Appendix

A. Algorithm Derivation

In this subsection, the proposed Bayesian controller reduction algorithm as shown in Algorithm 1 will be explain in detail. As introduced before, the actor-critic algorithms based on deterministic policy gradient can be applied over continuous action spaces [9]. And the effective reduction for actor network would be beneficial to real-time inference. For clear explanation, we will take deep deterministic policy gradient algorithm (DDPG), one representative actor-critic algorithm, as the example to elaborate our method. We use $\mathbf{W}$ to represent the parameters of a single layer of actor network θ^{μ}. Firstly, the definition of optimization objective will be illustrated. we will give the proof of proposition 1.

Proof: Given the likelihood with exponential family distribution:

$$p(\mathbf{A} \mid \mathbf{W}, \mathbf{S}, \gamma) \sim \exp\left(-E_M(\mathbf{A}; \mathrm{Net}(\mathbf{S}; \mathbf{W}); \gamma)\right) \tag{A.1}$$

The sparse prior with Gaussian distribution for $\mathbf{W}$ is supposed to be:

$$\begin{aligned} p(\mathbf{W}) &= \prod_{i=1}^{\aleph} \mathcal{N}(\mathbf{W}|0, \Gamma)\varphi(\gamma_i) \\ &= \max_{\gamma \succ \mathbf{0}} \mathcal{N}(\mathbf{W}|0, \Gamma)\varphi(\gamma), \end{aligned} \tag{A.2}$$

where

$$\gamma = [\gamma_1, \ldots, \gamma_{\aleph}] \in \mathbb{R}^{\aleph}, \ \Gamma = \mathrm{diag}\,[\gamma]\,.$$

where $\aleph$ means the number of groups that $\mathbf{W}$ could be divided into for reduction; diag denote the operation to get diagonal elements of matrix. The marginal likelihood could be calculated as:

$$\begin{aligned} &\int p(\mathbf{A}|\mathbf{W})\mathcal{N}(\mathbf{W}|\mathbf{0}, \Gamma)\prod_{i=1}^{\aleph}\varphi(\gamma_i)d\mathbf{W} \\ =&\int \exp\{-E(\mathbf{W})\}\mathcal{N}(\mathbf{W}|\mathbf{0}, \Gamma)\prod_{i=1}^{\aleph}\varphi(\gamma_i)d\mathbf{W} \end{aligned} \tag{A.3}$$

However, it is intractable to achieve analytical solution for Eq A.3. $E_M(\mathbf{W})$ can be expanded around $\mathbf{W}^*$ by performing a Taylor series:

$$E(\mathbf{W}) \approx E(\mathbf{W}^*) + (\mathbf{W} - \mathbf{W}^*)^{\top}\mathbf{g}(\mathbf{W}^*) + \frac{1}{2}(\mathbf{W} - \mathbf{W}^*)^{\top}\mathbf{H}(\mathbf{W}^*)(\mathbf{W} - \mathbf{W}^*) \tag{A.4}$$

where

$$\mathbf{g}(\mathbf{W}^*) \triangleq \nabla E_M(\mathbf{W})|_{\mathbf{W}^*}, \tag{A.5a}$$

$$\mathbf{H}(\mathbf{W}^*) \triangleq \nabla\nabla E_M(\mathbf{W})|_{\mathbf{W}^*}. \tag{A.5b}$$

To derive the cost function in Eq (7), we introduce the posterior mean and covariance:

$$\mathbf{m} = \sigma^2 \cdot [\mathbf{g}(\mathbf{W}^*) + \mathbf{H}(\mathbf{W}^*)\mathbf{W}^*], \tag{A.6a}$$

$$\sigma^2 = \left[\mathbf{H}(\mathbf{W}^*) +^{\top} \Gamma^{-1}\right]^{-1}. \tag{A.6b}$$

Then define the following quantities

$$b(\mathbf{W}^*) \triangleq \exp\left\{-\left(\frac{1}{2}\mathbf{W}^{*\top}\mathbf{H}(\mathbf{W}^*)\mathbf{W}^* - \mathbf{W}^{*\top}\mathbf{g}(\mathbf{W}^*) + E(\mathbf{W}^*)\right)\right\}, \tag{A.7a}$$

$$c(\mathbf{W}^*) \triangleq \exp\left\{\frac{1}{2}\hat{\mathbf{g}}(\mathbf{W}^*)^{\top}\mathbf{H}(\mathbf{W}^*)\hat{\mathbf{g}}(\mathbf{W}^*)\right\}, \tag{A.7b}$$

$$d(\mathbf{W}^*) \triangleq \sqrt{|\mathbf{H}(\mathbf{W}^*)|}, \tag{A.7c}$$

$$\hat{\mathbf{g}}(\mathbf{W}^*) \triangleq \mathbf{g}(\mathbf{W}^*) - \mathbf{H}(\mathbf{W}^*)\mathbf{W}^*. \tag{A.7d}$$

Now the approximated likelihood $p(\mathbf{A}|\mathbf{W})$ is a exponential of quadratic:

$$\begin{aligned}
&p(\mathbf{A}|\mathbf{W})\\
=&\exp\{-E(\mathbf{W})\}\\
\approx&\exp\left\{-\left(\frac{1}{2}(\boldsymbol{W}-\boldsymbol{W}^*)^\top\mathbf{H}(\boldsymbol{W}^*)(\boldsymbol{W}-\boldsymbol{W}^*)+(\boldsymbol{W}-\boldsymbol{W}^*)^\top\mathbf{g}(\boldsymbol{W}^*)+E(\mathbf{W}^*)\right)\right\}\\
=&\exp\left\{-\left(\frac{1}{2}\mathbf{W}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{W}+\mathbf{W}^\top\left[\mathbf{g}(\mathbf{W}^*)-\mathbf{H}(\mathbf{W}^*)\mathbf{W}^*\right]\right)\right\}\\
&\cdot\exp\left\{-\left(\frac{1}{2}\mathbf{W}^{*\top}\mathbf{H}(\mathbf{W}^*)\mathbf{W}^*-\mathbf{W}^{*\top}\mathbf{g}(\mathbf{W}^*)+E(\mathbf{W}^*)\right)\right\}\\
=&b(\mathbf{W}^*)\cdot\exp\left\{-\left(\frac{1}{2}\mathbf{W}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{W}+\mathbf{W}^\top\hat{\mathbf{g}}(\mathbf{W}^*)\right)\right\}\\
&\cdot\exp\left\{\frac{1}{2}\hat{\mathbf{g}}(\mathbf{W}^*)^\top\mathbf{H}(\mathbf{W}^*)\hat{\mathbf{g}}(\mathbf{W}^*)-\frac{1}{2}\hat{\mathbf{g}}(\mathbf{W}^*)^\top\mathbf{H}(\mathbf{W}^*)\hat{\mathbf{g}}(\mathbf{W}^*)\right\}\\
=&b(\mathbf{W}^*)\cdot c(\mathbf{W}^*)\\
&\cdot\exp\left\{-\left(\frac{1}{2}\mathbf{W}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{W}+\mathbf{W}^\top\hat{\mathbf{g}}(\mathbf{W}^*)+\frac{1}{2}\hat{\mathbf{g}}(\mathbf{W}^*)^\top\mathbf{H}(\mathbf{W}^*)\hat{\mathbf{g}}(\mathbf{W}^*)\right)\right\}\\
=&(2\pi)^{M/2}b(\mathbf{W}^*)c(\mathbf{W}^*)d(\mathbf{W}^*)\cdot\mathscr{N}(\mathbf{W}|\hat{\mathbf{W}}^*,\mathbf{H}^{-1}(\mathbf{W}^*))\\
\triangleq&A(\mathbf{W}^*)\cdot\mathscr{N}(\mathbf{W}|\hat{\mathbf{W}}^*,\mathbf{H}^{-1}(\mathbf{W}^*)),
\end{aligned}\tag{A.8}$$

where

$$\begin{aligned}
A(\mathbf{W}^*)&=(2\pi)^{M/2}b(\mathbf{W}^*)c(\mathbf{W}^*)d(\mathbf{W}^*),\\
\hat{\mathbf{W}}^*&=-\mathbf{H}^{-1}(\mathbf{W}^*)\hat{\mathbf{g}}(\mathbf{W}^*)=\mathbf{W}^*-\mathbf{H}^{-1}(\mathbf{W}^*)\mathbf{g}(\mathbf{W}^*).
\end{aligned}$$

We can write the approximate marginal likelihood as

$$\begin{aligned}
&A(\mathbf{W}^*)\int\mathscr{N}(\mathbf{W}|\hat{\mathbf{W}}^*,\mathbf{H}^{-1}(\mathbf{W}^*))\cdot\mathscr{N}(\mathbf{W}|\mathbf{0},\Gamma)\prod_{i=1}^{\aleph}\varphi(\gamma_i)d\mathbf{W}\\
=&b(\mathbf{W}^*)\cdot\int\exp\left\{-\left(\frac{1}{2}\mathbf{W}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{W}+\mathbf{W}^\top\hat{\mathbf{g}}(\mathbf{W}^*)\right)\right\}\mathscr{N}(\mathbf{W}|\mathbf{0},\Gamma)\prod_{i=1}^{\aleph}\varphi(\gamma_i)d\mathbf{W}\\
=&\frac{b(\mathbf{W}^*)}{(2\pi)^{\aleph/2}|\Gamma|^{1/2}}\int\exp\{-\hat{E}(\mathbf{W})\}d\mathbf{W}\prod_{i=1}^{\aleph}\varphi(\gamma_i)
\end{aligned}\tag{A.9}$$

where

$$\hat{E}(\mathbf{W})=\frac{1}{2}\mathbf{W}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{W}+\mathbf{W}^\top\hat{\mathbf{g}}(\mathbf{W}^*)+\frac{1}{2}\mathbf{W}^\top\Gamma^{-1}\mathbf{W}.\tag{A.10}$$

Equivalently, we get

$$\hat{E}(\mathbf{W})=\frac{1}{2}(\mathbf{W}-\mathbf{m})^\top(\sigma^2)^{-1}(\mathbf{W}-\mathbf{m})+\hat{E}(\mathbf{A}),\tag{A.11}$$

From (A.6a) and (A.6b), the data-dependent term can be re-expressed as

$$\begin{aligned}
\hat{E}(\mathbf{A})=&\frac{1}{2}\mathbf{m}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{m}+\mathbf{m}^\top\ \mathbf{g}(\mathbf{W}^*)+\frac{1}{2}\mathbf{m}^\top\Gamma^{-1}\mathbf{m}\\
=&\min_{\mathbf{W}}\left[\frac{1}{2}\mathbf{W}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{W}+\mathbf{W}^\top\hat{\mathbf{g}}(\mathbf{W}^*)+\frac{1}{2}\mathbf{W}^\top\Gamma^{-1}\mathbf{W}\right]\\
=&\min_{\mathbf{W}}\left[\frac{1}{2}\mathbf{W}^\top\mathbf{H}(\mathbf{W}^*)\mathbf{W}+\mathbf{W}^\top\left(\mathbf{g}(\mathbf{W}^*)-\mathbf{H}(\mathbf{W}^*)\mathbf{W}^*\right)+\frac{1}{2}\mathbf{W}^\top\Gamma^{-1}\mathbf{W}\right].
\end{aligned}\tag{A.12}$$

Using (A.11), we can evaluate the integral in (A.9) to obtain

$$\int\exp\left\{-\hat{E}(\mathbf{W})\right\}d\mathbf{W}=\exp\left\{-\hat{E}(\mathbf{A})\right\}(2\pi)^{\aleph}|\sigma^2|^{1/2}.\tag{A.13}$$

Applying a $-2\log(\cdot)$ transformation to (A.9), we have

$$\begin{aligned}
& -2\log\left[\frac{b(\mathbf{W}^*)}{(2\pi)^{\aleph/2}|\Gamma|^{1/2}}\int \exp\{-\hat{E}(\mathbf{W})\}d\mathbf{W}\prod_{i=1}^{\aleph}\varphi(\gamma_i)\right] \\
\propto & -2\log\cdot b(\mathbf{W}^*)+\hat{E}(\mathbf{A})+\log|\Gamma|+\log|\mathbf{H}(\mathbf{W}^*)+^{\top}\Gamma^{-1}|-2\sum_{i=1}^{\aleph}\log\varphi(\gamma_i) \\
\propto & \mathbf{W}^{\top}\mathbf{H}(\mathbf{W}^*)\mathbf{W}+2\mathbf{W}^{\top}\hat{\mathbf{g}}(\mathbf{W}^*)+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W} \\
& +\log|\Gamma|+\log|\mathbf{H}(\mathbf{W}^*)+^{\top}\Gamma^{-1}|-2\log\cdot b(\mathbf{W}^*)-2\sum_{i=1}^{\aleph}\log\varphi(\gamma_i).
\end{aligned} \tag{A.14}$$

Therefore we get the following cost function to be minimized over $\mathbf{W},\gamma$

$$\begin{aligned}
\mathscr{L}(\mathbf{W},\gamma) = & \mathbf{W}^{\top}\mathbf{H}(\mathbf{W}^*)\mathbf{W}+2\mathbf{W}^{\top}[\mathbf{g}(\mathbf{W}^*)-\mathbf{H}(\mathbf{W}^*)\mathbf{W}^*]+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W} \\
& +\log|\Gamma|+\log|\mathbf{H}(\mathbf{W}^*)+^{\top}\Gamma^{-1}|-2\log b(\mathbf{W}^*)-2\sum_{i=1}^{\aleph}\log\varphi(\gamma_i).
\end{aligned}$$

It can be easily found that the first line of $\mathscr{L}$ is quadratic programming with ℓ_2 regularize. The second line is all about the hyperparameter γ. Once the estimation on $\mathbf{W}$ and γ are obtained, the cost function is alternatively optimized. The new estimated $\mathbf{W}$ can substitute $\mathbf{W}^*$ and repeat the estimation iteratively.

■

We note that in (A.5), $\mathbf{W}^*$ may not be the mode (i.e., the lowest energy state), which means the gradient term $\mathbf{g}$ may not be zero. Therefore, the selection of $\mathbf{W}_1^*$ remains to be problematic. We give the following Corollary to address this issue, which is more general.

Corollary 1: Suppose

$$\mathbf{W}^* = \arg\min_{\mathbf{W}} E(\mathbf{W})+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W}, \tag{A.15}$$

we define a new cost function

$$\hat{\mathscr{L}}(\mathbf{W},\Gamma) \triangleq E(\mathbf{W})+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W}+\log|\Gamma|+\log|\mathbf{H}(\mathbf{W}^*)+^{\top}\Gamma^{-1}|-2\log b(\mathbf{W}^*)-2\sum_{i=1}^{\aleph}\log\varphi(\gamma_i). \tag{A.16}$$

Instead of minimizing $\mathscr{L}(\mathbf{W},\gamma)$, we can solve the following optimization problem to get $\mathbf{W},\gamma$,

$$\min_{\mathbf{W},\gamma,}\hat{\mathscr{L}}(\mathbf{W},\gamma).$$

Proof: Since the likelihood is

$$p(\mathbf{A}|\mathbf{W}) = \exp\{-E_M(\mathbf{W})\},$$

then $\min_{\mathbf{W}} E(\mathbf{W})+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W}$ is exactly the regularized *maximum likelihood estimation* with ℓ_2 type regularize.

We look at the first part of $\mathscr{L}(\mathbf{W},\gamma)$ in Eq (7), and define them as

$$\mathscr{L}_0(\mathbf{W},\gamma) \triangleq \mathbf{W}^{\top}\mathbf{H}(\mathbf{W}^*)\mathbf{W}+2\mathbf{W}^{\top}[\mathbf{g}(\mathbf{W}^*)-\mathbf{H}(\mathbf{W}^*)\mathbf{W}^*]+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W},$$

then

$$\begin{aligned}
& \min_{\mathbf{W}}\mathscr{L}_0(\mathbf{W},\gamma) \\
= & \min_{\mathbf{W}}\frac{1}{2}(\boldsymbol{W}-\boldsymbol{W}^*)^{\top}\mathbf{H}(\boldsymbol{W}^*)(\boldsymbol{W}-\boldsymbol{W}^*)+(\boldsymbol{W}-\boldsymbol{W}^*)^{\top}\mathbf{g}(\boldsymbol{W}^*)+E_M(\mathbf{W}^*)+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W} \\
\approx & \min_{\mathbf{W}}E_M(\mathbf{W})+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W}
\end{aligned} \tag{A.17}$$

This provides the solution to find the optimal objective at iteration t

$$\begin{aligned}
& \min_{\mathbf{W}}E_M(\mathbf{W}^t)+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W} \\
= & \min_{\mathbf{W}}\frac{1}{2}(\mathbf{W}-\mathbf{W}^t)^{\top}\mathbf{H}(\mathbf{W}^t)(\mathbf{W}-\mathbf{W}^t)+(\mathbf{W}-\mathbf{W}^t)^{\top}\mathbf{g}(\mathbf{W}^t)+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W}
\end{aligned} \tag{A.18}$$

Suppose

$$\mathbf{W}^* = \arg\min_{\mathbf{W}} E(\mathbf{W})+\mathbf{W}^{\top}\Gamma^{-1}\mathbf{W},$$

then inject $\mathbf{W}^*$ into $\min_{\mathbf{W},\gamma,}\mathscr{L}(\mathbf{W},\gamma)$, the original optimization problem Eq (7),i.e could be substituted with (A.16)

■

B. *Updating parameter* $\mathbf{W}$ *and hyper-parameters* γ

In this Section, we propose iterative optimization algorithms to estimate $\mathbf{W}$ and γ alternatively. The $\mathbf{H}(\mathbf{W}^*)$ is a known positive semidefinite symmetric matrix. The proof for proposition 2 will be elaborated firstly.

Proof: Fact on convexity: the function

$$\begin{aligned} u(\mathbf{W},\Gamma) =& \mathbf{W}^\top \mathbf{H}^*(\mathbf{W})\mathbf{W} + 2\mathbf{W}^\top [\mathbf{g}(\mathbf{W}^*) - \mathbf{H}(\mathbf{W}^*)\mathbf{W}^*] + \mathbf{W}^\top \Gamma^{-1}\mathbf{W} \\ \propto& (\mathbf{W}-\mathbf{W}^*)^\top \mathbf{H}(\mathbf{W}^*)(\mathbf{W}-\mathbf{W}^*) + 2\mathbf{W}^\top \mathbf{g}(\mathbf{W}^*) + \mathbf{W}^\top \Gamma^{-1}\mathbf{W} \end{aligned} \tag{B.1}$$

is convex jointly in $\mathbf{W}$, Γ due to the fact that $f(\mathbf{S},Y) = \mathbf{S}\mathbf{A}^{-1}\mathbf{S}$ is jointly convex in $\mathbf{x}$, $\mathbf{A}$ (see, [**?**, p.76]).

Fact on concavity: the function

$$v(\Gamma) = \log|\Gamma| + \log|\Gamma^{-1} + \mathbf{H}(\mathbf{W}^*)| \tag{B.2}$$

is jointly concave in Γ, $\mathbf{\Pi}$. We exploit the properties of the determinant of a matrix

$$|A_{22}||A_{11} - A_{12}A_{22}^{-1}A_{21}| = \left|\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22}\end{pmatrix}\right| = |A_{11}||A_{22} - A_{21}A_{11}^{-1}A_{12}|.$$

Then we have

$$\begin{aligned} v(\Gamma) &= \log|\Gamma| + \log|\Gamma^{-1} + \mathbf{H}(\mathbf{W}^*)| \\ &= \log\left(|\Gamma||\Gamma^{-1} + \mathbf{H}(\mathbf{W}^*)|\right) \\ &= \log\left|\begin{pmatrix} \mathbf{H}(\mathbf{W}^*) & \\ & -\Gamma \end{pmatrix}\right| \\ &= \log\left|\Gamma + \mathbf{H}^{-1}(\mathbf{W}^*)^\top\right| + \log|\mathbf{H}(\mathbf{W}^*)| \end{aligned} \tag{B.3}$$

which is a log-determinant of an affine function of semidefinite matrices $\mathbf{\Pi}$, Γ and hence concave.

Therefore, we can derive the iterative algorithm solving the CCCP. We have the following iterative convex optimization program by calculating the gradient of concave part.

$$\mathbf{W}^t = \arg\min_{\mathbf{W}} u(\mathbf{W}, \Gamma^{t-1}, \mathbf{H}(\mathbf{W}^*)), \tag{B.4}$$

$$\gamma^t = \arg\min_{\gamma\succeq\mathbf{0}} u(\mathbf{W}, \Gamma^{t-1}, \mathbf{H}(\mathbf{W}^*)) + \nabla_\gamma v(\gamma^{t-1}, \mathbf{H}(\mathbf{W}^*))^\top \gamma^{t-1}. \tag{B.5}$$

■

Using basic principles in convex analysis, we then obtain the following analytic form for the negative gradient of $v(\gamma)$ at γ is (using chain rule):

$$\begin{aligned} \boldsymbol{\alpha}^t \triangleq& \nabla_\gamma v(\gamma, \mathbf{H}(\mathbf{W}^*))^\top |_{\gamma=\gamma^t} \\ =& \nabla_\gamma \left(\log|\Gamma^{-1} + \mathbf{H}(\mathbf{W}^*)| + \log|\Gamma|\right)^\top |_{\gamma=\gamma^t} \\ =& -\operatorname{diag}\left\{(\Gamma^t)^{-1}\right\} \circ \operatorname{diag}\left\{\left((\Gamma^t)^{-1} + \mathbf{H}(\mathbf{W}^*)\right)^{-1}\right\} \circ \operatorname{diag}\left\{(\Gamma^t)^{-1}\right\} \\ & + \operatorname{diag}\left\{(\Gamma^k)^{-1}\right\} \\ =& \begin{bmatrix} \alpha_1^t & \cdots & \alpha_{\aleph}^t \end{bmatrix} \end{aligned} \tag{B.6}$$

Then we have:

$$\alpha_i^t = -\frac{((\Gamma^t)^{-1} + \mathbf{H}(\mathbf{W}^*))^{-1}}{(\gamma_i^t)^2} + \frac{1}{\gamma_i^t}. \tag{B.7}$$

Therefore, the iterative procedures (B.4) and (B.5) for $\mathbf{W}^t$ and γ^t can be formulated as

$$\begin{aligned} &\left[\mathbf{W}^{t+1}, \gamma^{t+1}\right] \\ =& \arg\min_{\Gamma\succeq\mathbf{0},\mathbf{W}} (\mathbf{W}-\mathbf{W}^*)^\top \mathbf{H}(\mathbf{W}^*)(\mathbf{W}-\mathbf{W}^*) + 2\mathbf{W}^\top \mathbf{g}(\mathbf{W}^*) + \sum_{i=1}^{\aleph}\left(\frac{\mathbf{W}^\top\mathbf{W}}{\gamma_i} + \alpha_i^t\gamma_i\right) \\ =& \arg\min_{\mathbf{W}} \mathbf{W}^\top \mathbf{H}(\mathbf{W}^*)\mathbf{W} + 2\mathbf{W}^\top\left(\mathbf{g}(\mathbf{W}^*) - \mathbf{H}(\mathbf{W}^*)\mathbf{W}^*\right) + \sum_{i=1}^{\aleph}\left(\frac{\mathbf{W}^\top\mathbf{W}}{\gamma_i} + \alpha_i^t\gamma_i\right). \end{aligned} \tag{B.8}$$

Or in the compact form

$$\left[\mathbf{W}^{t+1}, \gamma^{t+1}\right] = \arg\min_{\mathbf{W}} \mathbf{W}^\top \mathbf{H}(\mathbf{W}^*)\mathbf{W} + 2\mathbf{W}^\top\left(\mathbf{g}(\mathbf{W}^*) - \mathbf{H}(\mathbf{W}^*)\mathbf{W}^*\right) + \mathbf{W}^\top\Gamma^{-1} + \sum_{i=1}^{\aleph}\alpha_i^t\gamma_i. \tag{B.9}$$

Since

$$\frac{\mathbf{W}^\top \mathbf{W}}{\gamma_i} + \alpha_i^t \gamma_i \geq 2 \left| \sqrt{\alpha_i^t} \cdot \mathbf{W} \right|,$$

the optimal γ can be obtained as:

$$\gamma_i = \frac{|\mathbf{W}|}{\sqrt{\alpha_i^t}}, \forall i. \tag{B.10}$$

If we define:

$$\omega_i^t \triangleq \sqrt{\alpha_i^t} = \sqrt{-\frac{((\Gamma^t)^{-1} + \mathbf{H}(\mathbf{W}^*)^t)^{-1}}{(\gamma_i^k)^2} + \frac{1}{\gamma_i^t}}. \tag{B.11}$$

$\mathbf{W}^t$ can be obtained as follows

$$\mathbf{W}^{t+1} = \arg\min_{\mathbf{W}} \frac{1}{2} \mathbf{W}^\top \mathbf{H}(\mathbf{W}^*) \mathbf{W} + \mathbf{W}^\top (\mathbf{g}(\mathbf{W}^*) - \mathbf{H}(\mathbf{W}^*) \mathbf{W}^*) + \sum_{i=1}^{\aleph} \| \omega_i^t \cdot \mathbf{W} \|_{\ell_1}. \tag{B.12}$$

It should be noted here that $\|\omega_i^t \cdot \mathbf{W}\|_{\ell_1}$ represents the regularization term in Eq A.16. We can then inject this into (B.10), which yields

$$\gamma_i^{t+1} = \frac{|\mathbf{W}^{t+1}|}{\omega_i^t}, \forall i. \tag{B.13}$$

We notice that the update for $\mathbf{W}^t$ is to use the *lasso* or ℓ_1-*regularised* regression type optimization. Referring to DDPG[9], the pseudo code combined with DDPG [9] is summarized in Algorithm 2.

Algorithm 2 Bayesian Controller Reduction Algorithm in DDPG [9]

Randomly initialize critic network $Q(s,a|\theta^Q)$ and actor $\mu(s|\theta^\mu)$ with weights θ^Q and θ^μ.
Initialize target network Q' and μ' with weights $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$, θ^μ is denoted as $\mathbf{W}$
Initialize ω and λ, $\forall l = 1, \ldots, L$, $(\omega^\ell)^0 = I$; $\lambda^\ell \in \mathbb{R}^+$;
Initialize replay buffer R
for episode = 1, M **do**
  Initialize a random process $\mathcal{N}$ for action exploration
  Receive initial observation state s_1
  for t = 1, T **do**
    Select action $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}_t$ according to the current policy and exploration noise
    Execute action a_t and observe reward r_t and observe new state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in R
    Sample a random minibatch of N transitions (s_i, a_i, r_i, s_{i+1}) from R
    Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
    Update critic by minimizing the loss: $L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i|\theta^Q))^2$
    Update the actor policy using the sampled policy gradient:

$$\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_i \nabla_a Q(s,a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s|\theta^\mu)|_{s_i} + \lambda^\ell \nabla_{\theta^\mu} R(\omega^\ell \circ \mathbf{W}^\ell)$$

    Update the target networks:

$$\theta^{Q'} \leftarrow \tau \theta^Q + (1-\tau)\theta^{Q'}$$

$$\theta^{\mu'} \leftarrow \tau \theta^\mu + (1-\tau)\theta^{\mu'} \lambda^\ell$$

    Update the Hessian recursively for each layer of the actor network.
    Update hyper-parameters:
    $(\gamma^\ell)^t \leftarrow \text{Update}((\omega^\ell)^t, (\boldsymbol{W}^\ell)^t), (\Gamma^\ell)^t = [(\gamma^\ell)^t]$ {update rules are in Table I}
    $(C^\ell)^t \leftarrow (((\Gamma^\ell)^t)^{-1} + \text{H}^\ell)^t)^{-1}$
    $(\alpha^\ell)^t$ is given by Eq B.7
    $(\omega^\ell)^{t+1} \leftarrow \text{Update}((\alpha^\ell)^t)$
  end for
end for

C. Hessian computation

1) Compute the Hessian of fc layer: The mathematical operation in a fully-connected (fc) layer could be formulated as:

$$h^\ell = a^{\ell-1}\mathbf{W}^\ell, \quad a^\ell = \sigma(h^\ell) \tag{C.1}$$

where h_ℓ is the pre-activation value for layer ℓ and a_ℓ is the activation value. $\sigma()$ is the element-wise activation function. In [25], a recursive method is proposed to compute the Hessian $\mathbf{H}$ in a fc layer:

$$\mathbf{H}^\ell = (a^{\ell-1})^\top \cdot a^{\ell-1} \otimes H^\ell \tag{C.2}$$

where $\otimes$ stands for Kronecker product; H^ℓ denotes the pre-activation Hessian and could be computed with known $H^{\ell+1}$:

$$H^\ell = B^\ell \mathbf{W}^{\ell+1} H^{\ell+1} (\mathbf{W}^{\ell+1})^T B^\ell + D^\ell \tag{C.3}$$

$$B^\ell = \mathrm{diag}(\sigma'(h^\ell)), \quad D^\ell = \mathrm{diag}(\sigma''(h^\ell)\frac{\partial L}{\partial a^\ell}) \tag{C.4}$$

diag() means the operation to expand the input vector into a square matrix by assigning the principal diagonal values with input variable. L stands for the loss function. It can be found that the pre-activation Hessian H^ℓ needs to be computed recursively for each layer before computing the Hessian H.

In order to reduce computation complexity, the original matrix multiplication in Eq C.2-C.3 could be reduced to the vector multiplication without accuracy deterioration:

$$\mathbf{H}^\ell = (a^{\ell-1})^2 \otimes H^\ell \tag{C.5}$$

$$H^\ell = (B^\ell)^2 \circ (H^{\ell+1}(\mathbf{W}^{\ell+1})^2) + D^\ell \tag{C.6}$$

$$B^\ell = \sigma'(h^\ell), \quad D^\ell = \sigma''(h^\ell) \circ \frac{\partial L}{\partial a^\ell} \tag{C.7}$$

2) Compute the Hessian of conv layer: It is difficult to compute the Hessian for conv layer compared to FC layer due to the unstraightforward convolution operation. However, as shown in [26], conv layers can be converted to FC layers thereafter the Hessian of the resulting equivalent FC layer is ready to be obtained. Specifically for layer l, the input vector, weight and output vector are denoted as $B_{\text{in}}^{l-1} \in \mathbb{R}^{b\times C_l \times H_{\text{in}} \times W_{\text{in}}}$, $W^l \in \mathbb{R}^{N_l \times C_l \times m_l \times k_l}$ and $B_{\text{out}}^l \in \mathbb{R}^{b\times N_l \times H_{\text{out}} \times W_{\text{out}}}$ respectively (b is the batch size and $H_{\text{in/out}}$, $W_{\text{in/out}}$ and $C_{\text{in/out}}$ are the sizes of height, width and channel respectively).

As in [26], B_{in} is converted to two dimensional matrix for FC layer with dimension $(bH_{\text{out}}W_{\text{out}}) \times (m_l k_l C_l)$. Similarly, the dimension of B_{out} is changed from $b \times N_l \times H_{\text{out}} \times W_{\text{out}}$ to $(bH_{\text{out}}W_{\text{out}}) \times N_l$. W is changed to $\mathbb{R}^{(C_l k_l m_l)\times N_l}$. The input vector, output vector and weight or the FC layer are denoted as M_{in}, M_{out} and M_k.

Secondly, M_{in} and M_{out} are decomposed into a total of $bH_{\text{out}}W_{\text{out}}$ row vectors $M_{\text{in}}^i \in \mathbb{R}^{bH_{\text{out}}W_{\text{out}}}$ and $M_{\text{out}}^i \in \mathbb{R}^{bH_{\text{out}}W_{\text{out}}}$ respectively. Then we can obtain the Hessian $(\mathbf{H}^l)^i$ for M_k as follows:

$$(\mathbf{H}^l)^i = M_{\text{in}}^i \cdot (M_{\text{in}}^i)^\top \otimes (H^l)^i, \tag{C.8}$$

similar to Eq. (C.3), we have:

$$(H^l)^i = B^l {M_k^{l+1}}^\top (H^{l+1})^i M_k^{l+1} B^l + D^l, \quad B^l = \mathrm{diag}\left(\sigma'(h_{\text{in}}^i)\right), \quad D^l = \mathrm{diag}\left(\sigma''(h_{\text{in}}^i)\frac{\partial L}{\partial M_{\text{in}}^i}\right) \tag{C.9}$$

where h_{in}^i is the pre-activation value. After concatenating all $(H^l)^i$ as

$$H^l = [(H^l)^1; \ldots; (H^l)^{bH_{\text{out}}W_{\text{out}}}], \tag{C.10}$$

the Hessian $\mathbf{H}^l$ for M_k can be obtained as:

$$\mathbf{H}^l = \frac{1}{bH_{\text{out}}\boldsymbol{W}_{\text{out}}} \sum_{i=1}^{bH_{\text{out}}\boldsymbol{W}_{\text{out}}} (\mathbf{H}^l)^i. \tag{C.11}$$

It should be noted that as pre-activation Hessian is a recursive variable for convolutional layer and Hessian will be used for updating hyper-parameters which will be introduced later, both H^l and $\mathbf{H}^l$ should be converted back to conv layer before imparting to next layer with dimension $\mathbb{R}^{b\times C_l \times H_{\text{in}} \times W_{\text{in}}}$ and $\mathbb{R}^{\mathbb{R}^{N_l \times C_l \times m_l \times k_l}}$.