Solving dynamic stochastic general equilibrium models using k-order perturbation: what Dynare does

Willi Mutschler

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Dynare Team

Dynare Model Framework

$$E\left[f_\theta\left(y_{t-1}, y_t, y_{t+1}, u_t \,|\, \Omega_t\right)\right] = 0$$
$$u_s \sim WN(0, \Sigma_u)$$

$t, s \in \mathbb{T}$: discrete time set, typically $\mathbb{N}$ or $\mathbb{Z}$

y_t: n endogenous variables (declared in *var* block)

u_t: n_u exogenous variables (declared in *varexo* block)

Σ_u: covariance matrix of invariant distribution of exogenous variables (declared in *shocks* block)

θ: n_θ model parameters (declared in *parameters* block)

f: n model equations (declared in *model* block)

f_θ is a continuous non-linear function indexed by a vector of parameters θ

$$E\left[f_\theta\left(y_{t-1}, y_t, y_{t+1}, u_t \,|\, \Omega_t\right)\right] = 0$$
$$u_s \sim WN(0,\Sigma_u)$$

Ω_t: information set (*filtration,* i.e. $\Omega_t \subseteq \Omega_{t+s} \forall s \geq 0$)

$E[\,\cdot\,|\,\Omega_t]$: conditional expectation operator

- information set includes model equations f, value of parameters θ, value of current state y_{t-1}, value of current exogenous variables u_t, invariant distribution (but not values!) of future exogenous variables u_{t+s}

- $\Omega_t = \{f, \theta, y_{t-1}, u_t, u_{t+s} \sim N(0,\Sigma)\}$ for all $t \in \mathbb{T}$, $s > 0$

- typically we use shorthand E_t

Dynare Model Framework

$$E_t\left[f\left(y_{t-1}, y_t, y_{t+1}, u_t\right)\right] = 0$$

Typology and Ordering of Variables

Typology and Ordering of Variables

$$E_t \left[f\left(y_{t-1}, y_t, y_{t+1}, u_t \right) \right] = 0$$

- y_t denotes vector of all n endogenous variables

Typology and Ordering of Variables

$$n = n^{static} + n^{pred} + n^{fwrd} + n^{both}$$

static: appear only at t, not at $t-1$, not at $t+1$

predetermined: appear at $t-1$, not at $t+1$, *possibly* at t

forward: appear at $t+1$, not at $t-1$, *possibly* at t

mixed: appear at $t-1$ and $t+1$, *possibly* at t

Typology and Ordering of Variables

y_t^* are the *state variables*: predetermined and mixed variables (n^{spred})

y_t^{**} are the *jumper variables*: mixed and forward variables (n^{sfwrd})

Typology and Ordering of Variables

declaration order: as you declare in *var* block

decision-rule (DR) order: used for perturbation

$$y_t = \begin{pmatrix} static \\ predetermind \\ mixed \\ forward \end{pmatrix} \quad y_t^* = \begin{pmatrix} predetermind \\ mixed \end{pmatrix} \quad y_t^{**} = \begin{pmatrix} mixed \\ forward \end{pmatrix}$$

Typology and Ordering of Variables

$$E_t\left[f\left(y^{*}_{t-1}, y_t, y^{**}_{t+1}, u_t\right)\right] = 0$$

Perturbation Approach

General Idea

first ingredient: perturbation parameter σ

- scale u_t by a parameter $\sigma \geq 0$: $u_t = \sigma \ \eta_t$
- η_t is white noise with contemporaneous k-th order product moments:

$$\Sigma^{(k)} = \mathbb{E}\{\underbrace{\eta_t \otimes \eta_t \otimes \ldots \otimes \eta_t}_{\text{k times}}\}$$

- note that this implies $\Sigma_u^{(k)} = \sigma^k \ \Sigma_\eta^{(k)}$
- σ is called the *perturbation parameter*
    - non-stochastic, i.e. static model: $\sigma = 0$
    - stochastic, i.e. dynamic model: $\sigma > 0$

General Idea

second ingredient: dynamic solution is defined via a policy function

- find an invariant mapping between y_t and (y^*_{t-1}, u_t):

$$y_t = g(y^*_{t-1}, u_t, \sigma)$$

- $g(\cdot)$ is called the *policy-function* or *decision rule*

- $g(\cdot)$ is unknown, i.e. we need to solve a *functional equation*

General Idea

third ingredient: *implicit function theorem*

$$E_t\left[f\left(y^*_{t-1}, y_t, y^{**}_{t+1}, u_t\right)\right] = 0$$

<u>*implicitly*</u> defines

$$g(y^*_{t-1}, u_t, \sigma)$$

General Idea

fourth ingredient: Taylor approximation of order k

- compute the coefficients of a Taylor expansion of $g(y^*_{t-1}, u_t, \sigma)$ from a Taylor expansion of $E_t\left[f\left(y^*_{t-1}, y_t, y^{**}_{t+1}, u_t\right)\right] = 0$
- all evaluated at some known point; mostly non-stochastic steady-state (i.e. $\sigma = 0$)

Notation

$$u := u_t \, , \, u_+ := u_{t+1}$$

$$y_0 := y_t \, , \, y_0^* := y_t^*, \, y_0^{**} := y_t^{**}$$

$$y_- := y_{t-1}, \, y_-^* := y_{t-1}^*, \, y_-^{**} := y_{t-1}^{**}$$

$$y_+ := y_{t+1}, \, y_+^* := y_{t+1}^* \, , \, y_+^{**} := y_{t+1}^{**}$$

$x := y_{t-1}^*$ denotes *previous states*

$\bar{y}, \bar{y}^*, \bar{y}^{**}, \bar{x}$ denote non-stochastic steady-state

$\hat{x} := y_{t-1}^* - \bar{y}^*$ denotes deviation from steady-state

Notation

$$y_0 = g(x, u, \sigma) \qquad y_0^* = g^*(x, u, \sigma) \qquad y_0^{**} = g^{**}(x, u, \sigma)$$

$$y_+^{**} = g^{**}(y_0^*, u_+, \sigma) = g^{**}(g^*(x, u, \sigma), u_+, \sigma) \equiv G(x, u, u_+, \sigma)$$

Notation

dynamic model in terms of x, u, u_+ and σ:

$$f\left(y_-^*, y_0, y_+^{**}, u\right) = f\left(x, g(x, u, \sigma), g^{**}(g^*(x, u, \sigma), u_+, \sigma), u\right)$$

$$= f\left(x, g(x, u, \sigma), G(x, u, \sigma, u_+), u_t\right)$$

$$\equiv F(x, u, u_+, \sigma)$$

Notation

$$r := \begin{pmatrix} x \\ u \\ u_+ \\ \sigma \end{pmatrix}$$ [input vector for F and G]

$$w(r) := \begin{pmatrix} y_0^* \\ u_+ \\ \sigma \end{pmatrix} = \begin{pmatrix} g^*(x, u, \sigma) \\ u_+ \\ \sigma \end{pmatrix}$$ [input vector for g^{**}]

$$z(r) := \begin{pmatrix} y_-^* \\ y_0 \\ y_+^{**} \\ u \end{pmatrix} = \begin{pmatrix} x \\ g(x, u, \sigma) \\ G(x, u, u_+, \sigma) \\ u \end{pmatrix} = \begin{pmatrix} x \\ g(x, u, \sigma) \\ g^{**}(g^*(x, u, \sigma), u_+, \sigma) \\ u \end{pmatrix}$$ [input vector for f]

Objective

we know how to solve for the non-stochastic ($\sigma = 0$) steady-state $\bar{y}$ by solving the *static* model:

$$\bar{f}(\bar{y}) \equiv f(\bar{y}^*, \bar{y}, \bar{y}^{**}, 0) = F(\bar{y}^*, 0, 0, 0) = 0$$

which provides us with the non-stochastic steady-state for $\bar{y}$, $\bar{y}^*$ and $\bar{y}^{**}$

even though we do not know $g(\cdot)$ explicitly, we do know its value at $\bar{y}$, $\bar{y}^*$ and $\bar{y}^{**}$:

$$\bar{y}^* = g^*(\bar{y}^*, 0, 0) \quad \text{and} \quad \bar{y} = g(\bar{y}^*, 0, 0)$$

Objective

use a *k*-order Taylor expansion of *f* to recover the coefficients of the *k*-order Taylor expansion of *g*:

$$y = \bar{y} + g_x \hat{x} + g_u u + g_\sigma \sigma$$

$$+\frac{1}{2} g_{xx}(\hat{x} \otimes \hat{x}) + \frac{2}{2} g_{xu}(\hat{x} \otimes u) + \frac{2}{2} g_{x\sigma}(\hat{x} \otimes \sigma) + \frac{1}{2} g_{uu}(u \otimes u) + \frac{2}{2} g_{u\sigma}(u \otimes \sigma) + \frac{1}{2} g_{\sigma\sigma} \sigma^2$$

$$+\frac{1}{6} g_{xxx}(\hat{x} \otimes \hat{x} \otimes \hat{x}) + \frac{3}{6} g_{xxu}(\hat{x} \otimes \hat{x} \otimes u) + \frac{3}{6} g_{xx\sigma}(\hat{x} \otimes \hat{x})\sigma + \frac{3}{6} g_{xuu}(\hat{x} \otimes u \otimes u) + \frac{6}{6} g_{xu\sigma}(\hat{x} \otimes u)\sigma + \frac{3}{6} g_{x\sigma\sigma} \hat{x} \sigma^2 + \frac{1}{6} g_{uuu}(u \otimes u \otimes u) + \frac{3}{6} g_{uu\sigma}(u \otimes u \otimes)\sigma + \frac{3}{6} g_{u\sigma\sigma} u \sigma^2 + \frac{1}{6} g_{\sigma\sigma\sigma} \sigma^3$$

$$+\frac{1}{24} \cdots$$

Objective

find the coefficients of the k-order Taylor expansion of g:

$$g_{x^q u^p \sigma^{k-q-p}} := \frac{\partial^k g(\bar{x},0,0)}{\underbrace{\partial x \ldots \partial x}_{q \text{ times}} \cdot \underbrace{\partial u \ldots \partial u}_{p \text{ times}} \cdot \underbrace{\partial \sigma \ldots \partial \sigma}_{k-q-p \text{ times}}}$$

where $0 \leq p, q \leq k$ and $0 \leq p + q \leq k$

all evaluated at some known point, mostly the non-stochastic steady-state

Underlying Assumption

- *f* and *g* are *sufficiently differentiable so that the implicit function theorem (or its Banach space generalization) applies*.
- for *f* this assumption is easily checked
- for *g* we typically can only ASSUME that *g* behaves similarly to *f* (logical and credible, but not a formal proof)

Matrix vs Tensor Notation

Matrix vs Tensor Notation

higher-order perturbation

- based on multivariate version of implicit function theorem
- requires use of multidimensional chain rules
- many summation operators
- conventional matrix notation becomes unwieldy at $k > 2$ (unless you know what you're doing)
- tensor notation and Einstein summation notation is more concise (but requires getting used to)

Tensor Notation

- a tensor $\mathscr{A}$ is a multidimensional array, i.e. a collection of numbers, where we use indices $\alpha_j \in [1,...,n_j], 1 \leq j \leq d$, to access the elements in the array

- formally, it is defined as a mapping

$$\mathscr{A} : \{1,...,n_1\} \times \{1,...,n_2\} \times \ldots \times \{1,...,n_d\} \to \mathbb{R}$$
$$(\alpha_1, \alpha_2, \ldots, \alpha_d) \mapsto [\mathscr{A}]_{\alpha_1,\alpha_2,...,\alpha_d}$$

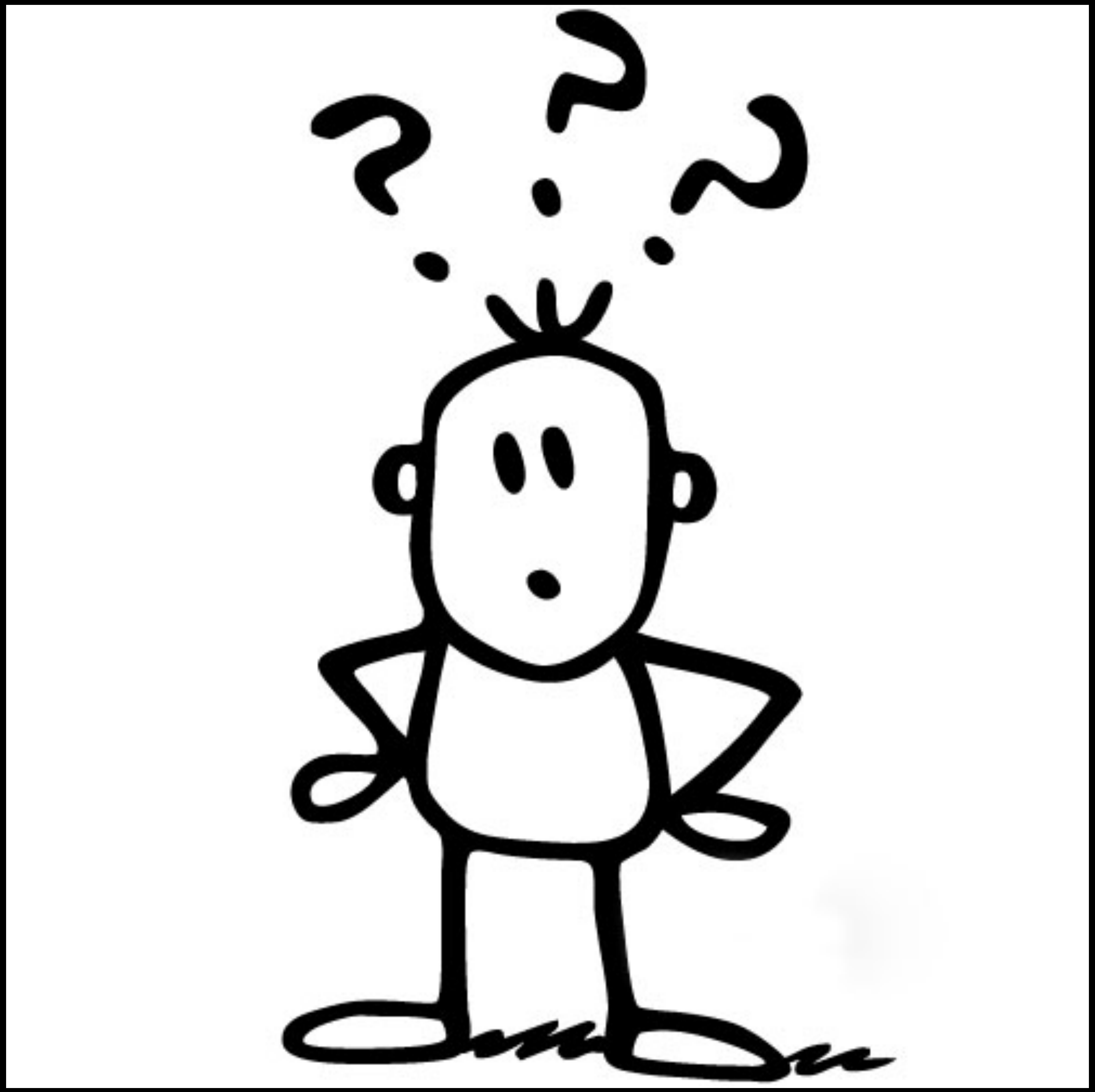
which assigns the real-valued entry $[\mathscr{A}]_{\alpha_1,\alpha_2,...,\alpha_d}$ to each index $(\alpha_1, \alpha_2, \ldots, \alpha_d)$ as function value

Einstein Summation Notation

Einstein summation notation allows to compactly express terms in a multivariate Taylor series expansion

- eliminates the summation symbols by making different use of the location of indices
- same index used first as subscript and then as superscript of two tensors implies summation of the products

Examples



Einstein Summation Notation

example for 1-dimensional tensors $\mathscr{A}$ (of size n) and $\mathscr{B}$ (of size n):

$$[\mathscr{A}]_{\alpha_1}[\mathscr{B}]^{\alpha_1} = \sum_{\alpha_1=1}^{n} [\mathscr{A}]_{\alpha_1}[\mathscr{B}]_{\alpha_1}$$

Einstein Summation Notation

example for 1-dimensional tensors $\mathscr{A}$ (of size n) and $\mathscr{B}$ (of size m), and 2-dimensional tensor $\mathscr{D}$ (of size $n \times m$):

$$[\mathscr{D}]_{\alpha_1\alpha_2}[\mathscr{A}]^{\alpha_1}[\mathscr{B}]^{\alpha_2} = \sum_{\alpha_1=1}^{n}\sum_{\alpha_2=1}^{m}[\mathscr{D}]_{\alpha_1\alpha_2}[\mathscr{A}]_{\alpha_1}[\mathscr{B}]_{\alpha_2}$$

Einstein Summation Notation

example for 2-dimensional tensors $\mathscr{D}$ (of size $n \times m$) and $\mathscr{E}$ (of size $n \times m$):

$$[\mathscr{D}]_{\alpha_1\alpha_2}[\mathscr{E}]^{\alpha_1\alpha_2} = \sum_{\alpha_1=1}^{n}\sum_{\alpha_2=1}^{m}[\mathscr{D}]_{\alpha_1\alpha_2}[\mathscr{E}]_{\alpha_1\alpha_2}$$

Einstein Summation Notation

example for 1-dimensional tensors $\mathscr{A}$ (of size n), $\mathscr{B}$ (of size m) and $\mathscr{C}$ (of size o), 3-dimensional tensor $\mathscr{F}$ (of size $n \times m \times o$):

$$[\mathscr{F}]_{\alpha_1\alpha_2,\alpha_3}[\mathscr{A}]^{\alpha_1}[\mathscr{B}]^{\alpha_2}[\mathscr{C}]^{\alpha_3} = \sum_{\alpha_1=1}^{n}\sum_{\alpha_2=1}^{m}\sum_{\alpha_3=1}^{o}[\mathscr{F}]_{\alpha_1\alpha_2\alpha_3}[\mathscr{A}]_{\alpha_1}[\mathscr{B}]_{\alpha_2}[\mathscr{C}]_{\alpha_3}$$

Einstein Summation Notation

example for 1-dimensional tensor $\mathscr{A}$ (of size n), 2-dimensional tensor $\mathscr{D}$ (of size $m \times o$) and 3-dimensional tensor $\mathscr{F}$ (of size $n \times m \times o$):

$$[\mathscr{F}]_{\alpha_1\alpha_2,\alpha_3}[\mathscr{A}]^{\alpha_1}[\mathscr{D}]^{\alpha_2\alpha_3} = \sum_{\alpha_1=1}^{n}\sum_{\alpha_2=1}^{m}\sum_{\alpha_3=1}^{o}[\mathscr{F}]_{\alpha_1\alpha_2\alpha_3}[\mathscr{A}]_{\alpha_1}[\mathscr{D}]_{\alpha_2\alpha_3}$$

Faà di Bruno's Formula

Faà di Bruno's Formula

- identity in mathematics generalizing the chain rule to higher derivatives
- in Einstein summation notation:

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

$$[G_{r^k}]^\iota_{\boldsymbol{\tau}_k} := \frac{\partial^k \left[G\right]_\iota}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [g^{**}_{w^l}]^\iota_{\boldsymbol{\phi}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [w_{r^{|c_m|}}]^{\phi_m}_{\boldsymbol{\tau}(c_m)}$$

- indices are compressed into bold vectors $\boldsymbol{\tau}_k := \tau_1, \ldots, \tau_k$, $\boldsymbol{\gamma}_l := \gamma_1, \ldots, \gamma_l$ and $\boldsymbol{\phi}_l := \phi_1, \ldots, \phi_l$

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

let $[F]_i$ denote the i-th dynamic model equation, then $[F_{r^k}]^i_{\boldsymbol{\tau}_k}$ is the k-th partial derivative of equation i with respect to variables in r selected by integers $\boldsymbol{\tau}_k$, where $[r]_{\tau_j}$ is the τ_j-th element of r

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k \, [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

let $[f]_i$ denote the i-th dynamic model equation, then $[f_{z^l}]^i_{\boldsymbol{\gamma}_l}$ is the l-th partial derivative of equation i with respect to dynamic variables z indexed by integers $\boldsymbol{\gamma}_l$, where $[z]_{\gamma_j}$ is the γ_j-th element of z:

$$[f_{z^l}]^i_{\boldsymbol{\gamma}_l} := \frac{\partial^l \, [f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}\cdots\partial[z]_{\gamma_l}}$$

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c \in \mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

let $[z]_{\gamma_m}$ denote the γ_m-th dynamic model variable, then $[z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$ is the $|c_m|$-th partial derivative of $[z]_{\gamma_m}$ with respect to variables in r indexed by integers $\boldsymbol{\tau}(c_m)$

$$[z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)} := \frac{\partial^{|c_m|} [z]_{\gamma_m}}{\partial[r]_{\tau(c_1)}\partial[r]_{\tau(c_2)}\cdots\partial[r]_{\tau(c_m)}}$$

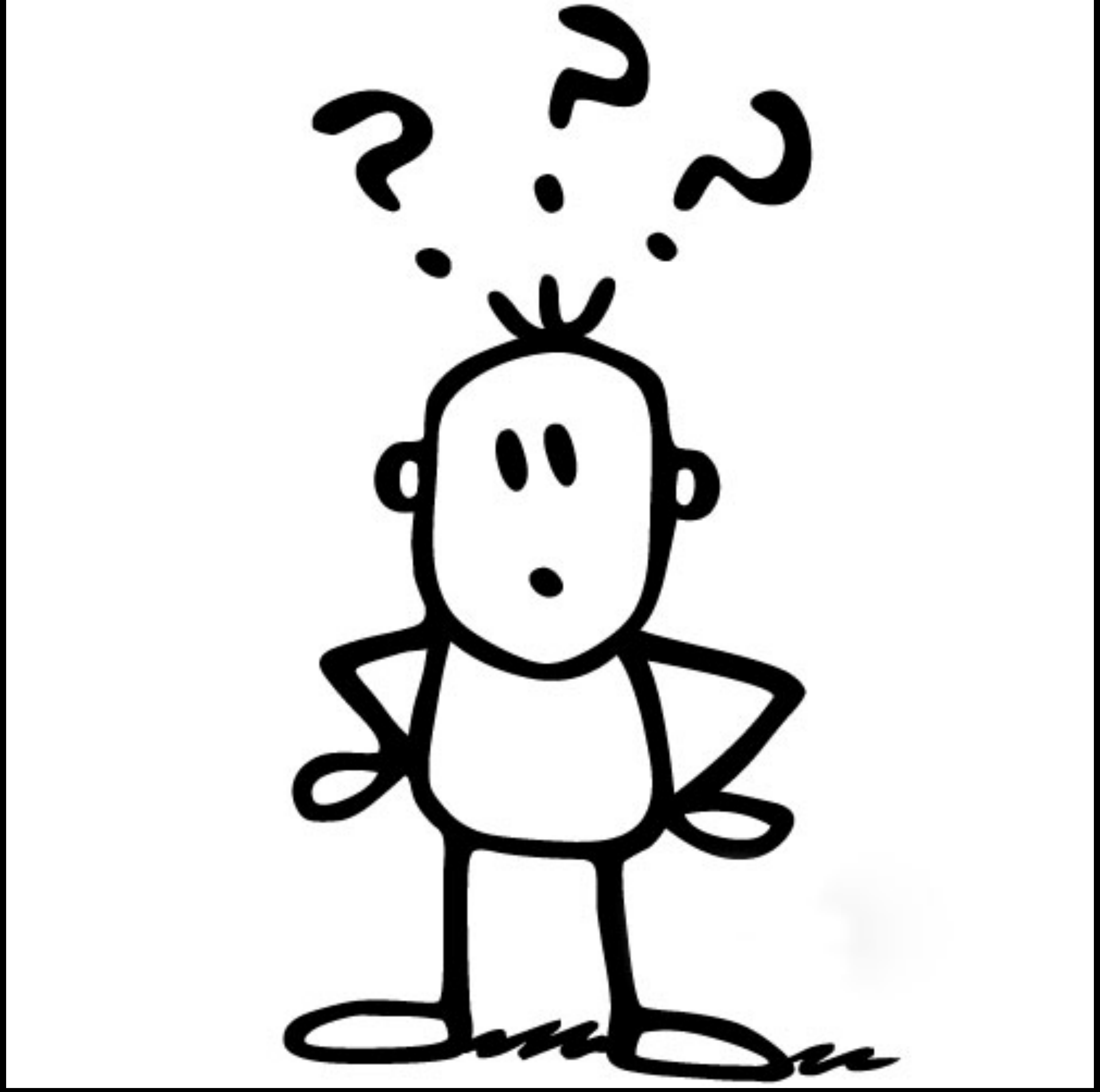
$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

combinatorics

- $\mathcal{M}_{l,k}$ is a so-called equivalence set (or Bell polynomial), which is defined as the set of all partitions c of the set of k indices with l classes
- c_m is the m-th class of partition c, $|c_m|$ its cardinality, and $\boldsymbol{\tau}(c_m)$ is a sequence of τ's indexed by c_m
- note: selection of $\boldsymbol{\tau}(c_m)$ ignores indices in c_m when they correspond to the perturbation parameter σ

example: $\mathcal{M}_{2,3} = \left\{ \underbrace{\{\overbrace{\{1\}}^{c_1}\ \overbrace{\{2,3\}}^{c_2}\}}_{c}\ \underbrace{\{\overbrace{\{2\}}^{c_1}\ \overbrace{\{1,3\}}^{c_2}\}}_{c}, \underbrace{\{\overbrace{\{3\}}^{c_1}\ \overbrace{\{1,2\}}^{c_2}\}}_{c}, \right\}$

Examples



$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

first-order derivative of equation i with respect to the α_1-th state variable:

$$[F_x]^i_{\alpha_1} = [f_z]^i_{\gamma_1}[z_x]^{\gamma_1}_{\alpha_1} = \sum_{\gamma_1=1}^{n_z} \frac{\partial[f]_i}{\partial[z]_{\gamma_1}} \frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}}$$

with $\mathcal{M}_{1,1} = \left\{ \underbrace{\overbrace{\{1\}}^{c_1}}_{c} \right\}$

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

second-order derivative of equation i with respect to the α_1-th state variable and to the β_1-th current shock variable:

$$[F_{xu}]^i_{\alpha_1\beta_1} = [f_z]^i_{\gamma_1}[z_{xu}]^{\gamma_1}_{\alpha_1\beta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}$$

$$= \sum_{\gamma_1=1}^{n_z} \frac{\partial[f]_i}{\partial[z]_{\gamma_1}} \frac{\partial^2[z]_{\gamma_1}}{\partial[x]_{\alpha_1}\partial[u]_{\beta_1}} + \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z} \frac{\partial^2[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}} \frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}} \frac{\partial[z]_{\gamma_2}}{\partial[u]_{\beta_1}}$$

with $\mathcal{M}_{1,2} = \left\{ \underbrace{\overbrace{\{1,2\}}^{c_1}}_{c} \right\}$, $\mathcal{M}_{2,2} = \left\{ \underbrace{\overbrace{\{1\}}^{c_1}, \overbrace{\{2\}}^{c_2}}_{c} \right\}$

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

third-order derivative of equation i with respect to the α_1-th and α_2-th state variables and to the β_1-th current shock variable:

$$[F_{xxu}]^i_{\alpha_1\alpha_2\beta_1} = [f_z]^i_{\gamma_1}[z_{xxu}]^{\gamma_1}_{\alpha_1\alpha_2\beta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}\left([z_x]^{\gamma_1}_{\alpha_1}[z_{xu}]^{\gamma_2}_{\alpha_2\beta_1} + [z_x]^{\gamma_1}_{\alpha_2}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1} + [z_u]^{\gamma_1}_{\beta_1}[z_{xx}]^{\gamma_2}_{\alpha_1\alpha_2}\right) + [f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_x]^{\gamma_1}_{\alpha_1}[z_x]^{\gamma_2}_{\alpha_2}[z_u]^{\gamma_3}_{\beta_1}$$

$$= \sum_{\gamma_1=1}^{n_z} \frac{\partial[f]_i}{\partial[z]_{\gamma_1}} \frac{\partial^3[z]_{\gamma_1}}{\partial[x]_{\alpha_1}\partial[x]_{\alpha_2}\partial[u]_{\beta_1}}$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z} \frac{\partial^2[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}} \left(\frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}}\frac{\partial^2[z]_{\gamma_2}}{\partial[x]_{\alpha_2}\partial[u]_{\beta_1}} + \frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_2}}\frac{\partial^2[z]_{\gamma_2}}{\partial[x]_{\alpha_1}\partial[u]_{\beta_1}} + \frac{\partial[z]_{\gamma_1}}{\partial[u]_{\beta_1}}\frac{\partial^2[z]_{\gamma_2}}{\partial[x]_{\alpha_1}\partial[x]_{\alpha_2}}\right)$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z}\sum_{\gamma_3=1}^{n_z} \frac{\partial^3[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}\partial[z]_{\gamma_3}} \frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}}\frac{\partial[z]_{\gamma_2}}{\partial[x]_{\alpha_2}}\frac{\partial[z]_{\gamma_3}}{\partial[u]_{\beta_1}}$$

with $\mathcal{M}_{1,3} = \left\{ \underbrace{\overbrace{\{1,2,3\}}^{c_1}}_{c} \right\}$, $\mathcal{M}_{2,3} = \left\{ \underbrace{\{\overbrace{\{1\}}^{c_1}\ \overbrace{\{2,3\}}^{c_2}\}}_{c}\ \underbrace{\{\overbrace{\{2\}}^{c_1}\ \overbrace{\{1,3\}}^{c_2}\}}_{c},\ \underbrace{\{\overbrace{\{3\}}^{c_1}\ \overbrace{\{1,2\}}^{c_2}\}}_{c}, \right\}$, $\mathcal{M}_{3,3} = \left\{ \underbrace{\overbrace{\{1\}}^{c_1}, \overbrace{\{2\}}^{c_2}, \overbrace{\{3\}}^{c_3}}_{c} \right\}$

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathscr{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

third-order derivative of equation i with respect to the α_1-th state variable and to the β_1-th and β_2-th current shock variables:

$$[F_{xuu}]^i_{\alpha_1\beta_1\beta_2} = [f_z]^i_{\gamma_1}[z_{xuu}]^{\gamma_1}_{\alpha_1\beta_1\beta_2} + [f_{zz}]^i_{\gamma_1\gamma_2}\left([z_x]^{\gamma_1}_{\alpha_1}[z_{uu}]^{\gamma_2}_{\beta_1\beta_2} + [z_u]^{\gamma_1}_{\beta_1}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_2} + [z_u]^{\gamma_1}_{\beta_2}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1}\right) + [f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}[z_u]^{\gamma_3}_{\beta_2}$$

$$= \sum_{\gamma_1=1}^{n_z} \frac{\partial[f]_i}{\partial[z]_{\gamma_1}} \frac{\partial^3[z]_{\gamma_1}}{\partial[x]_{\alpha_1}\partial[u]_{\beta_1}\partial[u]_{\beta_2}}$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z} \frac{\partial^2[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}} \left(\frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}} \frac{\partial^2[z]_{\gamma_2}}{\partial[u]_{\beta_1}\partial[u]_{\beta_2}} + \frac{\partial[z]_{\gamma_1}}{\partial[u]_{\beta_1}} \frac{\partial^2[z]_{\gamma_2}}{\partial[x]_{\alpha_1}\partial[u]_{\beta_2}} + \frac{\partial[z]_{\gamma_1}}{\partial[u]_{\beta_2}} \frac{\partial^2[z]_{\gamma_2}}{\partial[x]_{\alpha_1}\partial[u]_{\beta_1}} \right)$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z}\sum_{\gamma_3=1}^{n_z} \frac{\partial^3[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}\partial[z]_{\gamma_3}} \frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}} \frac{\partial[z]_{\gamma_2}}{\partial[u]_{\beta_1}} \frac{\partial[z]_{\gamma_3}}{\partial[u]_{\beta_2}}$$

with $\mathscr{M}_{1,3} = \left\{ \underbrace{\overbrace{\{1,2,3\}}^{c_1}}_{c} \right\}$, $\mathscr{M}_{2,3} = \left\{ \underbrace{\{\overbrace{\{1\}}^{c_1}\ \overbrace{\{2,3\}}^{c_2}\}}_{c}\ \underbrace{\{\overbrace{\{2\}}^{c_1}\ \overbrace{\{1,3\}}^{c_2}\}}_{c},\ \underbrace{\{\overbrace{\{3\}}^{c_1}\ \overbrace{\{1,2\}}^{c_2}\}}_{c}, \right\}$, $\mathscr{M}_{3,3} = \left\{ \underbrace{\overbrace{\{1\}}^{c_1}, \overbrace{\{2\}}^{c_2}, \overbrace{\{3\}}^{c_3}}_{c} \right\}$

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathcal{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

third-order derivative of equation i with respect to the α_1-th state, to the β_1-th current shock and to the δ_1-th future shock variables:

$$[F_{xuu_+}]^i_{\alpha_1\beta_1\delta_1} = [f_z]^i_{\gamma_1}[z_{xuu_+}]^{\gamma_1}_{\alpha_1\beta_1\delta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}\left([z_x]^{\gamma_1}_{\alpha_1}[z_{uu_+}]^{\gamma_2}_{\beta_1\delta_1} + [z_u]^{\gamma_1}_{\beta_1}[z_{xu_+}]^{\gamma_2}_{\alpha_1\delta_1} + [z_{u_+}]^{\gamma_1}_{\delta_1}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1}\right) + [f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}[z_{u_+}]^{\gamma_3}_{\delta_1}$$

$$= \sum_{\gamma_1=1}^{n_z} \frac{\partial[f]_i}{\partial[z]_{\gamma_1}} \frac{\partial^3[z]_{\gamma_1}}{\partial[x]_{\alpha_1}\partial[u]_{\beta_1}\partial[u_+]_{\delta_1}}$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z} \frac{\partial^2[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}} \left(\frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}} \frac{\partial^2[z]_{\gamma_2}}{\partial[u]_{\beta_1}\partial[u_+]_{\delta_1}} + \frac{\partial[z]_{\gamma_1}}{\partial[u]_{\beta_1}} \frac{\partial^2[z]_{\gamma_2}}{\partial[x]_{\alpha_1}\partial[u_+]_{\delta_1}} + \frac{\partial[z]_{\gamma_1}}{\partial[u_+]_{\delta_1}} \frac{\partial^2[z]_{\gamma_2}}{\partial[x]_{\alpha_1}\partial[u]_{\beta_1}} \right)$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z}\sum_{\gamma_3=1}^{n_z} \frac{\partial^3[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}\partial[z]_{\gamma_3}} \frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}} \frac{\partial[z]_{\gamma_2}}{\partial[u]_{\beta_1}} \frac{\partial[z]_{\gamma_3}}{\partial[u_+]_{\delta_1}}$$

with $\mathcal{M}_{1,3} = \left\{ \underbrace{\overbrace{\{1,2,3\}}^{c_1}}_{c} \right\}$, $\mathcal{M}_{2,3} = \left\{ \underbrace{\{\overbrace{\{1\}}^{c_1}\ \overbrace{\{2,3\}}^{c_2}\}}_{c} \underbrace{\{\overbrace{\{2\}}^{c_1}\ \overbrace{\{1,3\}}^{c_2}\}}_{c}, \underbrace{\{\overbrace{\{3\}}^{c_1}\ \overbrace{\{1,2\}}^{c_2}\}}_{c}, \right\}$, $\mathcal{M}_{3,3} = \left\{ \underbrace{\overbrace{\{1\}}^{c_1}, \overbrace{\{2\}}^{c_2}, \overbrace{\{3\}}^{c_3}}_{c} \right\}$

$$[F_{r^k}]^i_{\boldsymbol{\tau}_k} := \frac{\partial^k [F]_i}{\partial[r]_{\tau_1}\partial[r]_{\tau_2}\cdots\partial[r]_{\tau_k}} = \sum_{l=1}^{k} [f_{z^l}]^i_{\boldsymbol{\gamma}_l} \sum_{c\in\mathscr{M}_{l,k}} \prod_{m=1}^{l} [z_{r^{|c_m|}}]^{\gamma_m}_{\boldsymbol{\tau}(c_m)}$$

third-order derivative of equation i with respect to the α_1-th state and to two times the perturbation parameter:

$$[F_{x\sigma\sigma}]^i_{\alpha_1} = [f_z]^i_{\gamma_1}[z_{x\sigma\sigma}]^{\gamma_1}_{\alpha_1} + [f_{zz}]^i_{\gamma_1\gamma_2}\left([z_{x\sigma}]^{\gamma_1}_{\alpha_1}[z_\sigma]^{\gamma_2} + [z_{x\sigma}]^{\gamma_1}_{\alpha_1}[z_\sigma]^{\gamma_2} + [z_x]^{\gamma_1}_{\alpha_1}[z_{\sigma\sigma}]^{\gamma_2}\right) + [f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_x]^{\gamma_1}_{\alpha_1}[z_\sigma]^{\gamma_2}[z_\sigma]^{\gamma_3}$$

$$= \sum_{\gamma_1=1}^{n_z} \frac{\partial[f]_i}{\partial[z]_{\gamma_1}} \frac{\partial^3[z]_{\gamma_1}}{\partial[x]_{\alpha_1}\partial\sigma\partial\sigma}$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z} \frac{\partial^2[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}}\left(\frac{\partial^2[z]_{\gamma_1}}{\partial[x]_{\alpha_1}\partial\sigma}\frac{\partial[z]_{\gamma_2}}{\partial\sigma} + \frac{\partial^2[z]_{\gamma_1}}{\partial[x]_{\alpha_1}\partial\sigma}\frac{\partial[z]_{\gamma_2}}{\partial\sigma} + \frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}}\frac{\partial^2[z]_{\gamma_2}}{\partial\sigma\partial\sigma}\right)$$

$$+ \sum_{\gamma_1=1}^{n_z}\sum_{\gamma_2=1}^{n_z}\sum_{\gamma_3=1}^{n_z} \frac{\partial^3[f]_i}{\partial[z]_{\gamma_1}\partial[z]_{\gamma_2}\partial[z]_{\gamma_3}}\frac{\partial[z]_{\gamma_1}}{\partial[x]_{\alpha_1}}\frac{\partial[z]_{\gamma_2}}{\partial\sigma}\frac{\partial[z]_{\gamma_3}}{\partial\sigma}$$

with $\mathscr{M}_{1,3} = \left\{\underbrace{\overbrace{\{1,2,3\}}^{c_1}}_{c}\right\}$, $\mathscr{M}_{2,3} = \left\{\underbrace{\{\overbrace{\{1\}}^{c_1}\overbrace{\{2,3\}}^{c_2}\}}_{c}\ \underbrace{\{\overbrace{\{2\}}^{c_1}\overbrace{\{1,3\}}^{c_2}\}}_{c},\underbrace{\{\overbrace{\{3\}}^{c_1}\overbrace{\{1,2\}}^{c_2}\}}_{c},\right\}$, $\mathscr{M}_{3,3} = \left\{\underbrace{\overbrace{\{1\}}^{c_1},\overbrace{\{2\}}^{c_2},\overbrace{\{3\}}^{c_3}}_{c}\right\}$

Note that the selection of $\boldsymbol{\tau}(c_m)$ ignores indices in c_m when they correspond to the perturbation parameter σ.

Tensor Unfolding

Tensor Unfolding

- we can also express all tensors $[F_{r^k}]^i_{\boldsymbol{\tau}_k}$ by a matrix F_{r^k}
- idea is to map a multidimensional tensor to 2-dimensional matrix
- rows correspond to model equations i, columns correspond to specific ordering of individual tensors
  - natural approach for columns: let all $\boldsymbol{\tau}_k$ indices run from 1 to n_r and store computed values sequentially in rows and columns
  - example column ordering for $k = 3$ and $n_r = 3$:

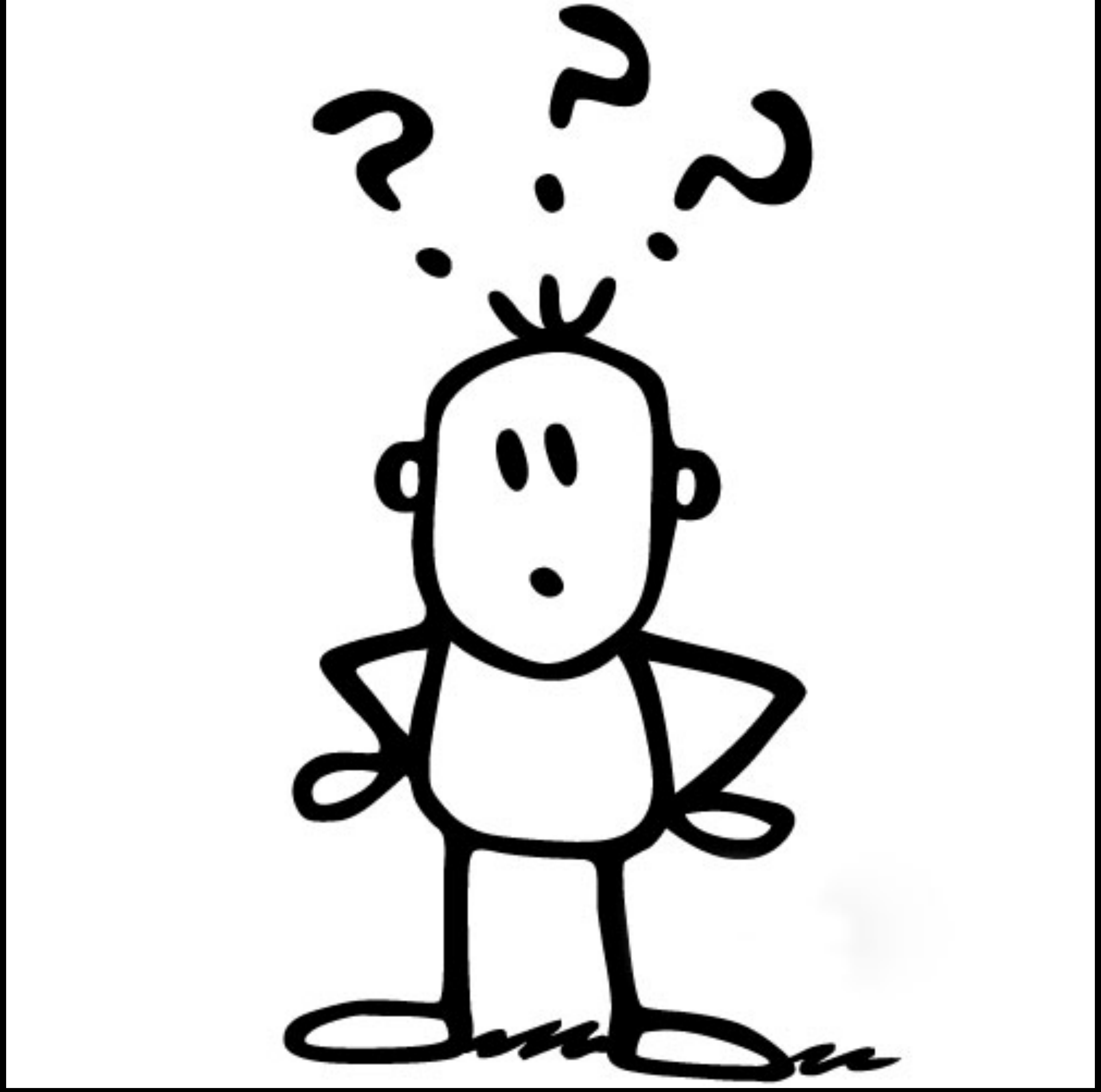
(1,1,1); (1,1,2); (1,1,3); (1,2,1); (1,2,2); (1,2,3); (1,3,1); (1,3,2); (1,3,3);...
(2,1,1); (2,1,2); (2,1,3); (2,2,1); (2,2,2); (2,2,3); (2,3,1); (2,3,2); (2,3,3);...
(3,1,1); (3,1,2); (3,1,3); (3,2,1); (3,2,2); (3,2,3); (3,3,1); (3,3,2); (3,3,3);...

- $\dfrac{\partial^3 [F]_5}{\partial [r]_3 \partial [r]_1 \partial [r]_2}$ would be the 5th row and 20th column of matrix F_{r^k}

Tensor Unfolding

- running loops for unfolding is computational inefficient, alternative:
  - basic matrix multiplication rules
  - Kronecker products
  - permutation matrices which perform the necessary reordering such that tensor summations are in accordance with matrix multiplications
- Dynare uses a dedicated and quite efficient *Multidimensional Tensor Library* written in C++ for $k > 2$

Examples



Tensor Unfolding $[F_x]^i_{\alpha_1}$

$$[F_x]^i_{\alpha_1} = [f_z]^i_{\gamma_1}[z_x]^{\gamma_1}_{\alpha_1}$$

$$F_x = f_z z_x$$

- basic matrix multiplication rules for matrix f_z and vector z_x

Tensor Unfolding $[F_{xu}]^i_{\alpha_1\beta_1}$

$$[F_{xu}]^i_{\alpha_1\beta_1} = [f_z]^i_{\gamma_1}[z_{xu}]^{\gamma_1}_{\alpha_1\beta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}$$

$$F_{xu} = f_z z_{xu} + f_{zz}\left(z_x \otimes z_u\right)$$

- Kronecker product $(z_x \otimes z_u)$ unfolds $[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}$ correctly, because it has required (α_1, β_1)-ordering
- basic matrix multiplication rules as both z_{xu} and $(z_x \otimes z_u)$ are vectors

Tensor Unfolding $[F_{xxu}]^i_{\alpha_1\alpha_2\beta_1}$

$$[F_{xxu}]^i_{\alpha_1\alpha_2\beta_1} = [f_z]^i_{\gamma_1}[z_{xxu}]^{\gamma_1}_{\alpha_1\alpha_2\beta_1}$$

$$+[f_{zz}]^i_{\gamma_1\gamma_2}\left([z_x]^{\gamma_1}_{\alpha_1}[z_{xu}]^{\gamma_2}_{\alpha_2\beta_1} + [z_x]^{\gamma_1}_{\alpha_2}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1} + [z_u]^{\gamma_1}_{\beta_1}[z_{xx}]^{\gamma_2}_{\alpha_1\alpha_2}\right)$$

$$+[f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_x]^{\gamma_1}_{\alpha_1}[z_x]^{\gamma_2}_{\alpha_2}[z_u]^{\gamma_3}_{\beta_1}$$

$$F_{xxu} = f_z z_{xxu} + f_{zz}\left(\left(z_x \otimes z_{xu}\right) P^2_{x\_xu} + \left(z_{xx} \otimes z_u\right)\right) + f_{zzz}\left(z_x \otimes z_x \otimes z_u\right)$$

Tensor Unfolding $[F_{xxu}]^i_{\alpha_1\alpha_2\beta_1}$

- red tensor terms contain same values but are summed in different ordering
  - $[z_x]^{\gamma_1}_{\alpha_1}[z_{xu}]^{\gamma_2}_{\alpha_2\beta_1}$ is consistent with $(\alpha_1, \alpha_2, \beta_1)$-ordering, can be unfolded by

$$\left(z_x \otimes z_{xu}\right)$$

  - $[z_x]^{\gamma_1}_{\alpha_2}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1}$ is not consistent with $(\alpha_1, \alpha_2, \beta_1)$-ordering, can be unfolded by

$$\left(z_x \otimes z_{xu}\right) P_{x\_xu}$$

    - P_{x_xu} is a permuted identity matrix
- $P^2_{x\_xu} = I + P_{x\_xu}$

Tensor Unfolding $[F_{xxu}]^i_{\alpha_1\alpha_2\beta_1}$

- green tensor is not consistent with $(\alpha_1, \alpha_2, \beta_1)$-ordering, but
  - due to symmetry $[f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_{xx}]^{\gamma_2}_{\alpha_1\alpha_2} = [f_{zz}]^i_{\gamma_1\gamma_2}[z_{xx}]^{\gamma_1}_{\alpha_1\alpha_2}[z_u]^{\gamma_2}_{\beta_1}$
  - which is consistent with $(\alpha_1, \alpha_2, \beta_1)$-ordering
  - can be unfolded by $f_{zz}\left(z_{xx} \otimes z_u\right)$

Tensor Unfolding $[F_{xuu}]^i_{\alpha_1\beta_1\beta_2}$

$$
\begin{aligned}
[F_{xuu}]^i_{\alpha_1\beta_1\beta_2} &= [f_z]^i_{\gamma_1}[z_{xuu}]^{\gamma_1}_{\alpha_1\beta_1\beta_2} \\
&+ [f_{zz}]^i_{\gamma_1\gamma_2}\left([z_x]^{\gamma_1}_{\alpha_1}[z_{uu}]^{\gamma_2}_{\beta_1\beta_2} + [z_u]^{\gamma_1}_{\beta_1}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_2} + [z_u]^{\gamma_1}_{\beta_2}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1}\right) \\
&+ [f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}[z_u]^{\gamma_3}_{\beta_2}
\end{aligned}
$$

$$
F_{xuu} = f_z z_{xxu} + f_{zz}\left(\left(z_x \otimes z_{uu}\right) + \left(z_{xu} \otimes z_u\right) P^2_{xu\_u}\right) + f_{zzz}\left(z_x \otimes z_u \otimes z_u\right)
$$

Tensor Unfolding $[F_{xuu}]^i_{\alpha_1\beta_1\beta_2}$

- green tensor is consistent with $(\alpha_1, \beta_1, \beta_2)$-ordering can be unfolded by $f_{zz}\left(z_x \otimes z_{uu}\right)$

Tensor Unfolding $[F_{xuu}]^{i}_{\alpha_1\beta_1\beta_2}$

- red tensor terms contain same values but are summed in different ordering
  - $[z_u]^{\gamma_1}_{\beta_2}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1}$ is not consistent with $(\alpha_1, \alpha_2, \beta_1)$-ordering, but due to symmetry of $[f_{zz}]^{i}_{\gamma_1\gamma_2} = [f_{zz}]^{i}_{\gamma_2\gamma_1}$, it can be unfolded by

$$\left(z_{xu} \otimes z_u\right)$$

  - $[z_u]^{\gamma_1}_{\beta_1}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_2}$ is not consistent with $(\alpha_1, \beta_1, \beta_2)$-ordering, but due to symmetry of $[f_{zz}]^{i}_{\gamma_1\gamma_2} = [f_{zz}]^{i}_{\gamma_2\gamma_1}$, it can be unfolded by

$$\left(z_{xu} \otimes z_u\right) P_{xu\_u}$$

  - P_{xu_u} is a permuted identity matrix
- $P^2_{xu\_u} = I + P_{xu\_u}$

Tensor Unfolding $[F_{xuu_+}]^i_{\alpha_1\beta_1\delta_1}$

$$[F_{xuu_+}]^i_{\alpha_1\beta_1\delta_1} = [f_z]^i_{\gamma_1}[z_{xuu_+}]^{\gamma_1}_{\alpha_1\beta_1\delta_1}$$

$$+[f_{zz}]^i_{\gamma_1\gamma_2}\left([z_x]^{\gamma_1}_{\alpha_1}[z_{uu_+}]^{\gamma_2}_{\beta_1\delta_1} + [z_u]^{\gamma_1}_{\beta_1}[z_{xu_+}]^{\gamma_2}_{\alpha_1\delta_1} + [z_{u_+}]^{\gamma_1}_{\delta_1}[z_{xu}]^{\gamma_2}_{\alpha_1\beta_1}\right)$$

$$+[f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}[z_{u_+}]^{\gamma_3}_{\delta_1}$$

$$F_{xuu_+} = f_z z_{xuu_+} + f_{zz}\left(\left(z_x \otimes z_{uu_+}\right) + \left(z_{xu_+} \otimes z_u\right)P^1_{xu+\_u} + \left(z_{xu} \otimes z_{u_+}\right)\right) + f_{zzz}\left(z_x \otimes z_u \otimes z_{u_+}\right)$$

Tensor Unfolding $[F_{xuu_+}]^i_{\alpha_1\beta_1\delta_1}$

- red tensor is consistent with $(\alpha_1, \beta_1, \delta_1)$-ordering can be unfolded by $(z_x \otimes z_{uu_+})$
- green tensor is not consistent with $(\alpha_1, \beta_1, \delta_1)$-ordering, but
  - due to symmetry of $[f_{zz}]^i_{\gamma_1\gamma_2}$, $[z_{xu_+}]^{\gamma_1}_{\alpha_1\delta_1}[z_u]^{\gamma_2}_{\beta_1}$ can be unfolded by $(z_{xu_+} \otimes z_u)P^1_{xu_+\_u}$
- blue tensor is not consistent with $(\alpha_1, \beta_1, \delta_1)$-ordering, but
  - due to symmetry of $[f_{zz}]^i_{\gamma_1\gamma_2}$, $[z_{xu}]^{\gamma_1}_{\alpha_1\beta_1}[z_{u_+}]^{\gamma_2}_{\delta_1}$ can be unfolded by $(z_{xu} \otimes z_{u_+})$

Tensor Unfolding $[F_{u\sigma\sigma}]^i_{\beta_1}$

$$[F_{u\sigma\sigma}]^i_{\beta_1} = [f_z]^i_{\gamma_1}[z_{u\sigma\sigma}]^{\gamma_1}_{\beta_1}$$

$$+[f_{zz}]^i_{\gamma_1\gamma_2}\left([z_u]^{\gamma_1}_{\beta_1}[z_{\sigma\sigma}]^{\gamma_2} + [z_\sigma]^{\gamma_1}[z_{u\sigma}]^{\gamma_2}_{\beta_1} + [z_\sigma]^{\gamma_1}[z_{u\sigma}]^{\gamma_2}_{\beta_1}\right)$$

$$+[f_{zzz}]^i_{\gamma_1\gamma_2\gamma_3}[z_u]^{\gamma_1}_{\beta_1}[z_\sigma]^{\gamma_2}[z_\sigma]^{\gamma_3}$$

$$F_{u\sigma\sigma} = f_z z_{u\sigma\sigma} + f_{zz}\left(\left(z_u \otimes z_{\sigma\sigma}\right) + \left(z_{u\sigma} \otimes z_\sigma\right) P^2_{u\sigma\_\sigma}\right) + f_{zzz}\left(z_u \otimes z_\sigma \otimes z_\sigma\right)$$

Tensor Unfolding $[F_{u\sigma\sigma}]^{i}_{\beta_1}$

- red tensor is consistent with (β_1)-ordering can be unfolded by $(z_u \otimes z_{\sigma\sigma})$
- green tensor is just a product of vectors, we could simply use Kronecker product, but to keep in the flow of the algorithm:
  - due to symmetry of $[f_{zz}]^{i}_{\gamma_1\gamma_2}$, $[z_{u\sigma}]^{\gamma_1}_{\beta_1}[z_\sigma]^{\gamma_2}$ can be unfolded by $(z_{u\sigma} \otimes z_\sigma)P_{u\sigma\_\sigma}$
  - $P^2_{u\sigma\_\sigma} = P_{u\sigma\_\sigma} + P_{u\sigma\_\sigma} = 2$

Perturbation Approximation

Algorithm

objective is to find the coefficients of the k-order Taylor expansion of g:

$$g_{x^q u^p \sigma^{k-q-p}} := \frac{\partial^k g(\bar{x},0,0)}{\underbrace{\partial x \ldots \partial x}_{q \text{ times}} \cdot \underbrace{\partial u \ldots \partial u}_{p \text{ times}} \cdot \underbrace{\partial \sigma \ldots \partial \sigma}_{k-q-p \text{ times}}}$$

where $0 \leq p, q \leq k$ and $0 \leq p + q \leq k$

algorithm is recursive:

- find all coefficients for $k = 1$, then find all coefficients for $k = 2$, then find all coefficients for $k = 3$, ...

First-order Approximation

First-order Approximation

- first-order Taylor expansion of the i-th equation of F around $\bar{r} = (\bar{x},0,0,0)$ is in tensor notation:

$$[F(r)]^i \approx [F(\bar{r})]^i + [F_x]^i_{\alpha_1}[\hat{x}]^{\alpha_1} + [F_u]^i_{\beta_1}[u]^{\beta_1} + [F_\sigma]^i\sigma + [F_{u_+}]^i_{\delta_1}[u_+]^{\delta_1}$$

- taking conditional expectation and setting it to zero yields:

$$0 = [F(\bar{r})]^i + [F_x]^i_{\alpha_1}[\hat{x}]^{\alpha_1} + [F_u]^i_{\beta_1}[u]^{\beta_1} + \left([F_\sigma]^i + [F_{u_+}]^i_{\delta_1}[\Sigma^{(1)}]^{\delta_1}\right)\sigma$$

- note that $\left[F(\bar{r})\right]^i = 0$ and $\left[\Sigma^{(1)}\right]^{\delta_1}$ is the δ_1 entry of $\Sigma^{(1)} = E_t\{\eta_{t+1}\}$

First-order Approximation

$$0 = [F_x]^i_{\alpha_1}[\hat{x}]^{\alpha_1} + [F_u]^i_{\beta_1}[u]^{\beta_1} + \left([F_\sigma]^i + [F_{u_+}]^i_{\delta_1}[\Sigma^{(1)}]^{\delta_1} \right) \sigma$$

- this equation needs to be satisfied for any value of $\hat{x}$, u and σ
- necessary and sufficient conditions to recover the first-order partial derivatives of g with respect to x, u and σ can be retrieved from:

$$\left[F_x\right]^i_{\alpha_1} = 0, \qquad \left[F_u\right]^i_{\beta_1} = 0, \qquad \left[F_\sigma\right]^i + \left[F_{u_+}\right]^i_{\delta_1} \left[\Sigma^{(1)}\right]^{\delta_1} = 0$$

First-order Approximation

$$\left[F_x\right]^i_{\alpha_1} = 0, \qquad \left[F_u\right]^i_{\beta_1} = 0, \qquad \left[F_\sigma\right]^i + \left[F_{u_+}\right]^i_{\delta_1} \left[\Sigma^{(1)}\right]^{\delta_1} = 0$$

computation is done in sequence:

- recover g_x
- recover g_u
- recover g_σ

First-order Approximation
Recovering g_x

Reminder

$$r := \begin{pmatrix} x \\ u \\ u_+ \\ \sigma \end{pmatrix}$$ [input vector for F and G]

$$w(r) := \begin{pmatrix} y_0^* \\ u_+ \\ \sigma \end{pmatrix} = \begin{pmatrix} g^*(x, u, \sigma) \\ u_+ \\ \sigma \end{pmatrix}$$ [input vector for g^{**}]

$$z(r) := \begin{pmatrix} y_-^* \\ y \\ y_+^{**} \\ u \end{pmatrix} = \begin{pmatrix} x \\ g(x, u, \sigma) \\ G(x, u, u_+, \sigma) \\ u \end{pmatrix} = \begin{pmatrix} x \\ g(x, u, \sigma) \\ g^{**}(g^*(x, u, \sigma), u_+, \sigma) \\ u \end{pmatrix}$$ [input vector for f]

Recovering g_x

Tensors

$$[w_x]_{\alpha_1} = \begin{bmatrix} [g_x^*]_{\alpha_1} \\ 0 \\ 0 \end{bmatrix}$$

$$[G_x]^{\iota}_{\alpha_1} = [g_w^{**}]^{\iota}_{\phi_1}[w_x]^{\phi_1}_{\alpha_1} = [g_x^{**}]^{\iota}_{\rho_1}[g_x^*]^{\rho_1}_{\alpha_1}$$

$$[z_x]_{\alpha_1} = \begin{bmatrix} [I_x]_{\alpha_1} \\ [g_x]_{\alpha_1} \\ [G_x]_{\alpha_1} \\ 0 \end{bmatrix}$$

$$[F_x]^{i}_{\alpha_1} = [f_z]^{i}_{\gamma_1}[z_x]^{\gamma_1}_{\alpha_1}$$

Matrix

$$w_x = \begin{pmatrix} g_x^* \\ 0 \\ 0 \end{pmatrix}$$

$$G_x = g_w^{**}w_x = g_x^{**}g_x^*$$

$$z_x = \begin{pmatrix} I_x \\ g_x \\ G_x \\ 0 \end{pmatrix}$$

$$F_x = f_z z_x$$

Recovering g_x

$$[F_x]^i_{\alpha_1} = [f_z]^i_{\gamma_1}[z_x]^{\gamma_1}_{\alpha_1} = [f_{y^*_-}]^i_{\alpha_1} + [f_{y_0}]^i_{\rho_1^0}[g_x]^{\rho_1^0}_{\alpha_1} + [f_{y^{**}_+}]^i_{\rho_1^+}[g^{**}_x]^{\rho_1^+}_{\rho_1}[g^*_x]^{\rho_1}_{\alpha_1} = 0$$

where $[f_{y^*_-}]^i_{\alpha_1}$, $[f_{y_0}]^i_{\rho_1^0}$ and $[f_{y^{**}_+}]^i_{\rho_1^+}$ are the first partial derivatives of equation i of f with respect to $[y^*_{t-1}]_{\alpha_1}$, $[y_t]_{\rho_1^0}$ and $[y^{**}_{t+1}]_{\rho_1^+}$, respectively.

Tensor Unfolding yields the corresponding matrix representation:

$$F_x = f_z z_x = f_{y^*_-} + f_{y_0} g_x + f_{y^{**}_+} g^{**}_x g^*_x = 0$$

Recovering g_x

$$[F_x]^i_{\alpha_1} = [f_z]^i_{\gamma_1}[z_x]^{\gamma_1}_{\alpha_1} = [f_{y^*_-}]^i_{\alpha_1} + [f_{y_0}]^i_{\rho^0_1}[g_x]^{\rho^0_1}_{\alpha_1} + [f_{y^{**}_+}]^i_{\rho^+_1}[g^{**}_x]^{\rho^+_1}_{\rho_1}[g^*_x]^{\rho_1}_{\alpha_1} = 0$$

$$F_x = f_z z_x = f_{y^*_-} + f_{y_0} g_x + f_{y^{**}_+} g^{**}_x g^*_x = 0$$

this is a quadratic matrix equation, solving it is equivalent to finding a solution to linearized rational expectations models for which different algorithms have been proposed.

Dynare uses algorithm outlined in Villemot (2011), see other presentation.

Perturbation Matrices

important auxiliary perturbation matrices:

$$A = f_{y_0} + \left(\underbrace{0}_{n \times n^{static}} \quad \vdots \quad \underbrace{f_{y_+^{**}} g_x^{**}}_{n \times n^{spred}} \quad \vdots \quad \underbrace{0}_{n \times n^{fwrd}} \right)$$

$$B = \left(\underbrace{0}_{n \times n^{static}} \quad \vdots \quad \underbrace{0}_{n \times n^{pred}} \quad \vdots \quad \underbrace{f_{y_+^{**}}}_{n \times n^{sfwrd}} \right)$$

First-order Approximation
Recovering g_u

Recovering g_u

Tensors

$$[w_u]_{\beta_1} = \begin{bmatrix} [g_u^*]_{\beta_1} \\ 0 \\ 0 \end{bmatrix}$$

$$[G_u]^{\iota}_{\beta_1} = [g_w^{**}]^{\iota}_{\phi_1}[w_u]^{\phi_1}_{\beta_1} = [g_x^{**}]^{\iota}_{\rho_1}[g_u^*]^{\rho_1}_{\beta_1}$$

$$[z_u]_{\beta_1} = \begin{bmatrix} 0 \\ [g_u]_{\beta_1} \\ [G_u]_{\beta_1} \\ [I_u]_{\beta_1} \end{bmatrix}$$

$$[F_u]^{i}_{\beta_1} = [f_z]^{i}_{\gamma_1}[z_u]^{\gamma_1}_{\beta_1}$$

Matrix

$$w_u = \begin{pmatrix} g_u^* \\ 0 \\ 0 \end{pmatrix}$$

$$G_u = g_w^{**}w_u = g_x^{**}g_u^*$$

$$z_u = \begin{pmatrix} 0 \\ g_u \\ G_u \\ I_u \end{pmatrix}$$

$$F_u = f_z z_u$$

Recovering g_u

$$[F_u]^i_{\beta_1} = [f_z]^i_{\gamma_1}[z_u]^{\gamma_1}_{\beta_1} = [f_{y_0}]^i_{\rho_1^0}[g_u]^{\rho_1^0}_{\beta_1} + [f_{y_+^{**}}]^i_{\rho_1^+}[g_x^{**}]^{\rho_1^+}_{\rho_1}[g_u^*]^{\rho_1}_{\beta_1} + [f_u]^i_{\beta_1} = 0$$

where $[f_u]^i_{\beta_1}$ is the first partial derivative of equation i of f with respect to $[u]_{\beta_1}$.

Tensor Unfolding yields the corresponding matrix representation:

$$F_u = f_z z_u = f_{y_0} g_u + f_{y_+^{**}} g_x^{**} g_u^* + f_u = A g_u + f_u = 0$$

taking the inverse of A yields g_u:

$$g_u = -A^{-1} f_u$$

First-order Approximation Recovering g_σ

Recovering g_σ

Tensors

$$[w_{u_+}]_{\delta_1} = \begin{bmatrix} 0 \\ [I_u]_{\delta_1} \\ 0 \end{bmatrix}, [w_\sigma] = \begin{bmatrix} [g_\sigma^*] \\ 0 \\ 1 \end{bmatrix}$$

$$[G_{u_+}]^\iota_{\delta_1} = [g_w^{**}]^\iota_{\phi_1}[w_{u_+}]^{\phi_1}_{\delta_1} = [g_u^{**}]^\iota_{\psi_1}[I_u]^{\psi_1}_{\delta_1}$$

$$[G_\sigma]^\iota = [g_w^{**}]^\iota_{\phi_1}[w_\sigma]^{\phi_1}_{\alpha_1} = [g_x^{**}]^\iota_{\rho_1}[g_\sigma^*]^{\rho_1} + [g_\sigma^{**}]^\iota$$

$$[z_{u_+}]_{\delta_1} = \begin{bmatrix} 0 \\ 0 \\ [G_{u_+}]_{\delta_1} \\ 0 \end{bmatrix}, [z_\sigma] = \begin{bmatrix} 0 \\ [g_\sigma] \\ [G_\sigma] \\ 0 \end{bmatrix}$$

$$[F_{u_+}]^i_{\delta_1} = [f_z]^i_{\gamma_1}[z_{u_+}]^{\gamma_1}_{\delta_1}$$

$$[F_\sigma]^i = [f_z]^i_{\gamma_1}[z_\sigma]^{\gamma_1}$$

Matrix

$$w_{u_+} = \begin{pmatrix} 0 \\ I_u \\ 0 \end{pmatrix}, w_\sigma = \begin{pmatrix} g_\sigma^* \\ 0 \\ 1 \end{pmatrix}$$

$$G_{u_+} = g_w^{**}w_{u_+} = g_u^{**}$$

$$G_\sigma = g_w^{**}w_\sigma = g_x^{**}g_\sigma^* + g_\sigma^{**}$$

$$z_{u_+} = \begin{pmatrix} 0 \\ 0 \\ G_{u_+} \\ 0 \end{pmatrix}, z_\sigma = \begin{pmatrix} 0 \\ g_\sigma \\ G_\sigma \\ 0 \end{pmatrix}$$

$$F_{u_+} = f_z z_{u_+}$$

$$F_\sigma = f_z z_\sigma$$

Recovering g_σ

$$0 = [F_\sigma]^i + [F_{u_+}]^i_{\delta_1}[\Sigma^{(1)}]^{\delta_1} = [f_z]^i_{\gamma_1}[z_\sigma]^{\gamma_1} + [f_z]^i_{\gamma_1}[z_{u_+}]^{\gamma_1}_{\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

$$0 = [f_{y_0}]^i_{\rho_1^0}[g_\sigma]^{\rho_1^0} + [f_{y_+^{**}}]^i_{\rho_1^+}\left([g_x^{**}]^{\rho_1^+}_{\rho_1}[g_\sigma^*]^{\rho_1} + [g_\sigma^{**}]^{\rho_1^+}\right) + [f_{y_+^{**}}]^i_{\rho_1^+}[g_u^{**}]^{\rho_1^+}_{\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

Tensor Unfolding yields the corresponding matrix representation:

$$0 = F_\sigma + F_{u_+}\Sigma^{(1)} = f_z z_\sigma + f_z z_{u_+}\Sigma^{(1)}$$

$$0 = f_{y_0}g_\sigma + f_{y_+^{**}}\left(g_x^{**}g_\sigma^* + g_\sigma^{**}\right) + f_{y_+^{**}}g_u^{**}\Sigma^{(1)} = (A+B)g_\sigma + f_{y_+^{**}}g_u^{**}\Sigma^{(1)}$$

taking the inverse of $(A + B)$ yields

$$g_\sigma = -(A+B)^{-1}\left(f_{y_+^{**}}g_u^{**}\Sigma^{(1)}\right)$$

because the first moment $\Sigma^{(1)}$ is zero by assumption, we get: $g_\sigma = 0$

Certainty Equivalence $g_\sigma = 0$

when we derived the optimality conditions (aka model equations) agents do take into account the effect of future uncertainty when optimizing their objective functions.

BUT: the first-order approximated policy function is independent of the size of the stochastic innovations:

$$y_t = g_x y^*_{t-1} + g_u u_t$$

future uncertainty does not matter for the decision rules of the agents at first order

certainty equivalence is a result of the first-order perturbation approximation, we can break it with e.g. higher-order perturbation approximation

Second-order Approximation

Second-order Approximation

second-order Taylor expansion of the i-th equation of F around $\bar{r} = (\bar{x}, 0, 0, 0)$ is in tensor notation:

$$[F(r)]^i \approx [F(\bar{r})]^i + [F_x]^i_{\alpha_1}[\hat{x}]^{\alpha_1} + [F_u]^i_{\beta_1}[u]^{\beta_1} + [F_{u_+}]^i_{\delta_1}[u_+]^{\delta_1} + [F_\sigma]^i\sigma$$

$$+\frac{1}{2}\left([F_{xx}]^i_{\alpha_1\alpha_2}[\hat{x}]^{\alpha_1}[\hat{x}]^{\alpha_2} + [F_{uu}]^i_{\beta_1\beta_2}[u]^{\beta_1}[u]^{\beta_2} + [F_{u_+u_+}]^i_{\delta_1\delta_2}[u_+]^{\delta_1}[u_+]^{\delta_2} + [F_{\sigma\sigma}]^i\sigma\sigma \right)$$

$$+\frac{2}{2}\left([F_{xu}]^i_{\alpha_1\beta_1}[\hat{x}]^{\alpha_1}[u]^{\beta_1} + [F_{xu_+}]^i_{\alpha_1\delta_1}[\hat{x}]^{\alpha_1}[u_+]^{\delta_1} + [F_{x\sigma}]^i_{\alpha_1}[\hat{x}]^{\alpha_1}\sigma + [F_{uu_+}]^i_{\beta_1\delta_1}[u]^{\beta_1}[u_+]^{\delta_1} + [F_{u\sigma}]^i_{\beta_1}[u]^{\beta_1}\sigma + [F_{u_+\sigma}]^i_{\delta_1}[u_+]^{\delta_1}\sigma \right)$$

Second-order Approximation

taking conditional expectation and setting it to zero yields:

$$0 = [F(\bar{r})]^i + [F_x]^i_{\alpha_1}[\hat{x}]^{\alpha_1} + [F_u]^i_{\beta_1}[u]^{\beta_1} + \left([F_\sigma]^i + [F_{u_+}]^i_{\delta_1}[\Sigma^{(1)}]^{\delta_1}\right)\sigma$$

$$+\frac{1}{2}\left([F_{xx}]^i_{\alpha_1\alpha_2}[\hat{x}]^{\alpha_1}[\hat{x}]^{\alpha_2} + [F_{uu}]^i_{\beta_1\beta_2}[u]^{\beta_1}[u]^{\beta_2} + \left([F_{\sigma\sigma}]^i + [F_{u_+u_+}]^i_{\delta_1\delta_2}[\Sigma^{(2)}]^{\delta_1\delta_2} + 2[F_{u_+\sigma}]^i_{\delta_1}[\Sigma^{(1)}]^{\delta_1}\right)\sigma\sigma \right)$$

$$+\frac{2}{2}\left([F_{xu}]^i_{\alpha_1\beta_1}[\hat{x}]^{\alpha_1}[u]^{\beta_1} + \left([F_{x\sigma}]^i_{\alpha_1} + [F_{xu_+}]^i_{\alpha_1\delta_1}[\Sigma^{(1)}]^{\delta_1}\right)[\hat{x}]^{\alpha_1}\sigma + \left([F_{u\sigma}]^i_{\beta_1} + [F_{uu_+}]^i_{\beta_1\delta_1}[\Sigma^{(1)}]^{\delta_1}\right)[u]^{\beta_1}\sigma \right)$$

note that $\left[F(\bar{r})\right]^i = 0$ and $\left[\Sigma^{(1)}\right]^{\delta_1}$ is the δ_1 entry of $\Sigma^{(1)} = E_t\{\eta_{t+1}\}$and $\left[\Sigma^{(2)}\right]^{\delta_1\delta_2}$ denotes the covariance between $[\eta_t]_{\delta_1}$ and $[\eta_t]_{\delta_2}$

this equation needs to be satisfied for any value of $\hat{x}$, u and σ

Second-order Approximation

necessary and sufficient conditions to recover the second-order partial derivatives of g with respect to xx, xu, $x\sigma$, uu, $u\sigma$ and $\sigma\sigma$:

for g_{xx}: $0 = \left[F_{xx}\right]^{i}_{\alpha_1\alpha_2}$

for g_{uu}: $0 = \left[F_{uu}\right]^{i}_{\beta_1\beta_2}$

for g_{xu}: $0 = \left[F_{xu}\right]^{i}_{\alpha_1\beta_1}$

for $g_{x\sigma}$: $0 = \left[F_{x\sigma}\right]^{i}_{\alpha_1} + \left[F_{xu_+}\right]^{i}_{\alpha_1\delta_1}\left[\Sigma^{(1)}\right]^{\delta_1}$

for $g_{u\sigma}$: $0 = \left[F_{u\sigma}\right]^{i}_{\beta_1} + \left[F_{uu_+}\right]^{i}_{\beta_1\delta_1}\left[\Sigma^{(1)}\right]^{\delta_1}$

for $g_{\sigma\sigma}$: $0 = \left[F_{\sigma\sigma}\right]^{i} + \left[F_{u_+u_+}\right]^{i}_{\delta_1\delta_2}\left[\Sigma^{(2)}\right]^{\delta_1\delta_2} + 2\left[F_{u_+\sigma}\right]^{i}_{\delta_1}\left[\Sigma^{(1)}\right]^{\delta_1}$

Second-order Approximation
Recovering g_{xx}

Recovering g_{xx}

Tensors

$$[w_{xx}]_{\alpha_1\alpha_2} = \begin{bmatrix} [g^*_{xx}]_{\alpha_1\alpha_2} \\ 0 \\ 0 \end{bmatrix}$$

$$[G_{xx}]^{\iota}_{\alpha_1\alpha_2} = [g^{**}_w]^{\iota}_{\phi_1}[w_{xx}]^{\phi_1}_{\alpha_1\alpha_2} + [g^{**}_{ww}]^{\iota}_{\phi_1\phi_2}[w_x]^{\phi_1}_{\alpha_1}[w_x]^{\phi_2}_{\alpha_2}$$

$$= [g^{**}_x]^{\iota}_{\rho_1}[g^*_{xx}]^{\rho_1}_{\alpha_1\alpha_2} + [g^{**}_{xx}]^{\iota}_{\rho_1\rho_2}[g^*_x]^{\rho_1}_{\alpha_1}[g^*_x]^{\rho_2}_{\alpha_2}$$

$$[z_{xx}]_{\alpha_1\alpha_2} = \begin{bmatrix} 0 \\ [g_{xx}]_{\alpha_1\alpha_2} \\ [G_{xx}]_{\alpha_1\alpha_2} \\ 0 \end{bmatrix}$$

$$[F_{xx}]^{i}_{\alpha_1\alpha_2} = [f_z]^{i}_{\gamma_1}[z_{xx}]^{\gamma_1}_{\alpha_1\alpha_2} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_x]^{\gamma_2}_{\alpha_2}$$

Matrix

$$w_{xx} = \begin{pmatrix} g^*_{xx} \\ 0 \\ 0 \end{pmatrix}$$

$$G_{xx} = g^{**}_w w_{xx} + g^{**}_{ww}\left(w_x \otimes w_x\right)$$

$$= g^{**}_x g^*_{xx} + g^{**}_{xx}\left(g^*_x \otimes g^*_x\right)$$

$$z_{xx} = \begin{pmatrix} 0 \\ g_{xx} \\ G_{xx} \\ 0 \end{pmatrix}$$

$$F_{xx} = f_z z_{xx} + f_{zz}\left(z_x \otimes z_x\right)$$

Recovering g_{xx}

$$[F_{xx}]^i_{\alpha_1\alpha_2} = [f_z]^i_{\gamma_1}[z_{xx}]^{\gamma_1}_{\alpha_1\alpha_2} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_x]^{\gamma_2}_{\alpha_2} = 0$$

Tensor Unfolding yields the corresponding matrix representation:

$$F_{xx} = f_z z_{xx} + f_{zz}\left(z_x \otimes z_x\right) = 0$$

developing terms, we can simplify this using perturbation matrices A and B:

$$A g_{xx} + B g_{xx}\left(g_x^* \otimes g_x^*\right) = -f_{zz}\left(z_x \otimes z_x\right)$$

Recovering g_{xx}

$$A g_{xx} + B g_{xx} \left(g_x^* \otimes g_x^* \right) = -f_{zz} \left(z_x \otimes z_x \right)$$

this is a *Generalized Sylvester Equation* for which Dynare uses specialized and very efficient algorithms

Recovering g_{xx}

$$A g_{xx} + B g_{xx}\left(g_x^* \otimes g_x^*\right) = -f_{zz}\left(z_x \otimes z_x\right)$$

- $RHS = f_{zz}\left(z_x \otimes z_x\right)$ contains only first-order terms
- RHS can be computed by evaluating *Faà di Bruno's formula* for $[F_{xx}]^i_{\alpha_1\alpha_2}$ and $[G_{xx}]^\iota_{\alpha_1\alpha_2}$ **conditional** on $[g_{xx}]^i_{\alpha_1\alpha_2} = 0$:

$$[G_{xx}^{cond}]^\iota_{\alpha_1\alpha_2} = [g_x^{**}]^\iota_{\rho_1}[g_{xx}^*]^{\rho_1}_{\alpha_1\alpha_2} + [g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_x^*]^{\rho_1}_{\alpha_1}[g_x^*]^{\rho_2}_{\alpha_2} = 0$$

$$[z_{xx}^{cond}]_{\alpha_1\alpha_2} = \begin{bmatrix} 0 \\ [g_{xx}]_{\alpha_1\alpha_2} \\ [G_{xx}]_{\alpha_1\alpha_2} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$[F_{xx}^{cond}]^i_{\alpha_1\alpha_2} = [f_z]^i_{\gamma_1}[z_{xx}^{cond}]^{\gamma_1}_{\alpha_1\alpha_2} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_x]^{\gamma_2}_{\alpha_2} = [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_x]^{\gamma_2}_{\alpha_2}$$

- Tensor Unfolding: $F_{xx}^{cond} = f_{zz}\left(z_x \otimes z_x\right)$

Second-order Approximation
Recovering g_{uu}

Recovering g_{uu}

Tensors

$$[w_{uu}]_{\beta_1\beta_2} = \begin{bmatrix} [g^*_{uu}]_{\beta_1\beta_2} \\ 0 \\ 0 \end{bmatrix}$$

$$[G_{uu}]^{\iota}_{\beta_1\beta_2} = [g^{**}_w]^{\iota}_{\phi_1}[w_{uu}]^{\phi_1}_{\beta_1\beta_2} + [g^{**}_{ww}]^{\iota}_{\phi_1\phi_2}[w_u]^{\phi_1}_{\beta_1}[w_u]^{\phi_2}_{\beta_2}$$

$$= [g^{**}_x]^{\iota}_{\rho_1}[g^*_{uu}]^{\rho_1}_{\beta_1\beta_2} + [g^{**}_{xx}]^{\iota}_{\rho_1\rho_2}[g^*_u]^{\rho_1}_{\beta_1}[g^*_u]^{\rho_2}_{\beta_2}$$

$$[z_{uu}]_{\beta_1\beta_2} = \begin{bmatrix} 0 \\ [g_{uu}]_{\beta_1\beta_2} \\ [G_{uu}]_{\beta_1\beta_2} \\ 0 \end{bmatrix}$$

$$[F_{uu}]^{i}_{\beta_1\beta_2} = [f_z]^{i}_{\gamma_1}[z_{uu}]^{\gamma_1}_{\beta_1\beta_2} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_u]^{\gamma_2}_{\beta_2}$$

Matrix

$$w_{uu} = \begin{pmatrix} g^*_{uu} \\ 0 \\ 0 \end{pmatrix}$$

$$G_{uu} = g^{**}_w w_{uu} + g^{**}_{ww}\left(w_u \otimes w_u\right)$$

$$= g^{**}_x g^*_{uu} + g^{**}_{xx}\left(g^*_u \otimes g^*_u\right)$$

$$z_{uu} = \begin{pmatrix} 0 \\ g_{uu} \\ G_{uu} \\ 0 \end{pmatrix}$$

$$F_{uu} = f_z z_{uu} + f_{zz}\left(z_u \otimes z_u\right)$$

Recovering g_{uu}

$$[F_{uu}]^i_{\beta_1\beta_2} = [f_z]^i_{\gamma_1}[z_{uu}]^{\gamma_1}_{\beta_1\beta_2} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_u]^{\gamma_2}_{\beta_2} = 0$$

Tensor Unfolding yields the corresponding matrix representation:

$$F_{uu} = f_z z_{uu} + f_{zz}\left(z_u \otimes z_u\right) = 0$$

developing terms, we can simplify this using perturbation matrix A:

$$Ag_{uu} = -\left(f_{y_+^{**}} g_{xx}^{**}\left(g_u^* \otimes g_u^*\right) + f_{zz}\left(z_u \otimes z_u\right) \right)$$

Recovering g_{uu}

$$Ag_{uu} = -\left(f_{y_+^{**}} g_{xx}^{**} \left(g_u^* \otimes g_u^* \right) + f_{zz} \left(z_u \otimes z_u \right) \right)$$

- note that the right-hand side contains only objects that are already available from the first-order approximation and previously computed g_{xx}
- taking the inverse of A yields g_{uu}

Recovering g_{uu}

$$Ag_{uu} = -\left(f_{y_+^{**}} g_{xx}^{**} \left(g_u^* \otimes g_u^*\right) + f_{zz} \left(z_u \otimes z_u\right) \right)$$

- $RHS = f_{y_+^{**}} g_{xx}^{**} \left(g_u^* \otimes g_u^*\right) + f_{zz} \left(z_u \otimes z_u\right)$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{uu}]^i_{\beta_1\beta_2}$ and $[G_{uu}]^\iota_{\beta_1\beta_2}$ **conditional** on $[g_{uu}]^i_{\beta_1\beta_2} = 0$:

$$[G^{cond}_{uu}]^\iota_{\beta_1\beta_2} = [g^{**}_x]^\iota_{\rho_1}[g^*_{uu}]^{\rho_1}_{\beta_1\beta_2} + [g^{**}_{xx}]^\iota_{\rho_1\rho_2}[g^*_u]^{\rho_1}_{\beta_1}[g^*_u]^{\rho_2}_{\beta_2} = [g^{**}_{xx}]^\iota_{\rho_1\rho_2}[g^*_u]^{\rho_1}_{\beta_1}[g^*_u]^{\rho_2}_{\beta_2}$$

$$[z^{cond}_{uu}]_{\beta_1\beta_2} = \begin{bmatrix} 0 \\ [g_{uu}]_{\beta_1\beta_2} \\ [G_{uu}]_{\beta_1\beta_2} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ [G^{cond}_{uu}]_{\beta_1\beta_2} \\ 0 \end{bmatrix}$$

$$[F^{cond}_{uu}]^i_{\beta_1\beta_2} = [f_z]^i_{\gamma_1}[z^{cond}_{uu}]^{\gamma_1}_{\beta_1\beta_2} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_u]^{\gamma_2}_{\beta_2} = [f_{y_+^{**}}]^i_{\rho_1^+}[g^{**}_{xx}]^{\rho_1^+}_{\rho_1\rho_2}[g^*_u]^{\rho_1}_{\beta_1}[g^*_u]^{\rho_2}_{\beta_2} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_u]^{\gamma_2}_{\beta_2}$$

- Tensor Unfolding: $F^{cond}_{uu} = f_{y_+^{**}} g_{xx}^{**} \left(g_u^* \otimes g_u^*\right) + f_{zz} \left(z_u \otimes z_u\right)$

Second-order Approximation
Recovering g_{xu}

Recovering g_{xu}

Tensors

$$[w_{xu}]_{\alpha_1\beta_1} = \begin{bmatrix} [g^*_{xu}]_{\alpha_1\beta_1} \\ 0 \\ 0 \end{bmatrix}$$

$$[G_{xu}]^{\iota}_{\alpha_1\beta_1} = [g^{**}_w]^{\iota}_{\phi_1}[w_{xu}]^{\phi_1}_{\alpha_1\beta_1} + [g^{**}_{ww}]^{\iota}_{\phi_1\phi_2}[w_x]^{\phi_1}_{\alpha_1}[w_u]^{\phi_2}_{\beta_1}$$

$$= [g^{**}_x]^{\iota}_{\rho_1}[g^*_{xu}]^{\rho_1}_{\alpha_1\beta_1} + [g^{**}_{xx}]^{\iota}_{\rho_1\rho_2}[g^*_x]^{\rho_1}_{\alpha_1}[g^*_u]^{\rho_2}_{\beta_1}$$

$$[z_{xu}]_{\alpha_1\beta_1} = \begin{bmatrix} 0 \\ [g_{xu}]_{\alpha_1\beta_1} \\ [G_{xu}]_{\alpha_1\beta_1} \\ 0 \end{bmatrix}$$

$$[F_{xu}]^{i}_{\alpha_1\beta_1} = [f_z]^{i}_{\gamma_1}[z_{xu}]^{\gamma_1}_{\alpha_1\beta_1} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}$$

Matrix

$$w_{xu} = \begin{pmatrix} g^*_{xu} \\ 0 \\ 0 \end{pmatrix}$$

$$G_{xu} = g^{**}_w w_{xu} + g^{**}_{ww}\left(w_x \otimes w_u\right)$$

$$= g^{**}_x g^*_{xu} + g^{**}_{xx}\left(g^*_x \otimes g^*_u\right)$$

$$z_{xu} = \begin{pmatrix} 0 \\ g_{xu} \\ G_{xu} \\ 0 \end{pmatrix}$$

$$F_{xu} = f_z z_{xu} + f_{zz}\left(z_x \otimes z_u\right)$$

Recovering g_{xu}

$$[F_{xu}]^{i}_{\alpha_1\beta_1} = [f_z]^{i}_{\gamma_1}[z_{xu}]^{\gamma_1}_{\alpha_1\beta_1} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1} = 0$$

Tensor Unfolding yields the corresponding matrix representation:

$$F_{xu} = f_z z_{xu} + f_{zz}\left(z_x \otimes z_u\right) = 0$$

developing terms, we can simplify this using perturbation matrix A:

$$Ag_{xu} = -\left(f_{y_+^{**}} g_{xx}^{**}\left(g_x^* \otimes g_u^*\right) + f_{zz}\left(z_x \otimes z_u\right) \right)$$

Recovering g_{xu}

$$Ag_{xu} = -\left(f_{y_+^{**}} g_{xx}^{**} \left(g_x^* \otimes g_u^* \right) + f_{zz} \left(z_x \otimes z_u \right) \right)$$

- note that the right-hand side contains only objects that are already available from the first-order approximation and previously computed g_{xx}
- taking the inverse of A yields g_{xu}

Recovering g_{xu}

$$Ag_{xu} = -\left(f_{y_+^{**}} g_{xx}^{**} \left(g_x^* \otimes g_u^*\right) + f_{zz} \left(z_x \otimes z_u\right) \right)$$

- $RHS = f_{y_+^{**}} g_{xx}^{**} \left(g_x^* \otimes g_u^*\right) + f_{zz} \left(z_x \otimes z_u\right)$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{xu}]^i_{\alpha_1\beta_1}$ and $[G_{xu}]^\iota_{\alpha_1\beta_1}$ **<u>conditional</u>** on $[g_{xu}]^i_{\alpha_1\beta_1} = 0$:

$$[G_{xu}^{cond}]^\iota_{\alpha_1\beta_1} = [g_x^{**}]^\iota_{\rho_1}[g_{xu}^*]^{\rho_1}_{\alpha_1\beta_1} + [g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_x^*]^{\rho_1}_{\alpha_1}[g_u^*]^{\rho_2}_{\beta_1} = [g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_x^*]^{\rho_1}_{\alpha_1}[g_u^*]^{\rho_2}_{\beta_1}$$

$$[z_{xu}^{cond}]_{\alpha_1\beta_1} = \begin{bmatrix} 0 \\ [g_{xu}]_{\alpha_1\beta_1} \\ [G_{xu}]_{\alpha_1\beta_1} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ [G_{xu}^{cond}]_{\alpha_1\beta_1} \\ 0 \end{bmatrix}$$

$$[F_{xu}^{cond}]^i_{\alpha_1\beta_1} = [f_z]^i_{\gamma_1}[z_{xu}^{cond}]^{\gamma_1}_{\alpha_1\beta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1} = [f_{y_+^{**}}]^i_{\rho_1^+}[g_{xx}^{**}]^{\rho_1^+}_{\rho_1\rho_2}[g_x^*]^{\rho_1}_{\alpha_1}[g_u^*]^{\rho_2}_{\beta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_u]^{\gamma_2}_{\beta_1}$$

- Tensor Unfolding: $F_{xu}^{cond} = f_{y_+^{**}} g_{xx}^{**} \left(g_x^* \otimes g_u^*\right) + f_{zz} \left(z_x \otimes z_u\right)$

Second-order Approximation
Recovering $g_{x\sigma}$

Recovering $g_{x\sigma}$

Tensors

$$[w_{xu_+}]_{\alpha_1\delta_1} = \begin{bmatrix}0\\0\\0\end{bmatrix}, [w_{x\sigma}]_{\alpha_1} = \begin{bmatrix}[g^*_{x\sigma}]_{\alpha_1}\\0\\0\end{bmatrix}$$

$$[G_{xu_+}]^{\iota}_{\alpha_1\delta_1} = [g^{**}_w]^{\iota}_{\phi_1}[w_{xu_+}]^{\phi_1}_{\alpha_1\delta_1} + [g^{**}_{ww}]^{\iota}_{\phi_1\phi_2}[w_x]^{\phi_1}_{\alpha_1}[w_{u_+}]^{\phi_2}_{\delta_1}$$

$$= [g^{**}_{xu}]^{\iota}_{\rho_1\psi_1}[g^*_x]^{\rho_1}_{\alpha_1}[I_u]^{\psi_1}_{\delta_1}$$

$$[G_{x\sigma}]^{\iota}_{\alpha_1} = [g^{**}_w]^{\iota}_{\phi_1}[w_{x\sigma}]^{\phi_1}_{\alpha_1} + [g^{**}_{ww}]^{\iota}_{\phi_1\phi_2}[w_x]^{\phi_1}_{\alpha_1}[w_\sigma]^{\phi_2}$$

$$= [g^{**}_x]^{\iota}_{\rho_1}[g^*_{x\sigma}]^{\rho_1}_{\alpha_1} + [g^{**}_{xx}]^{\iota}_{\rho_1\rho_2}[g^*_x]^{\rho_1}_{\alpha_1}[g^*_\sigma]^{\rho_2} + [g^{**}_{x\sigma}]^{\iota}_{\rho_1}[g^*_x]^{\rho_1}_{\alpha_1}$$

$$[z_{xu_+}]_{\alpha_1\delta_1} = \begin{bmatrix}0\\0\\ [G_{xu_+}]_{\alpha_1\delta_1}\\0\end{bmatrix}, [z_{x\sigma}]_{\alpha_1} = \begin{bmatrix}0\\ [g_{x\sigma}]_{\alpha_1}\\ [G_{x\sigma}]_{\alpha_1}\\0\end{bmatrix}$$

$$[F_{xu_+}]^{i}_{\alpha_1\delta_1} = [f_z]^{i}_{\gamma_1}[z_{xu_+}]^{\gamma_1}_{\alpha_1\delta_1} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_{u_+}]^{\gamma_2}_{\delta_1}$$

$$[F_{x\sigma}]^{i}_{\alpha_1} = [f_z]^{i}_{\gamma_1}[z_{x\sigma}]^{\gamma_1}_{\alpha_1} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_\sigma]^{\gamma_2}$$

Matrix

$$w_{xu_+} = \begin{pmatrix}0\\0\\0\end{pmatrix}, w_{x\sigma} = \begin{pmatrix}g^*_{x\sigma}\\0\\0\end{pmatrix}$$

$$G_{xu_+} = g^{**}_w w_{xu_+} + g^{**}_{ww}\left(w_x \otimes w_{u_+}\right)$$

$$= g^{**}_{xu}\left(g^*_x \otimes I_u\right)$$

$$G_{x\sigma} = g^{**}_w w_{x\sigma} + g^{**}_{ww}\left(w_x \otimes w_\sigma\right)$$

$$= g^{**}_x g^*_{x\sigma} + g^{**}_{xx}\left(g^*_x \otimes g^*_\sigma\right) + g^{**}_{x\sigma}g^*_x$$

$$z_{xu_+} = \begin{pmatrix}0\\0\\G_{xu_+}\\0\end{pmatrix}, z_{x\sigma} = \begin{pmatrix}0\\g_{x\sigma}\\G_{x\sigma}\\0\end{pmatrix}$$

$$F_{xu_+} = f_z z_{xu_+} + f_{zz}\left(z_x \otimes z_{u_+}\right)$$

$$F_{x\sigma} = f_z z_{x\sigma} + f_{zz}\left(z_x \otimes z_\sigma\right)$$

Recovering $g_{x\sigma}$

$$\left[F_{x\sigma}\right]^{i}_{\alpha_1} + \underbrace{\left[F_{xu_+}\right]^{i}_{\alpha_1\delta_1}\left[\Sigma^{(1)}\right]^{\delta_1}}_{=:[D_{101}]^{i}_{\alpha_1}} = [f_z]^{i}_{\gamma_1}[z_{x\sigma}]^{\gamma_1}_{\alpha_1} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_\sigma]^{\gamma_2} + [D_{101}]^{i}_{\alpha_1} = 0$$

Tensor Unfolding yields the corresponding matrix representation:

$$F_{x\sigma} + \underbrace{F_{xu_+}\left(I_x \otimes \Sigma^{(1)}\right)}_{=:D_{101}} = f_z z_{x\sigma} + f_{zz}\left(z_x \otimes z_\sigma\right) + D_{101} = 0$$

developing terms, we can simplify this using perturbation matrices A and B:

$$Ag_{x\sigma} + Bg_{x\sigma}g^{*}_{x} = -\left(f_{y^{**}_{+}} g^{**}_{xx}(g^{*}_{x} \otimes g^{*}_{\sigma}) + f_{zz}\left(z_x \otimes z_\sigma\right) + D_{101} \right)$$

Recovering $g_{x\sigma}$

$$A g_{x\sigma} + B g_{x\sigma} g_x^* = -\left(f_{y_+^{**}} g_{xx}^{**} (g_x^* \otimes g_\sigma^*) + f_{zz} \left(z_x \otimes z_\sigma \right) + D_{101} \right)$$

- this is a *Generalized Sylvester Equation*
- note that the right-hand side contains only objects that are already available from previously computed terms
  - but due to certainty equivalence: $g_\sigma^* = 0$, $z_\sigma = 0$
  - $D_{101} = 0$ because $\Sigma^{(1)} = 0$
- therefore: $g_{x\sigma} = 0$
- a second-order approximation does not imply a correction for uncertainty in terms which are linear in the state vector

Recovering $g_{x\sigma}$

$$A g_{x\sigma} + B g_{x\sigma} g_x^* = -\left(f_{y_+^{**}} g_{xx}^{**} (g_x^* \otimes g_\sigma^*) + f_{zz}\left(z_x \otimes z_\sigma\right) + D_{101} \right)$$

- $f_{y_+^{**}} g_{xx}^{**} (g_x^* \otimes g_\sigma^*) + f_{zz}\left(z_x \otimes z_\sigma\right)$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{x\sigma}]^i_{\alpha_1}$ and $[G_{x\sigma}]^\iota_{\alpha_1}$ **conditional** on $[g_{x\sigma}]^i_{\alpha_1} = 0$:

$$[G^{cond}_{x\sigma}]^\iota_{\alpha_1} = [g_x^{**}]^\iota_{\rho_1}[g^*_{x\sigma}]^{\rho_1}_{\alpha_1} + [g^{**}_{xx}]^\iota_{\rho_1\rho_2}[g^*_x]^{\rho_1}_{\alpha_1}[g^*_\sigma]^{\rho_2} + [g^{**}_{x\sigma}]^\iota_{\rho_1}[g^*_x]^{\rho_1}_{\alpha_1} = [g^{**}_{xx}]^\iota_{\rho_1\rho_2}[g^*_x]^{\rho_1}_{\alpha_1}[g^*_\sigma]^{\rho_2}$$

$$[z^{cond}_{x\sigma}]_{\alpha_1} = \begin{bmatrix} 0 \\ [g_{x\sigma}]_{\alpha_1} \\ [G_{x\sigma}]_{\alpha_1} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ [G_{x\sigma}]_{\alpha_1} \\ 0 \end{bmatrix}$$

$$[F^{cond}_{x\sigma}]^i_{\alpha_1} = [f_z]^i_{\gamma_1}[z^{cond}_{x\sigma}]^{\gamma_1}_{\alpha_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_\sigma]^{\gamma_2} = [f_{y_+^{**}}]^i_{\rho_1^+}[g^{**}_{xx}]^{\rho_1^+}_{\rho_1\rho_2}[g^*_x]^{\rho_1}_{\alpha_1}[g^*_\sigma]^{\rho_2} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_x]^{\gamma_1}_{\alpha_1}[z_\sigma]^{\gamma_2}$$

- Tensor Unfolding: $F^{cond}_{x\sigma} = f_{y_+^{**}} g^{**}_{xx}\left(g^*_x \otimes g^*_\sigma\right) + f_{zz}\left(z_x \otimes z_\sigma\right)$

- $D_{101} = F_{xu_+}\left(I_x \otimes \Sigma^{(1)}\right)$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{xu_+}]^i_{\alpha_1\delta_1}$ and $[G_{xu_+}]^\iota_{\alpha_1\delta_1}$ **directly** as all terms are available

Second-order Approximation
Recovering $g_{u\sigma}$

Recovering $g_{u\sigma}$

Tensors

$$[w_{uu_+}]_{\beta_1\delta_1} = \begin{bmatrix}0\\0\\0\end{bmatrix}, [w_{u\sigma}]_{\beta_1} = \begin{bmatrix}[g^*_{u\sigma}]_{\beta_1}\\0\\0\end{bmatrix}$$

$$[G_{uu_+}]^\iota_{\beta_1\delta_1} = [g^{**}_w]^\iota_{\phi_1}[w_{uu_+}]^{\phi_1}_{\beta_1\delta_1} + [g^{**}_{ww}]^\iota_{\phi_1\phi_2}[w_u]^{\phi_1}_{\beta_1}[w_{u_+}]^{\phi_2}_{\delta_1}$$

$$= [g^{**}_{xu}]^\iota_{\rho_1\psi_1}[g^*_u]^{\rho_1}_{\beta_1}[I_u]^{\psi_1}_{\delta_1}$$

$$[G_{u\sigma}]^\iota_{\beta_1} = [g^{**}_w]^\iota_{\phi_1}[w_{u\sigma}]^{\phi_1}_{\beta_1} + [g^{**}_{ww}]^\iota_{\phi_1\phi_2}[w_u]^{\phi_1}_{\beta_1}[w_\sigma]^{\phi_2}$$

$$= [g^{**}_x]^\iota_{\rho_1}[g^*_{u\sigma}]^{\rho_1}_{\beta_1} + [g^{**}_{xx}]^\iota_{\rho_1\rho_2}[g^*_u]^{\rho_1}_{\beta_1}[g^*_\sigma]^{\rho_2} + [g^{**}_{x\sigma}]^\iota_{\rho_1}[g^*_u]^{\rho_1}_{\beta_1}$$

$$[z_{uu_+}]_{\beta_1\delta_1} = \begin{bmatrix}0\\0\\ [G_{uu_+}]_{\beta_1\delta_1}\\0\end{bmatrix}, [z_{u\sigma}]_{\beta_1} = \begin{bmatrix}0\\ [g_{u\sigma}]_{\beta_1}\\ [G_{u\sigma}]_{\beta_1}\\0\end{bmatrix}$$

$$[F_{uu_+}]^i_{\beta_1\delta_1} = [f_z]^i_{\gamma_1}[z_{uu_+}]^{\gamma_1}_{\beta_1\delta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_{u_+}]^{\gamma_2}_{\delta_1}$$

$$[F_{u\sigma}]^i_{\beta_1} = [f_z]^i_{\gamma_1}[z_{u\sigma}]^{\gamma_1}_{\beta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_\sigma]^{\gamma_2}$$

Matrix

$$w_{uu_+} = \begin{pmatrix}0\\0\\0\end{pmatrix}, w_{u\sigma} = \begin{pmatrix}g^*_{u\sigma}\\0\\0\end{pmatrix}$$

$$G_{uu_+} = g^{**}_w w_{uu_+} + g^{**}_{ww}\left(w_u \otimes w_{u_+}\right)$$

$$= g^{**}_{xu}\left(g^*_u \otimes I_u\right)$$

$$G_{u\sigma} = g^{**}_w w_{u\sigma} + g^{**}_{ww}\left(w_u \otimes w_\sigma\right)$$

$$= g^{**}_x g^*_{u\sigma} + g^{**}_{xx}\left(g^*_u \otimes g^*_\sigma\right) + g^{**}_{x\sigma} g^*_u$$

$$z_{uu_+} = \begin{pmatrix}0\\0\\G_{uu_+}\\0\end{pmatrix}, z_{u\sigma} = \begin{pmatrix}0\\g_{u\sigma}\\G_{u\sigma}\\0\end{pmatrix}$$

$$F_{uu_+} = f_z z_{uu_+} + f_{zz}\left(z_u \otimes z_{u_+}\right)$$

$$F_{u\sigma} = f_z z_{u\sigma} + f_{zz}\left(z_u \otimes z_\sigma\right)$$

Recovering $g_{u\sigma}$

$$\left[F_{u\sigma}\right]^{i}_{\beta_1} + \underbrace{\left[F_{uu_+}\right]^{i}_{\beta_1\delta_1}\left[\Sigma^{(1)}\right]^{\delta_1}}_{=:[D_{011}]^{i}_{\beta_1}} = [f_z]^{i}_{\gamma_1}[z_{u\sigma}]^{\gamma_1}_{\beta_1} + [f_{zz}]^{i}_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_\sigma]^{\gamma_2} + [D_{011}]^{i}_{\beta_1} = 0$$

Tensor Unfolding yields the corresponding matrix representation:

$$F_{u\sigma} + \underbrace{F_{uu_+}\left(I_u \otimes \Sigma^{(1)}\right)}_{=:D_{011}} = f_z z_{u\sigma} + f_{zz}\left(z_u \otimes z_\sigma\right) + D_{011} = 0$$

developing terms, we can simplify this using perturbation matrix A:

$$Ag_{u\sigma} = -\left(f_{y_+^{**}}\left(g_{xx}^{**}(g_u^{*} \otimes g_\sigma^{*}) + g_{x\sigma}^{**}g_u^{*}\right) + f_{zz}\left(z_u \otimes z_\sigma\right) + D_{011} \right)$$

Recovering $g_{u\sigma}$

$$A g_{u\sigma} = -\left(f_{y_+^{**}} \left(g_{xx}^{**} (g_u^* \otimes g_\sigma^*) + g_{x\sigma}^{**} g_u^* \right) + f_{zz} \left(z_u \otimes z_\sigma \right) + D_{011} \right)$$

- taking the inverse of A gives us $g_{u\sigma}$
- note that the right-hand side contains only objects that are already available from previously computed terms
- but due to certainty equivalence and $\Sigma^{(1)} = 0$ we get

$$g_{u\sigma} = 0$$

- second-order approximation does not imply a correction for uncertainty in terms which are linear in the innovations vector

Recovering $g_{u\sigma}$

$$A g_{u\sigma} = -\left(f_{y_+^{**}}\left(g_{xx}^{**}(g_u^* \otimes g_\sigma^*) + g_{x\sigma}^{**} g_u^*\right) + f_{zz}\left(z_u \otimes z_\sigma\right) + D_{011} \right)$$

- $f_{y_+^{**}}\left(g_{xx}^{**}(g_u^* \otimes g_\sigma^*) + g_{x\sigma}^{**} g_u^*\right) + f_{zz}\left(z_u \otimes z_\sigma\right)$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{u\sigma}]^i_{\beta_1}$ and $[G_{u\sigma}]^\iota_{\beta_1}$ **conditional** on $[g_{u\sigma}]^i_{\beta_1} = 0$:

$$[G_{u\sigma}^{cond}]^\iota_{\beta_1} = [g_x^{**}]^\iota_{\rho_1}[g_{u\sigma}^*]^{\rho_1}_{\beta_1} + [g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_u^*]^{\rho_1}_{\beta_1}[g_\sigma^*]^{\rho_2} + [g_{x\sigma}^{**}]^\iota_{\rho_1}[g_u^*]^{\rho_1}_{\beta_1} = [g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_u^*]^{\rho_1}_{\beta_1}[g_\sigma^*]^{\rho_2} + [g_{x\sigma}^{**}]^\iota_{\rho_1}[g_u^*]^{\rho_1}_{\beta_1}$$

$$[z_{u\sigma}^{cond}]_{\beta_1} = \begin{bmatrix} 0 \\ [g_{u\sigma}]_{\beta_1} \\ [G_{u\sigma}]_{\beta_1} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ [G_{u\sigma}^{cond}]_{\beta_1} \\ 0 \end{bmatrix}$$

$$[F_{u\sigma}^{cond}]^i_{\beta_1} = [f_z]^i_{\gamma_1}[z_{u\sigma}^{cond}]^{\gamma_1}_{\beta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_\sigma]^{\gamma_2} = [f_{y_+^{**}}]^i_{\rho_1^+}\left([g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_u^*]^{\rho_1}_{\beta_1}[g_\sigma^*]^{\rho_2} + [g_{x\sigma}^{**}]^\iota_{\rho_1}[g_u^*]^{\rho_1}_{\beta_1}\right) + [f_{zz}]^i_{\gamma_1\gamma_2}[z_u]^{\gamma_1}_{\beta_1}[z_\sigma]^{\gamma_2}$$

- Tensor Unfolding: $F_{u\sigma}^{cond} = f_{y_+^{**}}\left(g_{xx}^{**}(g_u^* \otimes g_\sigma^*) + g_{x\sigma}^{**} g_u^*\right) + f_{zz}\left(z_u \otimes z_\sigma\right)$
- $D_{011} = F_{uu_+}\left(I_u \otimes \Sigma^{(1)}\right)$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{uu_+}]^i_{\beta_1\delta_1}$ and $[G_{uu_+}]^\iota_{\beta_1\delta_1}$ **directly** as all terms are available

Second-order Approximation Recovering $g_{\sigma\sigma}$

Recovering $g_{\sigma\sigma}$

Tensors

$$[w_{\sigma\sigma}] = \begin{bmatrix} [g^*_{\sigma\sigma}] \\ 0 \\ 0 \end{bmatrix}, [w_{u_+\sigma}]_{\delta_1} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, [w_{u_+u_+}]_{\delta_1\delta_2} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$[G_{\sigma\sigma}]^\iota = [g^{**}_w]^\iota_{\phi_1}[w_{\sigma\sigma}]^{\phi_1} + [g^{**}_{ww}]^\iota_{\phi_1\phi_2}[w_\sigma]^{\phi_1}[w_\sigma]^{\phi_2}$$

$$= [g^{**}_x]^\iota_{\rho_1}[g^*_{\sigma\sigma}]^{\rho_1} + [g^{**}_{xx}]^\iota_{\rho_1\rho_2}[g^*_\sigma]^{\rho_1}[g^*_\sigma]^{\rho_2} + [g^{**}_{x\sigma}]^\iota_{\rho_1}[g^*_\sigma]^{\rho_1} + [g^{**}_{x\sigma}]^\iota_{\rho_1}[g^*_\sigma]^{\rho_1} + [g^{**}_{\sigma\sigma}]^\iota$$

$$[G_{u_+\sigma}]^\iota_{\delta_1} = [g^{**}_w]^\iota_{\phi_1}[w_{u_+\sigma}]^{\phi_1}_{\delta_1} + [g^{**}_{ww}]^\iota_{\phi_1\phi_2}[w_{u_+}]^{\phi_1}_{\delta_1}[w_\sigma]^{\phi_2}$$

$$= [g^{**}_{xu}]^\iota_{\rho_1\psi_1}[g^*_\sigma]^{\rho_1}[I_u]^{\psi_1}_{\delta_1} + [g^{**}_{u\sigma}]^\iota_{\psi_1}[I_u]^{\psi_1}_{\delta_1}$$

$$[G_{u_+u_+}]^\iota_{\delta_1\delta_2} = [g^{**}_w]^\iota_{\phi_1}[w_{u_+u_+}]^{\phi_1}_{\delta_1\delta_2} + [g^{**}_{ww}]^\iota_{\phi_1\phi_2}[w_{u_+}]^{\phi_1}_{\delta_1}[w_{u_+}]^{\phi_2}_{\delta_2}$$

$$= [g^{**}_{uu}]^\iota_{\psi_1\psi_2}[I_u]^{\psi_1}_{\delta_1}[I_u]^{\psi_2}_{\delta_2}$$

Matrix

$$w_{\sigma\sigma} = \begin{pmatrix} g^*_{\sigma\sigma} \\ 0 \\ 0 \end{pmatrix}, w_{u_+\sigma} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, w_{u_+u_+} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$G_{\sigma\sigma} = g^{**}_w w_{\sigma\sigma} + g^{**}_{ww}\left(w_\sigma \otimes w_\sigma\right)$$

$$= g^{**}_x g^*_{\sigma\sigma} + g^{**}_{xx}\left(g^*_\sigma \otimes g^*_\sigma\right) + g^{**}_{x\sigma}g^*_\sigma + g^{**}_{x\sigma}g^*_\sigma + g^{**}_{\sigma\sigma}$$

$$G_{u_+\sigma} = g^{**}_w w_{u_+\sigma} + g^{**}_{ww}\left(w_{u_+} \otimes w_\sigma\right)$$

$$= g^{**}_{xu}\left(g^*_\sigma \otimes I_u\right) + g^{**}_{u\sigma}$$

$$G_{u_+u_+} = g^{**}_w w_{u_+u_+} + g^{**}_{ww}\left(w_{u_+} \otimes w_{u_+}\right)$$

$$= g^{**}_{uu}\left(I_u \otimes I_u\right)$$

Recovering $g_{\sigma\sigma}$

Tensors

$$[z_{\sigma\sigma}] = \begin{bmatrix} 0 \\ [g_{\sigma\sigma}] \\ [G_{\sigma\sigma}] \\ 0 \end{bmatrix}, [z_{u_+\sigma}]_{\delta_1} = \begin{bmatrix} 0 \\ 0 \\ [G_{u_+\sigma}]_{\delta_1} \\ 0 \end{bmatrix},$$

$$[z_{u_+u_+}]_{\delta_1\delta_2} = \begin{bmatrix} 0 \\ 0 \\ [G_{u_+u_+}]_{\delta_1\delta_2} \\ 0 \end{bmatrix}$$

$$[F_{\sigma\sigma}]^i = [f_z]^i_{\gamma_1}[z_{\sigma\sigma}]^{\gamma_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_\sigma]^{\gamma_1}[z_\sigma]^{\gamma_2}$$

$$[F_{u_+\sigma}]^i_{\delta_1} = [f_z]^i_{\gamma_1}[z_{u_+\sigma}]^{\gamma_1}_{\delta_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_{u_+}]^{\gamma_1}_{\delta_1}[z_\sigma]^{\gamma_2}$$

$$[F_{u_+u_+}]^i_{\delta_1\delta_2} = [f_z]^i_{\gamma_1}[z_{u_+u_+}]^{\gamma_1}_{\delta_1\delta_2} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_{u_+}]^{\gamma_1}_{\delta_1}[z_{u_+}]^{\gamma_2}_{\delta_2}$$

Matrix

$$z_{\sigma\sigma} = \begin{pmatrix} 0 \\ g_{\sigma\sigma} \\ G_{\sigma\sigma} \\ 0 \end{pmatrix}, z_{u_+\sigma} = \begin{pmatrix} 0 \\ 0 \\ G_{u_+\sigma} \\ 0 \end{pmatrix}, z_{u_+u_+} = \begin{pmatrix} 0 \\ 0 \\ G_{u_+u_+} \\ 0 \end{pmatrix}$$

$$F_{\sigma\sigma} = f_z z_{\sigma\sigma} + f_{zz}\left(z_\sigma \otimes z_\sigma\right)$$

$$F_{u_+\sigma} = f_z z_{u_+\sigma} + f_{zz}\left(z_{u_+} \otimes z_\sigma\right)$$

$$F_{u_+u_+} = f_z z_{u_+u_+} + f_{zz}\left(z_{u_+} \otimes z_{u_+}\right)$$

Recovering $g_{\sigma\sigma}$

$$[F_{\sigma\sigma}]^i + \underbrace{[F_{u_+u_+}]^i_{\delta_1\delta_2}[\Sigma^{(2)}]^{\delta_1\delta_2}}_{[D_{002}]^i} + 2\underbrace{[F_{u_+\sigma}]^i_{\delta_1}[\Sigma^{(1)}]^{\delta_1}}_{[E_{002}]^i} = [f_z]^i_{\gamma_1}[z_{\sigma\sigma}]^{\gamma_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_\sigma]^{\gamma_1}[z_\sigma]^{\gamma_2} + [D_{002}]^i + [E_{002}]^i = 0$$

Tensor Unfolding yields the corresponding matrix representation:

$$F_{\sigma\sigma} + \underbrace{F_{u_+u_+}\Sigma^{(2)}}_{D_{002}} + \underbrace{2F_{u_+\sigma}\Sigma^{(1)}}_{E_{002}} = f_z z_{\sigma\sigma} + f_{zz}\left(z_\sigma \otimes z_\sigma\right) + D_{002} + E_{002} = 0$$

developing terms, we can simplify this using perturbation matrices A and B:

$$(A+B)g_{\sigma\sigma} = -\left(f_{y_+^{**}}\left(g_{xx}^{**}(g_\sigma^* \otimes g_\sigma^*) + 2g_{x\sigma}^{**}g_\sigma^*\right) + f_{zz}\left(z_\sigma \otimes z_\sigma\right) + D_{002} + E_{002} \right)$$

Recovering $g_{\sigma\sigma}$

$$(A+B)g_{\sigma\sigma} = -\left(f_{y_+^{**}}\left(g_{xx}^{**}(g_\sigma^* \otimes g_\sigma^*) + 2g_{x\sigma}^{**}g_\sigma^*\right) + f_{zz}\left(z_\sigma \otimes z_\sigma\right) + D_{002} + E_{002} \right)$$

- note that the right-hand side contains only objects that are already available from previously computed terms
- due to certainty equivalence, $\Sigma^{(1)} = 0$, and previously computed terms this simplifies to

$$(A+B)g_{\sigma\sigma} = -\left(f_{y_+^{**}}g_{uu}^{**} + f_{y_+^{**}y_+^{**}}\left(g_u^{**} \otimes g_u^{**}\right) \right)\Sigma^{(2)}$$

- taking the inverse of $(A+B)$ gives us $g_{\sigma\sigma}$
- as $g_{\sigma\sigma}$ is nonzero, a second-order approximation adds a level correction for uncertainty to the approximated decision rule of agents (this breaks with certainty equivalence!)

Recovering $g_{\sigma\sigma}$

$$(A+B)g_{\sigma\sigma} = -\left(f_{y_+^{**}}\left(g_{xx}^{**}(g_\sigma^* \otimes g_\sigma^*) + 2g_{x\sigma}^{**}g_\sigma^*\right) + f_{zz}\left(z_\sigma \otimes z_\sigma\right) + D_{002} + E_{002} \right)$$

- $f_{y_+^{**}}\left(g_{xx}^{**}(g_\sigma^* \otimes g_\sigma^*) + 2g_{x\sigma}^{**}g_\sigma^*\right) + f_{zz}\left(z_\sigma \otimes z_\sigma\right)$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{\sigma\sigma}]^i$ and $[G_{\sigma\sigma}]^i$ **conditional** on $[g_{\sigma\sigma}]^i = 0$:

$$[G_{\sigma\sigma}^{cond}]^\iota = [g_x^{**}]^\iota_{\rho_1}[g_{\sigma\sigma}^*]^{\rho_1} + [g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_\sigma^*]^{\rho_1}[g_\sigma^*]^{\rho_2} + 2[g_{x\sigma}^{**}]^\iota_{\rho_1}[g_\sigma^*]^{\rho_1} + [g_{\sigma\sigma}^{**}]^\iota = [g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_\sigma^*]^{\rho_1}[g_\sigma^*]^{\rho_2} + 2[g_{x\sigma}^{**}]^\iota_{\rho_1}[g_\sigma^*]^{\rho_1}$$

$$[z_{\sigma\sigma}^{cond}] = \begin{bmatrix} 0 \\ [g_{\sigma\sigma}] \\ [G_{\sigma\sigma}] \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ [G_{\sigma\sigma}^{cond}] \\ 0 \end{bmatrix}$$

$$[F_{\sigma\sigma}^{cond}]^i = [f_z]^i_{\gamma_1}[z_{\sigma\sigma}^{cond}]^{\gamma_1} + [f_{zz}]^i_{\gamma_1\gamma_2}[z_\sigma]^{\gamma_1}[z_\sigma]^{\gamma_2} = [f_{y_+^{**}}]^i_{\rho_1^+}\left([g_{xx}^{**}]^\iota_{\rho_1\rho_2}[g_\sigma^*]^{\rho_1}[g_\sigma^*]^{\rho_2} + 2[g_{x\sigma}^{**}]^\iota_{\rho_1}[g_\sigma^*]^{\rho_1}\right) + [f_{zz}]^i_{\gamma_1\gamma_2}[z_\sigma]^{\gamma_1}[z_\sigma]^{\gamma_2}$$

  - Tensor Unfolding: $F_{\sigma\sigma}^{cond} = f_{y_+^{**}}\left(g_{xx}^{**}(g_\sigma^* \otimes g_\sigma^*) + 2g_{x\sigma}^{**}g_\sigma^*\right) + f_{zz}\left(z_\sigma \otimes z_\sigma\right)$

  - $D_{002} = F_{u_+u_+}\Sigma^{(2)}$ and $E_{002} = 2F_{u_+\sigma}\Sigma^{(1)}$ can be computed by evaluating *Faà di Bruno's formula* for $[F_{uu_+}]^i_{\beta_1\delta_1}$ and $[G_{uu_+}]^\iota_{\beta_1\delta_1}$ **directly** as all terms are available

Third-order Approximation

Third-order Approximation

Necessary and sufficient conditions to recover the third-order partial derivatives of g with respect to *xxx, xxu, xxσ, xuu, xuσ, xσσ,uuu, uuσ, uσσ,* and *σσσ*:

$$[g_{xxx}] : 0 = [F_{xxx}]^i_{\alpha_1\alpha_2\alpha_3}$$

$$[g_{uuu}] : 0 = [F_{uuu}]^i_{\beta_1\beta_2\beta_3}$$

$$[g_{xuu}] : 0 = [F_{xuu}]^i_{\alpha_1\beta_1\beta_2}$$

$$[g_{xxu}] : 0 = [F_{xxu}]^i_{\alpha_1\alpha_2\beta_1}$$

$$[g_{xx\sigma}] : 0 = [F_{xx\sigma}]^i_{\alpha_1\alpha_2} + [F_{xxu_+}]^i_{\alpha_1\alpha_2\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

$$[g_{xu\sigma}] : 0 = [F_{xu\sigma}]^i_{\alpha_1\beta_1} + [F_{xuu_+}]^i_{\alpha_1\beta_1\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

$$[g_{x\sigma\sigma}] : 0 = [F_{x\sigma\sigma}]^i_{\alpha_1} + [F_{xu_+u_+}]^i_{\alpha_1\delta_1\delta_2}[\Sigma^{(2)}]^{\delta_1\delta_2} + 2[F_{xu_+\sigma}]^i_{\alpha_1\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

$$[g_{uu\sigma}] : 0 = [F_{uu\sigma}]^i_{\beta_1\beta_2} + [F_{uuu_+}]^i_{\beta_1\beta_2\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

$$[g_{u\sigma\sigma}] : 0 = [F_{u\sigma\sigma}]^i_{\beta_1} + [F_{uu_+u_+}]^i_{\beta_1\delta_1\delta_2}[\Sigma^{(2)}]^{\delta_1\delta_2} + 2[F_{uu_+\sigma}]^i_{\beta_1\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

$$[g_{\sigma\sigma\sigma}] : 0 = [F_{\sigma\sigma\sigma}]^i + [F_{u_+u_+u_+}]^i_{\delta_1\delta_2\delta_3}[\Sigma^{(3)}]^{\delta_1\delta_2\delta_3} + 3[F_{u_+u_+\sigma}]^i_{\delta_1\delta_2}[\Sigma^{(2)}]^{\delta_1\delta_2} + 3[F_{u_+\sigma\sigma}]^i_{\delta_1}[\Sigma^{(1)}]^{\delta_1}$$

Third-order Approximation

$$Ag_{xxx} + Bg_{xxx}\left(g_x^* \otimes g_x^* \otimes g_x^*\right) = -F_{xxx}^{cond}$$

$$Ag_{uuu} = -F_{uuu}^{cond}$$

$$Ag_{xuu} = -F_{xuu}^{cond}$$

$$Ag_{xxu} = -F_{xuu}^{cond}$$

$$Ag_{x\sigma\sigma} + Bg_{x\sigma\sigma}g_x^* = -F_{x\sigma\sigma}^{cond} - D_{102} - E_{102}$$

$$Ag_{xu\sigma} = -F_{xu\sigma}^{cond} - D_{111}$$

$$Ag_{x\sigma\sigma} + Bg_{x\sigma\sigma}g_x^* = -F_{x\sigma\sigma}^{cond} - D_{102} - E_{102}$$

$$Ag_{uu\sigma} = -F_{uu\sigma}^{cond} - D_{021}$$

$$(A + B)g_{\sigma\sigma\sigma} = -F_{\sigma\sigma\sigma}^{cond} - D_{003} - E_{003}$$

Third-order approximation

$g_{xx\sigma} = 0, g_{uu\sigma} = 0, g_{xu\sigma} = 0$
no correction for uncertainty in terms which are quadratic in x and u

$g_{x\sigma\sigma} \neq 0, g_{u\sigma\sigma} \neq 0$
correction for uncertainty in terms which are linear in x and u

$g_{\sigma\sigma\sigma} \neq 0$
correction only iff third moments $\Sigma^{(3)} \neq 0$ (not in Dynare)

k-order Approximation

k-order Approximation

$$0 = \left[F_{x^i}\right]^i_{\boldsymbol{\alpha}_i} \text{ for } g_{x^i}$$

$$0 = \left[F_{x^i u^j}\right]^i_{\boldsymbol{\alpha}_i \boldsymbol{\beta}_j} \text{ for } g_{x^i u^j} \text{ and } j > 0$$

$$0 = \left[F_{x^i \sigma^j}\right]^i_{\boldsymbol{\alpha}_i} + [D_{ij}] + [E_{ij}] \text{ for } g_{x^i \sigma^j}$$

$$0 = \left[F_{x^i u^j \sigma^k}\right]^i_{\boldsymbol{\alpha}_i \boldsymbol{\beta}_j} + [D_{ijk}] + [E_{ijk}] \text{ for } g_{x^i u^j \sigma^k}$$

$$0 = \left[F_{\sigma^i}\right]^i + [D_i] + [E_i] \text{ for } g_{\sigma^i}$$

$$[D_{ijk}]_{\boldsymbol{\alpha}_i \boldsymbol{\beta}_j} = [F_{x^i u^j u_+^k}]_{\boldsymbol{\alpha}_i \boldsymbol{\beta}_j \boldsymbol{\delta}_k} [\Sigma^{(k)}]^{\boldsymbol{\delta}_k}$$

$$[E_{ijk}]_{\boldsymbol{\alpha}_i \boldsymbol{\beta}_j} = \sum_{m=1}^{k-1} \binom{k}{m} [F_{x^i u^j u_+^m \sigma^{k-m}}]_{\boldsymbol{\alpha}_i \boldsymbol{\beta}_j \boldsymbol{\delta}_m} [\Sigma^{(m)}]^{\boldsymbol{\delta}_m}$$

Order of Computation

recover g_{x^k}

for j=1:1:(k-1)

for i=(j-1):-1:1

recover $g_{x^{k-j}u^i\sigma^{j-i}}$

end

recover $g_{x^{k-j}\sigma^j}$

end

for i=(k-1):-1:1

recover $g_{u^i\sigma^{k-i}}$

end

recover g_{σ^k}

Computational Remarks (as of Dynare 5.1)

- order=2, we use unfolded matrix equations and optimized mex code
- order>2, we use multi-threaded and multidimensional tensor library implemented in C++
  - allows for folded/unfolded, dense/sparse tensor representations
  - implements Faà di Bruno's formula very efficiently
  - updates conditional Faa Di Bruno's formulas efficiently
- might change in future version to make use of more optimized code and/or Fortran re-implementation