studies (see Davis *et al.* (1999) and the references therein for more background knowledge), the extensions to multivariate cases are underdeveloped. A brief literature review of these extensions is discussed below, with their model properties summarized in Table 1.

1.1. Multivariate Poisson distribution

As we know, the Poisson distribution or its variants is often used to model count data. To date, there are three main types of methods to construct multivariate Poisson distributions for multivariate counts $\mathbf{Y} = [Y_1, \ldots, Y_d]$ with dimension d. We will discuss them in detail below. The first approach directly extends the bivariate Poisson distribution (Holgate, 1964) to d-dimensions by retaining the expression of each dimension as a sum of two independent variables; that is,

CONTACT Nan Chen isecn@nus.edu.sg
Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/uiie.
Supplemental data for this article can be accessed on the publisher's website.

Table 1. Summary of some state-of-the-art Poisson models of multivariate counts. "+": allow for positive correlations; "−": allow for negative correlations; ×: cannot describe the feature; √: can describe the feature.

Literature	*Method*	*Cross-correlation*	*Auto-correlation*	*Over-dispersion*
Karlis (2003)	Sum of Poisson	+	×	×
Karlis and Meligkotsidou (2005)	Sum of Poisson	+	×	×
Song (2000)	Normal copulas	+,−	×	×
Nikoloulopoulos and Karlis (2009)	Discrete copulas	+,−	×	×
Karlis and Meligkotsidou (2007)	Finite mixture	+,−	×	×
Arbous and Kerrich (1951)	Poisson-gamma mixture	+	×	√
Steyn (1976)	Poisson-normal mixture	+,−	×	√
Aitchison and Ho (1989)	Poisson-lognormal mixture	+,−	×	√
Sarabia and Gómez-Déniz (2011)	Poisson-beta mixture	+,−	×	√
Heinen and Rengifo (2007)	Multivariate INGARCH, copulas	+,−	+	√
Latour (1997)	Multivariate GINAR	+	+	√
Pedeli and Karlis (2013b)	Multivariate GINAR	+	+	√

$$Y_i = Z_i + Z_0, \quad i = 1, \ldots, d, \tag{1}$$

where Z_i and Z_0 follow Poisson distributions with rates λ_i and λ_0, respectively. Given the parameters $\mathbf{\Lambda} = \{\lambda_1, \ldots, \lambda_d, \lambda_0\}$, the joint density of the multivariate Poisson distribution is defined as $p(\mathbf{Y}|\mathbf{\Lambda}) = \prod_{i=1}^{d} p(Y_i|\lambda_i, \lambda_0)$, whose marginal distribution of every dimension is the Poisson distribution. Clearly, it is Z_0 that introduces the same cross-correlation between different dimensions. Later extensions to allow different cross-correlations are also discussed in Kocherlakota and Kocherlakota (1992). The second approach is to construct the multivariate Poisson distribution using copula (Song, 2000). A copula is a general way to introduce dependence between variables when their marginal distributions are given. Its idea is that a d-dimensional distribution can be written in terms of d marginal distributions and a copula that describes the dependence structure of these dimensions. However, although copula modeling provides useful tools for analyzing the cross-correlations between multiple variables and has been extensively used for continuous distributed data, its primary difficulty in the discrete case is the lack of uniqueness of Sklar's representation and the unidentifiability of the copula. This difficulty indicates that many of the convenient properties of a copula cannot carry over from the continuous case to the discrete case. Therefore, modeling and interpreting dependence for count data through copulas is still underdeveloped and subjects to caution (see Genest and Nešlehová (2007) for a comprehensive discussion). Furthermore, unfortunately, most models in the above two categories can only support limited positive cross-correlations of multivariate counts. Furthermore, they have limited flexibilities in accounting for overdispersion.

A more flexible method is based on the mixture model by placing a distribution on the mean vector $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_d]$ of d-dimensional Poisson counts. In this way, the method is able to allow cross-correlations of either sign, as well as overdispersion in a large range. This method can be further divided into two groups. The first considers a finite mixture, implying that the mean vector is chosen from finite K components (Karlis and Meligkotsidou, 2007) with the corresponding probabilities π_k, $k = 1, \ldots, K$. Every component is a d-dimensional Poisson distribution defined in Equation (1) with the parameters $\mathbf{\Lambda}_k = \{\lambda_{k1}, \ldots, \lambda_{kd}, \lambda_{k0}\}$. Then the joint probability mass function (pmf) of $\mathbf{Y}$ is given by

$$p(\mathbf{Y}|\mathbf{\Lambda}_{1:K}, \pi_{1:K}) = \sum_{k=1}^{K} \pi_k p(\mathbf{Y}|\mathbf{\Lambda}_k). \tag{2}$$

However, the assessment of the unknown K often requires a considerable amount of work. The second method imposes a continuous distribution $g(\boldsymbol{\lambda}|\Theta)$ on the mean vector $\boldsymbol{\lambda}$. Then the unconditional multivariate Poisson distribution is a marginalization, integrating out the mean vector distribution as

$$p(\mathbf{Y}|\Theta) = \int_{\mathbf{R}_+^d} \prod_{i=1}^{d} p(Y_i|\lambda_i) g(\boldsymbol{\lambda}|\Theta) \mathrm{d}\boldsymbol{\lambda}, \tag{3}$$

where $p(Y_i|\lambda_i)$ is the Poisson pmf with rate λ_i for $i = 1, \ldots, d$. This idea is generally adopted by many models with different forms of $g(\boldsymbol{\lambda}|\Theta)$, such as the gamma distribution (Arbous and Kerrich, 1951; Nelson, 1985), the normal distribution (Steyn, 1976), the log-normal distribution (Aitchison and Ho, 1989), and the beta distribution (Sarabia and Gómez-Déniz, 2011). Among them, the one using the log-normal mixture is the most powerful one. On one hand, it brings the rich vein of cross-correlation structures of the multivariate normal distribution into the multivariate Poisson distribution; on the other, it ensures that the Poisson mean vector is always positive, based on the logarithmic transformation.

1.2. Multivariate time series of counts

Although many multivariate Poisson distributions have been proposed in literature as discussed above, extensions accounting for the autocorrelation are still in their infancy. One notable study is the Multivariate Autoregressive Conditional Poisson (MACP) Model (Heinen and Rengifo, 2007). This model extends the univariate INteGer-valued AutoRegressive Conditional Heteroskedasticity time series model (INGARCH; Ferland *et al.*, 2006) to multivariate cases. Specifically, for a d-dimensional count $\mathbf{Y}_t = [Y_{t1}, \ldots, Y_{td}]$ at time t with the mean vector $\boldsymbol{\lambda}_t = [\lambda_{t1}, \ldots, \lambda_{td}]$, MACP assumes that $\boldsymbol{\lambda}_t$ follows a vector autoregressive moving average type model with order p and q as

$$\boldsymbol{\lambda}_t = \boldsymbol{\lambda}_0 + \sum_{j=1}^{p} \mathbf{A}_j \mathbf{Y}_{t-j} + \sum_{j=1}^{q} \mathbf{B}_j \boldsymbol{\lambda}_{t-j}. \tag{4}$$

Then the joint pmf of $\mathbf{Y}_t$ given $\boldsymbol{\lambda}_t$ is $p(\mathbf{Y}_t|\boldsymbol{\lambda}_t) = \prod_{i=1}^{d} p(Y_{ti}|\lambda_{ti})$ where $p(Y_{ti}|\lambda_{ti})$ is the Poisson pmf with rate λ_{ti}. To further allow for overdispersion, MACP suggests replacing the Poisson distribution $p(Y_{ti}|\lambda_{ti})$ with the double-Poisson distribution $p(Y_{ti}|\lambda_{ti}, \phi)$ where ϕ is the common overdispersion parameter. In addition, to conquer the limitation that INGARCH can

only support positive cross-correlation structures, MACP further imposes a multivariate normal copula on $\mathbf{Y}_t$ to allow for negative ones. Then the joint cumulative distribution function (cdf) of $\mathbf{Y}_t$ is defined as a copula function $\mathcal{C}$ of the cdfs of d marginal double Poisson distributions $F_i(Y_i)$, $i = 1, \ldots, d$; that is,

$$F(\mathbf{Y}_t) = \mathcal{C}(F_1(Y_1), \ldots, F_d(Y_d)). \tag{5}$$

However, MACP still suffers from the inherent limitation of INGARCH, that it can only describe *positive* autocorrelated count data (Jung *et al.*, 2006). In particular, as defined in Heinen and Rengifo (2007), MACP(1, 1) is stationary only if the eigenvalues of $\mathbf{I} - \mathbf{A}_1 - \mathbf{B}_1$ lie within the unit circle. Then according to Proposition 2.1 and Proposition 2.2 of Heinen and Rengifo (2007), the autocovariance matrix of MACP(1, 1) can only take positive values. Furthermore, as mentioned earlier, since the copula for discrete data is no longer identifiable, further theoretical properties and assumptions of MACP are still to be fully investigated. In particular, inference (and particularly rank-based inference) for the copula parameters is fraught with difficulties. Another type of notable work is the extension of univariate Integer-Value AutoRegressive (INAR) models to multivariate (MINAR) cases by generalizing the *binomial thinning* operator in the INAR models to a *thinning matrix* (Pedeli and Karlis, 2013a, 2013b); that is,

$$\mathbf{Y}_t = \mathbf{A} \circ \mathbf{Y}_{t-1} + \mathbf{R}_t. \tag{6}$$

The $d \times d$ matrix $\mathbf{A} = \{a_{ij}, i, j = 1, \ldots, d\}$ acts as the usual matrix multiplication but keeps the properties of the binomial thinning operator (Weiß, 2008). Specifically, the operators $a_{ij}\circ$ are mutually independent. Each operator is defined as $a_{ij} \circ Y_j = \sum_{k=1}^{Y_j} X_k$, where $\{X_k\}_{k=1}^{Y_j}$ is a sequence of independent and identically distributed Bernoulli random variables such that $p(X_k = 1) = a_{ij} = 1 - p(X_k = 0)$ and $a_{ij} \in [0, 1]$. Currently most work in this field focuses on first-order MINAR models, denoted as MINAR(1), for bivariate counts with $\mathbf{R}_t$ defined as a bivariate Poisson distribution in Equation (1). This is because, as analyzed earlier, defining $\mathbf{R}_t$ for higher dimensions with flexible cross-correlation structures is not easy. Currently, the only MINAR(1) model considering more than two dimensions is Pedeli and Karlis (2013a), which considers $\mathbf{R}_t$ with flexible cross-correlations between different dimensions. However, this model assumes $\mathbf{A}$ only has diagonal components. Another limitation of the MINAR models is that they can only support positive cross-correlations and autocorrelations of count data and do not allow for large overdispersion (Pedeli and Karlis, 2013b). This is because that the autocovariance matrix of MINAR models can be written as

$$\boldsymbol{\gamma}(h) = \mathbf{A}\boldsymbol{\gamma}(h-1) = \mathbf{A}^h\boldsymbol{\gamma}(0), \quad h \geq 1, \tag{7}$$

where $\boldsymbol{\gamma}(0)$ is the cross-correlation matrix. Since both $\mathbf{A}$ and $\boldsymbol{\gamma}(0)$ can only have positive values according to Equation (6) of Pedeli and Karlis (2013b), Equation (7) can only achieve positive autocorrelations as well.

In time series analysis, the two models mentioned above belong to the class of observation-driven models. As mentioned in Davis *et al.* (1999), while the observation-driven model is advantageous for easily calculating the forecasting density function, it is not good at characterizing the evolutionary properties of time series. An alternative is the parameter-driven model, which assumes that the serial correlation is induced by a latent variable. Then the evolutionary properties can be typically inherited by those assumed for this latent variable. Usually, we call this latent variable as the *hidden state* and can resort to state space approaches for analysis (Durbin and Koopman, 2000).

In the case of univariate count series, state space approaches have been widely used in Zeger (1988), Harvey and Fernandes (1989), and Chan and Ledolter (1995). For more discussions about univariate count series modeling, please refer to Fokianos (2012). For multivariate cases, Jørgensen *et al.* (1999) and Jung *et al.* (2011) propose two-factor models to describe autocorrelated multivariate Poisson counts. Both models assume that the count data of each dimension follow a Poisson distribution whose mean value is driven by some common latent factors following gamma Markov processes (Jørgensen *et al.*, 1999) or Gaussian autoregressive processes (Jung *et al.*, 2011). The former allows for mere positive autocorrelations, whereas the latter allows for autocorrelations of either sign. As these factor models explain the interactions of different counts by regression models, they avoid directly analyzing their cross-correlation structure. However, the choice of latent factors usually requires domain-specific knowledge, and the criteria about how many factors are needed are not always clear. As a result, it is difficult to generalize these models to fields where no factor or ambiguous factors exist.

Motivated by the wide application of autocorrelated multivariate counts and the infancy of reasonable models to describe them, this article further explores this field with a twofold contribution. First, this article proposes an easy-to-interpret state space model to describe autocorrelated multivariate counts. This model allows for flexible cross-correlation and autocorrelation structures of count data and can handle a large range of overdispersion. Specifically, this model builds upon the log-normal mixture Poisson distribution of Aitchison and Ho (1989) and allows for serial dependence by considering the Poisson mean vector as a latent variable evolving according to a state space model. In this way, the model can describe the cross-correlations and autocorrelations of count data flexibly. By integrating out the latent variable distribution, this model can generate itself an over-dispersed unconditional distribution and hence can capture the overdispersion of count data. Second, this article presents an efficient estimation algorithm for the proposed model. The challenge is that the marginal unconditional likelihood function of the model has no closed form, so the model needs numerical integration methods for estimation. Here the Monte Carlo Expectation Maximization (MCEM) algorithm is used, where the MC part is done by particle filtering and smoothing methods. Numerical studies show that the MCEM algorithm presents accurate estimation results for the model parameters. Particle filtering also provides an asymptotically unbiased estimator for the latent variable in a sequential way with a small computational complexity.

The remainder of this article is organized as follows. Section 2 introduces our proposed state space multivariate Poisson model in detail. Section 3 discusses the model estimation procedure. Section 4 stresses the advantages of the proposed model by comparing it with some other state-of-the-art Poisson models of multivariate counts and demonstrates the proposal using a real data example from the power utility industry. Finally, Section 5

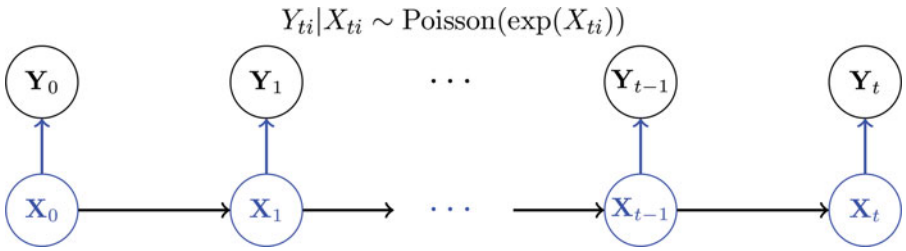


Figure 1. The evolution of SSMP.

concludes this article with remarks. Some technical details are provided in the Appendix.

2. A state space model for autocorrelated multivariate Poisson counts

In this section, we introduce a state space model to describe autocorrelated multivariate counts, which can not only flexibly describe the cross-correlation and autocorrelation structure of count data but also take overdispersion into consideration.

Consider d-dimensional count variables $\mathbf{Y}_t = [Y_{t1}, \ldots, Y_{td}]$ for time $t = 1, \ldots, T$. For each dimension at time t, conditional on its mean λ_{ti}, we assume Y_{ti} follows an independent Poisson distribution with the pmf as

$$p(Y_{ti}|\lambda_{ti}) = \frac{\exp(-\lambda_{ti})\lambda_{ti}^{Y_{ti}}}{Y_{ti}!}, i = 1, \ldots, d, t = 1, \ldots, T. \quad (8)$$

Then the joint conditional distribution of $\mathbf{Y}_t$ is

$$p(\mathbf{Y}_t|\boldsymbol{\lambda}_t) = \prod_{i=1}^{d} p(Y_{ti}|\lambda_{ti}), t = 1, \ldots, T, \quad (9)$$

where $\boldsymbol{\lambda}_t = [\lambda_{t1}, \ldots, \lambda_{td}]$. We assume $\mathbf{X}_t = \log(\boldsymbol{\lambda}_t) = [\log(\lambda_{t1}), \ldots, \log(\lambda_{td})]$ as a latent random variable following a multivariate normal distribution. It is $\mathbf{X}_t$ that introduces both cross-correlations and autocorrelations into $\mathbf{Y}_t$. Specifically, we consider $\mathbf{X}_t$ evolves according to a state space model as

$$p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{X}_{t-1}) : \mathbf{X}_t - \boldsymbol{\mu} = \boldsymbol{\Phi} \times (\mathbf{X}_{t-1} - \boldsymbol{\mu}) + \boldsymbol{\epsilon}_t, \quad (10)$$

where $\boldsymbol{\epsilon}_t$ is the white noise following a d-dimensional multivariate normal distribution with mean vector $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}$; i.e., $\mathbf{N}(\mathbf{0}, \boldsymbol{\Sigma})$. So far we have introduced all of the model parameters $\boldsymbol{\Theta} = \{\boldsymbol{\mu}, \boldsymbol{\Phi}, \boldsymbol{\Sigma}\}$.

As long as $\boldsymbol{\Phi}$ satisfies

$$\det(\mathbf{I} - z\boldsymbol{\Phi}) \neq 0, \quad \text{for all } |z| \leq 1, \ z \in \mathcal{C},$$

$\mathbf{X}_t$ is stationary, and the marginal distribution of $\mathbf{X}_t$ is multivariate normal with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}$ is the solution of $\boldsymbol{\Gamma} = \boldsymbol{\Phi}\boldsymbol{\Gamma}\boldsymbol{\Phi}' + \boldsymbol{\Sigma}$ according to the Yule–Walker relationship.

Marginalizing out $\mathbf{X}_t$, the unconditional distribution of $\mathbf{Y}_t$ can be expressed as

$$p_{\boldsymbol{\Theta}}(\mathbf{Y}_t) = \int_{\mathbf{R}_+^d} \prod_{i=1}^{d} \frac{\exp(-\exp(X_{ti}))\exp(X_{ti})^{Y_{ti}}}{Y_{ti}!} p_{\boldsymbol{\Theta}}(\mathbf{X}_t)\mathrm{d}\mathbf{X}_t, \quad (11)$$

where $p_{\boldsymbol{\Theta}}(\mathbf{X}_t)$ is the probability density function (pdf) of $\mathbf{N}(\boldsymbol{\mu}, \boldsymbol{\Gamma})$.

Figure 1 illustrates the evolution process of $\{\mathbf{X}_t, \mathbf{Y}_t\}$, which can be viewed as a nonlinear state space model. This hierarchical model allows for flexible cross-correlations and autocorrelations of multivariate counts. It ties multiple counts together but allows for individual stochastic components through the term $\boldsymbol{\epsilon}_t$. Higher-order autocorrelations of $\mathbf{X}_t$ can also be accommodated in the model. Hereafter, we call the proposed model as the State Space Multivariate Poisson model (SSMP). It is to be noted that when $d = 1$, SSMP degenerates to the univariate parameter-driven count series models of Chan and Ledolter (1995), Kuk and Cheng (1997), and Jung and Liesenfeld (2001). Although the unconditional distribution of $\mathbf{Y}_t$ in Equation (11) has no closed form, its moment properties can be obtained through conditional expectations and the properties of Poisson and normal distributions as shown in the following propositions. Detailed derivations are given in the Appendix.

Proposition 1. *(Mean of SSMP). Provided that $\mathbf{X}_t$ is stationary following Equation (10), $\mathbf{Y}_t$ is stationary with its unconditional mean as*

$$\mathrm{E}(Y_{ti}) \equiv \alpha_i = \exp\left(\mu_i + \frac{1}{2}\Gamma_{ii}\right) \quad (12)$$

for $i = 1, \ldots, d$, where Γ_{ii} is the ith diagonal component of $\boldsymbol{\Gamma}$.

This proposition shows that, as long as the latent variable is stationary, the process of counts is stationary whose mean vector is jointly decided by $\boldsymbol{\Theta} = \{\boldsymbol{\mu}, \boldsymbol{\Phi}, \boldsymbol{\Sigma}\}$.

Proposition 2. *(Variance of SSMP). Provided that $\mathbf{X}_t$ is stationary following Equation (10), the unconditional covariance matrix of $\mathbf{Y}_t$ can be expressed as*

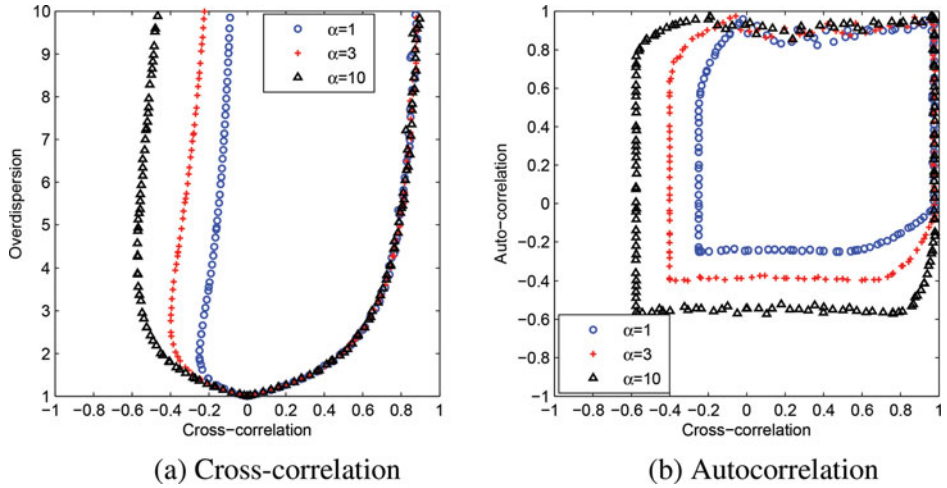
$$\mathrm{Var}(Y_{ti}) = \alpha_i\left[1 + \alpha_i(\exp(\Gamma_{ii}) - 1)\right], \quad \text{for} \quad i = 1, \ldots, d, \quad (13)$$

$$\mathrm{Cov}(Y_{ti}, Y_{tj}) = \alpha_i\alpha_j(\exp(\Gamma_{ij}) - 1), \quad \text{for} \quad i \neq j, i, j = 1, \ldots, d. \quad (14)$$

As a result,

$$\mathrm{Corr}(Y_{ti}, Y_{tj}) = \frac{\exp(\Gamma_{ij}) - 1}{\sqrt{\left[\alpha_i^{-1} + \exp(\Gamma_{ii}) - 1\right]\left[\alpha_j^{-1} + \exp(\Gamma_{jj}) - 1\right]}}, \quad (15)$$

where Γ_{ij} is the (i, j) component of $\boldsymbol{\Gamma}$.



(a) Cross-correlation (b) Autocorrelation

Figure 2. The regions of cross-correlation, overdispersion, and autocorrelation attainable for SSMP with fixed α and tunable $\{\mu, \gamma_0, \Gamma_{12}, \Gamma_{21}\}$ for bivariate counts.

Equation (13) shows that the unconditional variance of Y_{ti} for every dimension of SSMP exhibits overdispersion. The amount of overdispersion increases with Γ_{ii} and α_i. Only if $\Gamma_{ii} = 0$ indicating that X_{ti} degenerates to a fixed value, the overdispersion disappears, and consequently Y_{ti} follows the traditional Poisson distribution with no serial correlations. The cross-correlation structure of $\mathbf{Y}_t$ also depends on Γ_{ij} and can take either positive or negative values. However, as mentioned in Aitchison and Ho (1989), because

$$|\text{Corr}(Y_{ti}, Y_{tj})| < |\text{Corr}(X_{ti}, X_{tj})|, \tag{16}$$

the range of possible cross-correlations of $\mathbf{Y}_t$ is not as wide as those of $\mathbf{X}_t$. However, their gap becomes smaller when α_i and α_j become larger. Figure 2(a) provides the descriptive power of SSMP for bivariate counts with respect to attainable cross-correlations and overdispersion. Here we only consider a special case with $\mu_1 = \mu_2 = \mu$, $\Gamma_{11} = \Gamma_{22} = \gamma_0$, and hence $\alpha_1 = \alpha_2 = \alpha$ according to Equation (12). We fix α but vary μ, γ_0, Γ_{12}, and Γ_{21} to get different realizations of the cross-correlation structure with the same mean vector. The enclosed areas are the attainable regions for SSMP with $\alpha = 1$, 3, and 10, respectively. We can see that SSMP has the flexibility to accommodate different overdispersion magnitudes, which can be controlled by choosing appropriate γ_0 while maintaining the same α. On the other hand, for a small α, the model cannot realistically describe circumstances with moderate or strong negative cross-correlations. As α increases, the model is able to describe larger negative cross-correlations.

Proposition 3. *(Autocorrelation of SSMP). Provided that* $\mathbf{X}_t$ *is stationary, the unconditional autocorrelations of* $\mathbf{Y}_t$ *can be expressed as*

$$\text{Corr}(Y_{ti}, Y_{(t-\tau)j}) = \frac{E[\exp(X_{ti} + X_{(t-\tau)j})] - \alpha_i\alpha_j}{\alpha_i\alpha_j\sqrt{[\alpha_i^{-1} + \exp(\tau_{ii}) - 1][\alpha_j^{-1} + \exp(\Gamma_{jj}) - 1]}} \tag{17}$$

for $\tau = 0, 1, \ldots, i, j = 1, \ldots, d$.

The unconditional autocorrelations of $\mathbf{Y}_t$ depend on $E[\exp(X_{ti} + X_{(t-\tau)j}]$, which can be calculated from the characteristic function of $\mathbf{X}_t$ (see Appendix for details). The autocorrelations can be either positive or negative, but still we have

$$|\text{Corr}(Y_{ti}, Y_{(t-\tau)j})| < |\text{Corr}(X_{ti}, X_{(t-\tau)j})|. \tag{18}$$

Similarly, in Figure 2(b) we plot the descriptive power of SSMP with respect to attainable cross-correlations and first-order autocorrelations with the same settings as Figure 2(a). We see that SSMP can describe processes of counts with certain negative autocorrelations but not as flexible as positive ones. With the increase in α, the attainable regions increase for both kinds of correlations. In summary, Propositions 2 and 3 indicate that SSMP is more suitable to describe count data with moderate or big mean values.

3. Parameter estimation and inference

3.1. Prediction and inference

Now we study how to make inference and predictions based on SSMP. Here we temporarily assume that the model parameters $\boldsymbol{\Theta}$ are known and will discuss how to estimate them in Section 3.2.

Since in SSMP, $\{\mathbf{Y}_{1:T}\}$ are observable whereas $\{\mathbf{X}_{1:T}\}$ are latent, most of the inference problems focus on estimating the latent process $p_{\boldsymbol{\Theta}}(\mathbf{X}_{1:T}|\mathbf{Y}_{1:T})$ given the total T observations $\{\mathbf{Y}_{1:T}\}$. More specifically, we first focus on predicting $\mathbf{X}_{t+1}$ based on the previous and current observations $\{\mathbf{Y}_{1:t}\}$; i.e., $p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{Y}_{1:t})$. Second, we study the inference of $\mathbf{X}_t$ based on the total observations $\{\mathbf{Y}_{1:T}\}$; i.e., $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:T})$ for $t = 1, \ldots, T$. The challenge involved is due to the posterior distribution $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:t})$ having no closed form for arbitrary $\boldsymbol{\Theta}$; thus we need to resort to numerical methods, such as numerical integration or Markov Chain Monte Carlo (MCMC). In addition, we often need to update the predictions or estimations upon the arrival of new observations; e.g., to obtain $p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+2}|\mathbf{Y}_{1:(t+1)})$ from $p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{Y}_{1:t})$. As a result, it is desirable to have computationally efficient algorithms to make inference and predictions sequentially.

Considering the performance and computational efficiency, in state space model analysis, Particle Filtering (PF, also called

Sequential Monte Carlo) and Particle Smoothing (PS) methods are particularly useful. PF is designed to approximate $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:t})$ in a sequential manner with acceptably small computational complexity (Hürzeler and Künsch, 1998; Liu and Chen, 1998; Doucet *et al.*, 2000). Its basic idea is to compute $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:t})$ by *importance sampling*; i.e., approximating $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:t})$ by a set of samples, called *particles*, with their associated weights. The weight assigned to each particle is proportional to its probability of being sampled from the posterior distribution. When new data are observed, new particles and their associated weights can be efficiently obtained at an affordable computational burden.

In particular, for the prediction task, we assume at time t, the posterior density is approximated by weighted Dirac delta functions as

$$p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:t}) \approx \sum_{i=1}^{N_p} W_t^i \cdot \delta(\mathbf{X}_t - \mathbf{x}_t^i). \tag{19}$$

Here in Equation (19), δ is the Dirac delta function; N_p is the number of particles; and the normalized weights W_t^i satisfy $\sum_{i=1}^{N_p} W_t^i = 1$. To obtain the prediction distribution $p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{Y}_{1:t})$, we first generate samples of $\mathbf{X}_{t+1}$ from $p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{Y}_{1:t})$. For this purpose, each particle $\mathbf{x}_{t+1}^i$ is propagated following the state equation in Equation (10) with a random noise $\boldsymbol{\epsilon}_{t+1}^i$ drawn from the state noise distribution $\mathbf{N}(\mathbf{0}, \boldsymbol{\Sigma})$; i.e., $\mathbf{x}_{t+1}^i = \boldsymbol{\mu} + \boldsymbol{\Phi}(\mathbf{x}_t^i - \boldsymbol{\mu}) + \boldsymbol{\epsilon}_{t+1}^i$. Then we have

$$p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{Y}_{1:t}) \approx \sum_{i=1}^{N_p} W_t^i \cdot \delta(\mathbf{X}_{t+1} - \mathbf{x}_{t+1}^i). \tag{20}$$

When the new observation $\mathbf{Y}_{t+1}$ arrives, we update the conditional distribution of $\mathbf{X}_{t+1}$ and approximate $p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{Y}_{1:t+1})$. In fact, the updated distribution takes the same form as Equation (20) and uses the same set of particles. It only needs to update every particle's weight based on the likelihood $p_{\boldsymbol{\Theta}}(\mathbf{Y}_{t+1}|\mathbf{x}_{t+1}^i)$ based on the Bayes rule; that is,

$$p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{Y}_{1:(t+1)}) \approx \sum_{i=1}^{N_p} W_{t+1}^i \times \delta(\mathbf{X}_{t+1} - \mathbf{x}_{t+1}^i), \quad \text{where}$$
$$W_{t+1}^i \propto W_t^i \times p(\mathbf{Y}_{t+1}|\mathbf{x}_{t+1}^i).$$

The convergence of the approximated distribution by PF to the true $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:t})$ is guaranteed by the Central Limit Theorem (Liu, 2008), which ensures its estimation accuracy. Due to its computational efficiency and sequential nature, PF has been widely applied for nonlinear state space model inference and latent process tracking. See Doucet *et al.* (2001) and the references therein for a detailed introduction.

One problem of PF is that the distribution of the particles' weights becomes more and more skewed as t increases. Hence, after some iterations, only very few particles have non-zero weights. This phenomenon is called *degeneracy*. We can evaluate it in terms of the so-called Effective Sample Size (ESS; Liu, 2008), which is given by ESS $= (\sum_{i=1}^{N_p}(W_t^i)^2)^{-1}$. An intuitive solution for degeneracy is to multiply the particles with higher normalized weights and discard the particles with lower weights. This can be done by adding a resampling step. Specifically, if ESS is smaller than a pre-specified threshold α, we resample from the set $\{(W_t^i, \mathbf{x}_t^i), i = 1, \ldots, N_p\}$ with the probabilities $p(\hat{\mathbf{x}}_t^j = \mathbf{x}_t^i) = W_t^i$, $i = 1, \ldots, N_p$ with replacement N_p times, to get a new set $\{(\frac{1}{N_p}, \hat{\mathbf{x}}_t^j), j = 1, \ldots, N_p\}$. In this way, the skewness of the weights' distribution can be reduced. The detailed PF procedure involving the resampling step is summarized in Algorithm 1.

Algorithm 1 PF

At time t = 1

1: Initialization: sample $\mathbf{x}_1^i \sim p_0(\mathbf{X}_t)$ for $i = 1, \ldots, N_p$.
2: Compute the weights $w_1^i = p(\mathbf{Y}_1|\mathbf{x}_1^i)$ for $i = 1, \ldots, N_p$ and normalize the weights $W_1^i = \frac{w_1^i}{\sum_{i=1}^{N_p} w_1^i}$, $i = 1, \ldots, N_p$.
3: Calculate the filtered distribution $p(\mathbf{X}_1|\mathbf{Y}_1) \approx \sum_{i=1}^{N_p} W_1^i \delta(\mathbf{X}_1 - \mathbf{x}_1^i)$.

At time t ≥ 2

4: Sample $\mathbf{x}_t^i \sim p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{x}_{t-1}^i)$ for $i = 1, \ldots, N_p$.
5: Compute the weights $w_t^i = W_{t-1}^i \cdot p(\mathbf{Y}_t|\mathbf{x}_t^i)$ for $i = 1, \ldots, N_p$, and normalize the weights $W_t^i = \frac{w_t^i}{\sum_{i=1}^{N_p} w_t^i}$, $i = 1, \ldots, N_p$.
6: Calculate the filtered distribution $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:t}) \approx \sum_{i=1}^{N_p} W_t^i \delta(\mathbf{X}_t - \mathbf{x}_t^i)$.
7: If the resample criterion is satisfied—i.e., ESS $= (\sum_{i=1}^{N_p}(W_t^i)^2)^{-1} < a$—then resample with replacement N_p times from $\{\mathbf{x}_t^i, i = 1 : N_p\}$ with the probabilities $p(\hat{\mathbf{x}}_t^j = \mathbf{x}_t^i) = W_t^i$, $i = 1, \ldots, N_p$, and replace the previous set $\{(W_t^i, \mathbf{x}_t^i), i = 1, \ldots, N_p\}$ by $\{(\frac{1}{N_p}, \hat{\mathbf{x}}_t^j), j = 1, \ldots, N_p\}$.
8: Terminate when $t = T$; otherwise, $t = t + 1$, and go back to 4.

Now we consider estimating the latent process given the total T observations—i.e., $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:T})$, $t = 1, \ldots, T$—by PS. Its underpinning concept is to approximate $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:T})$ with the same particles as filtering but to readjust their weights by considering the information of the future observations $\{\mathbf{Y}_{t+1:T}\}$; that is,

$$p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:T}) \approx \sum_{i=1}^{N_p} W_{t|T}^i \delta(\mathbf{X}_t - \mathbf{x}_t^i), \tag{21}$$

for $t = 1, \ldots, T$, where

$$W_{t|T}^i = W_t^i \sum_{j=1}^{N_p} \frac{W_{t+1|T}^j p_{\boldsymbol{\Theta}}(\mathbf{x}_{t+1}^j|\mathbf{x}_t^i)}{\sum_{l=1}^{N_p} W_t^l p_{\boldsymbol{\Theta}}(\mathbf{x}_{t+1}^j|\mathbf{x}_t^l)},$$
$$t = 1, \ldots, T-1, i = 1, \ldots, N_p, \tag{22}$$

and $W_{T|T}^i = W_T^i$, $i = 1, \ldots, N_p$. The detailed PS procedure is summarized in Algorithm 2.

The computational complexities of filtering algorithms are generally much lower compared with other inference procedures, as they allow sequential updating when samples are observed incrementally. In particular, the PF updates the conditional distribution $P(\mathbf{X}_{t+1}|\mathbf{Y}_{1:(t+1)})$ from $P(\mathbf{X}_t|\mathbf{Y}_{1:t})$ when observing the new sample $\mathbf{Y}_{t+1}$. Given the particle size N_p, the complexity of PF at each step is $O(N_p)$. On the other hand, PS updates the distribution $P(\mathbf{X}_i|\mathbf{Y}_{1:(t+1)})$ from $P(\mathbf{X}_i|\mathbf{Y}_{1:t})$ for $i = 1, \ldots, t$. It has complexity $O(tN_p^2)$ to update all state estimations at time t. In our experiments using a laptop with an Intel i5 CPU, it took 0.015 seconds for one PF iteration to obtain

Algorithm 2 PS

1: Start by setting $W^i_{T|T} = W^i_T$ for $i = 1, \ldots, N_p$.
2: For each $t = T-1, \ldots, 1$, compute the smoothed weights by

$$W^i_{t|T} = W^i_t \sum_{j=1}^{N_p} \frac{W^j_{t+1|T} p_{\boldsymbol{\Theta}}(\mathbf{x}^j_{t+1}|\mathbf{x}^i_t)}{\sum_{l=1}^{N_p} W^l_t p_{\boldsymbol{\Theta}}(\mathbf{x}^j_{t+1}|\mathbf{x}^l_t)}, \; i = 1, \ldots, N_p.$$

3: Calculate the smoothed distribution $p_{\boldsymbol{\Theta}}(\mathbf{X}_t|\mathbf{Y}_{1:T}) \approx \sum_{i=1}^{N_p} W^i_{t|T} \delta(\mathbf{X}_t - \mathbf{x}^i_t)$ for $t = 1, \ldots, T$.

$P(\mathbf{X}_i|\mathbf{Y}_{1:i})$, $i = 1, \ldots, 500$, and 15 seconds for PS to obtain $P(\mathbf{X}_i|\mathbf{Y}_{1:500})$, $i = 1, \ldots, 500$ for a two-dimensional SSMP process with $N_p = 500$. In general, we believe that the computational load with increasing N_p should not be a major concern, due to the fast development of high-performance parallel computing. In fact, PF is inherently parallel, as it essentially consists of exploration of the state space by random, but independent, particles. The particles only interact when their weights need normalization. As a result, *parallel particle filtering* can be developed to take advantage of high-performance computing environments. We refer the readers to Brun *et al.* (2002) and Durham and Geweke (2011) and subsequent references for more details on these developments. In addition, PF algorithms have been developed that are scalable in the ultra-high-dimensional cases with small computation complexity (e.g., Beskos *et al.*, 2014). These algorithms may shed light on the application of SSMP in high-dimensional cases, which we will explore further in our future studies.

It should also be noted that in general, PF requires more particle samples for higher-dimensional state space model estimation. This is due to the space to be sampled increasing drastically with the dimension d, which makes it much harder for particles to efficiently propagate to the subspace with identifiably nonzero pdf. Therefore, weight degeneracy is the fundamental obstacle for particle filtering in high-dimensional models. To avoid collapse, the particle sample size N_p should be large enough. For Gaussian state space models, some PF methods, such as the bootstrap particle filter, can remain stable and consequently converge as long as N_p grows exponentially fast of d (Bengtsson *et al.*, 2008). The corresponding estimation error is bounded in the order of $c_t/\sqrt{N_p}$, where c_t is a constant independent of d (Theorem 4.3.1, Smith *et al.*, 2013). In addition, it should be noted that c_t increases exponentially with t. Therefore, N_p should also increase exponentially in t in order to achieve a given accuracy at time t Considering these features, we suggest N_p being exponential in p and T (the total time series length) to ensure the performance of the algorithm. Based on some additional simulation results, we found that for SSMP with $d = 10$ and $T = 500$, $N_p = 1000$ is sufficient to guarantee the convergence of PF and consequently the accuracy of the inference (in terms of parameter estimation and prediction). For extremely long time series, recalibration using full Bayesian MCMC can be performed periodically with interval T^*. T^* is the maximum time length for which the approximation of PF can achieve a pre-specified accuracy. The full Bayesian MCMC can guarantee obtaining accurate samples from the posterior distributions $P(\mathbf{X}_{kT^*}|\mathbf{Y}_{1:kT^*})$, $k = 1, 2, \ldots$ Consequently, these samples can be used as particles for future filtering before the next calibration, which can effectively avoid accumulation of errors without substantially increasing computational load.

3.2. Parameter estimation

This section considers estimation of the parameters $\boldsymbol{\Theta}$ of SSMP. In the Maximum Likelihood Estimation (MLE) framework, a natural and efficient estimation method to deal with latent variables is the Expectation Maximization (EM) algorithm. The EM algorithm is an iterative procedure that seeks $\boldsymbol{\Theta}^{(q)}$ in the qth iteration such that the likelihood is increased from that in the $(q-1)$st iteration. Its key idea is to postulate the "missing" data $\{\mathbf{X}_{1:T}\}$ and to consider maximizing the likelihood function given the complete data $\{\mathbf{X}_{1:T}, \mathbf{Y}_{1:T}\}$. Underlying this strategy is the idea that maximizing the "complete" log-likelihood $\log p_{\boldsymbol{\Theta}}(\mathbf{X}_{1:T}, \mathbf{Y}_{1:T})$ is easier than maximizing the incomplete one $\log p_{\boldsymbol{\Theta}}(\mathbf{Y}_{1:T})$. Here, due to the Markovian structure of SSMP, the complete data log-likelihood has the form

$$\log p_{\boldsymbol{\Theta}}(\mathbf{X}_{1:T}, \mathbf{Y}_{1:T}) = \log p_0(\mathbf{X}_1) + \sum_{t=1}^{T-1} \log p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{X}_t) + \sum_{t=1}^{T} \log p(\mathbf{Y}_t|\mathbf{X}_t). \quad (23)$$

However, because $\{\mathbf{X}_{1:T}\}$ are unavailable, the EM algorithm replaces Equation (23) by $\mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)})$, which is the conditional expectation of Equation (23) with respect to $\{\mathbf{X}_{1:T}\}$ given the observations $\{\mathbf{Y}_{1:T}\}$ using the parameters $\boldsymbol{\Theta}^{(q)}$ in the current iteration; that is,

$$\text{E step:} \quad \mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)}) = \int p_{\boldsymbol{\Theta}^{(q)}}(\mathbf{X}_{1:T}|\mathbf{Y}_{1:T}) \times \log p_{\boldsymbol{\Theta}}(\mathbf{X}_{1:T}, \mathbf{Y}_{1:T}) \mathrm{d}\mathbf{X}_{1:T}. \quad (24)$$

Then we want to find the revised parameter estimates $\boldsymbol{\Theta}^{(q+1)}$ that maximize the function

$$\text{M step:} \quad \boldsymbol{\Theta}^{(q+1)} = \arg\max_{\boldsymbol{\Theta}} \mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)}). \quad (25)$$

For SSMP, we can get $\mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)})$ in Equation (24) as

$$\mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)}) = \sum_{t=1}^{T-1} \iint p_{\boldsymbol{\Theta}^{(q)}}(\mathbf{X}_t, \mathbf{X}_{t+1}|\mathbf{Y}_{1:T}) \times \log p_{\boldsymbol{\Theta}}(\mathbf{X}_{t+1}|\mathbf{X}_t) \mathrm{d}\mathbf{X}_t \mathrm{d}\mathbf{X}_{t+1}. \quad (26)$$

Unfortunately, here $p_{\boldsymbol{\Theta}^{(q)}}(\mathbf{X}_t, \mathbf{X}_{t+1}|\mathbf{Y}_{1:T})$ is not analytical and consequently $\mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)})$ is intractable. However, on the other hand, the particles used in PF and PS can be viewed as samples from the conditional distribution. As a result, we can use these particles to approximate $p_{\boldsymbol{\Theta}^{(q)}}(\mathbf{X}_t, \mathbf{X}_{t+1}|\mathbf{Y}_{1:T})$ and consequently to implement the MCEM algorithm for parameter estimation. In more details, from Equations (19) and (21), we obtain

$$\hat{\mathcal{Q}}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)}) \approx \sum_{t=1}^{T-1} \sum_{i=1}^{N_p} \sum_{j=1}^{N_p} W_{t,t+1|T}^{ij} \log p_{\boldsymbol{\Theta}}(\mathbf{x}_{t+1}^j | \mathbf{x}_t^i), \text{ where}$$

$$W_{t,t+1|T}^{ij} = W_t^i \frac{W_{t+1|T}^j p_{\boldsymbol{\Theta}^{(q)}}(\mathbf{x}_{t+1}^j | \mathbf{x}_t^i)}{\sum_{l=1}^{N_p} W_t^l p_{\boldsymbol{\Theta}^{(q)}}(\mathbf{x}_{t+1}^j | \mathbf{x}_t^l)}. \quad (27)$$

For more detailed derivations together with the convergence properties for PF- and PS-based MCEM, please refer to Schön *et al.* (2011).

In the M step, with the gradient available for Equation (27), we get $\boldsymbol{\Theta}^{(q+1)} = \{\boldsymbol{\mu}^{(q+1)}, \boldsymbol{\Phi}^{(q+1)}, \boldsymbol{\Sigma}^{(q+1)}\}$ as

$$\begin{aligned} \boldsymbol{\Pi}^{(q+1)} &= [(\mathbf{I} - \boldsymbol{\Phi}^{(q+1)})\boldsymbol{\mu}^{(q+1)}, \boldsymbol{\Phi}^{(q+1)}]' \\ &= \left(\sum_{t=1}^{T-1} \sum_{i=1}^{N_p} \sum_{j=1}^{N_p} W_{t,t+1|T}^{ij} \mathbf{x}_{t+1}^j \mathbf{z}_t^{i'} \right) \\ &\quad \times \left(\sum_{t=1}^{T-1} \sum_{i=1}^{N_p} \sum_{j=1}^{N_p} W_{t,t+1|T}^{ij} \mathbf{z}_t^i \mathbf{z}_t^{i'} \right)^{-1}, \quad (28) \end{aligned}$$

$$\begin{aligned} \boldsymbol{\Sigma}^{(q+1)} = \frac{1}{T-1} \sum_{t=1}^{T-1} \sum_{i=1}^{N_p} \sum_{j=1}^{N_p} W_{t,t+1|T}^{ij} \left(\mathbf{x}_{t+1}^j - \boldsymbol{\Pi}^{(q+1)} \mathbf{z}_t^i \right) \\ \times \left(\mathbf{x}_{t+1}^j - \boldsymbol{\Pi}^{(q+1)} \mathbf{z}_t^i \right)', \quad (29) \end{aligned}$$

where $\mathbf{z}_t^i = [1, \mathbf{x}_t^i]$. For detailed derivation of the EM algorithm, please refer to the Appendix. The procedure of the particle EM-based estimation is summarized in Algorithm 3.

Algorithm 3 MCEM

1: Set $q = 0$ and initialize $\boldsymbol{\Theta}^{(q)}$ such that $\log p_{\boldsymbol{\Theta}^{(q)}}(\mathbf{Y}_{1:T})$ is finite.
2: Expectation (E) Step:
- Run Algorithms 1 and 2 to obtain the filtered and smoothed distributions for $t = 1, \ldots, T$.
- Calculate $\hat{\mathcal{Q}}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(q)})$ according to Equation (27).

3: Maximization (M) Step:
- Compute $\boldsymbol{\Theta}^{(q+1)}$ according to Equations (28) and (29).

4: Check the non-termination condition $\hat{\mathcal{Q}}(\boldsymbol{\Theta}^{(q+1)}, \boldsymbol{\Theta}^{(q)}) - \hat{\mathcal{Q}}(\boldsymbol{\Theta}^{(q)}, \boldsymbol{\Theta}^{(q)}) \geq \epsilon$ for some $\epsilon \geq 0$. If satisfied, update $q \rightarrow q + 1$ and return to 2; otherwise, terminate.

The following simulation was conducted to illustrate the performance. We set $N_p = 500$ to estimate a two-dimensional SSMP process with series length $T = 500$. The parameters were set to be $\boldsymbol{\Phi} = [0.6, 0.1; 0.2, 0.7]$, $\boldsymbol{\mu} = [4, 4]$, and $\boldsymbol{\Sigma} = 0.25\mathbf{I}_{2\times 2}$. We replicated the simulation 200 times. Each replication included data generation, estimation, and prediction, to evaluate the performance of the proposed method. For every replication, we randomly picked an initial value of $\boldsymbol{\Theta}^{(0)}$ and estimated the parameters iteratively based on the MCEM algorithm. Table 2 lists the estimation results. We observe that both the bias and the Root Mean Square Error (RMSE) of the estimators are acceptably small, illustrating the satisfactory estimation accuracy and stability of the MCEM algorithm. Then we used the estimated parameters $\boldsymbol{\Theta}$ to track the latent states $\{\mathbf{X}_{T+1:T+100}\}$ for the subsequent 100 observations $\{\mathbf{Y}_{T+1:T+100}\}$ based on Algorithm 1. The filtering results in one replication are shown in Figure 3. We can see that the tracked states (red crosses) based on PF almost overlap with the true states (the blue circles) with slight differences.

Table 2. Estimation bias and RMSE of Algorithm 3 based on 200 replications.

	True value	*Estimate bias*	*RMSE*
ϕ_1	0.6	− 0.0098	0.0409
ϕ_2	0.2	− 0.0035	0.0414
ϕ_3	0.1	− 0.0020	0.0290
ϕ_4	0.7	− 0.0077	0.0334
μ_1	4.0	0.0104	0.0786
μ_2	4.0	0.0104	0.1025
σ_{11}	0.25	− 0.0036	0.0174
σ_{12}	0	0.002	0.0140
σ_{22}	0.25	− 0.0031	0.0197

4. Case studies

4.1. Simulation studies

As emphasized earlier, the most advantageous property of SSMP is that it allows for more flexible cross-correlation and auto-correlation structures of count data compared with other state-of-the-art models. Here we demonstrate this point using some numerical studies. We compare SSMP with the log-normal mixture Poisson model (LP) of Aitchison and Ho (1989), which is a sub-model of SSMP, by setting the autocorrelation structure $\boldsymbol{\Phi} = \mathbf{0}$, and the other two time series models, MACP of Heinen and Rengifo (2007) with orders $p = 1$ and $q = 1$ (shorted as MACP (1, 1)), and MINAR(1) of Pedeli and Karlis (2013b). In our experiments, 500 observations were generated from each model and then fitted by all of these models using MLE methods separately. Here we adopt the Bayesian Information Criterion (BIC) to evaluate their fitting performances. In particular,

$$\text{BIC} = -2 \ln \hat{L} + k \ln(n),$$

where $\hat{L}$ is the fitted likelihood, k is the number of parameters in the model, and n is the number of observations. The parameters estimated in every model are listed in Table 3 with the same notations as those in the original papers, together with the corresponding k for a two-dimensional Poisson process.

For illustration purposes, here we simply consider processes of bivariate counts with either positive or negative cross-correlations or autocorrelations. Table S.2 (online) summarizes the true parameters and the fitted ones of all of these models, and Table 4 presents their BIC values with the corresponding logarithm of the fitted likelihood in the parenthesis.

Furthermore, we consider the prediction power of the fitted models. Following Czado *et al.* (2009), we adopt the Dawid–Sebastiani (DS) score to evaluate the prediction power. In particular, for every fitted model, we derived the one-step-ahead prediction probability $P(\mathbf{Y}_t | \mathbf{Y}_{1:t-1}, \boldsymbol{\Theta})$ for $t = 2, \ldots, T$ and calculated the DS score for every dimension. Denote $\boldsymbol{\mu}_t = E(\mathbf{Y}_t | \mathbf{Y}_{1:t-1}) \equiv [\mu_{t,1}, \mu_{t,2}, \ldots, \mu_{t,p}]$ and $\boldsymbol{\sigma}_t = Cov(\mathbf{Y}_t | \mathbf{Y}_{1:t-1})$ with diagonal element $[\sigma_{t,1}, \sigma_{t,2}, \ldots, \sigma_{t,p}]$. Then the DS score for dimension i is defined as

$$DSS_{t,i}(\mathbf{Y}_{t,i}) = \frac{Y_{t,i} - \mu_{t,i}}{\sigma_{t,i}} + 2 \log(\sigma_{t,i}).$$

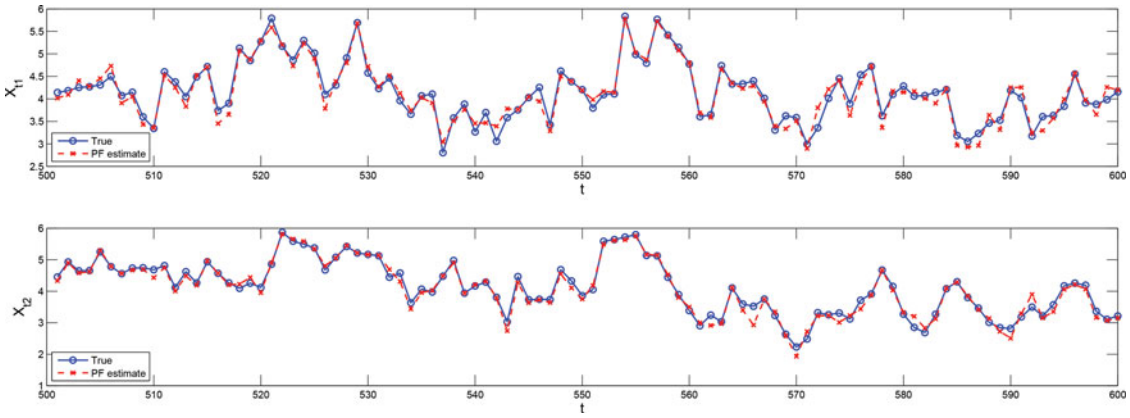


Figure 3. The tracked states $\{\mathbf{X}_{T+1:T+100}\}$ based on Algorithm 1 with parameters estimated by Algorithm 3.

Table 5 reports the mean of the DS scores over the $T-2$ samples for different fitted models, with the Mean Squared Prediction Error (MSPE) shown in the parentheses. The results are consistent with those in Table 4. In both tables, the first sign in SSMP indicates the sign of the cross-correlation, and the second sign indicates the sign of the autocorrelation.

We can see that, generally, the, results are consistent with the results based on BIC values. The fitted model that has the same form as the true model has the lowest DS score, meaning the best prediction performance. However, SSMP is always the second best with a slight higher DS score or BIC value. This demonstrates that SSMP is able to describe count data generated by different models (mechanisms). Furthermore, for data coming from MINAR(1, 1) or MACP(1, 1) or positive autocorrelated and cross-correlated SSMP (the fifth row), the DS scores of different fitted models are similar. This indicates that all three models can describe these processes equally well. On the other hand, MINAR(1, 1) has the largest BIC values and DS scores for data coming from LP or negative cross-correlated or autocorrelated SSMP data (the fourth or sixth row). The reason for the former is that the LP data have large overdispersion, and the reason for the latter is that MINAR cannot describe either negative cross-correlations or negative autocorrelations. On the contrary, although MACP(1, 1) can fit the LP data or negative cross-correlated SSMP data, it still fails to predict data with negative autocorrelations. As to LP, it has comparatively small BIC values but large DS scores for almost all cases, indicating that LP may overfit the data.

Table 3. Parameters estimated for different models together with k for a two-dimensional Poisson process.

Model	*MINAR(1)*	*MACP(1, 1)*	*SSMP*	*LP*
Parameters	$\mathbf{A}, \lambda_1, \lambda_2, \phi$	$\mathbf{A}, \mathbf{B}, \boldsymbol{\omega}, \phi, (\boldsymbol{\Sigma})$	$\boldsymbol{\Phi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$	$\boldsymbol{\mu}, \boldsymbol{\Sigma}$
k	7	11(14)	9	5

Table 4. The BIC values of the fitted models with the corresponding logarithm of the fitted likelihood in the parenthesis.

	Fitted models			
Data Model	*MINAR(1)*	*MACP(1, 1)*	*SSMP*	*LP*
MINAR(1)	5324 (−2640)	5378 (−2655)	5442 (−2693)	5499 (−2730)
MACP(1, 1)	5742 (−2849)	5624 (−2781)	5636 (−2790)	6634 (−3296)
LP	6241 (−3099)	6036 (−2984)	5924 (−2934)	5867 (−2915)
SSMP (−+)	5800 (−2878)	5456 (−2694)	5452 (−2698)	5565 (−2764)
SSMP(++)	5634 (−2790)	5432 (−2682)	5426 (−2685)	5469 (−2720)
SSMP(+−)	6466 (−3211)	8352 (−4142)	5518 (−2731)	5809 (−2886)

Table 5. The DS score for different models (with the MSPE shown in the parenthesis).

	Fitted models			
Data model	*MINAR(1)*	*MACP(1, 1)*	*SSMP*	*LP*
MINAR(1)	3.45 (10.74)	3.48 (12.03)	3.43 (10.98)	3.66 (14.15)
MACP(1, 1)	3.71 (15.03)	3.73 (15.39)	3.79 (15.97)	4.70 (40.50)
LP	5.05 (30.44)	4.39 (27.86)	4.46 (27.92)	4.30 (27.17)
SSMP(−+)	3.85 (16.01)	3.59 (13.96)	3.42 (10.49)	3.74 (15.54)
SSMP(++)	3.68 (14.65)	3.63 (14.01)	3.61 (12.69)	3.72 (15.17)
SSMP(+−)	4.58 (24.89)	22.36 (75.58)	3.68 (12.35)	3.95 (14.68)

4.2. A real application in the power utility industry

Now we use the proposed SSMP as well as the other three models to analyze the dynamic interactions of different types of damage that can occur in a power utility system. The dataset records the counts of three types of damage in a region every day for a period of $T = 100$ days. Due to confidentiality reasons, we reserve the detailed information of these three types of damage but simply denote them as type A, type B, and type C. Different types of damage may be caused by common weather-related conditions or accidents. Therefore, they may have certain contemporaneous correlations with each other and serial correlations with their previous observations. Figure 4 shows the logarithm of the count data over time. Their similar change patterns reveal their cross-correlation structure to some degree. Figure S.1 shows the autocorrelation function of these counts, from which we can see strong serial correlations. Descriptive statistics for the count data are also summarized in the left part of Table 6. We can see that the empirical marginal distributions of the count data are clearly over-dispersed, and their sample correlations are all positive.

We use these samples to fit SSMP. The parameter estimates based on Algorithm 3 with $N = 21$ replications are illustrated in Table 7. In every replication we initialized $\boldsymbol{\Theta}^{(0)}$ by adding some random noise to the calculated $\{\boldsymbol{\mu}, \boldsymbol{\Phi}, \boldsymbol{\Sigma}\}$ in the left part of Table 6. We set $N_p = 1000$ to ensure the accuracy of the E-step. Each iteration took around 200 seconds to complete on

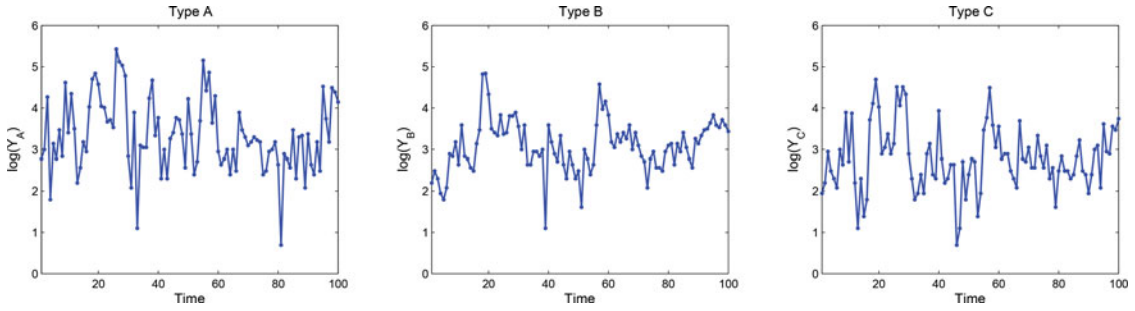


Figure 4. Logarithm of the count data in the power utility system.

Table 6. Descriptive statistics for the count data in the power utility system.

	Data			*Fitted SSMP*			*Fitted LP*			*Fitted MACP(1, 1)*			*Fitted MINAR(1)*		
	Type A	*Type B*	*Type C*	*Type A*	*Type B*	*Type C*	*Type A*	*Type B*	*Type C*	*Type A*	*Type B*	*Type C*	*Type A*	*Type B*	*Type C*
Mean	40.1	26.2	21.6	39.9	27.4	21.8	40.2	25.7	20.8	43.1	30.7	26.3	64.7	42.0	38.8
Standard dev.	40.6	19.5	20.6	38.7	16.5	18.3	40.0	15.6	17.2	28.2	15.5	17.1	64.7	42.0	38.8
Overdispersion	41.1	14.5	19.6	37.5	9.92	15.4	39.7	9.5	14.2	18.4	7.8	11.1	1	1	1
Cross-	1.00			1.00			1.00			1.00			1.00		
correlation	0.48	1.00		0.37	1.00		0.37	1.00		0.48	1.00		0.12	1.00	
	0.78	0.74	1.00	0.58	0.58	1.00	0.61	0.62	1.00	0.63	0.60	1.00	0.37	0.18	1.00
Lag 1	0.45	0.26	0.40	0.33	0.17	0.28				0.36	0.14	0.40	0.29	0.03	0.11
Autocorrelation	0.34	0.65	0.53	0.27	0.57	0.43				0.24	0.36	0.54	0.04	0.35	0.06
	0.46	0.44	0.50	0.35	0.36	0.41				0.39	0.51	0.59	0.12	0.06	0.32
loglikelihood					−769			−815			−1258			−1045	
DSS(MSPE)					6.79 (559)			7.51 (809)			7.11 (587)			14.2 (817)	
Residual mean				0.027	0.008	0.028	−0.003	0.033	0.048	−0.047	−0.152	−0.133	1.527	1.458	1.148
Standard dev.				0.942	0.904	0.954	1.016	1.249	1.197	1.281	0.957	0.915	4.671	2.287	2.904
Lag 1 Autocorrelation				0.050	−0.018	0.020	0.453	0.255	0.396	0.058	0.095	0.005	−0.327	−0.152	−0.154
				−0.040	0.131	0.045	0.342	0.647	0.532	0.156	0.367	0.091	−0.137	0.040	0.064
				0.000	0.090	0.023	0.464	0.442	0.503	−0.024	−0.013	−0.071	−0.250	0.010	−0.186
Q_3				6.601	2.765	0.827	33.020	56.070	33.86	6.286	22.123	1.8526	15.08	1.69	3.5
				(0.09)	(0.43)	(0.84)	(0.00)	(0.00)	(0.00)	(0.10)	(0.00)	(0.60)	(0.00)	(0.64)	(0.32)
Q_5				8.100	3.163	7.065	33.697	56.480	36.700	6.646	22.628	7.077	16.782	2.371	8.016
				(0.15)	(0.67)	(0.22)	(0.00)	(0.00)	(0.00)	(0.25)	(0.00)	(0.22)	(0.01)	(0.80)	(0.16)

a single-core personal computer. Figure S.2 shows the convergence process of every replication. We see generally that the estimates converge fast in the first $n = 5$ iterations. Based on the fitted model, we draw the one-step ahead prediction of $\log(\mathbf{Y}_t)$ in Figure S.3. We also fit the LP, MACP(1, 1), and MINAR(1) models using the data. Their corresponding parameter estimates are shown in Table S.1. It should be noted that because the dimension of the count data is three, we adopted Pedeli and Karlis (2013a), the only MINAR(1) model dealing with counts with more than two dimensions, for comparison. However, this work assumes that the binomial thinning matrix only has diagonal components. Their fitted log-likelihood is shown in Table 6.

Table 7. The parameter estimates for SSMP based on Algorithm (3) with $N = 21$ replicates (the standard deviations are shown in parentheses).

	type A	*type B*	*type C*
μ_i	3.361 (0.0006)	3.170 (0.0007)	2.830 (0.0007)
ϕ_{ij}	0.315 (0.0015)	−0.043 (0.0037)	0.182 (0.0035)
	−0.019 (0.0015)	0.579 (0.0028)	0.108 (0.0024)
	0.154 (0.0011)	0.246 (0.0029)	0.258 (0.0022)
σ_{ij}	0.532 (0.0009)		
	0.124 (0.0007)	0.154 (0.0009)	
	0.272 (0.0007)	0.138 (0.0009)	0.361 (0.0015)

From the parameter estimates in Table 7 and Table S.1, we computed the implied estimates for the moments of the unconditional distribution of the count data for every fitted model and compared them with their sample counterparts as shown in the left part of Table 6. We can see that for SSMP, its calculated moments and overdispersion closely match the sample counterparts, indicating that SSMP provides good representations of the cross-correlations and autocorrelations of the count data. Although MACP has poor fitted log-likelihood values, it has almost as good moment results as SSMP. This good result is due to the use of the Gaussian copula, which successfully captures the left correlation structures in addition to the conditional model. As to LP, it can still describe the cross-correlations satisfactorily; however, it cannot describe the autocorrelations at all. As for MINAR(1), it fails to provide convincing fitting results for any statistic, which is mainly due to its prohibition of overdispersion and off-diagonal autocorrelations. We also compared the prediction power of these fitted models in terms of DS scores as well as the MSPE as shown in Table 6, where SSMP gives the smallest DS score and MSPE.

We further dignose the models. If a model is well-specified, its normalized Pearson residuals for every dimension should have a zero mean and unit variance and be serially uncorrelated.

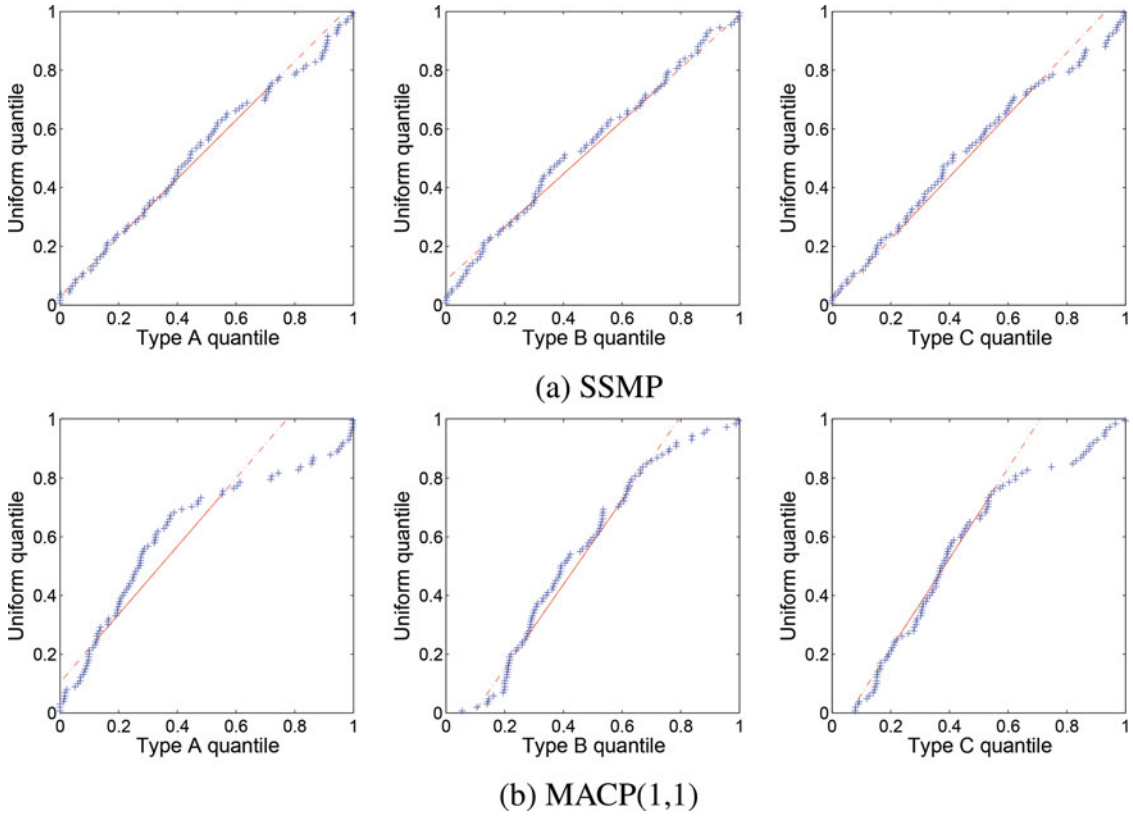


Figure 5. Quantile-quantile plots of the normalized PIT residuals for SSMP and MACP(1, 1). The red lines are the quantiles of the uniform distribution, and the blue marked dots are the quantiles of the sorted PIT residuals.

From the bottom part of Table 6, we can conclude that SSMP and MACP(1, 1) better describe the count data compared with LP and MINAR(1). For SSMP, in every dimension its residual mean is close to zero and its variance is close to one. Its lag 1 autocorrelation matrix has almost zero values. The Ljung–Box statistics for the residuals and their substantial p-values also demonstrate their serial independence to some degree. For MACP(1, 1), its residuals also have almost zero mean and unit variance; however, they tend to be slightly autocorrelated with the Ljung–Box statistic Q_3 for the second dimension bigger than the p- value of the 95% confidence interval. For the other two models, the results are even worse, with their residuals significantly autocorrelated. To further check the distributional assumptions of SSMP and MACP(1, 1), we use the "randomized" version of the Probability Integral Transform (PIT) proposed by Liesenfeld *et al.* (2008) for diagnosis (Jung *et al.*, 2011). If a model is correctly specified, its randomized PIT values should follow the standard uniform distribution. Figure 5 shows the corresponding quantile—quantile plots of the randomized PIT values for SSMP and MACP(1, 1). The PIT values of SSMP in every dimension nearly coincide with the 45° line, indicating their similarity to the uniform distribution. The formal Kolmogorov–Smirnov (KS) test for every dimension also does not reject the uniformity assumption, although the plots of MACP(1, 1) derive more from the 45° line, with the KS tests rejecting the uniformity assumption. All in all, these results show that SSMP provides a much better description of the dynamic interactions of the count data than the other three models.

5. Concluding remarks

Although multivariate time series of counts are very common in practice, models to describe them allowing for flexible cross-correlation and autocorrelation structures are yet to be addressed. To fill this gap, this article proposes an easy-to-interpret state space model to describe autocorrelated multivariate counts. This model can represent the contemporaneous and serial correlations of counts in a flexible way and also capture the overdispersion. Hence, this model provides a useful framework for multivariate count series analysis. A stable and efficient estimation procedure for this model is provided based on the MCEM algorithm together with PF and PS methods. PF can also track the latent states of the model accurately in a sequential way at the cost of a small computational complexity. Comparisons with other state-of-the-art models of multivariate counts demonstrate the superiority and more generality of our proposed model. This point is also illustrated by applying the proposed model to a real dataset from the power utility industry.

Along this research direction, we can next explore the following aspects. First, athough efficient Statistical Process Control (SPC) for autocorrelated multivariate counts is in high demand, to the best of our knowledge, there is no work targeted on it. Our proposed model may shed light on this field by constructing an SPC scheme to monitor model parameters. Second, in some applications count data involve spatial information and are further spatially correlated. It will be interesting to extend the current model by taking the spatial interdependence of count data into consideration; i.e., construct a multi-layer time

series model that aims to analyze not only the lead-lag relations within and between different time series but also those within and between different spatial layers. Last but not least, from the implementable point of view, it is reasonable to consider how to deal with missing data or how to add regression covariates into the model for a better explanation of count data.

Acknowledgment

The authors are grateful for the numerous valuable comments provided by the editors and referees.

Funding

Nan Chen was partially supported by Singapore AcRF Funding R-266-000-085-112 and National Research Foundation Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE).

Notes on contributors

Chen Zhang is a Ph.D candidate in the Department of Industrial and Systems Engineering at the National University of Singapore. She received her B.Eng. degree in Electronic Science and Technology (Optics) from Tianjin University. Her research interests include developing new approaches for modeling and monitoring of engineering systems with complex data. She is a member of IIE and INFORMS.

Nan Chen is an Assistant Professor in the Department of Industrial and Systems Engineering at National University of Singapore. He obtained his B.S. degree in Automation from Tsinghua University, M.S. degree in Computer Science, M.S. degree in Statistics, and Ph.D. degree in Industrial Engineering from the University of Wisconsin–Madison. His research interests include statistical modeling and surveillance of complex systems, simulation modeling and design, condition monitoring, and degradation modeling. He is a member of INFORMS, IIE, and IEEE.

Zhiguo Li is a Research Staff Member in statistics and data science with IBM T. J. Watson Research Center. He holds a Ph.D. in Industrial Engineering and an M.S. in Statistics from the University of Wisconsin–Madison and an M.S. and a B.S. in Automotive Engineering from Tsinghua University. He has many years of experience in industrial statistics, data mining, reliability and quality engineering, and system dynamics with applications to different engineering and business processes/systems including product management/supply chain, energy and utilities, printing systems, medical imaging systems, automobiles, manufacturing processes, and computer networks. He is a member of INFORMS, IIE, and IEEE.

References

Aitchison, J. and Ho, C. (1989) The multivariate Poisson-log normal distribution. *Biometrika*, **76**(4), 643–653.

Arbous, A.G. and Kerrich, J. (1951) Accident statistics and the concept of accident-proneness. *Biometrics*, **7**(4), 340–432.

Bengtsson, T., Bickel, P., Li, B., et al. (2008) Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems, in *Probability and Statistics: Essays in Honor of David A. Freedman*, pp. 316–334, Institute of Mathematical Statistics, Bethesda, MD.

Beskos, A., Crisan, D., Jasra, A., Kamatani, K. and Zhou, Y. (2014b) A stable particle filter in high-dimensions. *arXiv preprint arXiv:1412.3501*.

Billheimer, D., Guttorp, P. and Fagan, W.F. (2001) Statistical interpretation of species composition. *Journal of the American Statistical Association*, **96**(456), 1205–1214.

Brun, O., Teuliere, V. and Garcia, J.-M. (2002) Parallel particle filtering. *Journal of Parallel and Distributed Computing*, **62**(7), 1186–1202.

Chan, K. and Ledolter, J. (1995) Monte Carlo em estimation for time series models involving counts. *Journal of the American Statistical Association*, **90**(429), 242–252.

Chen, N., Li, Z. and Ou, Y. (2015) Multivariate exponentially weighted moving-average chart for monitoring Poisson observations. *Journal of Quality Technology*, **47**(3), 252–263.

Cox, D.R. and Isham, V. (1980) *Point Processes*, volume 12, CRC Press, Boca Raton, FL.

Czado, C., Gneiting, T. and Held, L. (2009) Predictive model assessment for count data. *Biometrics*, **65**(4), 1254–1261.

Davis, R.A., Dunsmuir, W. and Wang, Y. (1999) Modeling time series of count data. *Statistics Textbooks and Monographs*, **158**, 63–114.

Doucet, A., De Freitas, N. and Gordon, N. (2001) *An Introduction to Sequential Monte Carlo Methods*, Springer, New York, NY.

Doucet, A., Godsill, S. and Andrieu, C. (2000) On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, **10**(3), 197–208.

Durbin, J. and Koopman, S.J. (2000) Time series analysis of non-Gaussian observations based on state space models from both classical and Bayesian perspectives. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **62**(1), 3–56.

Durham, G. and Geweke, J. (eds), (2011) Massively parallel sequential Monte Carlo for Bayesian inference. Available at: http://www.censoc.uts.edu.au/pdfs/geweke_papers/gp_working_9.pdf. [14 November 2011]

Ferland, R.E., Latour, A. and Oraichi, D. (2006) Integer-valued GARCH process. *Journal of Time Series Analysis*, **27**(6), 923–942.

Fokianos, K. (2012) Count time series models, in T. Subba Rao, S. Subba Rao, & C. R. Rao (eds), *Time Series–Methods and Applications, Handbook of Statistics*, pp. 315–347, Elsevier, BV, Amsterdam.

Genest, C. and Nešlehová, J. (2007) A primer on copulas for count data. *Astin Bulletin*, **37**(2), 475–515.

Harvey, A.C. and Fernandes, C. (1989) Time series models for count or qualitative observations. *Journal of Business & Economic Statistics*, **7**(4), 407–417.

Heinen, A. and Rengifo, E. (2007) Multivariate autoregressive modeling of time series count data using copulas. *Journal of Empirical Finance*, **14**(4), 564–583.

Holgate, P. (1964) Estimation for the bivariate Poisson distribution. *Biometrika*, **51**(1–2), 241–287.

Hürzeler, M. and Künsch, H.R. (1998) Monte Carlo approximations for general state-space models. *Journal of Computational and Graphical Statistics*, **7**(2), 175–193.

Jørgensen, B., Lundbye-Christensen, S., Song, P.-K. and Sun, L. (1999) A state space model for multivariate longitudinal count data. *Biometrika*, **86**(1), 169–181.

Jung, R.C., Kukuk, M. and Liesenfeld, R. (2006) Time series of count data: Modeling, estimation and diagnostics. *Computational Statistics & Data Analysis*, **51**(4), 2350–2364.

Jung, R.C. and Liesenfeld, R. (2001) Estimating time series models for count data using efficient importance sampling. *Advances in Statistical Analysis*, **4**(85), 387–407.

Jung, R.C., Liesenfeld, R. and Richard, J.-F. (2011) Dynamic factor models for multivariate count data: An application to stock-market trading activity. *Journal of Business & Economic Statistics*, **29**(1), 73–85.

Karlis, D. (2003) An EM algorithm for multivariate Poisson distribution and related models. *Journal of Applied Statistics*, **30**(1), 63–77.

Karlis, D. and Meligkotsidou, L. (2005) Multivariate Poisson regression with covariance structure. *Statistics and Computing*, **15**(4), 255–265.

Karlis, D. and Meligkotsidou, L. (2007) Finite mixtures of multivariate Poisson distributions with application. *Journal of Statistical Planning and Inference*, **137**(6), 1942–1960.

Kocherlakota, S. and Kocherlakota, K. (1992) *Bivariate Discrete Distribution*, Wiley, New York, NY.

Kuk, A.Y. and Cheng, Y.W. (1997) The Monte Carlo Newton–Raphson algorithm. *Journal of Statistical Computation and Simulation*, **59**(3), 233–250.

Latour, A. (1997) The multivariate GINAR (*p*) process. *Advances in Applied Probability*, **29**(1), 228–248.

Liesenfeld, R., Nolte, I. and Pohlmeier, W. (2008) *Modelling Financial Transaction Price Movements: A Dynamic Integer Count Data Model*, Springer, New York, NY.

Liu, J.S. (2008) *Monte Carlo Strategies in Scientific Computing*, Springer Science & Business Media, New York, NY.
Liu, J.S. and Chen, R. (1998) Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, **93**(443), 1032–1044.
Nelson, J.F. (1985) Multivariate gamma-Poisson models. *Journal of the American Statistical Association*, **80**(392), 828–834.
Nikoloulopoulos, A.K. and Karlis, D. (2009) Modeling multivariate count data using copulas. *Communications in Statistics-Simulation and Computation*, **39**(1), 172–187.
Paul, M., Held, L. and Toschke, A.M. (2008) Multivariate modelling of infectious disease surveillance data. *Statistics in Medicine*, **27**(29), 6250–6267.
Pedeli, X. and Karlis, D. (2013a) On composite likelihood estimation of a multivariate INAR (1) model. *Journal of Time Series Analysis*, **34**(2), 206–220.
Pedeli, X. and Karlis, D. (2013b) Some properties of multivariate INAR(1) processes. *Computational Statistics & Data Analysis*, **67**, 213–225.
Sarabia, J.M. and Gómez-Déniz, E. (2011) Multivariate Poisson-beta distributions with applications. *Communications in Statistics–Theory and Methods*, **40**(6), 1093–1108.
Schön, T.B., Wills, A. and Ninness, B. (2011) System identification of nonlinear state-space models. *Automatica*, **47**(1), 39–49.
Smith, A., Doucet, A., de Freitas, N. and Gordon, N. (2013) *Sequential Monte Carlo Methods in Practice*, Springer Science & Business Media, New York, NY.
Song, X. (2000) Multivariate dispersion models generated from Gaussian copula. *Scandinavian Journal of Statistics*, **27**(2), 305–320.
Steyn, H. (1976) On the multivariate Poisson normal distribution. *Journal of the American Statistical Association*, **71**(353), 233–236.
Weiß, C.H. (2008) Thinning operations for modeling time series of counts—A survey. *Advances in Statistical Analysis*, **92**(3), 319–341.
Zeger, S.L. (1988) A regression model for time series of counts. *Biometrika*, **75**(4), 621–629.

Appendixes

A.1 The moment properties of SSMP

A.11. Mean

The mean of $\mathbf{Y}_t$ is

$$\mathrm{E}(\mathbf{Y}_t) = \mathrm{E}\left[\mathrm{E}(\mathbf{Y}_t|\mathbf{X}_t)\right] = \mathrm{E}[\exp(\mathbf{X}_t)] = \exp\left(\boldsymbol{\mu} + \frac{1}{2}\Gamma\right), \quad \text{(A1)}$$

with

$$\mathrm{E}(Y_{ti}) = \exp\left(\mu_i + \frac{1}{2}\Gamma_{ii}\right). \quad \text{(A2)}$$

A.12. Covariance matrix

The variance of $\mathbf{Y}_t$ is

$$\begin{aligned}\mathrm{Var}\left[\mathbf{Y}_t\right] &= \mathrm{E}\left[\mathrm{Var}(\mathbf{Y}_t|\mathbf{X}_t)\right] + \mathrm{Var}\left[\mathrm{E}(\mathbf{Y}_t|\mathbf{X}_t)\right] \\ &= \mathrm{E}\left[\mathrm{diag}(\exp(\mathbf{X}_t))\right] + \mathrm{Var}\left[\exp(\mathbf{X}_t)\right]; \quad \text{(A3)}\end{aligned}$$

therefore,

$$\begin{aligned}\mathrm{Var}(Y_{ti}) &= \mathrm{E}(Y_{ti}) + \mathrm{e}^{2\mu_i+\Gamma_{ii}}(\mathrm{e}^{\Gamma_{ii}} - 1), \\ \mathrm{Cov}(Y_{ti}, Y_{tj}) &= \mathrm{e}^{\mu_i+\mu_j+\frac{1}{2}(\Gamma_{ii}+\Gamma_{jj})}(\mathrm{e}^{\Gamma_{ij}} - 1).\end{aligned}$$

A.13. Autocorrelation structure

The autocorrelation of $\mathbf{Y}_t$ is

$$\begin{aligned}\mathrm{Cov}(\mathbf{Y}_t, \mathbf{Y}_{t-\tau}) &= \mathrm{E}\left[\mathrm{Cov}(\mathbf{Y}_t, \mathbf{Y}_{t-\tau})|\mathbf{X}_t, \mathbf{X}_{t-\tau}\right] \\ &\quad + \mathrm{Cov}\left[\mathrm{E}(\mathbf{Y}_t|\mathbf{X}_t), \mathrm{E}(\mathbf{Y}_{t-\tau}|\mathbf{X}_{t-\tau})\right] \\ &= \mathrm{Cov}(\exp(\mathbf{X}_t), \exp(\mathbf{X}_{t-\tau})), \quad \text{(A4)}\end{aligned}$$

because of

$$\mathrm{E}\left[\mathrm{Cov}(\mathbf{Y}_t, \mathbf{Y}_{t-\tau})|\mathbf{X}_t, \mathbf{X}_{t-\tau}\right] = 0,$$

then we have

$$\begin{aligned}\mathrm{Cov}(Y_{ti}, Y_{(t-\tau)j}) &= \mathrm{Cov}(\exp(X_{ti}), \exp(X_{(t-\tau)j})) \\ &= \mathrm{E}[\exp(X_{ti})\exp(X_{(t-\tau)j})] \\ &\quad - \mathrm{E}[\exp(X_{ti})]\mathrm{E}[\exp(X_{(t-\tau)j})] \\ &= \mathrm{E}[\exp(X_{ti} + X_{(t-\tau)j}))] \\ &\quad - \mathrm{E}[\exp(X_{ti})]\mathrm{E}[\exp(X_{(t-\tau)j})] \\ &= \exp(\mu_i + \mu_j) \\ &\quad \times \left\{\mathrm{E}[\exp(X_{it} - \mu_i + X_{(t-\tau)j} - \mu_j)]\right. \\ &\quad \left. - \exp\left(\frac{\Gamma_{ii}}{2} + \frac{\Gamma_{jj}}{2}\right)\right\}. \quad \text{(A5)}\end{aligned}$$

Since

$$\begin{aligned}\mathbf{X}_t - \boldsymbol{\mu} &= \Phi(\Phi(\mathbf{X}_{t-2} - \boldsymbol{\mu}) + \boldsymbol{\epsilon}_{t-1}) + \boldsymbol{\epsilon}_t \\ &= \Phi^\tau(\mathbf{X}_{t-\tau} - \boldsymbol{\mu}) + \Phi^{\tau-1}\boldsymbol{\epsilon}_{t-\tau+1} + \cdots + \Phi^1\boldsymbol{\epsilon}_{t-1} + \boldsymbol{\epsilon}_t,\end{aligned}$$

we define $\boldsymbol{\phi}_i^{\tau-k} = [\phi_{i,1}^{\tau-k}, \ldots, \phi_{i,d}^{\tau-k}]'$ as a row vector which represents the i^{th} row of $\Phi^{\tau-k}$. Then

$$\begin{aligned}X_{ti} + X_{t-\tau j} - \mu_i - \mu_j &= \boldsymbol{\phi}_i^\tau(\mathbf{X}_{t-\tau} - \boldsymbol{\mu}) + \boldsymbol{\phi}_i^{\tau-1}\boldsymbol{\epsilon}_{t-\tau+1} + \cdots \\ &\quad + \boldsymbol{\phi}_i^1\boldsymbol{\epsilon}_{t-1} + \mathbf{e}_i\boldsymbol{\epsilon}_t + X_{t-\tau j}\end{aligned}$$

We define $\boldsymbol{\phi}_i^{*\tau} = [\phi_{i,1}^\tau, \ldots, \phi_{i,j}^\tau + 1, \ldots, \phi_{i,d}^\tau]'$, then

$$\begin{aligned}X_{ti} + X_{t-\tau j} - \mu_i - \mu_j &= \boldsymbol{\phi}_i^{*\tau}(\mathbf{X}_{t-\tau} - \boldsymbol{\mu}) + \boldsymbol{\phi}_i^{\tau-1}\boldsymbol{\epsilon}_{t-\tau+1} + \cdots \\ &\quad + \boldsymbol{\phi}_i^1\boldsymbol{\epsilon}_{t-1} + \mathbf{e}_i\boldsymbol{\epsilon}_t,\end{aligned}$$

and, consequently,

$$\begin{aligned}\mathrm{E}[\exp(X_{ti} + X_{(t-\tau)j} - \mu_i - \mu_j)] &= \mathrm{E}[\exp(\boldsymbol{\phi}_i^{*\tau}(\mathbf{X}_{t-\tau} - \boldsymbol{\mu}))] \\ &\quad \times \mathrm{E}[\exp(\boldsymbol{\phi}_i^{\tau-1}\boldsymbol{\epsilon}_{t-\tau+1})] \\ &\quad \times \mathrm{E}[\exp(\boldsymbol{\phi}_i^1\boldsymbol{\epsilon}_{t-1})] \\ &\quad \times \mathrm{E}[\exp(\epsilon_{i,t})].\end{aligned}$$

According to the moment-generating function of the multivariate normal distribution, if $\mathbf{X} \sim \mathbf{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then the moment-generating function is given by $m_{\mathbf{x}}(\mathbf{t}) = \mathrm{E}\{\exp(\mathbf{t}'\mathbf{X})\} =$

$\exp(\mathbf{t}'\boldsymbol{\mu} + \frac{1}{2}\mathbf{t}'\Sigma\mathbf{t})$. Then we have

$$\begin{aligned}
&\mathrm{E}\left[\exp(X_{ti} + X_{(t-\tau)j} - \mu_i - \mu_j)\right] \\
&= \exp\left(\frac{1}{2}\boldsymbol{\phi}_i^{*\tau'}\Gamma(0)\boldsymbol{\phi}_i^{*\tau} + \frac{1}{2}\boldsymbol{\phi}_i^{\tau-1'}\Sigma\boldsymbol{\phi}_i^{\tau-1} + \cdots \right. \\
&\quad \left. + \frac{1}{2}\boldsymbol{\phi}_i^{1'}\Sigma\boldsymbol{\phi}_i^{1} + \frac{1}{2}\Sigma_{ii}\right). \qquad \text{(A6)}
\end{aligned}$$

Plug Equation (A5) into Equation (A6). Finally, we have the covariance of Y_{ti} and $Y_{t-\tau j}$. Then the autocorrelation could be obtained by

$$\rho(Y_{ti}, Y_{(t-\tau)j}) = \frac{\mathrm{Cov}(Y_{ti}, Y_{(t-\tau)j})}{\sqrt{\mathrm{Var}(Y_{ti})\mathrm{Var}(Y_{(t-\tau)j}))}}. \qquad \text{(A7)}$$

A.2 The MCEM algorithm based on PF & PS methods

For the M step, to get

$$\boldsymbol{\Theta}^* = \arg\max_{\boldsymbol{\Theta}} \mathcal{Q}(\boldsymbol{\Theta}^*, \boldsymbol{\Theta}), \qquad \text{(A8)}$$

we need take the gradient of $\hat{\mathcal{Q}}(\boldsymbol{\Theta}^*, \boldsymbol{\Theta})$ with respect to $\boldsymbol{\Theta}^*$; that is,

$$\frac{\partial\hat{\mathcal{Q}}(\boldsymbol{\Theta}^*, \boldsymbol{\Theta})}{\partial\boldsymbol{\Theta}^*} = \sum_{t=1}^{T-1}\sum_{i=1}^{N_p}\sum_{j=1}^{N_p} W_{t,t+1|T}^{ij}\frac{\partial\log p_{\boldsymbol{\Theta}^*}(\mathbf{x}_{t+1}^j|\mathbf{x}_t^i)}{\partial\boldsymbol{\Theta}^*}. \qquad \text{(A9)}$$

By rewriting Equation (10) as

$$\mathbf{X}_{t+1} = \boldsymbol{\Phi}\mathbf{X}_t + (\mathbf{I} - \boldsymbol{\Phi})\boldsymbol{\mu} + \boldsymbol{\epsilon}_t = \boldsymbol{\Phi}\mathbf{X}_t + \mathbf{c} + \boldsymbol{\epsilon}_t, \qquad \text{(A10)}$$

where $\mathbf{c} = (\mathbf{I} - \boldsymbol{\Phi})\boldsymbol{\mu}$ and dropping the constant, we have

$$\begin{aligned}
\log p_{\boldsymbol{\Theta}^*}(\mathbf{X}_{t+1}|\mathbf{X}_t) &= -\log|\boldsymbol{\Sigma}^*| - \frac{1}{2}(\mathbf{X}_{t+1} - \boldsymbol{\Phi}^*\mathbf{X}_t - \mathbf{c}^*)'\boldsymbol{\Sigma}^{*-1} \\
&\quad \times(\mathbf{X}_{t+1} - \boldsymbol{\Phi}^*\mathbf{X}_t - \mathbf{c}^*) \\
&= -\log|\boldsymbol{\Sigma}^*| - \frac{1}{2}(\mathbf{X}_{t+1} - \boldsymbol{\Pi}^*\mathbf{Z}_t)'\boldsymbol{\Sigma}^{*-1} \\
&\quad \times(\mathbf{X}_{t+1} - \boldsymbol{\Pi}^*\mathbf{Z}_t) \\
&= -\log|\boldsymbol{\Sigma}^*| - \frac{1}{2}\mathrm{tr} \\
&\quad \times((\mathbf{X}_{t+1} - \boldsymbol{\Pi}^*\mathbf{Z}_t)'\boldsymbol{\Sigma}^{*-1} \\
&\quad \times(\mathbf{X}_{t+1} - \boldsymbol{\Pi}^*\mathbf{Z}_t)), \qquad \text{(A11)}
\end{aligned}$$

where $\boldsymbol{\Pi}^* = [\mathbf{c}^*, \boldsymbol{\Phi}^*]'$, and $\mathbf{Z}_t = [1, \mathbf{X}_t]$.

Taking the derivative of $\log p_{\boldsymbol{\Theta}^*}(\mathbf{X}_{t+1}|\mathbf{X}_t)$, we have

$$\begin{aligned}
\frac{\partial\log p_{\boldsymbol{\Theta}^*}(\mathbf{X}_{t+1}|\mathbf{X}_t)}{\partial\boldsymbol{\Pi}^*} &= \boldsymbol{\Sigma}^{*-1}(\mathbf{X}_{t+1}\mathbf{Z}_t' - \boldsymbol{\Pi}^*\mathbf{Z}_t\mathbf{Z}_t'), \\
\frac{\partial\log p_{\boldsymbol{\Theta}^*}(\mathbf{X}_{t+1}|\mathbf{X}_t)}{\partial\boldsymbol{\Sigma}^{*-1}} &= \frac{\boldsymbol{\Sigma}^*}{2} - \frac{1}{2}(\mathbf{X}_{t+1} - \boldsymbol{\Pi}^*\mathbf{Z}_t) \\
&\quad \times(\mathbf{X}_{t+1} - \boldsymbol{\Pi}^*\mathbf{Z}_t)'. \qquad \text{(A12)}
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\frac{\partial\hat{\mathcal{Q}}(\boldsymbol{\Theta}^*, \boldsymbol{\Theta})}{\partial\boldsymbol{\Pi}^*} &= \boldsymbol{\Sigma}^{*-1}\sum_{t=1}^{T-1}\sum_{i=1}^{N_p}\sum_{j=1}^{N_p} W_{t,t+1|T}^{ij} \\
&\quad \times\left(\mathbf{x}_{t+1}^j - \boldsymbol{\Pi}^*\mathbf{z}_t^i\right)\mathbf{z}_t^{i'} = 0, \\
\frac{\partial\hat{\mathcal{Q}}(\boldsymbol{\Theta}^*, \boldsymbol{\Theta})}{\partial\boldsymbol{\Sigma}^{-1*}} &= \frac{T-1}{2}\boldsymbol{\Sigma}^* - \frac{1}{2}\sum_{t=1}^{T-1}\sum_{i=1}^{N_p}\sum_{j=1}^{N_p} W_{t,t+1|T}^{ij} \\
&\quad \times\left(\mathbf{x}_{t+1}^j - \boldsymbol{\Pi}^*\mathbf{z}_t^i\right)\left(\mathbf{x}_{t+1}^j - \boldsymbol{\Pi}^*\mathbf{z}_t^i\right)' = 0, \qquad \text{(A13)}
\end{aligned}$$

with its solution

$$\begin{aligned}
\boldsymbol{\Pi}^* = [(\mathbf{I} - \boldsymbol{\Phi}^*)\boldsymbol{\mu}^*, \boldsymbol{\Phi}^*]' &= \left(\sum_{t=1}^{T-1}\sum_{i=1}^{N_p}\sum_{j=1}^{N_p} W_{t,t+1|T}^{ij}\mathbf{x}_{t+1}^j\mathbf{z}_t^{i'}\right) \\
&\quad \times\left(\sum_{t=1}^{T-1}\sum_{i=1}^{N_p}\sum_{j=1}^{N_p} W_{t,t+1|T}^{ij}\mathbf{z}_t^i\mathbf{z}_t^{i'}\right)^{-1}, \qquad \text{(A14)}
\end{aligned}$$

and

$$\begin{aligned}
\boldsymbol{\Sigma}^* &= \frac{1}{T-1}\Big(\sum_{t=1}^{T-1}\sum_{i=1}^{N_p}\sum_{j=1}^{N_p} W_{t,t+1|T}^{ij}\left(\mathbf{x}_{t+1}^j - \boldsymbol{\Pi}^*\mathbf{z}_t^i\right) \\
&\quad \times\left(\mathbf{x}_{t+1}^j - \boldsymbol{\Pi}^*\mathbf{z}_t^i\right)'. \qquad \text{(A15)}
\end{aligned}$$