Thanks all the reviewers for the detailed and thoughtful comments.

(R1, R2) Resemblance to HMM-based speech synthesizers and recognizers We found resemblance to the previous HMM-based works [1, 2, 3], all of which proposed methods to estimate alignments from unsegmented data. Thanks for commenting the missing references, we will mention about the previous works in the new version of the paper. However, we claim that the contribution of our proposed method is not only about the efficient alignment search, but also about enabling parallel sampling over the complex data distribution. We believe incorporating bijective flows for parallel sampling does not naturally come out from HMM-based works, as HMMs are inherently sequential models. Likewise, estimating alignments in the latent space through dynamic programming is distinctive from the previous parallel TTS works such as FastSpeech. Finally, compared to HMM-VAE [3], Glow-TTS differs in that 1) it focuses on generating samples in parallel not estimating sequences of latent discrete variables, 2) does not need an additional encoder to approximate the latent posterior nor 3) independence assumption of outputs across time.

(R1, R3) Deterministic duration prediction Our main goal was to build a parallel TTS model which jointly learns to model data distribution and align by itself in an efficient way, and the key components of Glow-TTS were the MAS and the flow-based decoder. We've not thoroughly explored to improve the duration predictor and simply follow the same architecture of FastSpeech. But as the feedback of Reviewer 1, exploring a joint model of durations would be promising in the future direction of our work and we expect it could be effective to model diverse speech.

(R1, R3) Grouped 1x1 Convolutions We design the grouped 1x1 convolutions to be able to mix channels. For each group, as Reviewer 3 commented, the same number of channels are extracted from one half of the feature map separated by coupling layers and the other half, respectively. For example, if a coupling layer divides a 8-channel feature map (a, b, c, d, e, f, g, h) into two halves (a, b, c, d) and (e, f, g, h), we implement to group them into (a, b, e, f) and (c, d, g, h) when the number of groups is 2. We will add the implementation details and update Figure 8c more clearer.

(R1) Importance of parallel TTS With parallel vocoders such as WaveGlow, the primary computational bottleneck in mel spectrogram-based end-to-end TTS systems became TTS models. For example, to generate a speech of 5.8 seconds on a GPU, WaveGlow spends 120 ms while Tacotron 2 and Glow-TTS require 640 ms and 40 ms, respectively. Therefore, adopting parallel TTS models significantly improves the sampling speed of end-to-end systems.

(R3, R4) Controllability and diversity of sampling We agree Glow-TTS shows restricted controllability and diversity of some properties of speech such as pitch or intonation as Reviewer 3 pointed out. But, as Glow-TTS estimates the distribution of speech given text tokens, it has the potential to generate diverse speech and control the characteristics of speech, which is not possible with the deterministic models like FastSpeech. The stochasticity comes from the prior distribution of the latent variable z, and we believe that synthesizing with careful sampling from the latent space enables us to control the characteristics of generated samples. In Section 5.3, we showed that varying temperature can change the pitch of generated samples and different zs correspond to speech samples with different intonations. To clarify our analysis, we will add more samples for a variety of text samples in our demo page. In this work, we provided the limited analysis on the controllability of Glow-TTS. Further analysis on how the latent space of z is related to the characteristics of generated samples would be helpful for emotional control, and is left as future work.

(R3) The iterative objective The reason why we demonstrate the iterative objective provides a good lower bound of the global solution is that the iterative procedure is actually one example of widely used Viterbi training [2], which maximizes log likelihood of the most likely hidden alignment. We also empirically showed that our system works well with the objective. We will add a reference about Viterbi training.

(R3) CMOS results We conducted CMOS evaluation between Tacotron 2 and Glow-TTS with the sampling temperature 0.333. Through 500 ratings on 50 items, Glow-TTS wins Tacotron 2 by a gap of 0.934, which shows preference towards our model over Tacotron 2. We will add this result in the new version of the paper.

(R2, R3) Multi-speaker MOS evaluations We will add comparison to Tacotron 2 with speaker embeddings as a baseline in the new version of the paper. As for methodology of MOS evaluations, we randomly sampled 50 sentence and speaker id pairs from the test set, which resulted in 40 unique speakers (one sentence for 31 speakers, two sentences for 8 speakers, and three sentences for one speaker). When compared to Tacotron 2 with speaker embeddings, we will sample one sentence for each speaker for the evaluation. We will also include the test phrases as well as speaker ids used for all MOS evaluation in the supplemental material.

[1] Tokuda, Keiichi, et al. "Speech synthesis based on hidden Markov models." Proceedings of the IEEE 101.5 (2013): 1234-1252.

[2] Rabiner, Lawrence R. "A tutorial on hidden Markov models and selected applications in speech recognition." Proceedings of the IEEE 77.2 (1989): 257-286.

[3] Ebbers, Janek, et al. "Hidden Markov Model Variational Autoencoder for Acoustic Unit Discovery." INTERSPEECH. 2017.