The University Of Sheffield.

Exploring the Use of Group Delay for Generalised VTS-based Noise Compensation

Erfan Loweimi,
Jon Barker and Thomas Hain
{e.loweimi, j.p.barker, t.hain}@sheffield.ac.uk

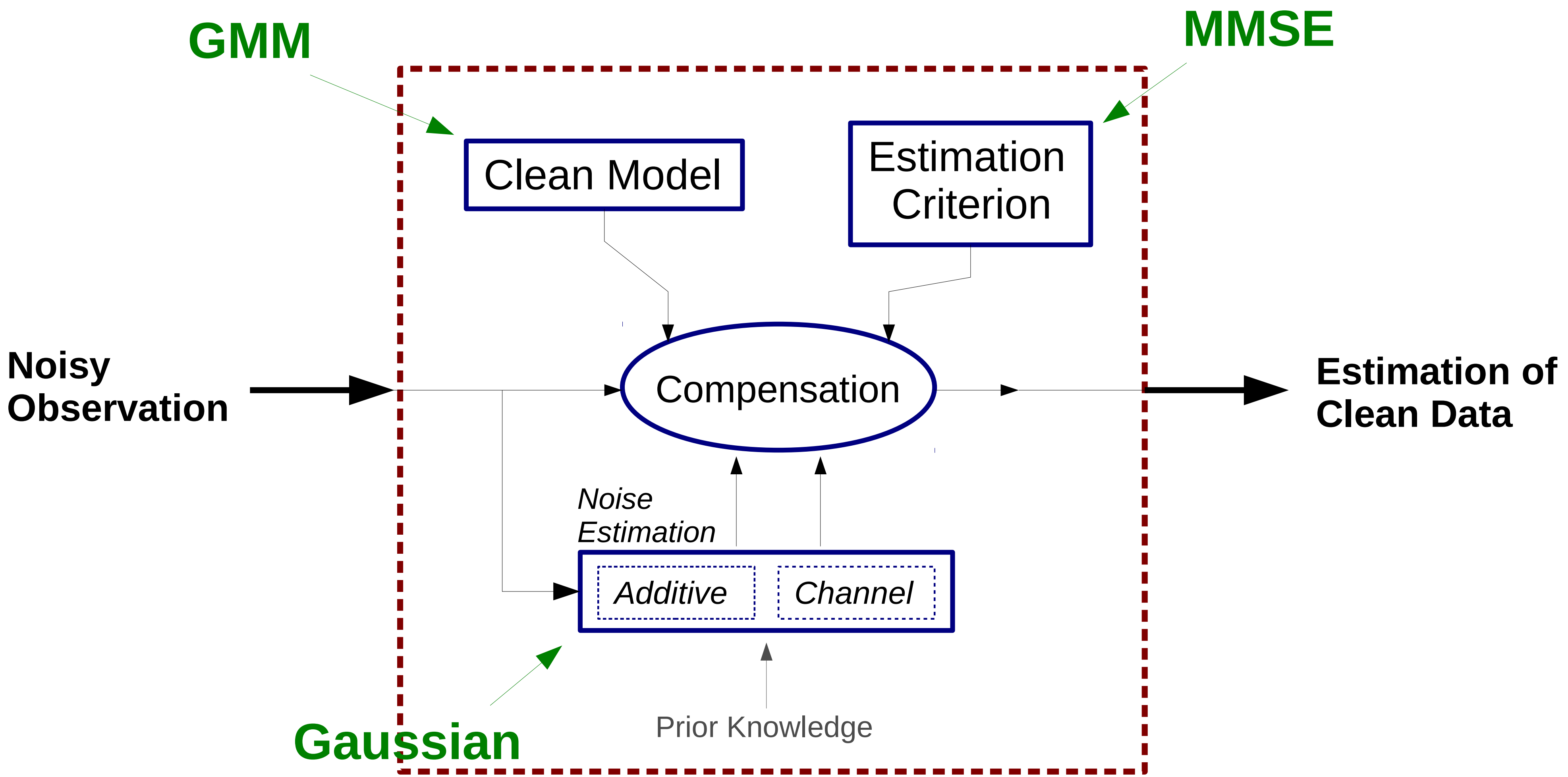
Presented by:
Yannis Stylianou

Speech and Hearing Research Group (**SPandH**)

Outline

- **G**eneralised **VTS** (gVTS) approach to Robust ASR
- Extension of the gVTS to group delay domain
  - Environment Model
  - Challenges
  - Proposed Solutions
  - Deriving Equations
- Experimental Results and Discussion

Model-based Noise Compensation



(g)VTS is a model-based technique for noise compensation.

(g)VTS Pseudocode

0. GMM of ***CLEAN***

– *For each utterance ...*

1. Compute the <u>environment model</u>
2. Apply the (Gen)Log
3. Factor out ***CLEAN*** part and compute the <u>distortion function</u>
4. Estimate ***Noise***
  4.1. Additive
  4.2. Channel
5. Linearise using **Taylor series**
  5.1. Points → means of Gaussians
  5.2. Jacobians → partial derivatives
6. Estimate ***CLEAN*** features using **MMSE**

Advantages of gVTS

- gVTS → replacing *Log* with *GenLog* in VTS
- One extra degree of freedom (*α*)
  - A non-linear transform with statistical effect [App. 1]
    - Can improve linearity, homoschodasticity and Gaussianty
  - Compensation is carried out in a space with a higher signal-to-noise ratio (SNR) [App. 2]
    - Further robustness
  - The optimal value for *α* → 0.05 – 0.1 [App. 3]

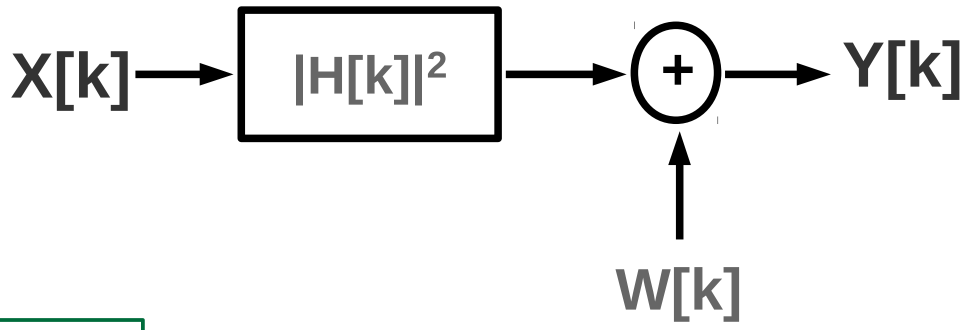
$$\begin{cases} GenLog(x;\alpha) = \frac{1}{\alpha}(x^{\alpha}-1), \quad x>0 \quad \alpha \neq 0 \\ \lim_{\alpha\to 0} GenLog(x;\alpha) = log(x) \end{cases}$$

Extension of the gVTS To Group Delay (GD) Domain

Environment Model

Periodogram domain

$$\boxed{|Y|^2 = |X|^2\,|H|^2 + |W|^2}$$



Group delay domain

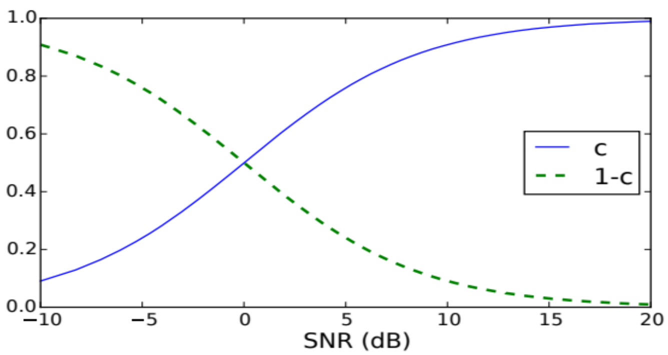
$$\boxed{\tau_Y = \frac{|X|^2|H|^2}{|Y|^2}\left(\tau_X + \tau_H\right) + \frac{|W|^2}{|Y|^2}\tau_W}$$

$$\tau_Y = \frac{\xi}{1+\xi}\,\tau_X + \frac{1}{1+\xi}\tau_W = \boxed{c\,\tau_X + (1-c)\,\tau_W}$$

convex combination

$$\xi = \frac{|X|^2}{|W|^2}:\ a\ priori\ \mathrm{SNR}$$

$$c = \frac{\xi}{1+\xi}$$



Challenges

1) Larger number of variables

- **4** in periodogram domain vs **8** in GD domain
  - For each variable a statistical model should be estimated
  - Noise compensation would be more complicated

2) Dynamic range compression using log and/or power transformation is problematic

- Group delay can be negative

4 variables →

$$|Y|^2 = |X|^2\,|H|^2 + |W|^2$$

8 variables →

$$\tau_Y = \frac{|X|^2|H|^2}{|Y|^2}\left(\tau_X + \tau_H\right) + \frac{|W|^2}{|Y|^2}\tau_W$$

Larger Number of Variables

- How to reduce number of variables?
  - Variables representing similar information and are added/multiplied may be encapsulated into one variable, e.g. group delay and power spectrum
  - Variables tends to zero in expected sense, may be removed, e.g. clean signal and noise cross-correlation

$$\boxed{|Y|^2\ \tau_Y} = |X|^2|H|^2\ (\tau_X + \tau_H)\ +\ \boxed{|W|^2\ \tau_W}$$

$$Q_Y = Q_X\ |H|^2 + Q_H\ |X|^2 + \ Q_W$$

Group delay-power
Product spectrum (PS)

#variables: 6
Still larger than periodogram domain which is 4!

Larger Number of Variables ...

$$|Y|^2 = |X|^2\,|H|^2 + |W|^2$$

← Periodogram domain

$$Q_Y = Q_X|H|^2 + \overbrace{\tau_H|H|^2}^{Q_H}|X|^2 + Q_W$$

← Product Spectrum domain

Extra *undesired* term!

– Without it the environment model resembles the periodogram domain

– If $\tau_H\,|H|^2\,|X|^2$ tends to zero, (g)VTS equations in periodogram domain can be used

– Obviously $|X|^2$ and $|H|^2$ are not zero!

– What about τ_H?

Phase and Group Delay of Channel

- Test sets A and C of Aurora-4 database may be used as *stereo* data to estimate <u>channel Fourier transform</u>

t: Frame index

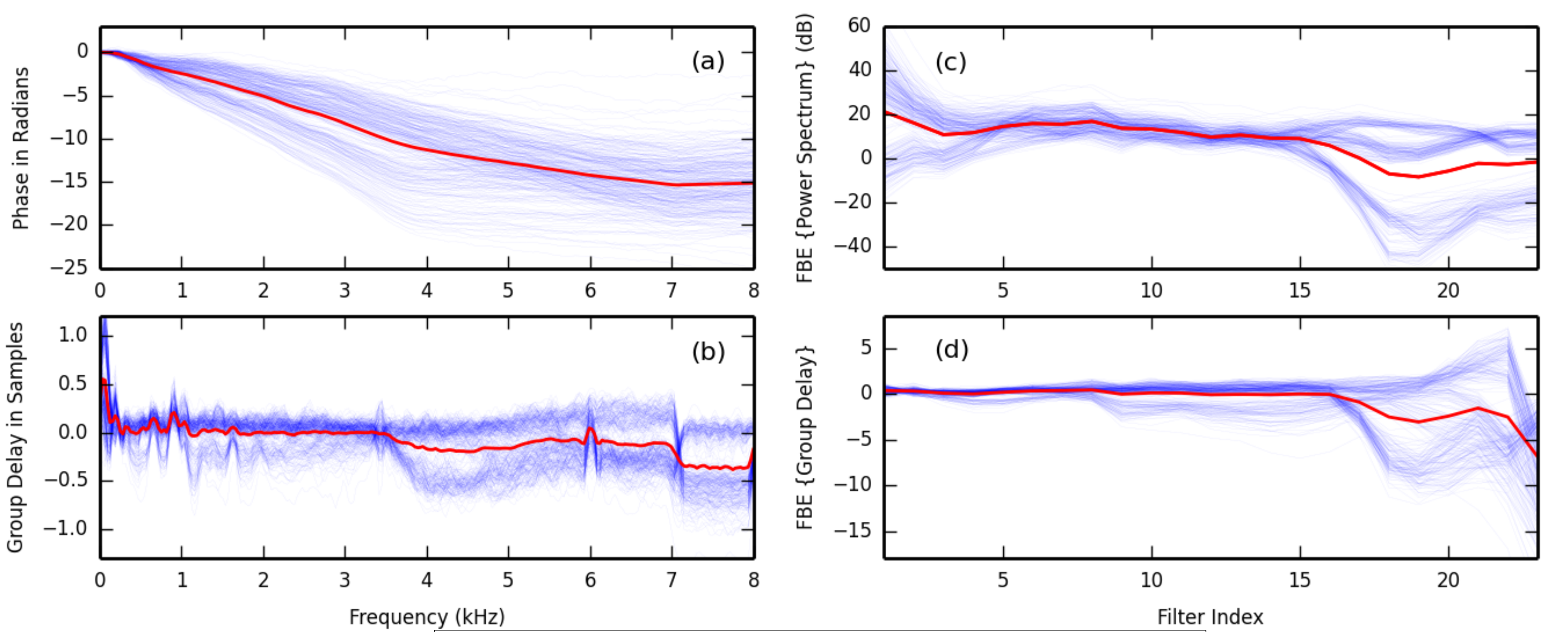
$$\begin{cases} \text{Test Set A} \Rightarrow Y^A = X \\ \text{Test Set C} \Rightarrow Y^C = X\ H \end{cases} \Rightarrow H_t = \frac{Y_t^C}{Y_t^A}$$

Channel phase spectrum

Channel group delay

Filterbank Energies (FBE) – Periodogram

(FBE) – group delay



Red curve: average over 330 utterances

Channel Phase Spectrum and Group Delay

- In the expected sense, the group delay of the channel tends to zero, as a result ...
  - *Undesired* term can be removed
  - Equation would be similar to periodogram domain

$$|Y|^2 = |X|^2\,|H|^2 + |W|^2$$

$$Q_Y = Q_X\,|H|^2 + \boxed{\cancelto{0}{\tau_H}|H|^2\,|X|^2} + Q_W$$

$$\Rightarrow \quad Q_Y \approx Q_X\,|H|^2 + Q_W$$

Dynamic Ranges Compression

- Dynamic range of product spectrum is comparable to power spectrum
  - Needs compression before statistical modelling
- (Gen)Log cannot be applied directly because group delay can be negative
- Possible solutions:

Absolute value $\rightarrow$ $|Q_Z|^\alpha$

$sign(Q_Z)|Q_Z|^\alpha \leftarrow$ Use sign function and abs

Add a constant $\rightarrow$ $(Q_Z + C)^\alpha$

Best solution

$\boxed{\breve{Q}_Z = max(Q_Z, 0)^\alpha \leftarrow \text{Flooring}}$

gVTS in the GD-power Product Spectrum Domain

1. Statistical models

$$\begin{cases} \breve{Q}_X \sim \sum_{m=1}^{M} p_m^{\breve{Q}_X} \mathcal{N}(\mu_m^{\breve{Q}_X}, \Sigma_m^{\breve{Q}_X}) \\ \breve{Q}_W \sim \mathcal{N}(\mu^{\breve{Q}_W}, \Sigma^{\breve{Q}_W}) \\ \breve{H} \sim \mathcal{N}(\mu^{\breve{H}}, \Sigma^{\breve{H}}) \end{cases}$$

2. Environment model
(after applying GenLog)

$$\breve{Q}_Y \approx \breve{Q}_X \underbrace{\breve{H} \left(1 + (\frac{\breve{Q}_W}{\breve{Q}_X \breve{H}})^{\frac{1}{\alpha}}\right)^{\alpha}}$$

Distortion function → $\breve{G}(\breve{Q}_X, \breve{Q}_W, \breve{H})$

3. Taylor series
(linearisation)

$$\breve{Q}_Y \approx \breve{Q}_{Y0} + J^{\breve{Q}_X}(\breve{Q}_X - \breve{Q}_{X0}) + \\ + J^{\breve{Q}_W}(\breve{Q}_W - \breve{Q}_{W0}) + J^{\breve{H}}(\breve{H} - \breve{H}_0)$$

gVTS in the GD-power Product Spectrum Domain

4. Compute the Jacobians (partial derivatives)

$$\breve{V}_m = \left(\frac{\mu^{\breve{Q}_W}}{\mu_m^{\breve{Q}_X}\ \mu^{\breve{H}}}\right)^{\frac{1}{\alpha}}$$

$$J_m^{\breve{Q}_X} = \frac{\partial \breve{Q}_Y}{\partial \breve{Q}_X} = diag\{\frac{\mu^{\breve{H}}}{(1+\breve{V}_m)^{1-\alpha}}\}$$

$$J_m^{\breve{Q}_W} = \frac{\partial \breve{Q}_Y}{\partial \breve{Q}_W} = diag\{(\frac{\breve{V}_m}{1+\breve{V}_m})^{1-\alpha}\}$$

$$J_m^{\breve{H}} = \frac{\partial \breve{Q}_Y}{\partial \breve{H}} = diag\{\frac{\mu_m^{\breve{Q}_X}}{(1+\breve{V}_m)^{1-\alpha}}\}$$

5. Compute noisy observations (Q_Y) statistics

$$\breve{Q}_Y \sim \sum_{m=1}^{M} p_m^{\breve{Q}_Y}\ \mathcal{N}(\mu_m^{\breve{Q}_Y}, \Sigma_m^{\breve{Q}_Y})$$

$$p_m^{\breve{Q}_Y} \approx p_m^{\breve{Q}_X}$$

$$\mu_m^{\breve{Q}_Y} \approx \mu_m^{\breve{Q}_X}\,\mu^{\breve{H}}(1+(\frac{\mu^{\breve{Q}_W}}{\mu_m^{\breve{Q}_X}\,\mu^{\breve{H}}})^{\frac{1}{\alpha}})^{\alpha}$$

$$\Sigma_m^{\breve{Q}_Y} \approx J_m^{\breve{Q}_X}\,\Sigma_m^{\breve{Q}_X}\,{J_m^{\breve{Q}_X}}^T + J_m^{\breve{Q}_W}\,\Sigma^{\breve{Q}_W}\,{J_m^{\breve{Q}_W}}^T + J_m^{\breve{H}}\,\Sigma^{\breve{H}}\,{J_m^{\breve{H}}}^T$$

gVTS in the GD-power Product Spectrum Domain

6. MMSE estimate

$$
\begin{aligned}
\hat{Q}_X^{MMSE} &= \mathbb{E}[\breve{Q}_X|\breve{Q}_Y] = \int \breve{Q}_X\, p(\breve{Q}_X|\breve{Q}_Y)d\breve{Q}_X \\
&\approx \breve{Q}_Y \sum_{m=1}^{M} p(m|\breve{Q}_Y)\,\frac{1}{\breve{G}(\mu_m^{\breve{Q}_X}, \mu^{\breve{Q}_W}, \mu^{\breve{H}})}
\end{aligned}
$$

Experimental Setup

- Database: Aurora-4
- Training sets: (each set: 7138 utterances, ~ 14 hours)
  - Clean-SI-84 → only clean data → **<u>CL</u>**
  - Noisy-SI-84 → clean and additive noise, SNR: 15 dB → **<u>M1</u>**
  - Multi-SI-84 → clean+additive+channel, SNR: 15 dB → **<u>M2</u>**
- Test set: Eval-92 → 330 utterances, ~ 40 minutes
  - 14 noise types artificially added using FaNT tool → 4620 utterances, grouped into
    - Test set A: Clean
    - Test set B: Additive noise, SNR: 10 dB
      - 6 noise types: Airport, Babble, Car, Restaurant, Street, Train Station
    - Test set C: Channel distortion
    - Test set D: Additive and Channel noise, SNR: 10 dB (6 additive noise types+channel distortion)
- Channel estimation → Method proposed in our earlier publication [App. 4]
- GMM/HMM → HTK → state-clustered triphones → 16 Gaussians, 4 iterations
- DNN (TNET) → 4 hidden layers (1300 nodes each)→bottleneck (26 nodes)→ output-layer [App. 5]
- Language model → bigram (perplexity: 147)

Experimental Results (WER)
Aurora4 – GMM/HMM

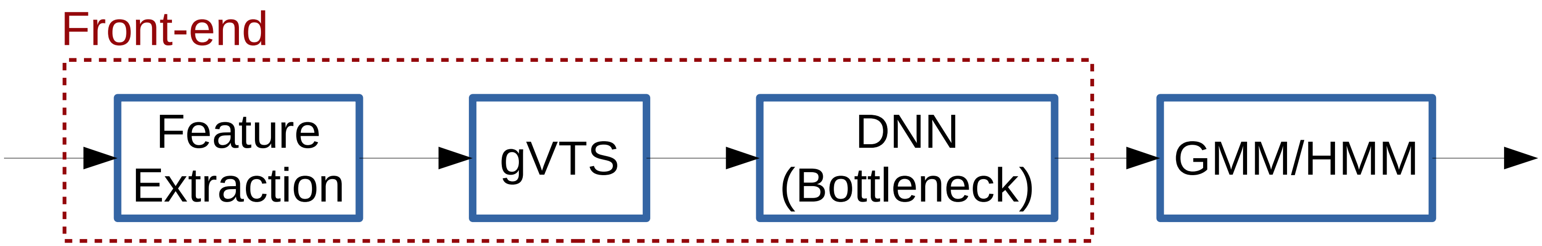
Feature	α	A	B	C	D	*Ave*
MFCC-CL	log	7.0	33.7	23.6	49.9	28.6
MFCC-M1	log	9.1	18.4	23.4	35.9	21.7
MFCC-M2	log	10.7	17.0	19.1	31.3	19.5
PS-CL	log	7.1	33.7	23.7	49.9	28.6
gPS-CL	0.05	7.0	25.3	23.2	42.9	24.6
gPS-CL	0.1	8.1	22.1	25.6	40.8	24.1
gVTS-CL	0.05	**6.5**	20.2	**13.9**	34.3	**18.7**
gVTS-CL	0.075	7.1	19.8	15.0	34.0	19.0
gVTS-CL	0.1	7.4	**19.6**	15.4	**33.9**	19.1

Relative WER reduction (relative to CL):
A → 7.7%
C → 41.1%
Ave → 34.8%
B → 41.8%
D → 32.1%

Relative to M1:
Ave → +13.8%

Relative to M2:
Ave → +4.1%

Experimental Results (WER)
Aurora4 – Bottleneck



Feature	α	A	B	C	D	*Ave*
BN{gPS}-CL	0.1	5.5	24.2	26.8	45.4	25.5
BN{gVTS}-CL	0.1	**4.6**	**20.6**	**16.0**	**36.7**	**19.5**
BN{gPS}-M1	0.1	5.5	11.1	23.5	32.3	18.1
BN{gVTS}-M1	0.1	**5.3**	12.4	**14.3**	**30.6**	**15.6**
BN{gPS}-M2	0.1	5.7	10.8	13.0	24.7	13.6
BN{gVTS}-M2	0.1	**5.6**	11.9	**12.3**	26.5	14.1

Relative WER reduction (relative to CL):
A → 16.4% C → 40.3% Ave → 23.5%
B → 18.2% D → 24.9%

Relative to M1:
Ave → +13.8%

Relative to M2:
Ave → -3.7%

Discussion and Conclusion

- gVTS and channel estimation techniques, proposed in earlier publications, successfully extended to product spectrum domain
- On average, the propose system trained by only clean data outperforms the MFCC-based system trained by Multi-style data (**M2**) in conventional GMM/HMM
- Combination of the gVTS and DNN is
  - Super-additive [App. 6] when there is a structural mismatch between the test and train conditions, e.g. **CL** or **M1** training conditions
    - Allows for building a robust system using DNN even when only clean data is available
  - sub-additive [App. 6] when all noise types with comparable SNR are available during training (**M2**)
    - In this case, discrimination is the main issue, not robustness

That's it!

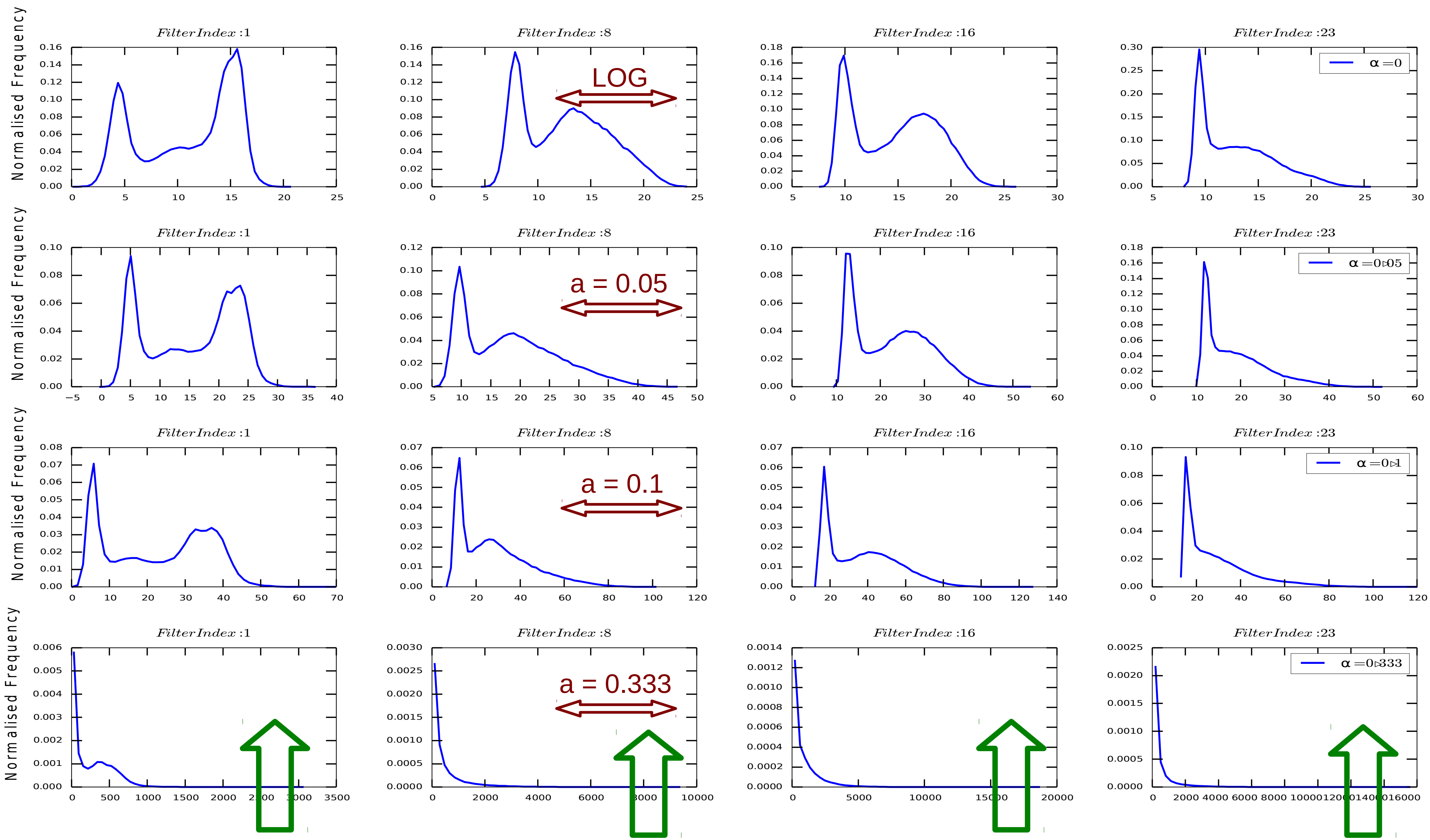
- Thanks for your attention
- Q&A

Appendices

1) Statistical effect of GenLog
2) GenLog can improve the SNR
3) Parameter adjustment in gVTS framework
4) Channel noise estimation
    1) Pseudocode
    2) Initialisation and iteration effects
5) DNN Architecture → Bottleneck
6) Super-additivity vs Sub-additivity

- NBins: 50
- 330 Utterances, WSJ
- #frames > 241 k

Statistical Effect of GenLog

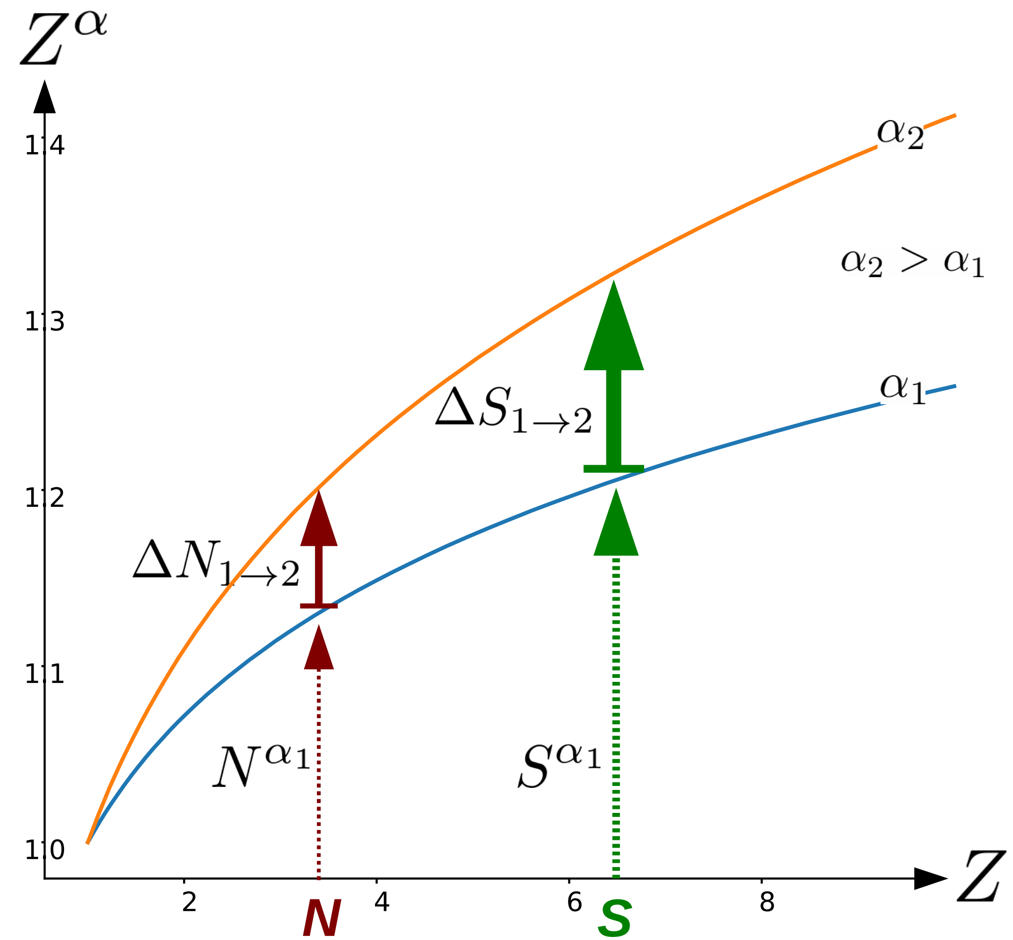


Filter Index: 1 **Filter Index: 8** **Filter Index: 16** **Filter Index: 23**

GenLog can improve the SNR ...

$$SNR_1 = \frac{S^{\alpha_1}}{N^{\alpha_1}}$$

$$SNR_2 = \frac{S^{\alpha_2}}{N^{\alpha_2}} = \frac{S^{\alpha_1} + \Delta S_{1\to 2}}{N^{\alpha_1} + \Delta N_{1\to 2}}$$



$$\alpha_1 < \alpha_2 \Rightarrow SNR_1 < SNR_2$$

$$\begin{cases} GenLog(x;\alpha) = \frac{1}{\alpha}(x^{\alpha} - 1), \quad x > 0 \quad \alpha \neq 0 \\ \lim_{\alpha \to 0} GenLog(x;\alpha) = log(x) \end{cases}$$

$\downarrow \alpha$

$\uparrow \alpha$

Statistical Distribution

Easier to be fitted by a GMM!

SNR Boost

$$0.05 \leq \alpha < 0.1$$

Channel Estimation Pseudocode

0. Initialise H

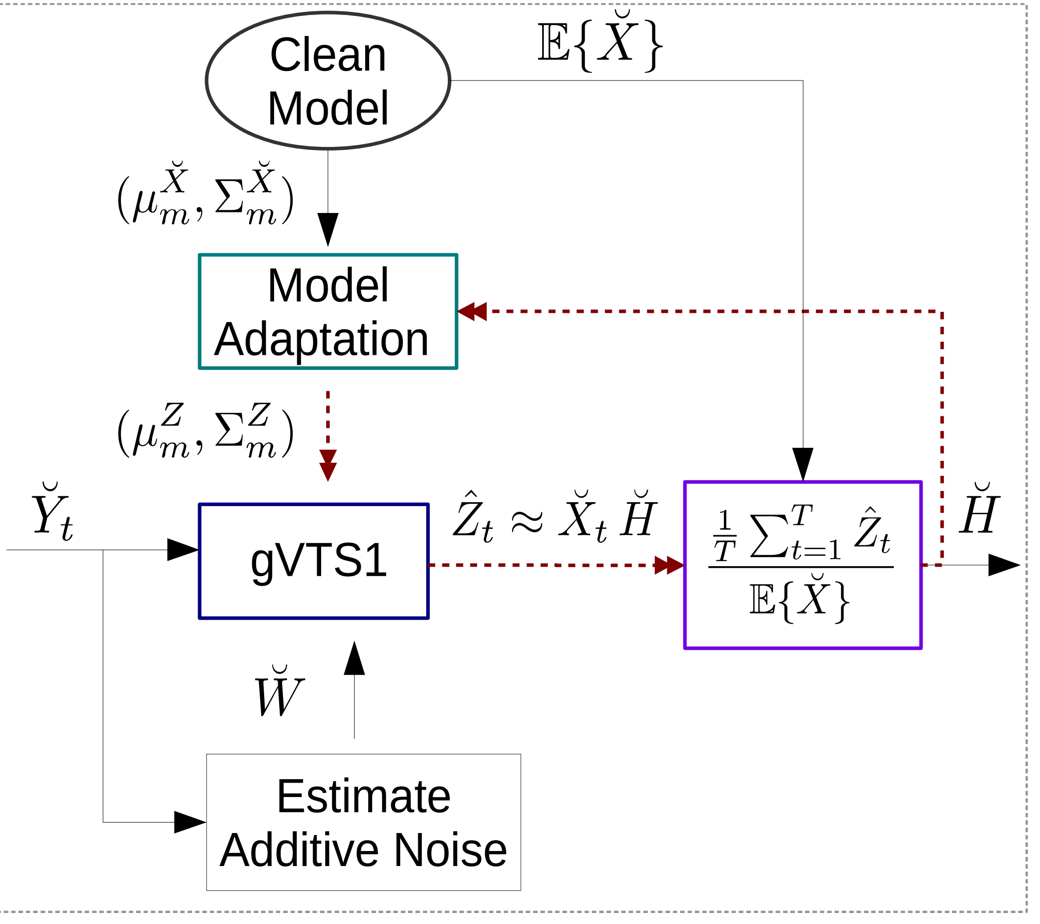
1. Adapt Clean Model with **H**

2. gVTS for Additive Noise

3. Update **H**

4. If not converged **GO TO 1**

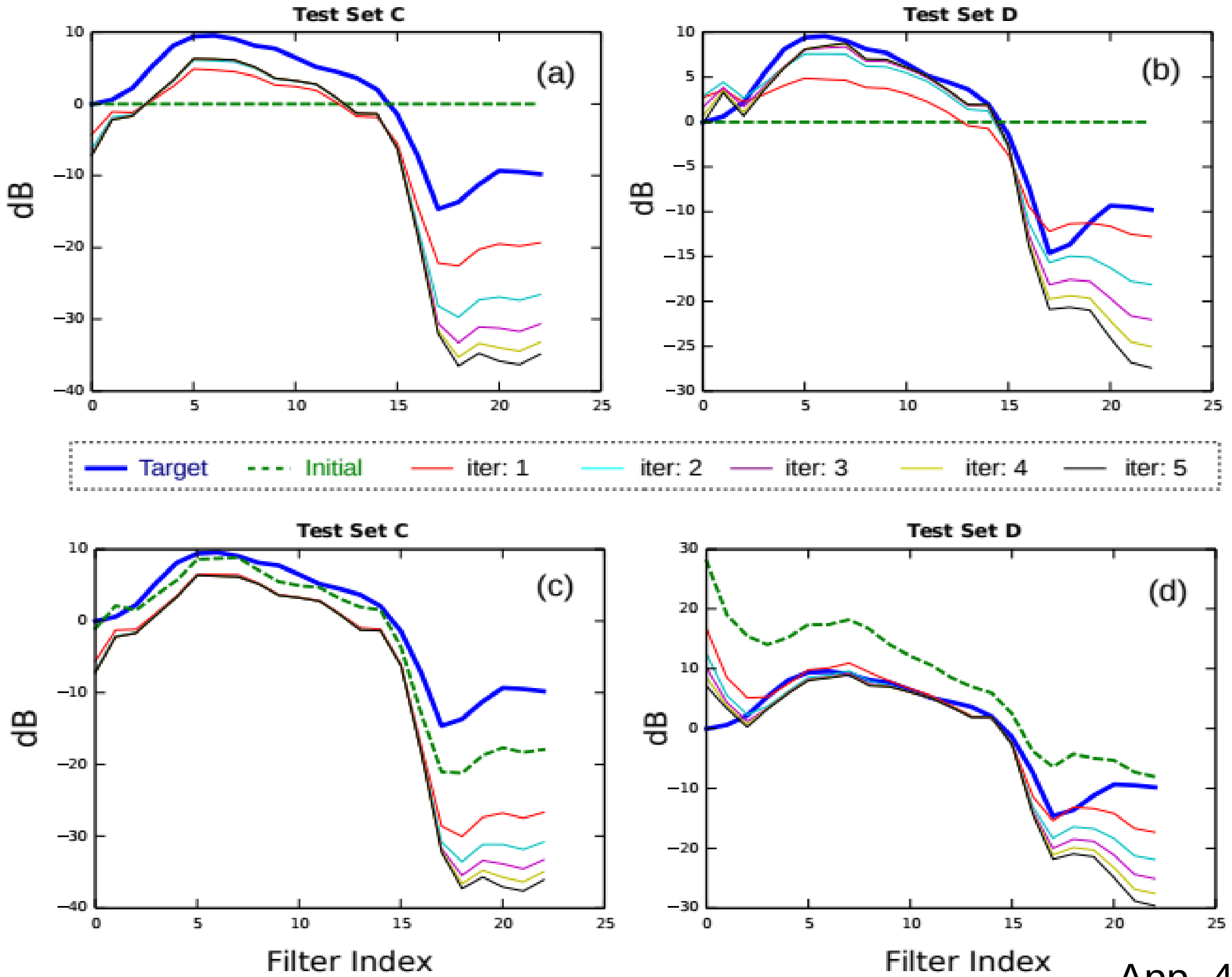
Return H



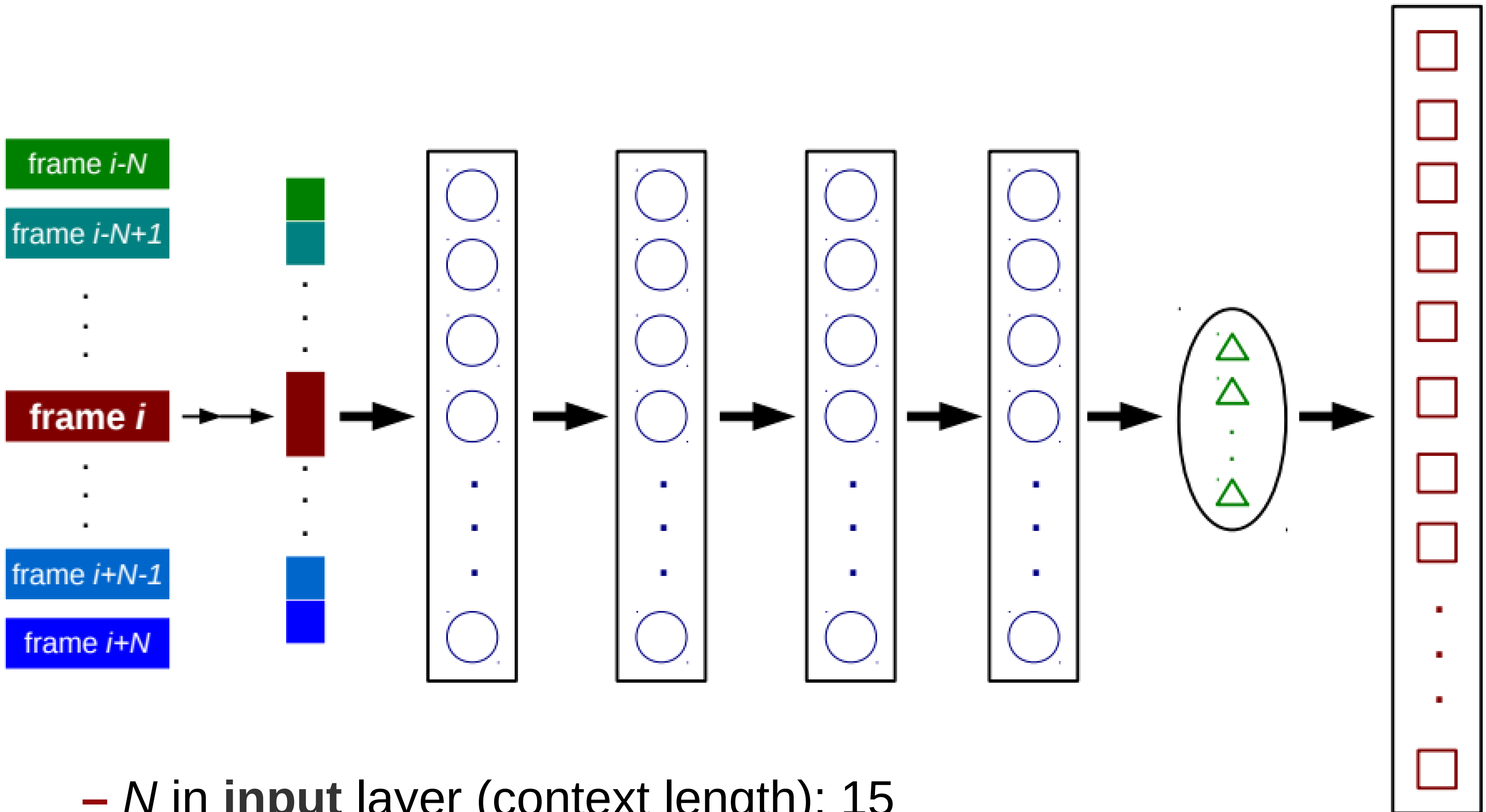
Channel Estimation -- Initialisation and Iteration effects

$$H_0 = 1$$

$$H_0 = \frac{Y}{\mathbb{E}\{X\}}$$



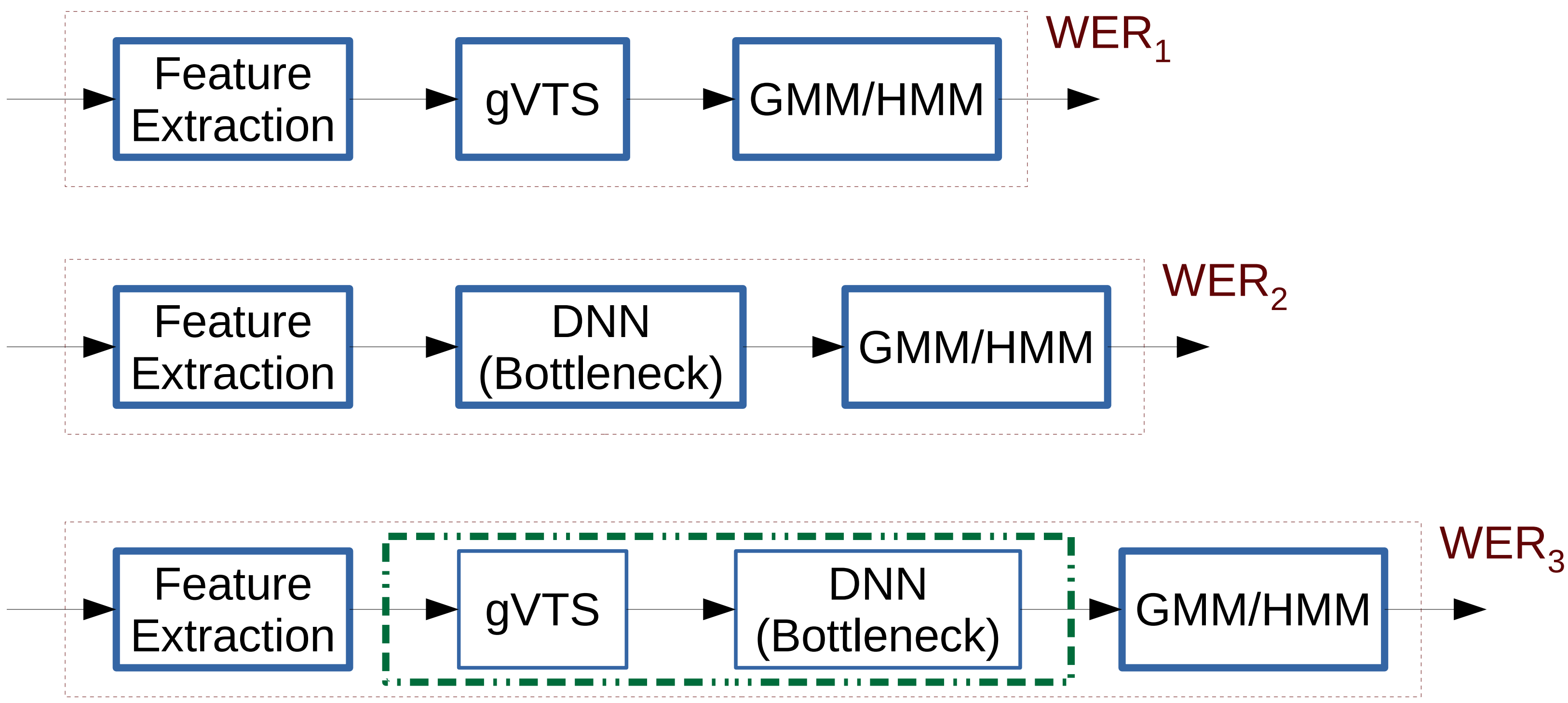
DNN Architecture → Bottleneck



- *N* in **input** layer (context length): 15
- #nodes in the **hidden** layers: 1300
- #nodes in the **bottleneck** layer: 26
- #nodes in the **output** layer: state-clustered triphones (~2000)

SPANDH

Super-additivity vs Sub-additivity of a Tandem gVTS-DNN System



– Super-additive: WER_3 is <u>better</u> than $\min(WER_1, WER_2)$

– Sub-additive: WER_3 is <u>worse</u> than $\min(WER_1, WER_2)$