Explanation of Equations (10)-(14) from Diffusion-DPO

The core idea is to adapt the Direct Preference Optimization (DPO) framework, originally designed for language models, to diffusion models for image generation. We assume familiarity with the preceding equations, particularly Eq. (5) (RLHF objective), Eq. (6) (Optimal DPO solution), Eq. (8) (DPO loss for language models), and Eq. (9) (Reward over diffusion paths).

Equation (10): Adapting the RLHF Objective to Diffusion Paths

The objective is to find a model p_θ that maximizes reward while staying close to a reference model p_{ref}, now considering entire diffusion paths $x_{0:T}$.

$$\max_{p_\theta} \mathbb{E}_{c\sim\mathcal{D}_c, x_{0:T}\sim p_\theta(x_{0:T}|c)}[r(c, x_0)] - \beta \mathrm{D}_{\text{KL}}[p_\theta(x_{0:T}|c)||p_{\text{ref}}(x_{0:T}|c)] \quad (10)$$

Motivation: This equation adapts the standard Reinforcement Learning from Human Feedback (RLHF) objective (Eq. 5) for diffusion models. Since the direct marginal likelihood $p_\theta(x_0|c)$ is intractable (it requires marginalizing over all diffusion paths $x_{1:T}$ leading to x_0), the optimization is framed over the joint distribution of paths $p_\theta(x_{0:T}|c)$.

Derivation/Explanation:

- **First Term (Expected Reward):** $\mathbb{E}_{c\sim\mathcal{D}_c, x_{0:T}\sim p_\theta(x_{0:T}|c)}[r(c, x_0)]$. Instead of an expectation over final samples $x_0 \sim p_\theta(x_0|c)$, we consider the expectation over full paths $x_{0:T} = (x_0, x_1, \ldots, x_T)$ generated by the model $p_\theta(x_{0:T}|c)$. The reward $r(c, x_0)$ is implicitly understood as $R(c, x_{0:T})$, the reward associated with the entire path that generated x_0, or as $E_{p_\theta(x'_{1:T}|x_0,c)}[R(c, x_0, x'_{1:T})]$ as defined in Eq. (9). The paper uses $r(c, x_0)$ as shorthand for the path-dependent reward concept.

- **Second Term (KL Regularization):** $\beta \mathrm{D}_{\text{KL}}[p_\theta(x_{0:T}|c)||p_{\text{ref}}(x_{0:T}|c)]$. The KL divergence term from the original RLHF objective, $\mathrm{D}_{\text{KL}}[p_\theta(x_0|c)||p_{\text{ref}}(x_0|c)]$, is also intractable for diffusion models. Following prior work, this is replaced by its upper bound, the joint KL divergence over entire diffusion paths. This joint KL divergence is more manageable because $p_\theta(x_{0:T}|c) = p_\theta(x_T|c)\prod_{k=1}^{T} p_\theta(x_{k-1}|x_k, c)$, where each term $p_\theta(x_{k-1}|x_k, c)$ is a Gaussian (from Eq. 1 of the paper).

In essence, Eq. (10) reformulates the RLHF objective for full diffusion paths, aiming to find a diffusion process p_θ that generates high-reward paths while staying close to a reference diffusion process p_{ref}.

Equation (11): The DPO Objective for Diffusion Paths

This equation translates the DPO objective (like Eq. 8 for language models) to the context of diffusion model paths, using the objective from Eq. (10). The paper omits c for compactness; x_w, x_l are x_0^w, x_0^l.

$$\mathcal{L}_{\text{DPO-Diffusion}}(\theta) = -\mathbb{E}_{(x_w,x_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\mathbb{E}_{x_{1:T}^w\sim p_\theta(x_{1:T}|x_0^w)}\left[\log\frac{p_\theta(x_{0:T}^w)}{p_{\text{ref}}(x_{0:T}^w)}\right] - \beta\mathbb{E}_{x_{1:T}^l\sim p_\theta(x_{1:T}|x_0^l)}\left[\log\frac{p_\theta(x_{0:T}^l)}{p_{\text{ref}}(x_{0:T}^l)}\right]\right)\right] \quad (11)$$

Motivation: The goal is to directly optimize the conditional path distribution $p_\theta(x_{0:T}|c)$ using the DPO framework. This avoids explicit reward modeling and subsequent reinforcement learning.

Derivation/Explanation: This equation parallels the structure of the DPO loss for language models (Eq. 8).

- $\mathbb{E}_{(x_w,x_l)\sim\mathcal{D}}$: Expectation over pairs of preferred (x_0^w) and dispreferred (x_0^l) final images from the human preference dataset $\mathcal{D}$.

- $\log\sigma(\cdot)$: The logistic sigmoid function, central to the Bradley-Terry model for pairwise preferences, determines the probability of x_0^w being preferred over x_0^l.

- The terms inside $\log\sigma(\cdot)$ represent the difference in implicit rewards. For the winning image x_0^w, the implicit reward is:

$$\beta\mathbb{E}_{x_{1:T}^w\sim p_\theta(x_{1:T}|x_0^w)}\left[\log\frac{p_\theta(x_{0:T}^w)}{p_{\text{ref}}(x_{0:T}^w)}\right]$$

  This term involves:

  1. Given x_0^w, considering all possible reverse diffusion paths $x_{1:T}^w$ that could lead to it, sampled according to the current model $p_\theta(x_{1:T}|x_0^w)$.

2. For each such path $x_{0:T}^w = (x_0^w, x_{1:T}^w)$, calculating the log-ratio of its probability under the model p_θ versus the reference model p_{ref}.
3. Taking the expectation of this log-ratio over all such paths originating from x_0^w.

A similar term exists for the losing image x_0^l.

- **Challenge:** The expectations $\mathbb{E}_{x_{1:T} \sim p_\theta(x_{1:T}|x_0)}$ are problematic because sampling from $p_\theta(x_{1:T}|x_0)$ (the reverse diffusion process starting from a known x_0) is:
  1. **Inefficient:** Requires T sampling steps (e.g., $T \approx 1000$).
  2. **Intractable for training in this form:** $p_\theta(x_{1:T}|x_0)$ depends on the model p_θ being trained. This makes it an on-policy objective, complex for gradient-based optimization, especially with the expectation inside the $\log \sigma$.

Equation (12): Bounding the DPO Objective for Tractability

To make Eq. (11) trainable, it's bounded. (Note: x_w, x_l are x_0^w, x_0^l. The notation $p_\theta(x_{t-1}, x_t|x_0)$ implies sampling a segment of the reverse path.)

$$\mathcal{L}_{\text{DPO-Diffusion}}(\theta) \leq -\mathbb{E}_{\substack{(x_w, x_l) \sim \mathcal{D}, t \sim U(0,T) \\ x_{t-1}^w, x_t^w \sim p_\theta(x_{t-1}, x_t | x_0^w) \\ x_{t-1}^l, x_t^l \sim p_\theta(x_{t-1}, x_t | x_0^l)}} \left[\log \sigma \left(\beta T \log \frac{p_\theta(x_{t-1}^w | x_t^w)}{p_{\text{ref}}(x_{t-1}^w | x_t^w)} - \beta T \log \frac{p_\theta(x_{t-1}^l | x_t^l)}{p_{\text{ref}}(x_{t-1}^l | x_t^l)} \right) \right] \tag{12}$$

Motivation: To address the intractability of Eq. (11), this step introduces simplifications using Jensen's inequality and a single-timestep sampling approximation.

Derivation/Explanation:

- **Step 1: Decomposing log-ratios and Jensen's Inequality.** The log-ratio of path probabilities can be decomposed:

$$\log \frac{p_\theta(x_{0:T})}{p_{\text{ref}}(x_{0:T})} = \log \frac{p_\theta(x_T)}{p_{\text{ref}}(x_T)} + \sum_{k=1}^{T} \log \frac{p_\theta(x_{k-1}|x_k)}{p_{\text{ref}}(x_{k-1}|x_k)}$$

  The paper states: "we substitute the reverse decompositions for p_θ and p_{ref}, and utilize Jensen's inequality and the convexity of function $-\log \sigma$ to push the expectation outside." Since $-\log \sigma(X)$ is convex, $E[-\log \sigma(X)] \geq -\log \sigma(E[X])$. This implies $E[\log \sigma(X)] \leq \log \sigma(E[X])$. Applying this to Eq. (11), the path expectations $E_{x_{1:T} \sim p_\theta(\cdot|x_0)}$ are moved outside the $\log \sigma$ function. This results in an upper bound on the loss because we are minimizing $-\mathbb{E}[\log \sigma(\Delta R)]$, and $-\mathbb{E}[\log \sigma(\Delta R)] \leq -\log \sigma(\mathbb{E}[\Delta R])$.

- **Step 2: Single Timestep Sampling.** The sum $\sum_{k=1}^{T} \log(\dots)$ is computationally expensive. A common approximation (used in standard diffusion model training) is to sample a single timestep t uniformly from $[1, T]$ and multiply its contribution by T:

$$\sum_{k=1}^{T} A_k \approx T \cdot \mathbb{E}_{t \sim U(1,T)}[A_t]$$

  The paper appears to drop the initial state term $\log(p_\theta(x_T)/p_{\text{ref}}(x_T))$, likely assuming its contribution is small or can be absorbed. Thus, the inner term $\beta \mathbb{E}_{\text{path}}[\log(p_\theta(\text{path})/p_{\text{ref}}(\text{path}))]$ from Eq. (11) is approximated by:

$$\beta T \mathbb{E}_{t \sim U(0,T), (x_{t-1}, x_t) \sim p_\theta(\cdot|x_0)} \left[\log \frac{p_\theta(x_{t-1}|x_t)}{p_{\text{ref}}(x_{t-1}|x_t)} \right]$$

  The expectation is now over (x_0^w, x_0^l) from the dataset $\mathcal{D}$, a random timestep t, and single transitions (x_t^w, x_{t-1}^w) and (x_t^l, x_{t-1}^l) sampled from the respective reverse processes conditioned on x_0^w and x_0^l. The factor T arises from this single-timestep approximation of the sum.

- **Remaining Challenge:** Sampling (x_{t-1}, x_t) from $p_\theta(\cdot|x_0)$ (a segment of the reverse path) is still on-policy and depends on the model p_θ being trained.

Equation (13): Approximating Reverse Process with Forward Process

The problematic sampling from p_θ in Eq. (12) is addressed by approximating with the forward process q. (Simplified notation for q: the paper implies $q(x_{t-1}|x_t, x_0)$ when writing $q(x_{t-1}|x_t, t)$.)

$$\mathcal{L}(\theta) = -\mathbb{E}_{\substack{(x_w,x_l)\sim\mathcal{D},t\sim U(0,T)\\ x_t^w\sim q(x_t|x_0^w),x_t^l\sim q(x_t|x_0^l)}}\Bigg[\log\sigma\Bigg(-\beta T\Bigg(+ \mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t^w, x_0^w)||p_\theta(x_{t-1}|x_t^w)) - \mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t^w, x_0^w)||p_{\mathrm{ref}}(x_{t-1}|x_t^w)) - \mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t^l, x_0^l)||p_\theta(x_{t-1}|x_t^l)) + \mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t^l, x_0^l)||p_{\mathrm{ref}}(x_{t-1}|x_t^l))\Bigg)\Bigg)\Bigg] \tag{13}$$

Motivation: To make the objective practical, the on-policy sampling from $p_\theta(x_{t-1}, x_t|x_0)$ in Eq. (12) is replaced. The key idea is to use the tractable forward noising process q.

Derivation/Explanation:

- **Key Approximation:** "So we approximate the reverse process $p_\theta(x_{1:T}|x_0)$ with the forward $q(x_{1:T}|x_0)$." This means that for sampling x_t, we use $x_t \sim q(x_t|x_0)$, which is a known Gaussian $N(x_t; \alpha_t x_0, \sigma_t^2 I)$ (using paper's α_t, σ_t from Eq. 1 context; or $\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I$ in DDPM terms). The "true" reverse step if x_0 were perfectly known is $q(x_{t-1}|x_t, x_0)$, also a known Gaussian.
- **Replacing Log-Ratios with KL Divergences:** This is the "some algebra" step. The intuition is that the difference of log-likelihood ratios $\log(p_\theta/p_{\mathrm{ref}})$ in Eq. (12) relates to how well p_θ and p_{ref} model the "ideal" denoising step, represented by $q(x_{t-1}|x_t, x_0)$. A term like $\log\frac{p_\theta(x_{t-1}|x_t)}{p_{\mathrm{ref}}(x_{t-1}|x_t)}$ can be rewritten using a reference distribution $Q = q(x_{t-1}|x_t, x_0)$:

  $$\log\frac{p_\theta(x_{t-1}|x_t)}{Q} - \log\frac{p_{\mathrm{ref}}(x_{t-1}|x_t)}{Q}$$

  Each $\log(P/Q)$ term is related to $-\mathrm{D}_{\mathrm{KL}}(Q||P)$ if an expectation with respect to Q is taken. The paper effectively substitutes these log-ratios with differences of KL divergences directly inside the $\log\sigma$.

  - For x_0^w, the term $\log\frac{p_\theta(x_{t-1}^w|x_t^w)}{p_{\mathrm{ref}}(x_{t-1}^w|x_t^w)}$ is replaced by the difference:

    $$-\mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t^w, x_0^w)||p_\theta(x_{t-1}|x_t^w)) + \mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t^w, x_0^w)||p_{\mathrm{ref}}(x_{t-1}|x_t^w))$$

    This is then negated due to the outer $-\beta T$ factor, leading to the signs in Eq. (13). The first KL term measures how well p_θ approximates the ideal single denoising step. The second measures how well p_{ref} does.

  The overall structure for the argument of $\log\sigma$ becomes $-\beta T\times[(\mathrm{KL}_{\theta,w} - \mathrm{KL}_{\mathrm{ref},w}) - (\mathrm{KL}_{\theta,l} - \mathrm{KL}_{\mathrm{ref},l})]$.
- **Advantage:** All x_t states are now sampled from the fixed, known forward process $q(x_t|x_0)$, which is off-policy and efficient.

Equation (14): Simplifying KL Divergences to Denoising Errors

This final form expresses the KL divergences from Eq. (13) in terms of the model's noise prediction errors. (Note: $x_t^* = \alpha_t x_0^* + \sigma_t\epsilon^*$ means $x_t^w = \alpha_t x_0^w + \sigma_t\epsilon$ and $x_t^l = \alpha_t x_0^l + \sigma_t\epsilon$ using the *same* noise sample ϵ for a given (x_0^w, x_0^l, t) triplet).

$$\mathcal{L}(\theta) = -\mathbb{E}_{\substack{(x_w,x_l)\sim\mathcal{D},t\sim U(0,T)\\ \epsilon\sim N(0,I)\\ x_t^w=\alpha_t x_0^w+\sigma_t\epsilon\\ x_t^l=\alpha_t x_0^l+\sigma_t\epsilon}}\Bigg[\log\sigma\Bigg(-\beta T w(t)\Bigg(||\epsilon-\epsilon_\theta(x_t^w, t)||^2 - ||\epsilon-\epsilon_{\mathrm{ref}}(x_t^w, t)||^2 - (||\epsilon-\epsilon_\theta(x_t^l, t)||^2 - ||\epsilon-\epsilon_{\mathrm{ref}}(x_t^l, t)||^2)\Bigg)\Bigg)\Bigg] \tag{14}$$

Motivation: To obtain a practical training objective, the KL divergences in Eq. (13) are converted into measurable quantities related to the neural network's output, which is typically a noise predictor $\epsilon_\theta(x_t, t)$.

Derivation/Explanation:

- **KL Divergence to MSE of Noise Predictors:** In diffusion models (like DDPM or as per Eq. 1 of the paper), the reverse transition $p_\theta(x_{t-1}|x_t)$ is a Gaussian. Its mean $\mu_\theta(x_t, t)$ is parameterized using the noise prediction $\epsilon_\theta(x_t, t)$. The "ideal" reverse transition $q(x_{t-1}|x_t, x_0)$ is also a Gaussian. Its mean $\mu_q(x_t, x_0, t)$ can be related to the original noise ϵ that produced x_t from x_0 (i.e., $x_t = \alpha_t x_0 + \sigma_t \epsilon$ as in the paper's notation, or $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ in DDPM). The KL divergence $\mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t, x_0)||p_\theta(x_{t-1}|x_t))$ between these two Gaussians (which have related variances) simplifies to be proportional to the squared difference of their means. This, in turn, is proportional to the squared error of the noise prediction:

  $$\mathrm{D}_{\mathrm{KL}}(q(x_{t-1}|x_t, x_0)||p_\theta(x_{t-1}|x_t)) \propto w(t)||\epsilon - \epsilon_\theta(x_t, t)||^2$$

  where $w(t)$ is a weighting factor (related to $w(\lambda_t)$ from Eq. 2 of the paper, often constant or related to SNR).

- **Applying to Eq. (13):** Each KL term in Eq. (13) is replaced by its corresponding weighted mean squared error term:

  - $\mathrm{D}_{\mathrm{KL}}(q(\cdot|x_t^w, x_0^w)||p_\theta(\cdot|x_t^w))$ becomes (up to a constant absorbed into $w(t)$) $w(t)||\epsilon - \epsilon_\theta(x_t^w, t)||^2$.
  - $\mathrm{D}_{\mathrm{KL}}(q(\cdot|x_t^w, x_0^w)||p_{\mathrm{ref}}(\cdot|x_t^w))$ becomes $w(t)||\epsilon - \epsilon_{\mathrm{ref}}(x_t^w, t)||^2$.

  Similar substitutions are made for x_t^l. The ϵ in $||\epsilon - \epsilon_\theta||^2$ is the ground truth noise sample used to generate x_t^w from x_0^w (and x_t^l from x_0^l).

- **Final Loss Form:** This equation is now a practical loss function. The training process involves:

  1. Sampling a preference pair (x_0^w, x_0^l) from the dataset $\mathcal{D}$.
  2. Sampling a timestep $t \sim U(0, T)$.
  3. Sampling a single noise vector $\epsilon \sim N(0, I)$.
  4. Constructing the noised samples: $x_t^w = \alpha_t x_0^w + \sigma_t \epsilon$ and $x_t^l = \alpha_t x_0^l + \sigma_t \epsilon$.
  5. Obtaining noise predictions from the current model ($\epsilon_\theta(x_t^w, t)$, $\epsilon_\theta(x_t^l, t)$) and the reference model ($\epsilon_{\mathrm{ref}}(x_t^w, t)$, $\epsilon_{\mathrm{ref}}(x_t^l, t)$).
  6. Plugging these values into the formula for Eq. (14).

  This loss encourages the model ϵ_θ to reduce its prediction error for preferred samples x_0^w (i.e., $||\epsilon - \epsilon_\theta(x_t^w, t)||^2$ should be smaller than $||\epsilon - \epsilon_{\mathrm{ref}}(x_t^w, t)||^2$) and/or increase its error for dispreferred samples x_0^l, relative to the reference model. The term $\beta T w(t)$ scales the impact of these differences.

This sequence of derivations transforms the abstract DPO principle into a concrete, trainable loss function for aligning diffusion models with human preferences.