AALBORG UNIVERSITY

icassp 2022

Sparse Modeling of The Early Part of Noisy Room Impulse Responses with Sparse Bayesian Learning

Maozhong Fu[1,2], Jesper Rindom Jensen[2], Yuhan Li[1,2], Mads Græsbøll Christensen[2]

[1]Xiamen University

[2]Aalborg University

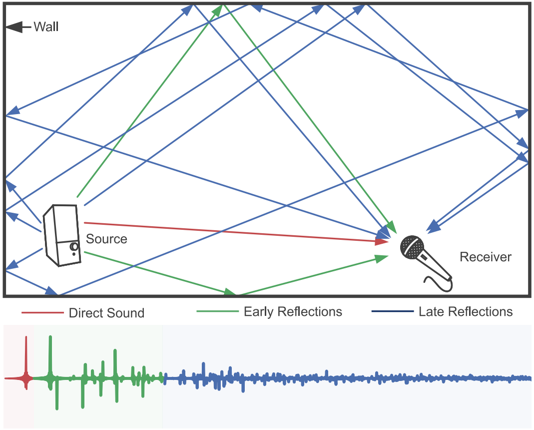
May 2022

Contents

1. Background

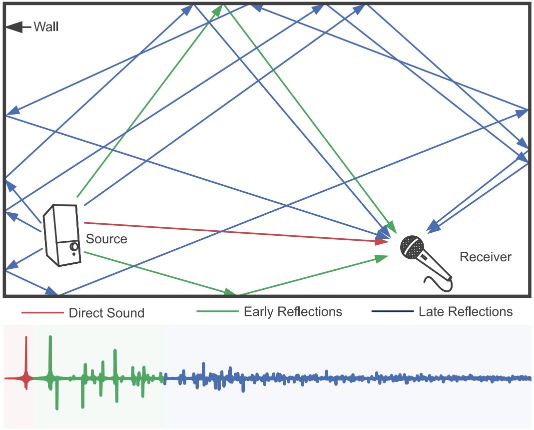
Room Impulse Response (RIR)

- A RIR in reverberation environment consists of
  - **Direct Sound**
  - **Early Reflections**
  - **Late Reflections**
- The early part (Direct Sound + Early Reflections) of a RIR is relatively sparse.
- Modeling of the early part of RIRs have many applications, such as
  - Room Geometry Reconstruction
  - Augmented and Virtual Reality
  - Dereverberation



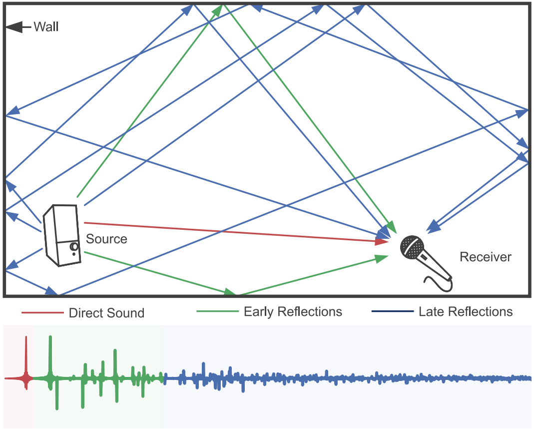
Room Impulse Response (RIR)

- A RIR in reverberation environment consists of
  - **Direct Sound**
  - **Early Reflections**
  - **Late Reflections**
- The early part (Direct Sound + Early Reflections) of a RIR is relatively sparse.
- Modeling of the early part of RIRs have many applications, such as
  - Room Geometry Reconstruction
  - Augmented and Virtual Reality
  - Dereverberation



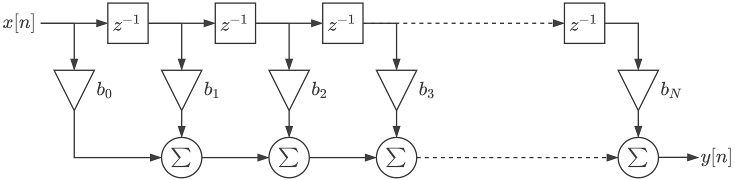
Room Impulse Response (RIR)

- A RIR in reverberation environment consists of
  - **Direct Sound**
  - **Early Reflections**
  - **Late Reflections**
- The early part (Direct Sound + Early Reflections) of a RIR is relatively sparse.
- Modeling of the early part of RIRs have many applications, such as
  - Room Geometry Reconstruction
  - Augmented and Virtual Reality
  - Dereverberation



Modeling of RIR

- RIRs typically are modeled as **long** FIR filters which may have thousands of coefficients.



- ℓ_1-regularization technology (LASSO) can be used to reduce the coefficient number of the early part of RIRs.

LASSO

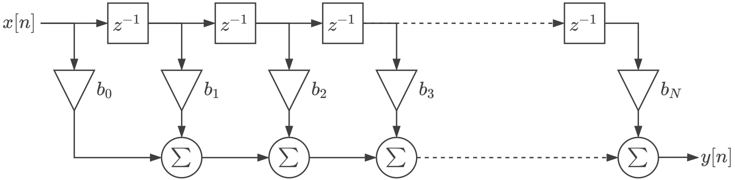
- Being **sensitive** to regularization parameters and requiring finding optimal regularization parameters.
  - Grid-search is **computational heavy**.
  - Cross-validation requires **training data**.

Sparse Bayesian Learning (SBL)

- Being **insensitive** to user parameters.
- Hyper-parameters are **learned** adaptively from data.
- Being computationally efficient and sparse-promoting.

Modeling of RIR

- RIRs typically are modeled as **long** FIR filters which may have thousands of coefficients.



- ℓ_1-regularization technology (LASSO) can be used to reduce the coefficient number of the early part of RIRs.

LASSO

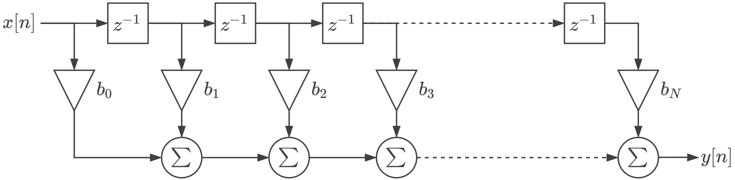
- Being **sensitive** to regularization parameters and requiring finding optimal regularization parameters.
  - Grid-search is **computational heavy**.
  - Cross-validation requires **training data**.

Sparse Bayesian Learning (SBL)

- Being **insensitive** to user parameters.
- Hyper-parameters are **learned** adaptively from data.
- Being computationally efficient and sparse-promoting.

Modeling of RIR

- RIRs typically are modeled as **long** FIR filters which may have thousands of coefficients.



- ℓ_1-regularization technology (LASSO) can be used to reduce the coefficient number of the early part of RIRs.

LASSO

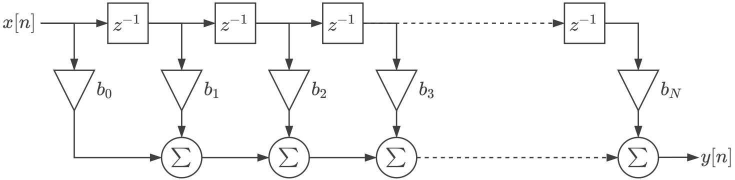
- Being **sensitive** to regularization parameters and requiring finding optimal regularization parameters.
  - Grid-search is **computational heavy**.
  - Cross-validation requires **training data**.

Sparse Bayesian Learning (SBL)

- Being **insensitive** to user parameters.
- Hyper-parameters are **learned** adaptively from data.
- Being computationally efficient and sparse-promoting.

Modeling of RIR

- RIRs typically are modeled as **long** FIR filters which may have thousands of coefficients.



- ℓ_1-regularization technology (LASSO) can be used to reduce the coefficient number of the early part of RIRs.

LASSO

- Being **sensitive** to regularization parameters and requiring finding optimal regularization parameters.
  - Grid-search is **computational heavy**.
  - Cross-validation requires **training data**.

Sparse Bayesian Learning (SBL)

- Being **insensitive** to user parameters.
- Hyper-parameters are **learned** adaptively from data.
- Being computationally efficient and sparse-promoting.

2. Proposed Method

Signal Model

- Discretizing the RIR measurements in the time-delay domain:

$$\overbrace{r((n-1)T_s)}^{\text{Measurements}} = \sum_{m=1}^{M} \underbrace{\beta_m}_{\text{Model Coefficients}} \overbrace{s((n-1)T_s-(m-1)T_\Delta)}^{\text{Source Signal}} + \underbrace{w((n-1)T_s)}_{\text{Noise}}$$

- Converting to matrix form:

Signal Model

- Discretizing the RIR measurements in the time-delay domain:

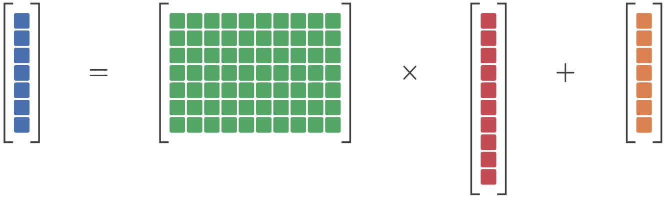
Measurements — Source Signal

$$r((n-1)T_s) = \sum_{m=1}^{M} \beta_m \, s((n-1)T_s - (m-1)T_\Delta) + w((n-1)T_s)$$

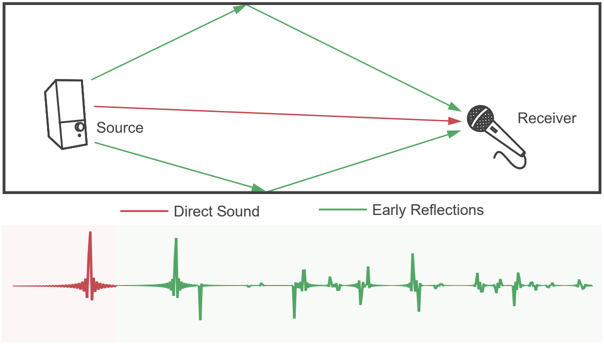
Model Coefficients — Noise

- Converting to matrix form:

Measurement $\mathbf{y}$ — Redundant Dictionary $\mathbf{\Phi}$ — Model Coefficients $\mathbf{x}$ — Noise $\mathbf{w}$

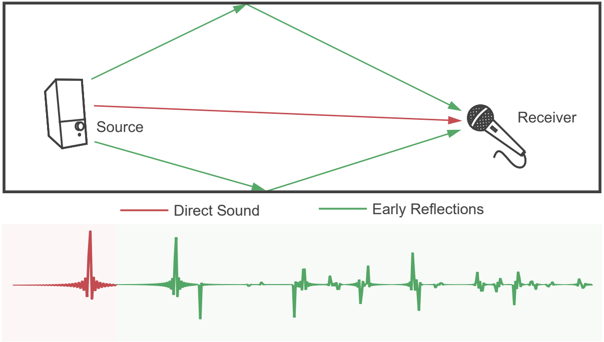


Scene Model



- Typical Scene: the early part of a RIR contains **very few** impulse responses.
- Scene is **sparse** i.e., $\mathbf{x}$ is mostly zeros.

Scene Model



- Typical Scene: the early part of a RIR contains **very few** impulse responses.
- Scene is **sparse** i.e., $\mathbf{x}$ is mostly zeros.

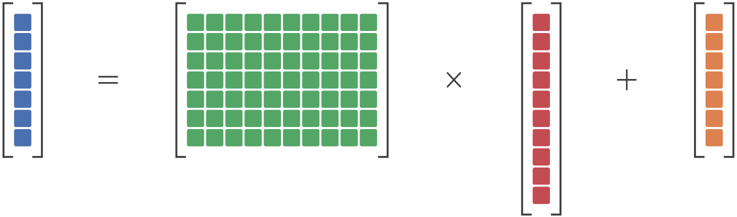
Problem Formulation

Measurement $\mathbf{y}$ **(Known)**

Redundant Dictionary $\mathbf{\Phi}$ **(Determined by the Source Signal)**

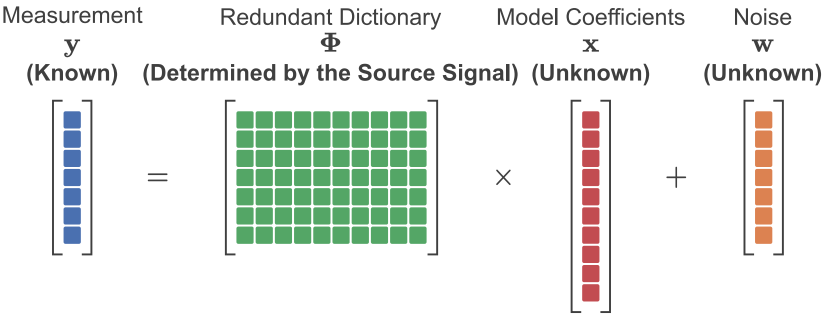
Model Coefficients $\mathbf{x}$ **(Unknown)**

Noise $\mathbf{w}$ **(Unknown)**



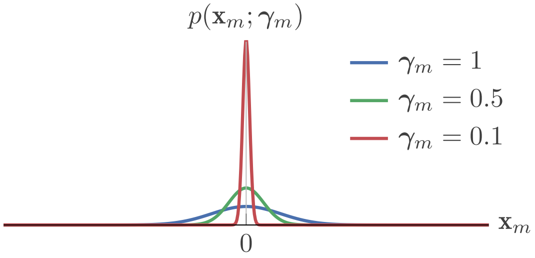
- **Problem 1:** Given $\mathbf{y}$, how to recover $\mathbf{x}$ with the a priori that $\mathbf{x}$ is sparse?
- **Problem 2:** If the source signal is unknown, how to determine $\mathbf{\Phi}$?

Problem Formulation



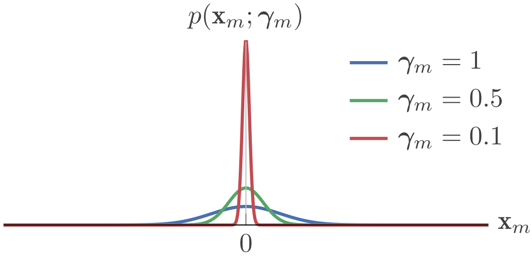
- **Problem 1:** Given $\mathbf{y}$, how to recover $\mathbf{x}$ with the a priori that $\mathbf{x}$ is sparse?
- **Problem 2:** If the source signal is unknown, how to determine $\mathbf{\Phi}$?

- **Prior:** SBL assumes the m-th component of $\mathbf{x}$ follows a zero-mean Gaussian distribution with variance γ_m.



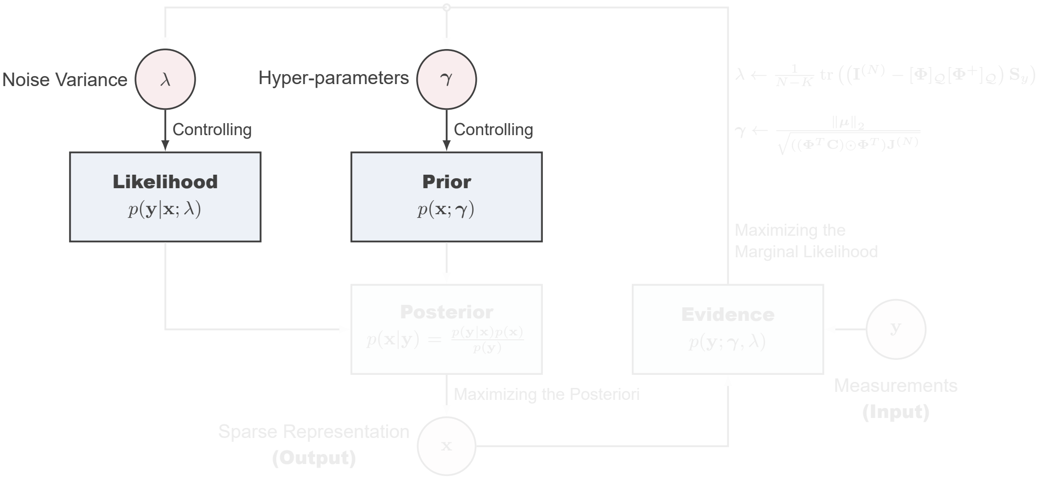
- **Likelihood:** SBL assumes the noise follows a zero-mean Gaussian distribution with variance λ.

- **Prior:** SBL assumes the m-th component of $\mathbf{x}$ follows a zero-mean Gaussian distribution with variance γ_m.

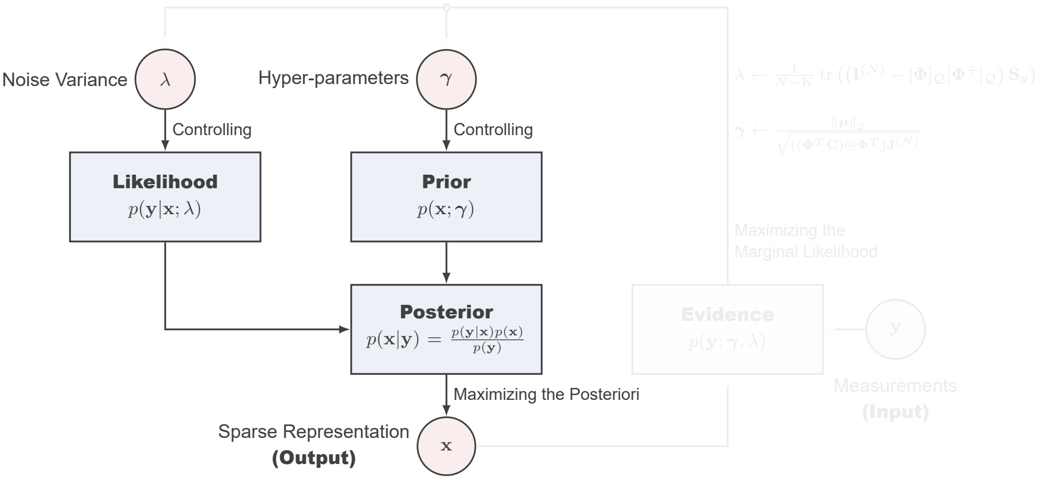


- **Likelihood:** SBL assumes the noise follows a zero-mean Gaussian distribution with variance λ.

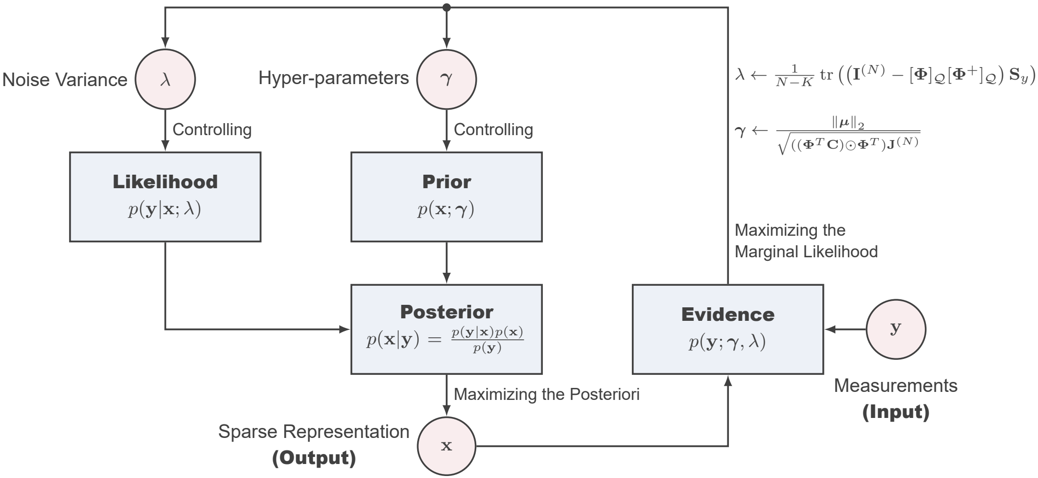
Recover x with SBL - Implement



Recover $\mathbf{x}$ with SBL - Implement

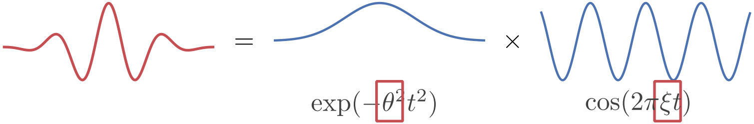


Recover x with SBL - Implement



Determine Φ - Assumptions

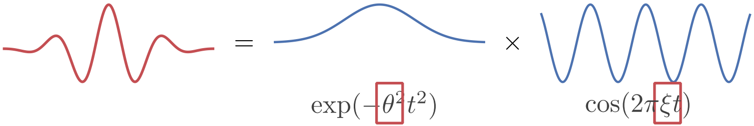
- **Source Signal:** Assume the source signal is a Gaussian-modulated pulse.



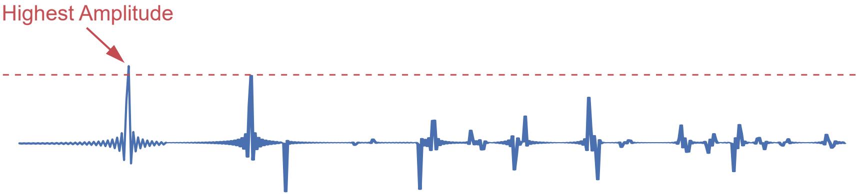
- **Direct Sound:** Assume the direct sound has the highest amplitude and is the time-delayed and scaled version of the source signal.

Determine Φ - Assumptions

- **Source Signal:** Assume the source signal is a Gaussian-modulated pulse.

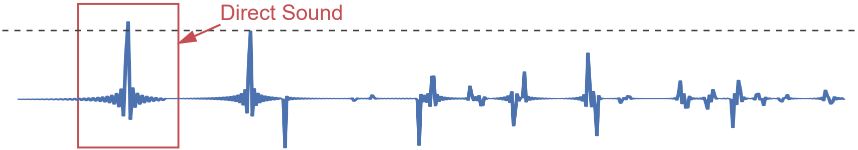


- **Direct Sound:** Assume the direct sound has the highest amplitude and is the time-delayed and scaled version of the source signal.



Determine Φ - Implement

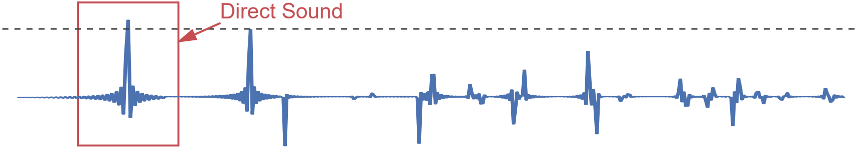
- The direct sound can be extracted by using a **rectangle window** around the peak.



- The parameters of the Gaussian-modulated pulse can be estimated by minimizing the **mean square error** between the measured and reconstructed direct sound.
- The dictionary can be built with the various **time-delayed** source signal.

Determine Φ - Implement

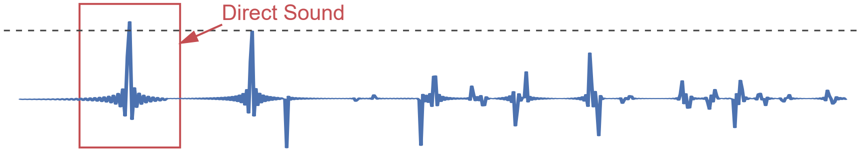
- The direct sound can be extracted by using a **rectangle window** around the peak.



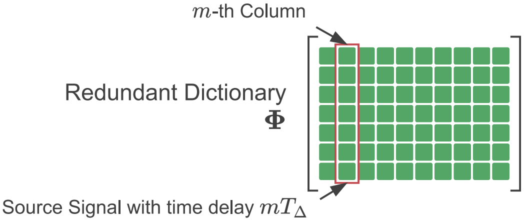
- The parameters of the Gaussian-modulated pulse can be estimated by minimizing the **mean square error** between the measured and reconstructed direct sound.
- The dictionary can be built with the various **time-delayed** source signal.

Determine Φ - Implement

- The direct sound can be extracted by using a **rectangle window** around the peak.

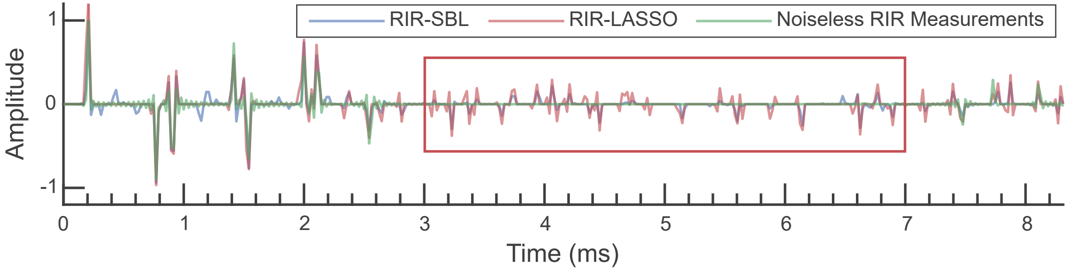


- The parameters of the Gaussian-modulated pulse can be estimated by minimizing the **mean square error** between the measured and reconstructed direct sound.
- The dictionary can be built with the various **time-delayed** source signal.



3. Results

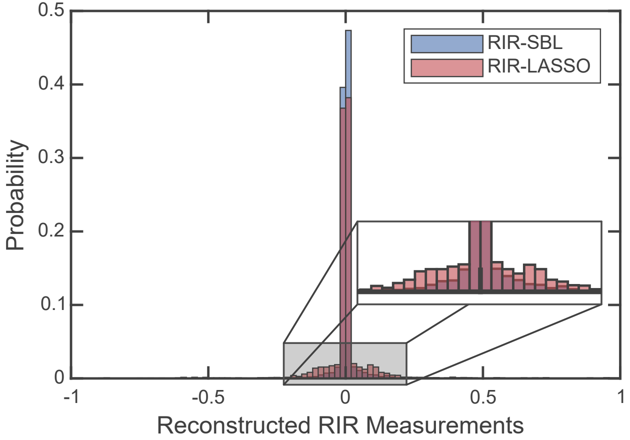
Sparsity - Recovered RIR



- For the simulated RIR measurements[1], RIR-SBL shows less noise sensitiveness to the noise than RIR-LASSO in the time interval [3,7] ms.

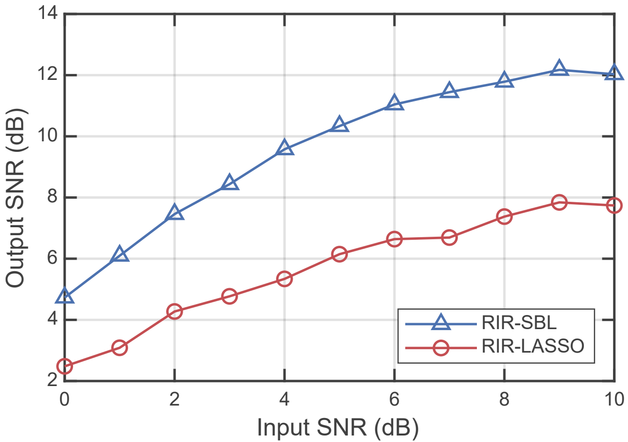
[1]The simulated noiseless RIR measurements are generated by the image method using the gpuRIR library.

Sparsity - Histogram



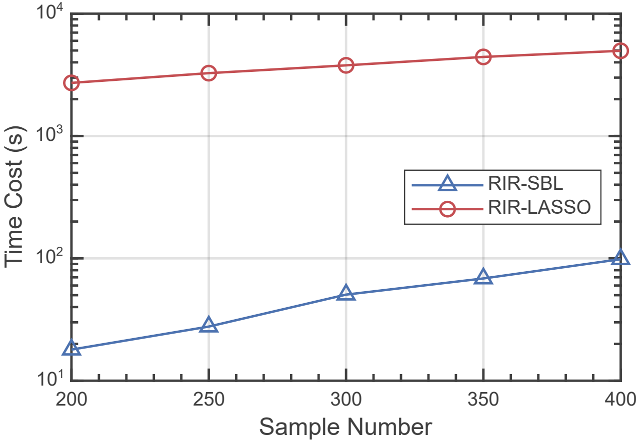
- The reconstruct RIR measurements of RIR-SBL is sparser than that of RIR-LASSO.

Output SNR

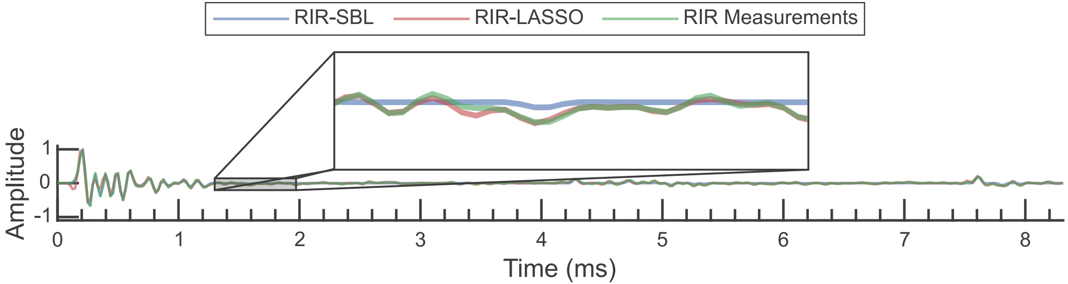


- RIR-SBL provides an improvement in the output SNR compared to RIR-LASSO.

Computational Cost



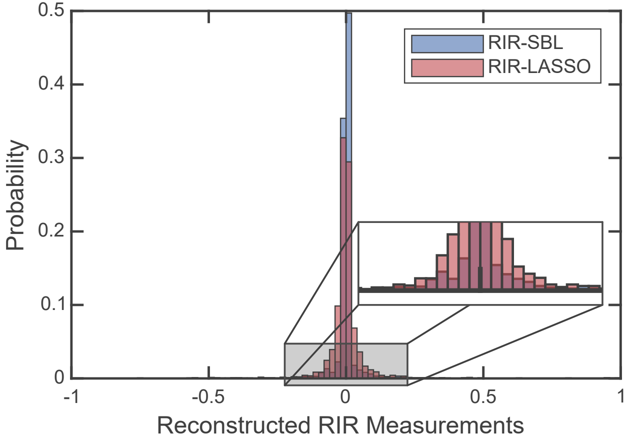
- RIR-SBL requires much less computation than RIR-LASSO.



- For the real-world RIR measurements[2], RIR-SBL performs better in noise reduction.

[2] The real-world RIR measurements are taken from the acoustic characterization of environments (ACE) database.

Real-World Data Processing



- RIR-SBL is more sparse-promoting than RIR-LASSO in real-world RIR processing.

4. Conclusions

Conclusions

- The proposed RIR-SBL works well in **noisy conditions** and can improve the output SNR.
- RIR-SBL is **computationally efficient**.
- RIR-SBL can effectively exploit the sparse structure of the early part of RIRs to promote the **sparsity** of the model.

Conclusions

- The proposed RIR-SBL works well in **noisy conditions** and can improve the output SNR.
- RIR-SBL is **computationally efficient**.
- RIR-SBL can effectively exploit the sparse structure of the early part of RIRs to promote the **sparsity** of the model.

Conclusions

- The proposed RIR-SBL works well in **noisy conditions** and can improve the output SNR.
- RIR-SBL is **computationally efficient**.
- RIR-SBL can effectively exploit the sparse structure of the early part of RIRs to promote the **sparsity** of the model.

Thank You!