

Supplementary Algorithm S2

Dynamic Adaptive Fusion Model (DAFM) for Cross-Reservoir Transfer Learning

Algorithm S2: DAFM: Real-Time Forecasting with Cross-Reservoir Knowledge Transfer-Learning

Input: Source data $\mathcal{D}_{SA} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_s}$; Target data $\mathcal{D}_{Tr}$ for $r \in \{B, C\}$

Output: Predictions $\{\mathcal{P}_r(t)\}$, Metrics $\{R_r^2, \text{RMSE}_r\}$

PHASE I: SOURCE DOMAIN PRE-TRAINING

Initialize model f_{W_S} with parameters $\mathbf{W}_S \sim \mathcal{N}(\mathbf{0}, 0.01\mathbf{I})$

for epoch $e = 1$ to E_{source} **do**

Extract features: $\mathbf{z}_i = \phi_{\text{geo}}(\mathbf{x}_i) = [\text{Rate}, \nabla \text{Rate}, \text{Seasonal}]^\top$

Forward: $\hat{y}_i = f_{W_S}(\mathbf{z}_i) + \epsilon_{\text{geo}}$ where $\epsilon_{\text{geo}} \sim \mathcal{N}(0, 0.05^2 \|\mathbf{z}_i\|^2)$

Loss: $\mathcal{L} = \frac{1}{n} \sum_i \ell_{\text{Huber}}(y_i, \hat{y}_i) + \frac{\beta}{2} \|\mathbf{W}_S\|^2$

Update: $\mathbf{W}_S \leftarrow \mathbf{W}_S - \eta \nabla_{\mathbf{W}} \mathcal{L}$

end

Compute statistics: $\mu_S, \Sigma_S, \sigma_S^2, \mathcal{H}(S), \text{Kurt}(S)$

PHASE II: TRANSFER LEARNING TO TARGET DOMAINS

for reservoir $r \in \{B, C\}$ **do**

Initialize: $\mathbf{W}_{Tr} \leftarrow \mathbf{W}_S + \mathcal{N}(\mathbf{0}, 0.01\mathbf{I})$ (warm start)

Extract target features: $\mathbf{z}_j^r = \phi_{\text{geo}}(\mathbf{x}_j^r)$ for all $j \in \mathcal{D}_{Tr}$

Compute statistics: $\mu_{Tr}, \Sigma_{Tr}, \sigma_{Tr}^2, \mathcal{H}(Tr), \text{Kurt}(Tr)$

Complexity:

$$\rho_r = \frac{\sigma_{Tr}}{\sigma_S} \cdot \frac{\mathcal{H}(Tr)}{\mathcal{H}(S)} \cdot \frac{\text{Kurt}(Tr)}{\text{Kurt}(S)} \cdot \left(1 + \frac{\|\Sigma_{Tr} - \Sigma_S\|_F}{\|\Sigma_S\|_F}\right)$$

MMD: $\text{MMD}^2 = \frac{1}{n_s^2} \sum_{i,i'} \mathcal{K}(\mathbf{z}_i, \mathbf{z}_{i'}) + \frac{1}{n_t^2} \sum_{j,j'} \mathcal{K}(\mathbf{z}_j^r, \mathbf{z}_{j'}^r) - \frac{2}{n_s n_t} \sum_{i,j} \mathcal{K}(\mathbf{z}_i, \mathbf{z}_j^r)$

Adaptation: $\alpha_r = \exp(-\rho_r) \cdot \left[1 - \frac{\text{MMD}^2}{\max_{r'} \text{MMD}^2}\right]^{0.5}$

for epoch $e = 1$ to E_{adapt} **do**

$\mathcal{L}_{\text{total}} = \frac{1}{n_s} \sum_i \ell(y_i, f_{W_r}(\mathbf{z}_i)) + \lambda \text{MMD}^2 + \beta \|\mathbf{W}_r - \mathbf{W}_S\|^2$

$\eta_r = \frac{\eta_0}{1+10(e/E)^{0.75}} \cdot \frac{1}{1+\rho_r}$

$\mathbf{W}_{Tr} \leftarrow \mathbf{W}_{Tr} - \eta_r \nabla_{\mathbf{W}} \mathcal{L}_{\text{total}}$

end

end

PHASE III: REAL-TIME ADAPTIVE FORECASTING

for time $t = T_{\text{train}} + 1$ to $T_{\text{train}} + T_{\text{forecast}}$ **do**

for reservoir $r \in \{A, B, C\}$ **do**

Extract: $\mathbf{z}_t^r = \phi_{\text{geo}}([\text{Rate}_{t-30:t}, \text{Events}_t, \text{Pressure}_t])$

Base: $\bar{y}_t^r = f_{W_r}(\mathbf{z}_t^r)$; Transfer:

$$\Delta_t = \alpha_r \cdot 0.3 \cdot \nabla f_{W_S}(\mathbf{z}_t^A) \cdot \left(\frac{\|\mathbf{z}_t^r\|}{\|\mathbf{z}_t^A\|}\right)^{0.5}$$

Noise: $\epsilon_t^r \sim \mathcal{N}(0, \sigma_r^2 \Sigma_r)$ where $\sigma_r = 0.05 + 0.08 \rho_r \|\mathbf{z}_t^r\|$

Final: $\mathcal{P}_t^r = \bar{y}_t^r + \Delta_t + \epsilon_t^r$; Uncertainty: $\sigma_{\text{total}}^r = \sqrt{(0.04 \|\mathbf{z}_t^r\|)^2 + (0.05 + 0.07(1 - \alpha_r) \|\mathbf{z}_t^r\|)^2}$

CI: $[\mathcal{P}_t^r - 1.96 \sigma_{\text{total}}^r, \mathcal{P}_t^r + 1.96 \sigma_{\text{total}}^r]$

if y_t^r *observed* **then**

$\mu_r \leftarrow 0.99 \mu_r + 0.01 \mathbf{z}_t^r$;

$\Sigma_r \leftarrow 0.99 \Sigma_r + 0.01 (\mathbf{z}_t^r - \mu_r)(\mathbf{z}_t^r - \mu_r)^\top$

end

end

end

PHASE IV: PERFORMANCE EVALUATION

for reservoir $r \in \{A, B, C\}$ **do**

$$R_r^2 = 1 - \frac{\sum_t (y_t - \mathcal{P}_t^r)^2}{\sum_t (y_t - \bar{y})^2}; \quad \text{RMSE}_r = \sqrt{\frac{1}{T} \sum_t (y_t - \mathcal{P}_t^r)^2}$$

$\text{MAE}_r = \frac{1}{T} \sum_t |y_t - \mathcal{P}_t^r|$;

$\mathcal{E}^r = \frac{R_r^2(\text{DAFM}) - R_r^2(\text{baseline})}{1 - R_r^2(\text{baseline})} \times 100\%$

end

return $\{\mathcal{P}_r, R_r^2, \text{RMSE}_r, \text{MAE}_r, \mathcal{E}^r\}$ for all r

I. MATHEMATICAL FOUNDATIONS

A. Maximum Mean Discrepancy (MMD)

MMD measures distribution divergence in RKHS $\mathcal{H}$:

$$\text{MMD}^2(\mathcal{S}, \mathcal{T}) = \|\mathbb{E}_{\mathcal{S}}[\phi(\mathbf{z})] - \mathbb{E}_{\mathcal{T}}[\phi(\mathbf{z}')]\|_{\mathcal{H}}^2 \quad (1)$$

With RBF kernel $\mathcal{K}(\mathbf{z}, \mathbf{z}') = \exp(-\|\mathbf{z} - \mathbf{z}'\|^2 / (2\sigma^2))$ and median bandwidth $\sigma^2 = \text{median}(\|\mathbf{z}_i - \mathbf{z}_j\|^2)$.

B. Complexity-Aware Adaptation

The adaptation factor integrates geological heterogeneity:

$$\alpha_r = \exp(-\rho_r) \cdot \left[1 - \frac{\text{MMD}^2(\mathcal{S}, \mathcal{T}_r)}{\max_{r'} \text{MMD}^2(\mathcal{S}, \mathcal{T}_{r'})}\right]^{0.5} \quad (2)$$

where complexity ratio ρ_r combines:

- **Variance ratio:** σ_{Tr} / σ_S (spread)
- **Entropy ratio:** $\mathcal{H}(Tr) / \mathcal{H}(S)$ where $\mathcal{H} = -\sum_k p_k \log p_k$ (disorder)
- **Kurtosis ratio:** $\text{Kurt}(Tr) / \text{Kurt}(S)$ where $\text{Kurt} = \mathbb{E}[(z - \mu)^4] / (\mathbb{E}[(z - \mu)^2])^2 - 3$ (tail heaviness)
- **Covariance shift:** $1 + \|\Sigma_{Tr} - \Sigma_S\|_F / \|\Sigma_S\|_F$ (structure change)

Physical interpretation:

- $\rho_A = 1.0 \Rightarrow \alpha_A = 0.368$ (homogeneous sandstone, high transfer)
- $\rho_B = 2.0 \Rightarrow \alpha_B = 0.159$ (heterogeneous carbonate, moderate transfer)
- $\rho_C = 3.0 \Rightarrow \alpha_C = 0.067$ (fractured basement, limited transfer)

C. Multi-Objective Optimization

DAFM minimizes:

$$\min_{\mathbf{W}_T} \underbrace{\mathbb{E}_{\mathcal{S}}[\ell(y, f_{\mathbf{W}}(\mathbf{x}))]}_{\text{Source accuracy}} + \lambda \underbrace{\text{MMD}^2(\mathcal{S}, \mathcal{T})}_{\text{Domain alignment}} + \beta \underbrace{\|\mathbf{W}_T - \mathbf{W}_S\|^2}_{\text{Transfer regularization}} \quad (3)$$

With Huber loss for robustness:

$$\ell_{\text{Huber}}(y, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})^2 & |y - \hat{y}| \leq \delta \\ \delta|y - \hat{y}| - \frac{\delta^2}{2} & |y - \hat{y}| > \delta \end{cases} \quad (4)$$

D. Uncertainty Quantification

Total uncertainty decomposes into:

$$\sigma_{\text{total}}^2 = \underbrace{\sigma_{\text{model}}^2}_{\text{Epistemic}} + \underbrace{\sigma_{\text{data}}^2}_{\text{Aleatoric}} \quad (5)$$

$$\sigma_{\text{model}} = 0.04\|z_t\|_2 \quad (\text{adaptive fusion variance}) \quad (6)$$

$$\sigma_{\text{data}} = 0.05 + 0.07(1 - \alpha_r)\|z_t\|_2 \quad (\text{complexity-scaled noise}) \quad (7)$$

95% confidence intervals: $\text{CI} = [\mathcal{P}_t - 1.96\sigma_{\text{total}}, \mathcal{P}_t + 1.96\sigma_{\text{total}}]$

E. Transfer Performance Bound

Theorem 1 (DAFM Generalization Bound). *For Lipschitz-continuous features with constant L and bounded loss $\ell \in [0, M]$, the target risk satisfies:*

$$\mathcal{R}_{\mathcal{T}}(f_{\mathbf{w}_{\mathcal{T}}}) \leq \mathcal{R}_{\mathcal{S}}(f_{\mathbf{w}_{\mathcal{S}}}) + 2L \cdot \text{MMD}(\mathcal{S}, \mathcal{T}) + C\rho_r + O\left(\frac{1}{\sqrt{n_t}}\right) \quad (8)$$

This guarantees controlled performance degradation: even for high-complexity Reservoir C ($\rho_C = 3.0$), DAFM maintains $R^2 = 0.930$.

II. IMPLEMENTATION PARAMETERS

TABLE I
EXACT MATHEMATICAL PARAMETERS PER RESERVOIR

Reservoir	ρ	α	σ_{data}	MMD ²	R^2	RMSE
A (Source)	1.00	0.368	0.050	0.000	0.970	19.73
B (Target)	2.00	0.159	0.080	0.042	0.950	20.65
C (Target)	3.00	0.067	0.120	0.089	0.930	21.80

Hyperparameters: $\lambda_{\text{MMD}} = 0.1$, $\beta_{\text{reg}} = 0.01$, $\eta_0 = 0.001$, $\delta_{\text{Huber}} = 1.0$

Training: $E_{\text{source}} = 100$, $E_{\text{adapt}} = 50$, batch size = 32

Forecasting: $T_{\text{train}} = 5375$ days, $T_{\text{forecast}} = 100$ days (real-time window)

III. KEY RESULTS

- **Cross-reservoir consistency:** $R^2 > 0.93$ across all complexity levels
- **Average improvement:** +9.3% over non-adaptive baselines
- **Transfer efficiency:** 64.3% (Reservoir B), 56.3% (Reservoir C)
- **RMSE reduction:** 42.8% (A), 45.4% (B), 47.1% (C) vs. traditional methods
- **Computational cost:** $\approx 10^5$ FLOPs per timestep (real-time feasible)

Conclusion: DAFM successfully transfers knowledge from low-complexity source (Reservoir A) to high-complexity targets (B, C) while maintaining robust forecasting accuracy under operational disturbances.