We solve the phase shifter matrix optimization, i.e., P5 via the MM method.

At first, P5 can be transformed as

$$\begin{aligned} \text{PR1: } & \min_{\mathbf{q}} f(\mathbf{q}), \\ & \text{s.t. } \mathbf{q} = \text{vec}(\mathbf{\Phi}) = [\exp(j\theta_1), ..., \exp(j\theta_{N_s})]^{\mathrm{T}}, \end{aligned} \tag{1}$$

where

$$f(\mathbf{q}) = \frac{-\phi_1}{\beta^2 + |\mathbf{\Phi}\mathbf{H}\mathbf{b}|^2} = k\{\mathbf{q}^{\mathrm{H}}\mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{\Sigma}_1\mathbf{q} + 2\Re(\alpha\mathbf{q}^{\mathrm{H}}\mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{H}_b\mathbf{b}) - \alpha^2|\mathbf{H}_b\mathbf{b}|^2 - t\}, \tag{2}$$

$$\mathbf{\Sigma}_1 = -\mathbf{G}_r\text{diag}(\mathbf{H}\mathbf{b}). \tag{3}$$

Since

$$\mathbf{q}^{\mathrm{H}}\mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{\Sigma}_1\mathbf{q} \leq \mathbf{q}^{\mathrm{H}}\mathbf{X}\mathbf{q} - 2\Re\{\mathbf{q}^{\mathrm{H}}(\mathbf{X} - \mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{\Sigma}_1)\mathbf{q}(t)\} + \mathbf{q}(t)^{\mathrm{H}}(\mathbf{X} - \mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{\Sigma}_1)\mathbf{q}(t), \tag{4}$$

where $\mathbf{x} = \lambda_{\max}\mathbf{I}_{N_s}$ and $\lambda_{\max}$ is the maximum eigenvalue of $\mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{\Sigma}_1$. It is obvious that $\{\mathbf{q}^{\mathrm{H}}\mathbf{X}\mathbf{q} + \mathbf{q}(t)^{\mathrm{H}}(\mathbf{X} - \mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{\Sigma}_1)\mathbf{q}(t) - \alpha^2|\mathbf{H}_b\mathbf{b}|^2 - t\}$ is constant. Hence, PR1 can be rewritten as

$$\begin{aligned} \text{PR2: } & \max_{\mathbf{q}} 2\Re(\mathbf{q}^{\mathrm{H}}\mathbf{u}(t)), \\ & \text{s.t. } \mathbf{q} = [\exp(j\theta_1), ..., \exp(j\theta_{N_s})]^{\mathrm{T}}. \end{aligned} \tag{5}$$

where $\mathbf{u}(t) = (\lambda_{\max}\mathbf{I}_{N_s} - \mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{\Sigma}_1)\mathbf{q}(t) - \mathbf{\Sigma}_1^{\mathrm{H}}\mathbf{H}_b\mathbf{b}$. The optimal solution of PR2 is given as

$$\mathbf{q}(t+1) = \exp\{j\text{arg}[\mathbf{u}(t)]\}. \tag{6}$$

Refer to [21], we can get the optimal $\mathbf{q}$ via iteration algorithm with computational complexity $O[N_s^3 + N_{\text{iter}}(8N_s^2 + 4N_s)]$, where N_{iter} is the number of iterations. The computational complexity is similar to that of the manifold optimization method [14].