Joint Sampling and Optimisation for Inverse Rendering: Supplementary Material

Martin Balint
Max Planck Institute for Informatics
Saarbruecken, Germany
mbalint@mpi-inf.mpg.de

Karol Myszkowski
Max Planck Institute for Informatics
Saarbruecken, Germany
karol@mpi-inf.mpg.de

Hans-Peter Seidel
Max Planck Institute for Informatics
Saarbruecken, Germany
hpseidel@mpi-inf.mpg.de

Gurprit Singh
Max Planck Institute for Informatics
Saarbruecken, Germany
gsingh@mpi-inf.mpg.de

CCS CONCEPTS

• **Computing methodologies** → *Image processing*; **Ray tracing**.

ACM Reference Format:
Martin Balint, Karol Myszkowski, Hans-Peter Seidel, and Gurprit Singh. 2023. Joint Sampling and Optimisation for Inverse Rendering: Supplementary Material. In *SIGGRAPH Asia 2023 Conference Papers (SA Conference Papers '23), December 12–15, 2023, Sydney, NSW, Australia.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3610548.3618244

A START-UP BIAS

To show our start-up bias, we rewrite Equation 7 as:

$$\langle F_i \rangle_M = \sum_{j=1}^{i-1} \prod_{k=j+1}^{i} [\alpha_k](1-\alpha_j)\langle F_j \rangle + \prod_{k=1}^{i}[\alpha_k]\langle F_0 \rangle + (1-\alpha_i)\langle F_i \rangle + \sum_{j=1}^{i}\prod_{k=j}^{i}[\alpha_k]\langle \Delta F_j \rangle \, . \quad (1)$$

$\langle F_0 \rangle$'s contribution diminishes over time. As $\beta_F \to 1$, the correlation between α and $\langle F_i \rangle$ also diminishes, leaving us with correlation between α_k and $\langle \Delta F_j \rangle$ as the main source of bias. As optimisation converges we expect our step size to shrink and $\mathrm{Var}[\langle \Delta F_i \rangle]$ to decline. Thus, as $\mathrm{Var}[\langle F_i \rangle]$ dominates Equation 8, α_k and $\mathrm{Var}[\langle \Delta F_j \rangle]$, become increasingly less correlated, thus, by extension, reducing correlation with $\langle \Delta F_j \rangle$ (Equation 14).

Growing β_F over the optimisation process is a simple way to make our meta-estimators consistent. However, in practice, we have found no benefit to this approach compared to setting β_F to a fixed value that we experimentally fit to the noise characteristics of the gradient estimator $\langle F_i \rangle$.

B PSEUDOCODE

Variance approximation with zero-centred, second moment EMAs:

```
class Moment2:
    def __init__(self, shape, decay = 0.9):
        self.decay = decay
        self.count = 0
        self.m2 = 0

    def step(self, x):
        self.count += 1
        w = 1.0 - self.decay
        w_sum = 1.0 - self.decay ** self.count

        self.m2 = self.m2 + (w / w_sum) * (x ** 2 - self.m2)
```

Meta-estimation:

```
class Meta:
    def __init__(self, η = 0.001):
        self.η = η
        self.var = 0
        self.mean = 0
        self.α_{i-1} = -∞

    def step(self, ⟨F_i⟩, ⟨ΔF_i⟩, Var[⟨F_i⟩], Var[⟨ΔF_i⟩]):
        self.mean += ⟨ΔF_i⟩
        self.var += Var[⟨ΔF_i⟩]

        α = Var[⟨F_i⟩] / (Var[⟨F_i⟩] + self.var + 1e-30) # Inverse variance weighting
        α = min(α, 1 / (2 - self.α_{i-1})) # Alpha clipping
        self.α_{i-1} = α

        self.mean = α * self.mean + (1 - α) * ⟨F_i⟩
        self.var = α ** 2 * self.var + (1 - α) ** 2 * Var[⟨F_i⟩]

        return - self.η * self.mean / (sqrt(self.var) + 1e-8)
```

Optimisation loop snippet:

SA Conference Papers '23, December 12–15, 2023, Sydney, NSW, Australia

ACM ISBN 979-8-4007-0315-7/23/12.
https://doi.org/10.1145/3610548.3618244

```
Var[⟨F_i⟩] = Moment2()
Var[⟨ΔF_i⟩]_D = Moment2()
meta = Meta()

...

Var[⟨F_i⟩].step(⟨F_i⟩)
Var[⟨ΔF_i⟩]_D.step(⟨ΔF_i⟩ / sqrt(Δπ_i^2))
Δπ_{i+1} = meta.step(diff, ⟨F_i⟩, ⟨ΔF_i⟩, Var[⟨F_i⟩].m2,
↪  Var[⟨ΔF_i⟩]_D.m2 * Δπ_i^2)
π_{i+1} = π_i + Δπ_{i+1}
```