End-to-End 音声合成の Continual Learning における破滅的忘却の影響の調査*

○齋藤 佑樹，猿渡 洋（東大院・情報理工）

1 はじめに

Deep neural network (DNN) に基づく音声合成 [1, 2] の進展により，自然音声に肉薄する品質の合成音声を生成可能になりつつある．特に，入力テキストから音声を一気通貫方式で予測する End-to-End (E2E) 音声合成 [3] は，非専門家の音声合成分野への新規参入の障壁を取り除く技術として広く研究されている．このように，音声合成が機械学習タスクの一種としてみなされるようになり，高品質で多様なテキストと音声を含む大規模コーパスの構築・整備 [4, 5] や，少量データを用いた音響モデル・波形生成モデルの適応法に関する研究 [6, 7] が進められている．

現状の DNN 音声合成は，ドメインを限定して十分なデータを用意できれば高品質な音声を合成可能だが，人間のように複数ドメインの音声を継続的・逐次的に学習する能力を有さず，「一度の学習で用いられたドメインの音声データを高精度に再現する技術」に留まる．故に，実社会で容易に適応可能な人工知能エージェントの実現には，過去の学習で得られたドメイン固有の知識を忘却する現象である「DNN の破滅的忘却」（図 1(a)）[8] を回避しつつ，当該ドメイン固有の知識を獲得・蓄積可能な音声合成の学習法が要求される．

本稿では，DNN が継続的・階層的・追加的に知識を学習するための枠組みである continual learning [9] に基づく E2E 音声合成の学習法を検討する．特に，単一話者の多様なドメインのテキスト読み上げ音声が逐次的に与えられる想定のもとで，現状の E2E 音声合成の枠組みにおける破滅的忘却の影響を調査する．本稿では更に，破滅的忘却を回避するために，過去の学習で用いたデータの一部を再利用するリハーサル法（図 1(b)）の有効性も実験的に評価する．評価結果より，(1) 破滅的忘却の影響は，合成音声の韻律・スペクトル包絡特徴量の予測において特に顕著であること，(2) リハーサル法が破滅的忘却に起因する合成音声の品質劣化を緩和させることを示す．

2 関連研究

2.1 FastSpeech2

これまでに数多くの E2E 音声合成の手法が提案されているが，本稿では，学習速度や推論時の安定性の観点から，非自己回帰型 E2E 音声合成方式の一種である FastSpeech2 [10] を用いる．FastSpeech2 は，(1) duration predictor と length regulator による音素継続長予測と系列アラインメント，(2) 複数の variance adaptor による音声特徴量予測，(3) mel-spectrogram decoder によるメルスペクトログラム予測を行う．学習時に音素アラインメントの情報を用意しなければならないという制約があるが，Tacotron2 [3] などの自己

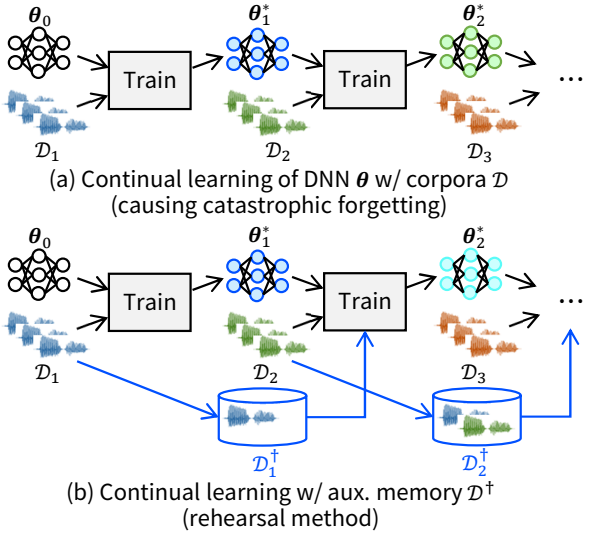

Fig. 1 本稿で検討する continual learning の概念図

回帰型 E2E 音声合成方式における推論時のエラー（合成音声のループや読み飛ばしなど）を発生させにくいという利点がある．学習時には，音素継続長予測誤差，音声特徴量予測誤差，メルスペクトログラム予測誤差の総和を最小化するように DNN のモデルパラメータを更新する．即ち，学習に用いるコーパス（N 個のテキスト $\boldsymbol{x}$ と音声 $\boldsymbol{y}$ のペア）を $\mathcal{D} = \{(\boldsymbol{x}_n, \boldsymbol{y}_n)\}_{n=1}^N$ とすると，学習時の損失関数は次式で与えられる．

$$L_{\mathrm{FS2}}(\mathcal{D}; \boldsymbol{\theta}) = \mathrm{MSE}(\boldsymbol{y}_{\mathrm{dur}}, \hat{\boldsymbol{y}}_{\mathrm{dur}}) + \mathrm{MSE}(\boldsymbol{y}_{\mathrm{feat}}, \hat{\boldsymbol{y}}_{\mathrm{feat}}) + \mathrm{MAE}(\boldsymbol{y}_{\mathrm{mel}}, \hat{\boldsymbol{y}}_{\mathrm{mel}}) \quad (1)$$

ここで，MAE(·) と MSE(·) はそれぞれ与えられた 2 変数間の mean absolute error と mean squared error を表す．変数の下付き添字 dur, feat, mel はそれぞれ学習データから得られる音素継続長，音声特徴量，メルスペクトログラムであり，$\boldsymbol{y}_*$ は自然音声を，$\hat{\boldsymbol{y}}_*$ は合成音声を表す．$\boldsymbol{\theta}$ は FastSpeech2 のモデルパラメータ（ネットワーク結合重みやバイアスなど）であり，勾配法ベースの学習によって $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} L_{\mathrm{FS2}}(\mathcal{D})$ として更新される．ただし，η は学習率である．

2.2 Continual Learning

Continual learning [9] は強化学習の分野で登場した研究領域であり，人工知能が継続的・階層的・追加的に学習可能な枠組みを提唱するものである．DNN の continual learning では，単一の DNN モデルを複数のデータセットで逐次的に学習し，学習された既知のドメインと未知のドメインの両方に対する性能改善を目的とする．通常の DNN 学習では，与えられたデータセットに対する適合のみを考慮するため，複数データセットでの逐次的学習においては，過去の学習で得られたドメイン固有の知識の忘却（DNN の破滅

*Investigation of effects caused by catastrophic forgetting in continual learning of end-to-end text-to-speech synthesis by SAITO, Yuki and SARUWATARI, Hiroshi (The University of Tokyo).

的忘却）が生じる．故に，continual learning の主な技術的課題として，有限の計算資源を用いた学習の過程での破滅的忘却を回避しつつ，未知のドメインに対する性能も改善可能な知識を学習することが挙げられる．

これまでの continual learning の研究は，認識タスクを対象としたものが多く [11]，生成タスクにおける研究は少数である [12]．音声分野においてもこの傾向は類似しており，continual learning を適用した音声認識 [13] や音響イベント分類 [14] の研究は存在するが，音声合成という時系列データを対象とした生成タスクにおける適用例は少ない [15]．

3 E2E 音声合成の Continual Learning

3.1 問題設定

単一の E2E 音声合成モデル（本稿では FastSpeech2）を，$\mathcal{T}$ 個の複数コーパス $\mathcal{D} = \{\mathcal{D}_\tau\}_{\tau=1}^{\mathcal{T}}$ で逐次的に学習することを考える．ここで，τ 番目のコーパス $\mathcal{D}_\tau$ を用いた学習を「タスク τ」と定義し，ディスク容量の制限やプライバシー保護などの観点から，タスク τ では以前の学習で用いたコーパス $\mathcal{D}_{<\tau} = \{\mathcal{D}_k\}_{k=1}^{\tau}$ には原則アクセス不可能とする．学習時の損失関数は通常と同じ（FastSpeech2 の場合は式（1））であり，タスク τ 終了後のモデルパラメータを $\boldsymbol{\theta}_\tau^*$ と表記する．$\boldsymbol{\theta}_\tau^*$ は $\nabla_{\boldsymbol{\theta}} L_{\mathrm{FS2}}(\mathcal{D}_\tau)$ のみを用いて推定されるため，タスク $\tau-1$ までに得られた知識を保持する保証はない．

3.2 リハーサル法を用いた Continual Learning

リハーサル法を用いた continual learning [16] では，前節での問題設定を緩和し，$\mathcal{D}_{<\tau}$ の一部を格納するための容量 M の補助メモリ $\mathcal{D}_{\tau-1}^{(M)}$（$\mathcal{D}_0^{(M)} = \varphi$ として初期化）を用意する．タスク τ では，$\mathcal{D}_\tau^\dagger = \mathcal{D}_\tau \cup \mathcal{D}_{\tau-1}^{(M)}$ を用いてモデルパラメータを推定し，タスク終了後，$\mathcal{D}_\tau^\dagger$ の中から容量 M に収まる範囲でデータを選択して $\mathcal{D}_\tau^{(M)}$ に格納する．なお，音声は可変長データであるため，メモリ容量はデータ数ではなく音声データのファイルサイズを基準として定める．

補助メモリに格納するデータの選択基準として，本稿では Chang らの先行研究 [13] にて提案されている median length を用いる．この基準では，まず，$\mathcal{D}_\tau^\dagger$ に含まれる各データの音素継続長 $\boldsymbol{y}_{\mathrm{dur},n} = [d_{1,n}, \ldots, d_{P_n,n}]^\top$ の総和（音声データ長の概算）$l_n = \sum_{p=2}^{P_n-1} d_{p,n}$ を計算する．ここで，先頭と末尾の音素は無音区間であることを想定し，継続長の総和計算から除外する．その後，l_n の中央値 $\tilde{l} = \mathrm{median}\{l_n\}_{n=1}^{|\mathcal{D}_\tau^\dagger|}$ を計算し，中央値との差の絶対値 $|\tilde{l} - l_n|$ が小さいデータから順に補助メモリに格納する．

3.3 考察

Continual learning は，これまでに音声合成分野で広く用いられてきたモデル適応 [6, 7] やマルチタスク学習 [17] と深く関連する研究領域である．前者は，大規模コーパスが利用可能なドメインで学習された DNN 音響モデルを目的ドメイン（比較的少量のデータしか用意できない話者の音声など）に微調整する手法であり，目的ドメインでの合成音声の品質改善を保証するが，適応前のドメインに関する知識の忘却が生じる．後者は，複数ドメインのコーパスを用いて DNN 音響モデルを一度に学習する手法であり，単一のモデルで複数ドメインの音声を合成可能だが，データ不均衡性の影響により，データ量の多い支配的なドメインへの過適合が生じる．本稿で検討する continual learning は，複数ドメインのデータを用いた逐次的な学習と補助メモリを用いた知識共有機構により，これらの技術における課題を解決しつつ，統一的な議論を可能とする．

Table 1　実験的評価に用いたデータと学習ステップ数（“Tr.” と “Val.” はそれぞれ学習と検証を意味する）

Task ID	Subset name	Data split Tr.	Val.	Test	# Tr. steps
1	basic5000	4,500	250	250	100k
2	travel1000	900	50	50	50k
3	utparaphrase512	484	16	14	15k
4	onomatopee300	270	15	15	15k
5	precedent130	110	10	10	10k
6	loanword128	99	14	14	10k
NA	voiceactress100	NA	NA	100	NA

4 実験的評価

4.1 音声コーパス

本稿では，単一話者の多数ドメインデータにおける continual learning を想定し，JSUT コーパス [5] を実験的評価に用いた．JSUT コーパスは女性話者 1 名による多様なドメインの日本語テキストの読み上げ音声を収録したものであり，本稿では，表 1 第 2 列に示す “basic5000” から “voiceactress100” までの 7 サブセットを用いた．各サブセットの学習・検証・評価データの分割は，表 1 の第 3–5 列に示すように行った．Continual learning では，データ数の多い順にタスク ID を割り当て，各タスクでは，表 1 の第 6 列に示すステップ数だけモデルパラメータの更新を行った．“voiceactress100” は学習データから除外し，open なサブセットに対する音声合成モデルの頑健性の評価に用いた．音声データは 48,000 Hz から 22,050 Hz にダウンサンプリングし，量子化ビット数は 16 とした．

4.2 E2E 音声合成の実験条件

本稿では，Wataru-Nakata により公開されている日本語音声合成向けの FastSpeech2 pytorch 実装[1]を用いて実験を行った．JSUT コーパスのテキストの読みと音素アラインメントの情報は，r9y9 により公開されている時間情報付きフルコンテキストラベル[2]から抽出した．DNN アーキテクチャの設定（隠れ層数や隠れユニット数など）は前述の実装から変更していないが，学習率のアニーリングは 50,000 ステップ毎に 0.3 倍するように変更した．メルスペクトログラムの次元は 80 とし，短時間フーリエ変換の分析パラメータは FFT 長 1,024，ホップサイズ 256，窓長を 1,024 サンプルとした．FastSpeech2 の variance adaptor で予測される音声特徴量は，音素継続長，基本周波数（F_0），energy とし，F_0 の推定には WORLD [18, 19] の DIO と StoneMask を用いた．オリジナルの Fast-

[1] https://github.com/Wataru-Nakata/FastSpeech2-JSUT
[2] https://github.com/r9y9/jsut-lab

Table 2　合成音声と自然音声の MCD [dB]

	Full		basic5000		Cont.		Cont. w/ rehearsal (M = 5MB)		(M = 40MB)		(M = 400MB)	
Test subset	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last
basic5000	3.95	**3.96**	4.00	4.02	3.96	4.29	3.96	4.19	3.97	4.03	3.99	3.99
travel1000	3.66	3.71	4.25	4.25	3.65	3.90	3.61	3.82	3.66	3.72	3.69	**3.70**
utparaphrase512	3.95	4.01	3.93	3.98	3.93	4.41	3.91	4.30	3.94	4.19	3.86	**3.92**
onomatopee300	3.90	**3.92**	3.92	**3.92**	3.82	4.24	3.79	4.13	3.78	3.99	3.89	3.96
precedent130	3.82	3.87	4.01	4.09	3.81	4.04	3.82	3.83	3.75	**3.82**	3.85	3.91
loanword128	4.10	4.22	4.29	4.33	4.06	4.06	4.06	4.06	3.99	**3.99**	4.11	4.11
voiceactress100	4.00	**4.01**	4.04	4.04	4.06	4.34	4.01	4.25	3.98	4.10	3.98	4.04

Table 3　合成音声と自然音声の F_0 RMSE [Hz]

	Full		basic5000		Cont.		Cont. w/ rehearsal (M = 5MB)		(M = 40MB)		(M = 400MB)	
Test subset	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last
basic5000	75.9	75.9	76.2	76.3	75.7	78.1	76.0	78.9	75.7	76.9	75.6	**75.6**
travel1000	73.1	75.3	74.8	77.6	75.1	77.5	73.5	77.5	75.2	80.3	73.8	**74.5**
utparaphrase512	59.6	**61.4**	64.6	66.5	61.7	62.2	60.1	67.1	60.2	65.3	62.3	62.3
onomatopee300	73.4	78.5	75.7	77.5	77.9	81.7	75.2	75.2	73.9	79.8	74.0	**74.0**
precedent130	65.7	68.6	72.3	72.3	64.8	69.1	69.3	71.0	62.9	**66.6**	71.4	71.4
loanword128	73.6	76.2	76.3	78.9	75.6	79.1	76.2	78.3	73.3	**73.3**	76.4	80.1
voiceactress100	73.8	74.4	73.5	**73.3**	75.5	76.2	74.0	76.8	74.0	75.0	74.1	74.2

Speech2 [10] では音声特徴量をフレーム毎を予測するが，本稿では FastPitch [20] を参考に音素単位で平均された音声特徴量を予測するように実装した．学習時の optimizer には Adam [21] を使用し，学習率 η の初期値は 0.0001，β_1 は 0.9，β_2 は 0.98 とした．音声波形生成には，jik876 により公開されている Hifi-GAN ボコーダ [22] の UNIVERSAL_V1 事前学習モデル[3] を用いた．

4.3　客観評価

本稿では，6 サブセット (“basic5000” から “loanword128”) を全て用いて 200,000 ステップの学習を行う “Full” と，全サブセットの中で最もデータ数が多い basic5000 のみを用いて 200,000 ステップの学習を行う “basic5000” と，表 1 に示す学習ステップ数で各サブセットを用いて continual learning を行う “Cont.” を比較した．“Cont.” に関しては，リハーサル法を用いる場合に “w/ reharsal” と追記し，補助メモリ容量は $M = \{5, 40, 400\}$ MB とした．客観評価指標としては，合成音声と自然音声の mel-cepstral distortion (MCD)，F_0 root mean squared error (RMSE)，energy RMSE，duration RMSE を計算した．これらの評価指標は，continual learning において各タスクが終了するステップ数毎に計算し，その中で最良の評価値を “Best” と，最後の評価値を “Last” とした．

表 2–5 にそれぞれ MCD, F_0 RMSE, energy RMSE, duration RMSE の評価結果を示す．ここで，表中の**太字**と下線はそれぞれ各行の “Last” での最良と最悪の評価値であることを意味する．

まず，“Full” の評価結果（表 2–5 の 2–3 列目）に着目すると，各評価指標について，多くの評価サブセットに対して “Last” の評価値が最良となっていることが確認できる．この結果は，学習に最も多くのデータを用いているという観点から妥当と言えるが，一方で，いくつかの評価指標・評価サブセット（例えば，表 4 の “utparaphrase512”）に対しては，他の比較手法の中で最悪の評価値となっていることも確認できる．この一因としては，複数サブセットを用いたマルチタスク学習における特定サブセットへの過適合が考えられる．別の観点から，MCD と F_0 RMSE の評価結果（表 2–3）には，“Best” と “Last” の評価値に差が生じているケースがあり，各サブセット内での学習データへの過適合も観測される．

次に，“basic5000” の評価結果（表 2–5 の 4–5 列目）に着目すると，各評価指標について，“basic5000” 以外のいくつかの評価サブセットに対して “Last” の評価値が最悪となっていることが確認できる．この結果は，学習に単一のサブセットのみを用いているという観点から妥当と言えるが，一方で，MCD と F_0 RMSE の評価結果（表 2–3）には，いくつかの評価サブセットに対する “Last” の評価値が他の比較手法の中で最良となっていることも確認できる．また，“Best” と “Last” の評価値の差に着目すると，前述した “Full” のケースほどの大きな差は観測されない．

最後に，“Cont.” 及び “Cont. w/ rehearsal” の評価結果（表 2–5 の 6–13 列目）に着目すると，リハーサル法を用いない “Cont.” の各評価指標について，多くの評価サブセットに対して “Last” の評価値が最悪となっていることが確認できる．特に，MCD と F_0 RMSE の評価結果（表 2–3）には，“Best” と “Last” の評価値に大きな差があるケースが多く観測され，continual learning による破滅的忘却が生じていることが示唆される．一方で，energy RMSE と duration RMSE の評価結果（表 4–5）ではその傾向は弱まり，FastSpeech2 型の E2E 音声合成における continual learning では，スペクトル包絡と F_0 の予測における破滅的忘却の影響が大きいということが示された．しかし，“Cont. w/rehearsal” の評価結果（表 2–5 の 8–13 列目）から，これらの破滅的忘却の影響はリハーサル法を用いることで緩和され，特に，F_0 RMSE の評価結果（表 4）では，“Last” の評価値が多くの評価サブセットに対

[3] https://drive.google.com/drive/folders/1YuOoV3lO2-Hhn1F2HJ2aQ4SOLC1JdKLd

Table 4　合成音声と自然音声の energy RMSE

							Cont. w/ rehearsal					
	Full		basic5000		Cont.		(M = 5MB)		(M = 40MB)		(M = 400MB)	
Test subset	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last
basic5000	25.1	**25.1**	25.0	25.2	24.9	25.5	25.2	25.3	24.9	25.6	25.1	25.3
travel1000	24.2	24.5	24.3	24.3	23.6	**23.8**	23.9	24.4	23.8	24.4	24.0	24.2
utparaphrase512	25.2	26.1	24.1	**24.9**	25.3	25.8	24.2	**24.9**	24.6	25.7	25.2	26.1
onomatopee300	27.5	27.7	28.3	28.6	27.4	28.0	27.6	**27.6**	27.3	28.2	27.7	28.5
precedent130	25.6	25.6	25.5	25.7	25.0	25.6	24.9	**25.2**	25.2	26.5	25.3	26.4
loanword128	26.9	27.0	26.9	26.9	25.0	26.8	26.9	27.2	25.9	**25.9**	26.8	26.8
voiceactress100	30.0	30.0	29.7	30.1	29.7	30.2	29.7	30.1	29.3	30.5	29.7	**29.7**

Table 5　合成音声と自然音声の duration RMSE [frame]

							Cont. w/ rehearsal					
	Full		basic5000		Cont.		(M = 5MB)		(M = 40MB)		(M = 400MB)	
Test subset	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last
basic5000	1.97	**1.98**	2.02	2.03	2.01	2.08	2.04	2.15	2.00	2.04	2.02	2.02
travel1000	1.90	1.90	2.06	2.08	1.78	**1.84**	1.81	1.87	1.87	1.89	1.84	1.87
utparaphrase512	2.13	2.18	2.37	2.37	2.13	2.35	2.07	**2.17**	2.12	2.20	2.20	2.23
onomatopee300	2.29	2.29	2.21	2.23	2.04	2.17	2.07	**2.10**	2.06	2.14	2.14	2.15
precedent130	2.68	2.72	2.52	2.52	2.47	2.78	2.58	2.58	2.57	2.61	2.45	**2.46**
loanword128	2.89	2.92	3.05	3.07	2.94	2.98	2.91	**2.91**	3.00	3.01	3.04	3.04
voiceactress100	3.40	3.41	3.51	3.51	3.45	3.55	3.33	**3.35**	3.38	3.47	3.51	3.51

して “Full” よりも小さくなっていることが確認できる．この結果は，複数のサブセットを逐次的に学習することにより，過去のタスクで得られた知識を保持しつつ，新たなタスクに対する性能も改善できることを示唆する．また，リハーサル法の補助メモリ容量 M に着目すると，破滅的忘却の影響が大きい MCD と F_0 RMSE の評価結果（表 2–3）では容量を大きくすることによる改善が顕著だが，energy RMSE と duration RMSE の評価結果（表 4–5）では小さい容量での改善が見られる．この結果は，E2E 音声合成の continual learning におけるリハーサル法では，最適な補助メモリ容量は予測対象となる音声特徴量によって異なることを示唆する．

5　結論

本稿では，End-to-End 音声合成を複数の音声コーパスで逐次的に学習するための continual learning の枠組みを検討し，実験的評価により，破滅的忘却の影響とリハーサル法の有効性を検証した．今後は，モデルパラメータに対する正則化に基づく continual learning [23] を導入した学習法を検討する．

謝辞: 本研究は，JST，ムーンショット型研究開発事業，JPMJMS2011 の支援を受けたものです．

参考文献

[1] H. Zen et al., *Proc. ICASSP,* pp. 3872–3876, Vancouver, Canada, May 2013.
[2] A. Oord et al., *Proc. SSW,* pp. 125–125, Sunnyvale, U.S.A., Sep. 2016.
[3] J. Shen et al., *Proc. ICASSP,* pp. 4779—4783, Calgary, Canada, Apr. 2018.
[4] H. Zen et al., *Proc. INTERSPEECH,* pp. 1526–1530, Graz, Austria, Sep. 2019.
[5] S. Takamichi et al., *Acoustical Science and Technology,* Vol. 41, No. 5, pp.761–768, Sep. 2020.
[6] M. Chen et al., *Proc. ICLR,* Virtual Conference, May 2021.
[7] Q. Huang et al., *Proc. APSIPA-ASC,* pp. 815–820, Auckland, New Zealand, Dec. 2020.
[8] R. M. French, *Trends in Cognitive Sciences,* Vol. 3, No. 4, pp. 128–135, June 1999.
[9] Z. Chen et al., *Synthesis Lectures on AI and ML,* Morgan & Claypool Publishers, Nov. 2016.
[10] Y. Ren et al., *Proc. ICLR,* Virtual Conference, May 2021.
[11] M. Delange et al., *IEEE Trans. on PAMI,* Feb. 2021. (Early Access)
[12] M. Zhan et al., *Proc. ICCV,* pp. 2759–2768, Seoul, Korea, Oct. 2019.
[13] H.-J. Chang et al., *Proc. INTERSPEECH,* Brno, Czech Republic, Sep. 2021. (accepted)
[14] Y. Wang et al., *Proc. ICASSP,* pp. 321–325, Toronto, Canada, June 2021.
[15] H. Hemati et al., *arXiv:2103.14512,* Mar 2021.
[16] A. Robins, *Connection Science,* Vol. 7, No. 2, pp. 123–146, July 1995.
[17] Z. Wu et al., *Proc. ICASSP,* pp. 4460–4464, South Brisbane, Australia, Apr. 2015.
[18] M. Morise *et al.*, *IEICE Trans. on Information and Systems,* vol. E99-D, no. 7, pp. 1877–1884, Jul. 2016.
[19] M. Morise, *Speech Communication,* vol. 84, pp. 57–65, Nov. 2016.
[20] A. Łańcucki, *Proc. ICASSP,* pp. 6588–6592, Toronto, Canada, June 2021.
[21] D. P. Kingma et al., *Proc. ICLR,* San Diego, U.S.A., May 2015.
[22] J. Kong et al., *Proc. NeurIPS,* Vancouver, Canada, Dec. 2020.
[23] J. Kirkpatrick et al., *Proc. of The National Academy of Sciences,* pp. 201611835, Mar. 2017.