Time series model fitting via Kalman smoothing and EM estimation in TimeModels.jl

Gord Stephen

Last updated: January 2016

Contents

1 Introduction

1.1 Motivation and Acknowledgements

While the expectation-maximization (EM) optimization approach is well-recognized in time-series analysis texts as a useful method for obtaining local maximum-likelihood (ML) parameter estimates of linear state-space model system matrices, avoiding the need to resort to general gradient-free optimization techniques, the procedure is generally only described as a means to generate maximum likelihood estimates of full system matrices.

In many practical cases, however, most elements of a state-space model's system matrices are set deterministically, with only a small subset requiring estimation (one notable example of such a case is the fitting of ARIMA models). In such situations, allowing all elements of the system matrix to vary during the fitting process is clearly undesirable.

A common solution to this challenge is to simply disregard analytical knowledge of the system and fall back on gradient-free nonlinear optimization techniques to compute maximum-likelihood estimates for the unknown parameters. This can be avoided, however, by parametrizing system matrices as reshaped linear transformations of smaller parameter vectors, then applying the EM estimation algorithm to the resulting system equations to generate ML estimates for the parameter vectors alone. This model fitting approach was first outlined by Holmes and implemented as the MARSS R package (`https://cran.r-project.org/web/packages/MARSS/index.html`).

This document re-derives these techniques and describes their implementation in the TimeModels.jl Julia package (`https://github.com/JuliaStats/TimeModels.jl`). The results presented here are based very heavily on Holmes' original derivations (available at `https://cran.r-project.org/web/packages/MARSS/vignettes/EMDerivation.pdf`), with some gaps in derivation steps filled, and changes to notation and the treatment of external system inputs that will be more familiar to readers approaching state-space modelling from a control-theory background.

Like TimeModels.jl, this document is very much a work in progress. In particular, general contraints for solvable systems (while usually alluded to in the various derivations) are not yet detailed explicitly. The reader is instead referred to section 10 of Holmes' derivations, which provides a relevant summary of such constraints, albeit with the different notation and external input expressions mentioned above. In some cases where derivations are identical to those given by Holmes (notational differences aside), only the final results are given.

1.2 Notation

We define a general multivariate state-space process with Gaussian noise according to the following notation:

$$x_t = A_{t-1}x_{t-1} + B_{t-1}u_{t-1} + v_{t-1},\ v_{t-1} \sim \mathcal{N}(0, V_{t-1}),\ t = 2 \dots n$$

$$y_t = C_t x_t + D_t u_t + w_t,\ w_t \sim \mathcal{N}(0, W_t),\ t = 1 \ldots n$$
$$x_1 \sim \mathcal{N}(x_1^0, P_1^0)$$

Here, $x_t \in \mathbb{R}^{n_x}$ is an unobserved state vector, $y_t \in \mathbb{R}^{n_y}$ is an observation vector with possible missing values, and $u_t \in \mathbb{R}^{n_u}$ is a deterministic input vector. A_t is the state transition matrix, B_t is the control input matrix, and V_t is the state transition noise covariance matrix. C_t is the observation matrix, D_t the feed-forward matrix, and W_t the observation noise covariance matrix. Each system matrix can be time-varying. x_1^0 and P_1^0 are initial state value and covariance conditions needed to initiate the recursive process evolution.

2 Kalman Smoothing

Given the observation time series y (possibly containing missing values), the input time series u, and values for the system matrices, we can applying Kalman smoothing to compute the probabilistic distribution of the latent state time series x.

After Kalman smoothing, the estimated latent state time series values will be distributed as $x_t \sim \mathcal{N}(\hat{x}_t, P_t)$. This distribution is obtained by taking into account all observations in y. Obtaining these estimates also requires computing and storing a *predicted* state distribution $x_{t+1} \sim \mathcal{N}(x_{t+1}^t, P_{t+1}^t)$, based only on observations $y_1 \ldots y_t$.

2.1 Prediction and Filtering

x_{t+1}^t, P_{t+1}^t, the Kalman gain matrix K_t, and the marginal log-likelihood δ_{llt} of the observed series given the system matrices can be computed via forward recursion starting from initial conditions x_1^0, P_1^0:

$$\epsilon_t = y_t - C_t x_t^{t-1} - D_t u_t$$
$$\Sigma_t = C_t P_t^{t-1} C_t^T + W_t$$
$$K_t = A_t P_t^{t-1} C_t^T \Sigma_t^{-1}$$
$$\delta_{llt} = \epsilon_t^T \Sigma_t^{-1} \epsilon_t + \ln |\Sigma_t|$$
$$x_{t+1}^t = A_t x_t^{t-1} + B_t u_t + K_t \epsilon_t$$
$$P_{t+1}^t = A_t P_t^{t-1} (A_t - K_t C_t)^T + V_t$$

2.2 Smoothing

Now, given x_t^{t-1}, P_t^{t-1}, K_t, and Σ_t^{-1}, smoothed values $\hat{x}_t$ and P_t can be computed via backwards recursion, with initial conditions $r_n = \mathbf{0}_{n_x}$ and $N_n = \mathbf{0}_{n_x,n_x}$:

$$
\begin{aligned}
L_t &= A_t - K_t C_t \\
r_{t-1} &= C_t^T \Sigma_t^{-1} \epsilon_t + L_t^T r_t \\
N_{t-1} &= C_t^T \Sigma_t^{-1} C_t + L_t^T N_t L_t \\
\hat{x}_t &= x_t^{t-1} + P_t^{t-1} r_{t-1} \\
P_t &= P_t^{t-1} - P_t^{t-1} N_{t-1} P_t^{t-1}
\end{aligned}
$$

2.3 Lag-1 Covariance Smoother

The EM parameter estimation procedure described below also requires $P_{t,t-1}$, the lag-1 state covariance between x_t and x_{t-1}. This can be computed by smoothing a corresponding "stacked" state space model given as follows:

$$
\tilde{x}_t = \begin{bmatrix} x_t \\ x_{t-1} \end{bmatrix}
$$

$$
\tilde{A}_t = \begin{bmatrix} A_t & \mathbf{0}_{n_x,n_x} \\ \mathbf{I}_{n_x} & \mathbf{0}_{n_x,n_x} \end{bmatrix}, \tilde{B}_t = \begin{bmatrix} B_t \\ \mathbf{0}_{n_x,n_u} \end{bmatrix}, \tilde{V}_t = \begin{bmatrix} V_t & \mathbf{0}_{n_x,n_x} \\ \mathbf{0}_{n_x,n_x} & \mathbf{0}_{n_x,n_x} \end{bmatrix}
$$

$$
\tilde{C}_t = \begin{bmatrix} C_t & \mathbf{0}_{n_y,n_x} \end{bmatrix}
$$

$$
\begin{aligned}
\tilde{x}_t &= \tilde{A}_{t-1} \tilde{x}_{t-1} + \tilde{B}_{t-1} u_{t-1} + \tilde{v}_{t-1}, \tilde{v}_{t-1} \sim \mathcal{N}(0, \tilde{V}_{t-1}) \\
y_t &= \tilde{C}_t \tilde{x}_t + D_t u_t + w_t, w_t \sim \mathcal{N}(0, W_t)
\end{aligned}
$$

The resulting $\tilde{x}_t$ state covariance $\tilde{P}_t$ will be a block matrix of the form:

$$
\tilde{P}_t = \begin{bmatrix} P_t & P_{t,t-1} \\ P_{t,t-1} & P_{t-1} \end{bmatrix}
$$

from which $P_{t,t-1}$ can be easily extracted.

3 Linear Constraint Parametrization

3.1 System Matrix Decomposition

To parametrize and estimate the time-dependent system matrices, we first factor them into components that are either time-independent and parametrized, or time-dependent and explicitly defined:

$$A_t = A_{1t}A_2|_{p_A}A_{3t}, B_t = B_{1t}B_2|_{p_B}, V_t = G_tQ|_{p_Q}G_t^T$$
$$C_t = C_{1t}C_2|_{p_C}C_{3t}, D_t = D_{1t}D_2|_{p_D}, W_t = H_tR|_{p_R}H_t^T$$

where p_A, p_B, p_Q, p_C, p_D, and p_R are parameter vectors to be estimated. We require that $Q|_{p_Q}$ and $R|_{p_R}$ be invertible in addition to being valid covariance matrices (symmetric and positive-semidefinite).

3.2 Linear Constraint Formulation

We formalize the linear constraints on the parametrized matrices as:

$$\mathrm{vec}(Z|_{p_Z}) = f_Z + D_Z p_Z$$

where f_Z and D_Z represent the constant and parameter coefficient terms making up Z.

3.3 System Equation Representation

We define $\Phi_t = (G_t^T G_t)^{-1} G_t^T$ such that:

$$\Phi_{t-1}x_t = \Phi_{t-1}A_{t-1}x_{t-1} + \Phi_{t-1}B_{t-1}u_{t-1} + \Phi_{t-1}G_{t-1}q_{t-1}$$
$$\Phi_{t-1}x_t = \Phi_{t-1}A_{t-1}x_{t-1} + \Phi_{t-1}B_{t-1}u_{t-1} + q_{t-1}$$

Using Kronecker product identities, we can now represent the general system equations in terms of the parameters to be estimated:

$$\mathrm{vec}(\Phi_{t-1}x_t) = \mathrm{vec}(\Phi_{t-1}A_{t-1}x_{t-1}) + \mathrm{vec}(\Phi_{t-1}B_{t-1}u_{t-1}) + \mathrm{vec}(q_{t-1})$$

Using $Ab = \mathrm{vec}(Ab) = (b^T \otimes \mathbf{I}_{n_a})\mathrm{vec}(A)$, where A is an $n_a \times n_b$ matrix and b is a length-n_b vector:

$$\Phi_{t-1}x_t = \Phi_{t-1}\mathrm{vec}(A_{t-1}x_{t-1}) + \Phi_{t-1}\mathrm{vec}(B_{t-1}u_{t-1}) + q_{t-1}$$

$$\Phi_{t-1}x_t = \Phi_{t-1}(x_{t-1}^T \otimes \mathbf{I}_{n_x})\text{vec}(A_{t-1}) + \Phi_{t-1}(u_{t-1}^T \otimes \mathbf{I}_{n_x})\text{vec}(B_{t-1}) + q_{t-1}$$
$$\Phi_{t-1}x_t = \Phi_{t-1}(x_{t-1}^T \otimes \mathbf{I}_{n_x})\text{vec}(A_{1t-1}A_2|_{p_A}A_{3t-1}) + \Phi_{t-1}(u_{t-1}^T \otimes \mathbf{I}_{n_x})\text{vec}(B_{1t-1}B_2|_{p_B}) + q_{t-1}$$

Using $\text{vec}(ABC) = (C^T \otimes A)\text{vec}(B)$:

$$\Phi_{t-1}x_t = \Phi_{t-1}(x_{t-1}^T \otimes \mathbf{I}_{n_x})(A_{3t-1}^T \otimes A_{1t-1})\text{vec}(A_2|_{p_A}) + \Phi_{t-1}(u_{t-1}^T \otimes \mathbf{I}_{n_x})\text{vec}(B_{1t-1}B_2|_{p_B}) + q_{t-1}$$

Using $\text{vec}(AB) = (\mathbf{I}_{n_c} \otimes A)\text{vec}(B)$ where B is an $n_b \times n_c$ matrix::

$$\Phi_{t-1}x_t = \Phi_{t-1}(x_{t-1}^T \otimes \mathbf{I}_{n_x})(A_{3t-1}^T \otimes A_{1t-1})\text{vec}(A_2|_{p_A}) + \Phi_{t-1}(u_{t-1}^T \otimes \mathbf{I}_{n_x})(\mathbf{I}_{n_u} \otimes B_{1t-1})\text{vec}(B_2|_{p_B}) + q_{t-1}$$

Using the linear constraint definition $\text{vec}(Z|_{p_Z}) = f_Z + D_Z p_Z$:

$$\Phi_{t-1}x_t = \Phi_{t-1}(x_{t-1}^T \otimes \mathbf{I}_{n_x})(A_{3t-1}^T \otimes A_{1t-1})(f_A + D_A p_A) + \Phi_{t-1}(u_{t-1}^T \otimes \mathbf{I}_{n_x})(\mathbf{I}_{n_u} \otimes B_{1t-1})(f_B + D_B p_B) + q_{t-1}$$

Finally, we can consolidate the deterministic elements by defining

$$\begin{aligned}
f_{A,t-1} &= (A_{3t-1}^T \otimes A_{1t-1})f_A \\
D_{A,t-1} &= (A_{3t-1}^T \otimes A_{1t-1})D_A \\
f_{B,t-1} &= (\mathbf{I}_{n_u} \otimes B_{1t-1})f_B \\
D_{B,t-1} &= (\mathbf{I}_{n_u} \otimes B_{1t-1})D_B
\end{aligned}$$

which results in

$$\Phi_{t-1}x_t = \Phi_{t-1}(x_{t-1}^T \otimes \mathbf{I}_{n_x})(f_{A,t-1} + D_{A,t-1}p_A) + \Phi_{t-1}(u_{t-1}^T \otimes \mathbf{I}_{n_x})(f_{B,t-1} + D_{B,t-1}p_B) + q_{t-1}$$

The second system equation follows from the same process and gives:

$$\begin{aligned}
\Xi_t &= (H_t^T H_t)^{-1}H_t^T \\
f_{C,t} &= (C_{3t}^T \otimes C_{1t})f_C \\
D_{C,t} &= (C_{3t}^T \otimes C_{1t})D_C \\
f_{D,t} &= (\mathbf{I}_{n_u} \otimes D_{1t})f_B \\
D_{D,t} &= (\mathbf{I}_{n_u} \otimes D_{1t})D_B
\end{aligned}$$
$$\Xi_t y_t = \Xi_t(x_t^T \otimes \mathbf{I}_{n_y})(f_{C,t} + D_{C,t}p_C) + \Xi_t(u_t^T \otimes \mathbf{I}_{n_y})(f_{D,t} + D_{D,t}p_D) + r_t$$

The parametrized initial state and error covariances are given similarly, as follows:

$$x_1 = x_1^0 + \zeta,\ \zeta \sim \mathcal{N}(0, P_1^0),\ P_1^0 = JSJ^\top$$
$$x_1 = x_1^0 + Js,\ s \sim \mathcal{N}(0, S)$$

Introducing the usual linear parameter constraints we get

$$x_1^0 = f_{x_1^0} + D_{x_1^0} p_{x_1^0},\ S = f_S + D_S p_S$$

Like Q and R, S must be both invertible and a valid covariance matrix. Finally, defining $\Pi = (J^\top J)^{-1} J^\top$ gives

$$\Pi x_1 = \Pi x_1^0 + s = \Pi(f_{x_1^0} + D_{x_1^0} p_{x_1^0}) + s$$

4 Expectation-Maximization Parameter Estimation

4.1 Basic Log-likelihood Derivation

Given the parametrized system equations and taking errors to be normally-distributed, the log-likelihood of the states and observations given the matrix parameters can be represented:

$$\epsilon_t = \Phi_{t-1} x_t - \Phi_{t-1}(x_{t-1}^T \otimes \mathbf{I}_{n_x})(f_{A,t-1} + D_{A,t-1} p_A) - \Phi_{t-1}(u_{t-1}^T \otimes \mathbf{I}_{n_x})(f_{B,t-1} + D_{B,t-1} p_B)$$
$$\epsilon_t = \Phi_{t-1}(x_t - (x_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{A,t-1} - (x_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{A,t-1} p_A - (u_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{B,t-1} - (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1} p_B)$$

$$f_t = x_t - (x_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{A,t-1} - (u_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{B,t-1}$$
$$a_t = (x_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{A,t-1}$$
$$b_t = (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1}$$
$$\epsilon_t = \Phi_{t-1}(f_t - a_t p_A - b_t p_B)$$

$$\eta_t = \Xi_t y_t - \Xi_t(x_t^T \otimes \mathbf{I}_{n_y})(f_{C,t} + D_{C,t} p_C) - \Xi_t(u_t^T \otimes \mathbf{I}_{n_y})(f_{D,t} + D_{D,t} p_D)$$
$$\eta_t = \Xi_t(y_t - (x_t^T \otimes \mathbf{I}_{n_y}) f_{C,t} - (x_t^T \otimes \mathbf{I}_{n_y}) D_{C,t} p_C - (u_t^T \otimes \mathbf{I}_{n_y}) f_{D,t} - (u_t^T \otimes \mathbf{I}_{n_y}) D_{D,t} p_D)$$

$$g_t = y_t - (x_t^T \otimes \mathbf{I}_{n_y}) f_{C,t} - (u_t^T \otimes \mathbf{I}_{n_y}) f_{D,t}$$
$$c_t = (x_t^T \otimes \mathbf{I}_{n_y}) D_{C,t}$$
$$d_t = (u_t^T \otimes \mathbf{I}_{n_y}) D_{D,t}$$

$$\eta_t = \Xi_t(g_t - c_t p_C - d_t p_D)$$

$$\xi = \Pi x_1 - \Pi(f_{x_1^0} + D_{x_1^0} p_{x_1^0})$$

$$\text{LL} = -\frac{1}{2}\sum_{t=2}^{n} \epsilon_t^\top Q_{t-1}^{-1}\epsilon_t - \frac{1}{2}\sum_{t=2}^{n}\ln|Q_{t-1}| - \frac{1}{2}\sum_{t=1}^{n}\eta_t^\top R_t^{-1}\eta_t - \frac{1}{2}\sum_{t=1}^{n}\ln|R_t| - \frac{1}{2}\xi^\top S^{-1}\xi - \frac{1}{2}\ln|S| - \frac{n}{2}\ln 2\pi$$

For this and the following sections, it will be useful to define $\tilde{V}_{t-1}^{-1} = \Phi_{t-1}^\top Q_{t-1}^{-1}\Phi_{t-1}$, $\tilde{W}_t^{-1} = \Xi_t^\top R_t^{-1}\Xi_t$, and $\tilde{P_1^0}^{-1} = \Pi^\top S^{-1}\Pi$. Expanding the above gives:

$$\begin{aligned}
-\text{LL} = \frac{1}{2}\sum_{t=2}^{n}(f_t + a_t p_A + b_t p_B)^T \tilde{V}_{t-1}^{-1}(f_t + a_t p_A + b_t p_B) + \frac{1}{2}\sum_{t=2}^{n}\ln|Q_{t-1}| \\
+\frac{1}{2}\sum_{t=1}^{n}(g_t + c_t p_C + d_t p_D)^T \tilde{W}_t^{-1}(g_t + c_t p_C + d_t p_D) + \frac{1}{2}\sum_{t=1}^{n}\ln|R_t| \\
+\frac{1}{2}(f_{x_1^0} + D_{x_1^0} p_{x_1^0})^\top \tilde{P_1^0}^{-1}(f_{x_1^0} + D_{x_1^0} p_{x_1^0}) + \frac{1}{2}\ln|S| + \frac{n}{2}\ln 2\pi
\end{aligned}$$

4.2 Log-likelihood Derivation with Zero-variance States or Observations

We also need to account for the possibility that some state or observation rows are deterministically-defined (with G_t or H_t having a one or more all-zero-valued rows), and thus do not make any probabilistic contribution to the log-likelihood calculation and cannot be estimated via maximum-likelihood methods. While the values of elements of G_t and H_t can vary in time, it is assumed their overall rows will remain consistently either all-zero or not-all-zero. There is additional nuance needed in considering that errors introduced in non-deterministic state or observation rows have the potential to propagate to seemingly-deterministic rows via the A and C matrices. Holmes provides the full details of this derivation (along with explanations for associated estimation limitations) and so for brevity only the broad strokes are presented here.

Given an $n_x \times n_x$ selection matrix $\mathbf{I}_t^d$ that zeros out any rows that are all-zero in G_t, the fully-deterministic states x_2^d can be expressed as:

$$x_2^d = \mathbf{I}_2^d x_2 = \mathbf{I}_2^d(A_1 x_1 + B_1 u_1)$$
$$x_2^d = \mathbf{I}_2^d(A_1 x_1 + (u_1^T \otimes \mathbf{I}_{n_x}) f_{B,1} + (u_1^T \otimes \mathbf{I}_{n_x}) D_{B,1} p_B)$$

Similarly:

$$x_3^d = \mathbf{I}_3^d(A_2 x_2 + (u_2^T \otimes \mathbf{I}_{n_x}) f_{B,2} + (u_2^T \otimes \mathbf{I}_{n_x}) D_{B,2} p_B)$$

Now, recursing on x_2:

$$x_3^d = \mathbf{I}_3^d(A_2(A_1x_1 + (u_1^T \otimes \mathbf{I}_{n_x})f_{B,1} + (u_1^T \otimes \mathbf{I}_{n_x})D_{B,1}p_B) + (u_2^T \otimes \mathbf{I}_{n_x})f_{B,2} + (u_2^T \otimes \mathbf{I}_{n_x})D_{B,2}p_B)$$

$$x_3^d = \mathbf{I}_3^d(A_2A_1x_1 + (u_2^T \otimes \mathbf{I}_{n_x})f_{B,2} + A_2(u_1^T \otimes \mathbf{I}_{n_x})f_{B,1} + ((u_2^T \otimes \mathbf{I}_{n_x})D_{B,2} + A_2(u_1^T \otimes \mathbf{I}_{n_x})D_{B,1})p_B)$$

More generally, x_t^d can be expressed via the recursion relations

$$x_t^d = \mathbf{I}_t^d(A_{t-1}^* x_1^0 + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B)$$
$$A_t^* = A_t A_{t-1}^*, \; A_0^* = \mathbf{I}_{n_x}$$
$$f_{Bu,t}^* = (u_t^\top \otimes \mathbf{I}_{n_x})f_{B,t} + A_t f_{Bu,t-1}^*, f_{Bu,0}^* = 0$$
$$D_{Bu,t}^* = (u_t^\top \otimes \mathbf{I}_{n_x})D_{B,t} + A_t D_{Bu,t-1}^*, D_{Bu,0}^* = 0$$

With this recursion we will also need to account for the probabilistic and non-deterministic components of the expectation of x_1. Specifically, defining $I_{x_1^0}^d$ as a selection matrix zeroing-out non-deterministic rows in x_1^0 (corresponding to non-zero rows in J):

$$\hat{x}_1^0 = (\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d x_1^0 = (\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d(f_{x_1^0} + D_{x_1^0}p_{x_1^0})$$

We can now consider the general log-likelihood formula with deterministic rows set to zero. Φ_{t-1} will automatically zero out likelihood contributions from deterministic x_t rows in the process evolution terms, while Ξ_t will do the same for y_t in the observation terms and Π for the initial state x_1^0. We can apply the x_t^d relation derived above to replace x_{t-1} and x_t in the process and observation terms respectively.

$$\epsilon_t = \Phi_{t-1}(x_t - A_{t-1}x_{t-1} - B_{t-1}u_{t-1})$$
$$\epsilon_t = \Phi_{t-1}(x_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)x_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d x_{t-1} - B_{t-1}u_{t-1})$$
$$\epsilon_t = \Phi_{t-1}(x_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)x_{t-1} - A_{t-1}x_{t-1}^d - B_{t-1}u_{t-1})$$
$$\epsilon_t = \Phi_{t-1}(x_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)x_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d(A_{t-2}^* x_1^0 + f_{Bu,t-2}^* + D_{Bu,t-2}^* p_B) - B_{t-1}u_{t-1})$$

$$\eta_t = \Xi_t(y_t - C_t x_t - D_t u_t)$$
$$\eta_t = \Xi_t(y_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)x_t - C_t\mathbf{I}_t^d x_t - D_t u_t)$$
$$\eta_t = \Xi_t(y_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)x_t - C_t x_t^d - D_t u_t)$$
$$\eta_t = \Xi_t(y_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)x_t - C_t\mathbf{I}_t^d(A_{t-1}^* x_1^0 + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B) - D_t u_t)$$

$$\xi = \Pi(x_1 - f_{x_1^0} - D_{x_1^0}p_{x_1^0})$$

$$-\text{LL} = \frac{1}{2}\sum_{t=2}^{n}\epsilon_t^T Q_{t-1}^{-1}\epsilon_t + \frac{1}{2}\sum_{t=2}^{n}\ln|Q_{t-1}| + \frac{1}{2}\sum_{t=1}^{n}\eta_t^T R_t^{-1}\eta_t + \frac{1}{2}\sum_{t=1}^{n}\ln|R_t|$$
$$+\frac{1}{2}\xi^\top S^{-1}\xi + \frac{1}{2}\ln|S| + \frac{n}{2}\ln 2\pi$$

From this point we can compute derivatives with respect to to-be-estimated parameter vectors, and set the expected values (expectation) to zero (maximization). Solving for all parameters and using them to recompute expected values gives a new model with improved fit log-likelihood. This process can be repeated iteratively until the log-likelihood value converges sufficiently.

4.3 Solving for p_A

Same for finite-variance rows, doesn't work for fixed rows. Final result is:

$$p_A = \Big(\sum_{t=1}^{n} D_{A,t}^\top(\widehat{x_{t-1}x_{t-1}^\top}\otimes\tilde{Q}_t)D_{A,t}\Big)^{-1}\sum_{t=1}^{n} D_{A,t}^\top\big(\text{vec}(\tilde{Q}_t\widehat{x_t x_{t-1}^\top}) - (\widehat{x_{t-1}x_{t-1}^\top}\otimes\tilde{Q}_t)f_{A,t} - \text{vec}(\tilde{Q}_t B_t u_t \hat{x}_{t-1}^\top)\big)$$

4.4 Solving for p_B

We start by taking the derivative of negative log-likelihood with respect to p_B. Using the matrix derivative rule $\frac{\delta}{\delta a}a^\top C a = 2a^\top C$:

$$\frac{\delta(-\text{LL})}{\delta p_B} = \frac{1}{2}\sum_{t=2}^{n}\frac{\delta}{\delta p_B}\epsilon_t^T Q^{-1}\epsilon_t + \frac{1}{2}\sum_{t=1}^{n}\frac{\delta}{\delta p_B}\eta_t^T R^{-1}\eta_t$$

$$\frac{\delta(-\text{LL})}{\delta p_B} = \sum_{t=2}^{n}\epsilon_t^T Q^{-1}\frac{\delta}{\delta p_B}\epsilon_t + \sum_{t=1}^{n}\eta_t^T R^{-1}\frac{\delta}{\delta p_B}\eta_t$$

$$\frac{\delta\epsilon_t}{\delta p_B} = -\Phi_{t-1}(A_{t-1}\mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T\otimes\mathbf{I}_{n_x})D_{B,t-1})$$
$$\frac{\delta\eta_t}{\delta p_B} = -\Xi_t C_t\mathbf{I}_t^d D_{Bu,t-1}^*$$

$$\frac{\delta(-\text{LL})}{\delta p_B} = -\sum_{t=2}^{n}\epsilon_t^T Q^{-1}\Phi_{t-1}(A_{t-1}\mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T\otimes\mathbf{I}_{n_x})D_{B,t-1}) - \sum_{t=1}^{n}\eta_t^T R^{-1}\Xi_t C_t\mathbf{I}_t^d D_{Bu,t-1}^*$$

$$\frac{\delta(-\text{LL})}{\delta p_B} = -\sum_{t=2}^{n}\big(x_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)x_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d(A_{t-2}^* x_1 + f_{Bu,t-2}^* + D_{Bu,t-2}^* p_B)$$

$$-(u_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{B,t-1} - (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1} p_B\big)^T \tilde{V}_t^{-1} (A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1})$$

$$-\sum_{t=1}^{n} \big(y_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d) x_t - C_t \mathbf{I}_t^d (A_{t-1}^* x_1 + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B) - D_t u_t\big)^T \tilde{W}_t^{-1} C_t \mathbf{I}_t^d D_{Bu,t-1}^*$$

Taking expectations and setting $E\Big[\frac{\delta(-\mathrm{LL})}{\delta p_B}\Big] = 0$:

$$0 = -\sum_{t=2}^{n} \big(\hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d) x_{t-1} - A_{t-1} \mathbf{I}_{t-1}^d (A_{t-2}^* \hat{x}_1^0 + f_{Bu,t-2}^* + D_{Bu,t-2}^* p_B)$$

$$-(u_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{B,t-1} - (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1} p_B\big)^T \tilde{V}_t^{-1} (A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1})$$

$$-\sum_{t=1}^{n} \big(\hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d) x_t - C_t \mathbf{I}_t^d (A_{t-1}^* \hat{x}_1^0 + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B) - D_t u_t\big)^T \tilde{W}_t^{-1} C_t \mathbf{I}_t^d D_{Bu,t-1}^*$$

$$0 = -\sum_{t=2}^{n} \big(\hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d) x_{t-1} - A_{t-1} \mathbf{I}_{t-1}^d (A_{t-2}^* \hat{x}_1^0 + f_{Bu,t-2}^*)$$

$$-(u_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{B,t-1}\big)^T \tilde{V}_t^{-1} (A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1})$$

$$+\sum_{t=2}^{n} \big(A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* p_B + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1} p_B\big)^T \tilde{V}_t^{-1} (A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1})$$

$$-\sum_{t=1}^{n} \big(\hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d) x_t - C_t \mathbf{I}_t^d (A_{t-1}^* \hat{x}_1^0 + f_{Bu,t-1}^*) - D_t u_t\big)^T \tilde{W}_t^{-1} C_t \mathbf{I}_t^d D_{Bu,t-1}^*$$

$$+\sum_{t=1}^{n} \big(C_t \mathbf{I}_t^d D_{Bu,t-1}^* p_B\big)^T \tilde{W}_t^{-1} C_t \mathbf{I}_t^d D_{Bu,t-1}^*$$

$$0 = -\sum_{t=2}^{n} \big(\hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d) x_{t-1} - A_{t-1} \mathbf{I}_{t-1}^d (A_{t-2}^* \hat{x}_1^0 + f_{Bu,t-2}^*)$$

$$-(u_{t-1}^T \otimes \mathbf{I}_{n_x}) f_{B,t-1}\big)^T \tilde{V}_t^{-1} (A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1})$$

$$+p_B^\top \sum_{t=2}^{n} \big(A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1}\big)^\top \tilde{V}_t^{-1} (A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1})$$

$$-\sum_{t=1}^{n} \big(\hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d) x_t - C_t \mathbf{I}_t^d (A_{t-1}^* \hat{x}_1^0 + f_{Bu,t-1}^*) - D_t u_t\big)^T \tilde{W}_t^{-1} C_t \mathbf{I}_t^d D_{Bu,t-1}^*$$

$$+p_B^\top \sum_{t=1}^{n} \big(C_t \mathbf{I}_t^d D_{Bu,t-1}^*\big)^T \tilde{W}_t^{-1} C_t \mathbf{I}_t^d D_{Bu,t-1}^*$$

For clarity we can define:

$$\Delta_{1,t} = A_{t-1} \mathbf{I}_{t-1}^d D_{Bu,t-2}^* + (u_{t-1}^T \otimes \mathbf{I}_{n_x}) D_{B,t-1}$$

$$\Delta_{2,t} = \hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)x_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d(A_{t-2}^*\hat{x}_1^0 + f_{Bu,t-2}^*) - (u_{t-1}^T \otimes \mathbf{I}_{n_x})f_{B,t-1}$$

$$\Delta_{3,t} = C_t\mathbf{I}_t^d D_{Bu,t-1}^*$$

$$\Delta_{4,t} = \hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)x_t - C_t\mathbf{I}_t^d(A_{t-1}^*\hat{x}_1^0 + f_{Bu,t-1}^*) - D_t u_t$$

This gives:

$$0 = -\sum_{t=2}^{n} \Delta_{2,t}^\top \tilde{V}_t^{-1}\Delta_{1,t} + p_B^\top \sum_{t=2}^{n} \Delta_{1,t}^\top \tilde{V}_t^{-1}\Delta_{1,t} - \sum_{t=1}^{n} \Delta_{4,t}^\top \tilde{W}_t^{-1}\Delta_{3,t} + p_B^\top \sum_{t=1}^{n} \Delta_{3,t}^\top \tilde{W}_t^{-1}\Delta_{3,t}$$

$$p_B = \Big(\sum_{t=2}^{n} \Delta_{1,t}^\top \tilde{V}_t^{-1}\Delta_{1,t} + \sum_{t=1}^{n} \Delta_{3,t}^\top \tilde{W}_t^{-1}\Delta_{3,t}\Big)^{-1}\Big(\sum_{t=2}^{n} \Delta_{1,t}^\top \tilde{V}_t^{-1}\Delta_{2,t} + \sum_{t=1}^{n} \Delta_{3,t}^\top \tilde{W}_t^{-1}\Delta_{4,t}\Big)$$

4.5 Solving for p_Q

The solution for p_Q is not general, but works in a wide range of special cases, most notably when Q is a simple diagonal matrix of parameters, which when multiplied with any arbitrary G_t can serve most purposes. Aside from notational differences, the derivation is identical to the one presented in Holmes, so for brevity only the final result is presented here:

$$p_Q = \Big(\sum_{t=1}^{n} D_{Q,t}^\top D_{Q,t}\Big)^{-1} \sum_{t=1}^{n} D_{Q,t}^\top \mathrm{vec}(S_t)$$

$$S_t = \Phi_t(\widehat{x_t x_t^\top} + \widehat{x_t x_{t-1}^\top} A_{t-1}^\top + A_{t-1}\widehat{x_{t-1} x_t^\top} - \hat{x}_t u_{t-1}^\top B_{t-1}^\top - B_{t-1}u_{t-1}\hat{x}_t^\top + A_{t-1}\widehat{x_{t-1}x_{t-1}^\top}A_{t-1}^\top$$
$$+ A_{t-1}\hat{x}_{t-1}u_{t-1}^\top B_{t-1}^\top + B_{t-1}u_{t-1}\hat{x}_{t-1}^\top A_{t-1}^\top + B_{t-1}u_{t-1}u_{t-1}^\top B_{t-1}^\top)\Phi_t^\top$$

4.6 Solving for p_C

The solution for p_C is analogous to p_A:

$$p_C = \Big(\sum_{t=1}^{n} D_{C,t}^\top (\widehat{x_t x_t^\top} \otimes \tilde{R}_t) D_{C,t}\Big)^{-1} \sum_{t=1}^{n} D_{C,t}^\top \big(\mathrm{vec}(\tilde{R}_t \widehat{y_t x_t^\top}) - (\widehat{x_t x_t^\top} \otimes \tilde{R}_t) f_{C,t} - \mathrm{vec}(\tilde{R}_t D_t u_t \hat{x}_t^\top)\big)$$

4.7 Solving for p_D

Finite-variance rows works as normal, with constraints

This derivation isn't in Holmes:

$$p_D = \Big(\sum_{t=1}^{n} D_{D,t}^\top (u_t^\top \otimes \mathbf{I}_{n_y})^\top \tilde{R}_t (u_t^\top \otimes \mathbf{I}_{n_y}) D_{D,t}\Big)^{-1} \sum_{t=1}^{n} D_{D,t}^\top (u_t^\top \otimes \mathbf{I}_{n_y})^\top \tilde{R}_t \big(\hat{y}_t - C_t \hat{x}_t - (u_t^\top \otimes \mathbf{I}_{n_y}) f_{A,t}\big)$$

4.8 Solving for p_R

p_R is fully analogous to p_Q, and the same matrix property constraints apply.

$$p_R = \Big(\sum_{t=1}^{n} D_{R,t} D_{R,t}^\top\Big)^{-1} \sum_{t=1}^{n} D_{R,t}^\top \mathrm{vec}(T_t)$$

$$\begin{aligned} T_t = \Xi_t (\widehat{y_t y_t^\top} + \widehat{y_t x_t^\top} C_t^\top + C_t \widehat{x_t y_t^\top} - \hat{y}_t u_t^\top D_t^\top - D_t u_t \hat{y}_t{}^\top + C_t \widehat{x_t x_t^\top} C_t^\top \\ + C_t \hat{x}_t u_t^\top D_t^\top + D_t u_t \hat{x}_t{}^\top C_t^\top + D_t u_t u_t^\top D_t^\top) \Xi_t^\top \end{aligned}$$

4.9 Solving for $p_{x_1^0}$

Solving for $p_{x_1^0}$ is particularly messy given how often it appears in the log-likelihood equation! The derivation for the final result is very similar to the one used to solve for p_B:

$$\frac{\delta(-\mathrm{LL})}{\delta p_{x_1^0}} = \frac{1}{2} \sum_{t=2}^{n} \frac{\delta}{\delta p_{x_1^0}} \epsilon_t^T Q_{t-1}^{-1} \epsilon_t + \frac{1}{2} \sum_{t=1}^{n} \frac{\delta}{\delta p_{x_1^0}} \eta_t^T R_t^{-1} \eta_t + \frac{1}{2} \frac{\delta}{\delta p_{x_1^0}} \xi^\top S^{-1} \xi$$

$$\frac{\delta(-\mathrm{LL})}{\delta p_{x_1^0}} = \sum_{t=2}^{n} \epsilon_t^T Q_{t-1}^{-1} \frac{\delta}{\delta p_{x_1^0}} \epsilon_t + \sum_{t=1}^{n} \eta_t^T R_t^{-1} \frac{\delta}{\delta p_{x_1^0}} \eta_t + \xi^\top S^{-1} \frac{\delta}{\delta p_{x_1^0}} \xi$$

$$\frac{\delta}{\delta p_{x_1^0}} \epsilon_t = -\Phi_{t-1} A_{t-1} \mathbf{I}_{t-1}^d A_{t-2}^* \mathbf{I}_{x_1^0}^d D_{x_1^0} \equiv -\Phi_{t-1} \Delta_{5,t}$$

$$\frac{\delta}{\delta p_{x_1^0}} \eta_t = -\Xi_t C_t \mathbf{I}_t^d A_{t-1}^* \mathbf{I}_{x_1^0}^d D_{x_1^0} \equiv -\Xi_t \Delta_{7,t}$$

$$\frac{\delta}{\delta p_{x_1^0}} \xi = -\Pi D_{x_1^0}$$

$$\frac{\delta(-\mathrm{LL})}{\delta p_{x_1^0}} = -\sum_{t=2}^{n} \epsilon_t^\top Q_{t-1}^{-1} \Phi_{t-1} \Delta_{5,t} - \sum_{t=1}^{n} \eta_t^\top R_t^{-1} \Xi_t \Delta_{7,t} - \xi^\top S^{-1} \Pi D_{x_1^0}$$

$$E\Big[\frac{\delta(-\text{LL})}{\delta p_{x_1^0}}\Big] = 0 = -\sum_{t=2}^{n} E[\epsilon_t]^\top Q_{t-1}^{-1}\Phi_{t-1}\Delta_{5,t} - \sum_{t=1}^{n} E[\eta_t]^\top R_t^{-1}\Xi_t\Delta_{7,t} - E[\xi]^\top S^{-1}\Pi D_{x_1^0}$$

$$E[\epsilon_t] = \Phi_{t-1}(\hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)\hat{x}_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d(A_{t-2}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d(f_{x_1^0} + D_{x_1^0}p_{x_1^0})\big) + f_{Bu,t-2}^* + D_{Bu,t-2}^* p_B) - B_{t-1}u_{t-1})$$

$$E[\eta_t] = \Xi_t(\hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)\hat{x}_t - C_t\mathbf{I}_t^d(A_{t-1}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d(f_{x_1^0} + D_{x_1^0}p_{x_1^0})\big) + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B) - D_t u_t)$$

$$E[\xi] = \Pi(\hat{x}_1 - f_{x_1^0} - D_{x_1^0}p_{x_1^0})$$

$$\begin{aligned}
0 = &-\sum_{t=2}^{n}(\hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)\hat{x}_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d(A_{t-2}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d(f_{x_1^0} + D_{x_1^0}p_{x_1^0})\big) + f_{Bu,t-2}^* + \\
&D_{Bu,t-2}^* p_B) - B_{t-1}u_{t-1})^\top \tilde{V}_{t-1}^{-1}\Delta_{5,t} \\
&-\sum_{t=1}^{n}(\hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)\hat{x}_t - C_t\mathbf{I}_t^d(A_{t-1}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d(f_{x_1^0} + D_{x_1^0}p_{x_1^0})\big) + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B) - D_t u_t)^\top \\
&\tilde{W}_t^{-1}\Delta_{7,t} - (\hat{x}_1 - f_{x_1^0} - D_{x_1^0}p_{x_1^0})^\top \tilde{P_1^0}^{-1} D_{x_1^0}
\end{aligned}$$

$$\begin{aligned}
0 = &-\sum_{t=2}^{n}(\hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)\hat{x}_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d(A_{t-2}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d f_{x_1^0}\big) + f_{Bu,t-2}^* + \\
&D_{Bu,t-2}^* p_B) - B_{t-1}u_{t-1})^\top \tilde{V}_{t-1}^{-1}\Delta_{5,t} + \sum_{t=2}^{n}(A_{t-1}\mathbf{I}_{t-1}^d A_{t-2}^*\mathbf{I}_{x_1^0}^d D_{x_1^0}p_{x_1^0})^\top \tilde{V}_{t-1}^{-1}\Delta_{5,t} \\
&-\sum_{t=1}^{n}(\hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)\hat{x}_t - C_t\mathbf{I}_t^d(A_{t-1}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d f_{x_1^0}\big) + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B) - D_t u_t)^\top \tilde{W}_t^{-1}\Delta_{7,t} \\
&+\sum_{t=1}^{n}(C_t\mathbf{I}_t^d A_{t-1}^*\mathbf{I}_{x_1^0}^d D_{x_1^0}p_{x_1^0})^\top \tilde{W}_t^{-1}\Delta_{7,t} \\
&-(\hat{x}_1 - f_{x_1^0})^\top \tilde{P_1^0}^{-1} D_{x_1^0} + (D_{x_1^0}p_{x_1^0})^\top \tilde{P_1^0}^{-1} D_{x_1^0}
\end{aligned}$$

$$\Delta_{6,t} \equiv \hat{x}_t - A_{t-1}(\mathbf{I}_{n_x} - \mathbf{I}_{t-1}^d)\hat{x}_{t-1} - A_{t-1}\mathbf{I}_{t-1}^d(A_{t-2}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d f_{x_1^0}\big) + f_{Bu,t-2}^* + D_{Bu,t-2}^* p_B) - B_{t-1}u_{t-1}$$

$$\Delta_{8,t} \equiv \hat{y}_t - C_t(\mathbf{I}_{n_x} - \mathbf{I}_t^d)\hat{x}_t - C_t\mathbf{I}_t^d(A_{t-1}^*\big((\mathbf{I}_{n_x} - \mathbf{I}_{x_1^0}^d)\hat{x}_1 + \mathbf{I}_{x_1^0}^d f_{x_1^0}\big) + f_{Bu,t-1}^* + D_{Bu,t-1}^* p_B) - D_t u_t$$

$$0 = -\sum_{t=2}^{n} \Delta_{6,t}^\top \tilde{V}_{t-1}^{-1} \Delta_{5,t} + \sum_{t=2}^{n} (\Delta_{5,t} p_{x_1^0})^\top \tilde{V}_{t-1}^{-1} \Delta_{5,t} - \sum_{t=1}^{n} \Delta_{8,t}^\top \tilde{W}_t^{-1} \Delta_{7,t}$$
$$+ \sum_{t=1}^{n} (\Delta_{7,t} p_{x_1^0})^\top \tilde{W}_t^{-1} \Delta_{7,t} - (\hat{x}_1 - f_{x_1^0})^\top \tilde{P_1^0}^{-1} D_{x_1^0} + (D_{x_1^0} p_{x_1^0})^\top \tilde{P_1^0}^{-1} D_{x_1^0}$$

$$\sum_{t=2}^{n} (\Delta_{5,t} p_{x_1^0})^\top \tilde{V}_{t-1}^{-1} \Delta_{5,t} + \sum_{t=1}^{n} (\Delta_{7,t} p_{x_1^0})^\top \tilde{W}_t^{-1} \Delta_{7,t} + D_{x_1^0} + (D_{x_1^0} p_{x_1^0})^\top \tilde{P_1^0}^{-1} D_{x_1^0} =$$
$$\sum_{t=2}^{n} \Delta_{6,t}^\top \tilde{V}_{t-1}^{-1} \Delta_{5,t} + \sum_{t=1}^{n} \Delta_{8,t}^\top \tilde{W}_t^{-1} \Delta_{7,t} + (\hat{x}_1 - f_{x_1^0})^\top \tilde{P_1^0}^{-1} D_{x_1^0}$$

Transposing the equation and isolating $p_{x_1^0}$:

$$\Big(\sum_{t=2}^{n} \Delta_{5,t}^\top \tilde{V}_{t-1}^{-1} \Delta_{5,t} + \sum_{t=1}^{n} \Delta_{7,t}^\top \tilde{W}_t^{-1} \Delta_{7,t} + D_{x_1^0}^\top \tilde{P_1^0}^{-1} D_{x_1^0}\Big) p_{x_1^0} =$$
$$\sum_{t=2}^{n} \Delta_{5,t}^\top \tilde{V}_{t-1}^{-1} \Delta_{6,t} + \sum_{t=1}^{n} \Delta_{7,t}^\top \tilde{W}_t^{-1} \Delta_{8,t} + D_{x_1^0}^\top \tilde{P_1^0}^{-1} (\hat{x}_1 - f_{x_1^0})$$

$$p_{x_1^0} = \Big(\sum_{t=2}^{n} \Delta_{5,t}^\top \tilde{V}_{t-1}^{-1} \Delta_{5,t} + \sum_{t=1}^{n} \Delta_{7,t}^\top \tilde{W}_t^{-1} \Delta_{7,t} + D_{x_1^0}^\top \tilde{P_1^0}^{-1} D_{x_1^0}\Big)^{-1} \times$$
$$\Big(\sum_{t=2}^{n} \Delta_{5,t}^\top \tilde{V}_{t-1}^{-1} \Delta_{6,t} + \sum_{t=1}^{n} \Delta_{7,t}^\top \tilde{W}_t^{-1} \Delta_{8,t} + D_{x_1^0}^\top \tilde{P_1^0}^{-1} (\hat{x}_1 - f_{x_1^0})\Big)$$

4.10 Solving for p_S

p_S can be solved for in a very similar fashion to p_Q and p_R, although this is not done explicitly in Holmes:

$$p_S = (D_S^\top D_S)^{-1} D_S^\top \mathrm{vec}\big(\Pi(\widehat{x_1 x_1^\top} - \hat{x}_1 x_1^{0\top} - x_1^0 \hat{x}_1^\top + x_1^0 x_1^{0\top})\Pi^\top\big)$$