An Efficient Technique for Bit Error Probability Estimation of MLSDA-Turbo Decoder

Presenter: Jia-Qin Pan (潘家斳)
Advisory by Prof. Po-Ning Chen
Dept. of Communications Engineering
National Chiao Tung University

Outline

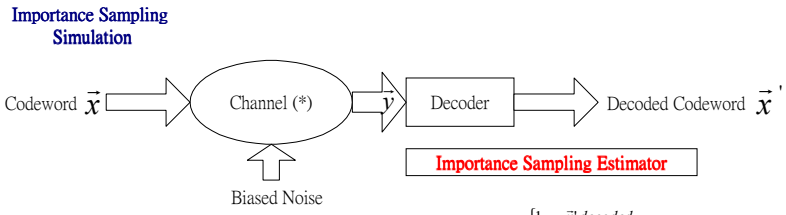
- Introduction and Background
- Preliminaries
  - Fixed-Length Codes and Maximum Metric Decoding
  - Importance Sampling
  - MLSDA for PCCC
- The Important Sampling Method on MLSDA-Turbo Decoder
- Simulation Results
- Conclusions

Introduction and Background

- MLSDA (Maximum-likelihood soft-decision sequential decoding algorithm) was proposed by Chen and Han in 1999. Later, it was applied to the decoding of PCCC codes. Unlike the iterative decoding algorithm, the algorithm turns out to be a maximum-likelihood decoding algorithm for PCCC.
- Unfortunately, for this ML decoder, brute-force MC often requires a very large number of simulation trials in order to achieve a meaningful estimate of system performance.

Introduction and Background

- Importance Sampling(IS) is a modified MC technique which can efficiently reduce the number of simulation trials required to achieve a specified accuracy.



$$I_{\vec{x}'}(\vec{y}) = \begin{cases} 1 & \vec{x}' \text{ decoded} \\ 0 & \text{otherwise.} \end{cases}$$

$$\hat{P}_L^*(\vec{x}'|\vec{x}) = \frac{1}{L}\sum_{l=1}^{L} w(\vec{Y}^{(l)}|\vec{x}) I_{\vec{x}'}(\vec{Y}^{(l)})$$

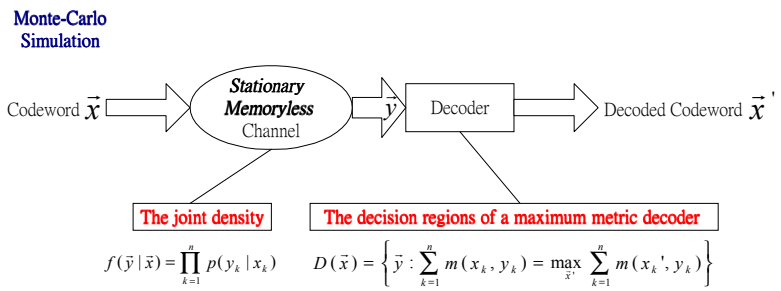
Introduction and Background

- In recent years, there has been considerable interest in applying IS to digital communications systems. Some of them are applied to ML decoder, e.g., Sadowsky developed an error event simulation(EES) method for the simulation of the Viterbi decoders; Khaled and Khurram developed a modified error event simulation(MEES) method for the simulation of the ZJ decoders.
- Here, a new application of the important sampling method for estimating BER of MLSDA-Turbo Decoder is presented.

Preliminaries

Fixed-length Codes and Maximum Metric Decoding

- System Model



Preliminaries

Fixed-length Codes and Maximum Metric Decoding

- System Model

The Indicator function of the decoding region $D(\vec{x}')$

$$I_{\vec{x}'}(\vec{y}) = \begin{cases} 1 & \vec{y} \in D(\vec{x}') \\ 0 & \vec{y} \notin D(\vec{x}'). \end{cases}$$

The specific decoding error probability

$$P(\vec{x}'|\vec{x}) = \oint_{D(\vec{x}')} f(\vec{y}|\vec{x})d\vec{y} = \oint I_{\vec{x}'}(\vec{y}) f(\vec{y}|\vec{x})d\vec{y}$$

The average bit error probability

$$P_b(\vec{x}) = \frac{1}{m} E[N_b | \vec{x} - transmitted] = \frac{1}{m} \sum_{\vec{x}'} n_b(\vec{x}, \vec{x}') P(\vec{x}'|\vec{x})$$

$N_b =$ the number of bits errors that occer after decoding.

$n_b(\vec{x}, \vec{x}') =$ the number of postdecoding information bit errors caused by decoding $\vec{x}'$ instead of $\vec{x}$.

$m = \log_2(|\, codeword \,|) =$ the number of information bits transmitted by a single codeword transmission.

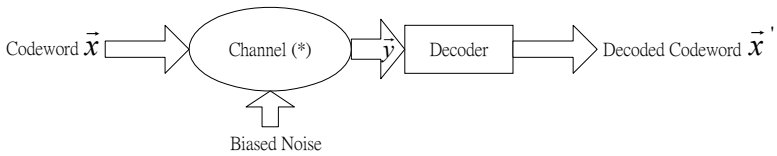
Preliminaries

Importance Sampling

- How to apply importance sampling to coded communications systems?

Importance Sampling Simulation

Importance Sampling Estimator	Importance Sampling Weight
$P(\vec{x}'|\vec{x}) = \oint \frac{f(\vec{y}|\vec{x})}{f^*(\vec{y}|\vec{x})} I_{\vec{x}'}(\vec{y}) f^*(\vec{y}|\vec{x})d\vec{y}$	$w(\vec{y}|\vec{x}) = \frac{f(\vec{y}|\vec{x})}{f^*(\vec{y}|\vec{x})}$
$\hat{P}_L^*(\vec{x}'|\vec{x}) = \frac{1}{L}\sum_{l=1}^{L} w(\vec{Y}^{(l)}|\vec{x}) I_{\vec{x}'}(\vec{Y}^{(l)})$	$I_{\vec{x}'}(\vec{y}) = \begin{cases} 1 & \vec{x}' decoded \\ 0 & otherwise. \end{cases}$



Preliminaries

Importance Sampling

- How to choose a good simulation channel?

The expectation operation for the simulation distribution

$$E^*[\hat{P}_L^*(\vec{x}'|\vec{x})]=E^*[w(\vec{Y}^{(l)}|\vec{x})I_{\vec{x}'}(\vec{Y}^{(l)})]=\oint I_{\vec{x}'}(\vec{y})w(\vec{y}|\vec{x})f^*(\vec{y}|\vec{x})d\vec{y}=\oint I_{\vec{x}'}(\vec{y})\frac{f(\vec{y}|\vec{x})}{f^*(\vec{y}|\vec{x})}f^*(\vec{y}|\vec{x})d\vec{y}=P(\vec{x}'|\vec{x})$$

The variance of the estimator

$$\mathrm{var}[\hat{P}_L^*(\vec{x}'|\vec{x})]=\frac{1}{L}\mathrm{var}^*[w(\vec{Y}|\vec{x})I_{\vec{x}'}(\vec{Y})]=\frac{1}{L}\left\{\oint_{D(\vec{x}')}\left[\frac{f(\vec{y}|\vec{x})}{f^*(\vec{y}|\vec{x})}\right]^2.f^*(\vec{y}|\vec{x})d\vec{y}-P(\vec{x}'|\vec{x})^2\right\}$$

The optimal importance sampling density

$$f_{\vec{x}'}^*(\vec{y}|\vec{x})=\frac{f(\vec{y}|\vec{x})I_{\vec{x}'}(\vec{y})}{P(\vec{x}'|\vec{x})}$$

Preliminaries

Importance Sampling

- How to estimate the precision of importance sampling estimates?

The corresponding estimate of $\mathrm{var}[\hat{P}_L^*(\vec{x}'|\vec{x})]$

$$Var=\mathrm{var}[\hat{P}_L^*(\vec{x}'|\vec{x})]=\frac{1}{L}\mathrm{var}^*[w(\vec{Y}|\vec{x})I_{\vec{x}'}(\vec{Y})]\hat{=}\frac{1}{L}\left[\frac{1}{L}\sum_{l=1}^{L}w(\vec{Y}^{(l)}|\vec{x})^2I_{\vec{x}'}(\vec{Y}^{(l)})-(\hat{P}_L^*(\vec{x}'|\vec{x}))^2\right]$$

The relative precision estimates of $\hat{P}_L^*(\vec{x}'|\vec{x})$

$$RPE_{\hat{P}_L^*}=\frac{\sqrt{Var}}{\hat{P}_L^*(\vec{x}'|\vec{x})}$$

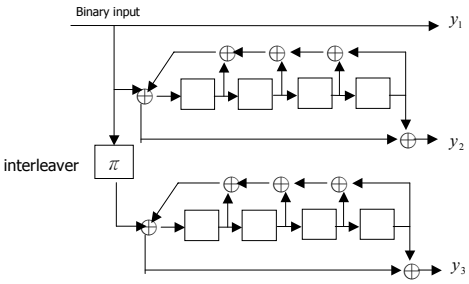
The relative precision estimates of $\hat{P}_b^*(\vec{x})$

$$RPE_{\hat{P}_b^*}=\frac{\sqrt{\mathrm{var}[\hat{P}_b^*(\vec{x})]}}{\hat{P}_b^*(\vec{x})}=\frac{\sqrt{\mathrm{var}[\frac{1}{m}\sum_{\vec{x}'}n_b(\vec{x},\vec{x}')\hat{P}_L^*(\vec{x}'|\vec{x})]}}{\hat{P}_b^*(\vec{x})}$$

Preliminaries
MLSDA for PCCC

- PCCC with G_1=37,G_2=21 as its component code.

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36



Block interleaver with N=36

Preliminaries
MLSDA for PCCC

- Tree-based *Maximum-likelihood* Soft-decision Sequential Decoding Algorithm on **AWGN** channels
  - The conventional sequential decoding algorithm with a new metric

$$M\left(x_i^{(j)} \middle| r_i^{(j)}\right) = \left(x_i^{(j)} \oplus y_i^{(j)}\right) \left| r_i^{(j)} \right|$$

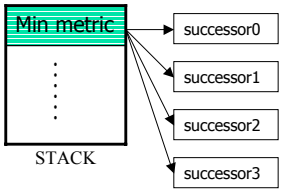
where $x_i^{(j)}$ *and* $r_i^{(j)}$ are respectively the ith bit of the jth transmitted block and ith received bit at jth received block, and

$$y_i^{(j)} = \begin{cases} 1, & if\ r_i^{(j)} < 0; \\ 0 & otherwise \end{cases}$$

Preliminaries

MLSDA for PCCC

- ML decoding algorithm
  - Step1



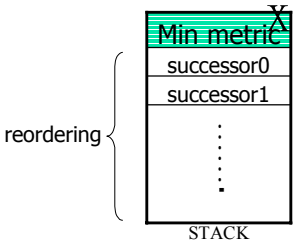
◆ Find the successors of the top path in the STACK

◆ There could be 1 or 2 or 4 successors.

Preliminaries

MLSDA for PCCC

- Step 2



◆ Delete the top path.

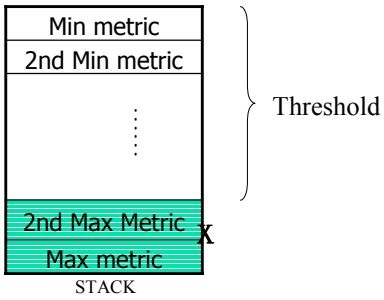
◆ Insert the successors into the STACK.

◆ Reordering the STACK according to the metric.

Preliminaries

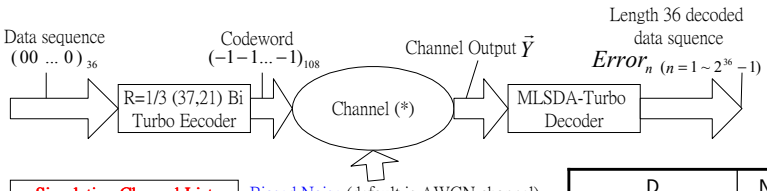
MLSDA for PCCC

- Step 3
  If the length of STACK exceeds the Threshold, delete the path with Max metric in the stack.



The Important Sampling Method on MLSDA-Turbo Decoder

- Simulation Flow



Simulation Channel List

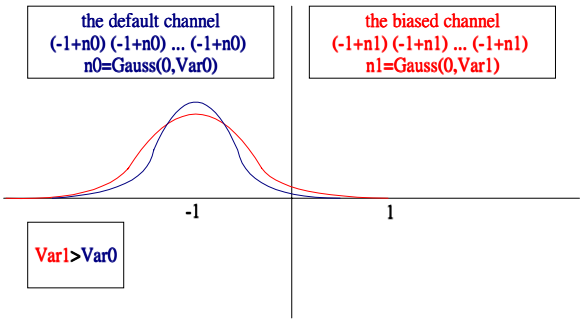
Biased Noise (default is AWGN channel)

I. stationary channels
II. nonstationary channels

D	N_d
3	1
6	2
7	2
9	3
10	25
11	13
12	13

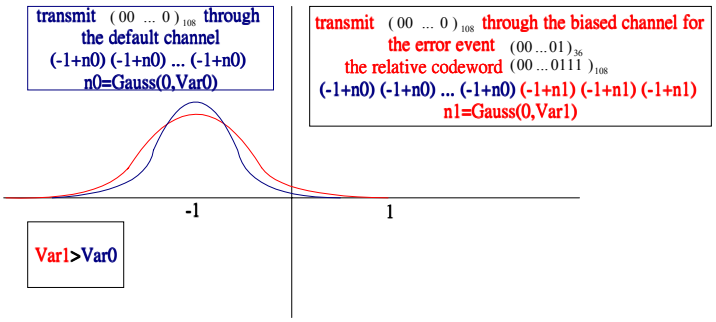
The Important Sampling Method on MLSDA-Turbo Decoder

- Stationary Channel



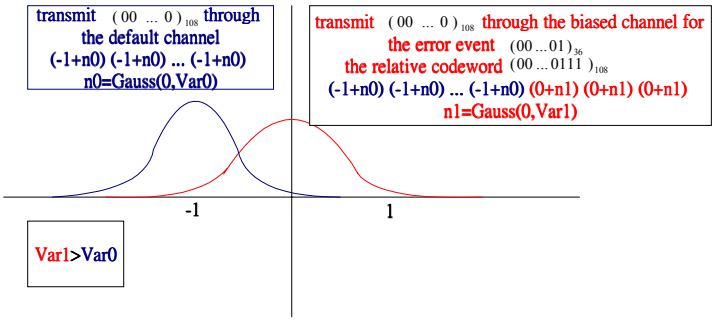
The Important Sampling Method on MLSDA-Turbo Decoder

- Nonstationary channels
  - Channel I



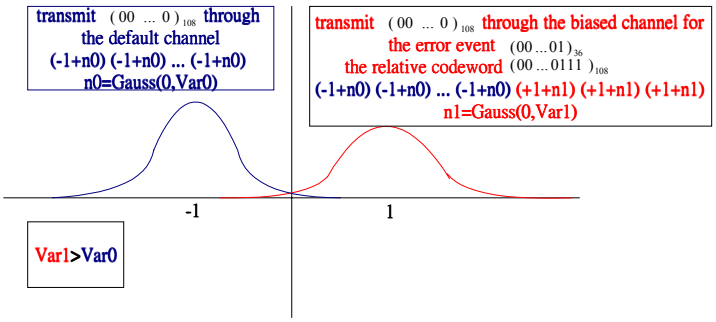
The Important Sampling Method on MLSDA-Turbo Decoder

- Channel II



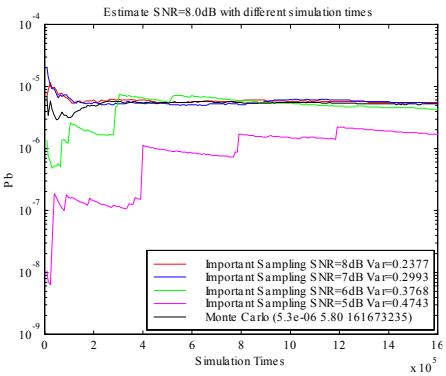
The Important Sampling Method on MLSDA-Turbo Decoder

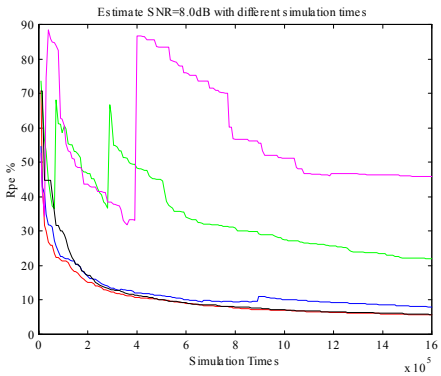
- Channel III



Simulation Results

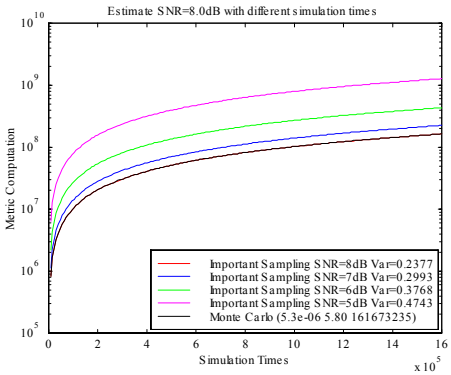
- Estimate the bit error probability of **SNR=8.0dB** by **Biased *Stationary* Channels**





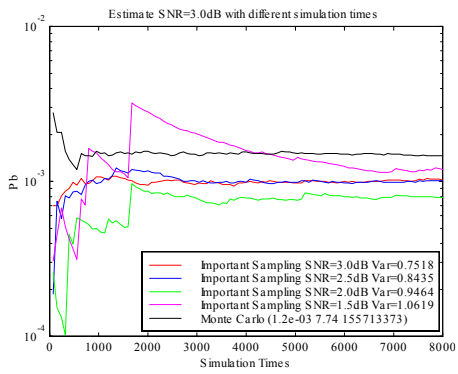
Simulation Results

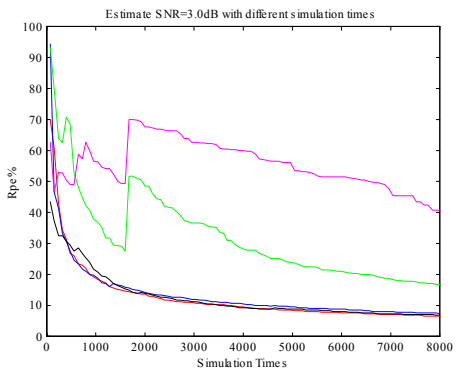
- Time-spent during estimating the bit error probability of **SNR=8.0dB** by **Biased *Stationary* Channels**



Simulation Results

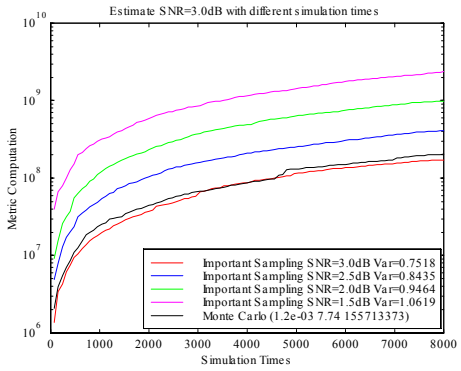
- Estimate the bit error probability of **SNR=3.0dB** by **Biased *Stationary* Channels**





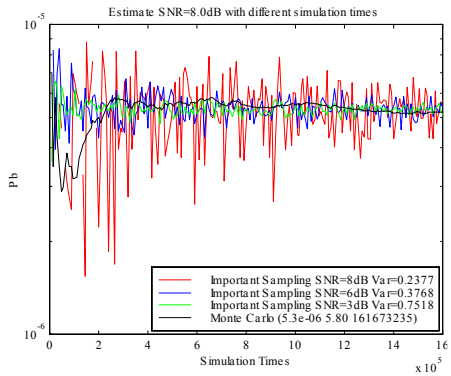
Simulation Results

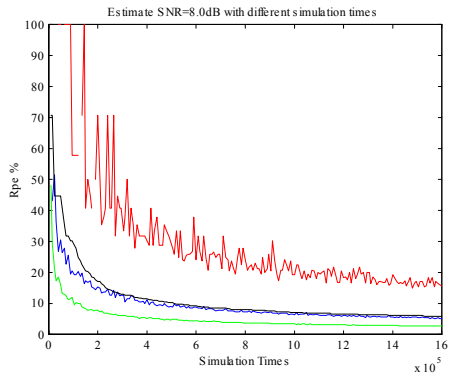
- Time-spent during estimating the bit error probability of **SNR=3.0dB** by **Biased *Stationary* Channels**



Simulation Results

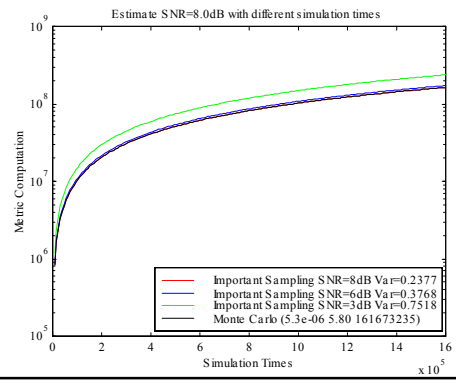
- Estimate the bit error probability of **SNR=8.0dB** by **the *first* Biased *Nonstationary* Channels**





Simulation Results

- Time-spent during estimating the bit error probability of **SNR=8.0dB** by **the *first Nonstationary* Channels**

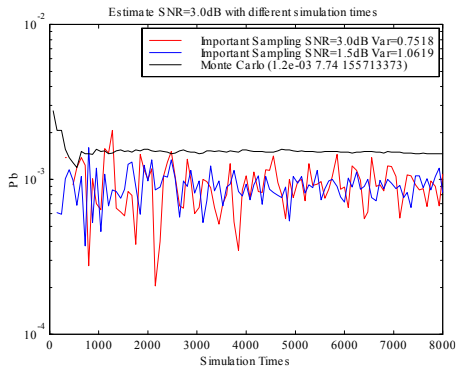


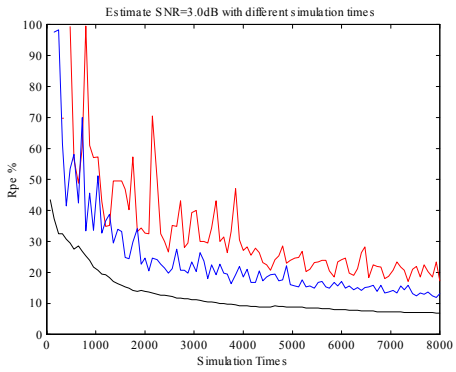
P_b	MC	SNR=6.0	SNR=3.0
	5.3e-06	6.2e-06	5.2e-06
Rpe %	5.80	5.75	5.84
Com	161673235	125344768	47607830
Ratio	1	0.7753	0.2945

$$Ratio = \frac{Com_{IS}}{}$$

Simulation Results

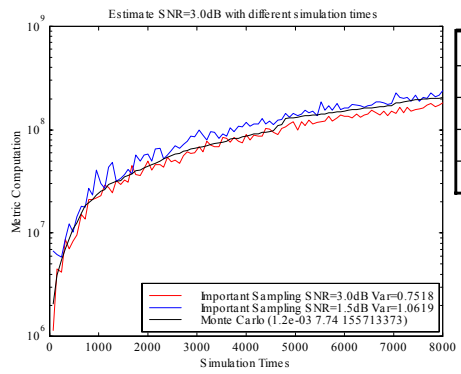
- Estimate the bit error probability of **SNR=3.0dB** by **the first Biased Nonstationary Channels**





Simulation Results

- Time-spent during estimating the bit error probability of **SNR=3.0dB** by **the first Nonstationary Channels**

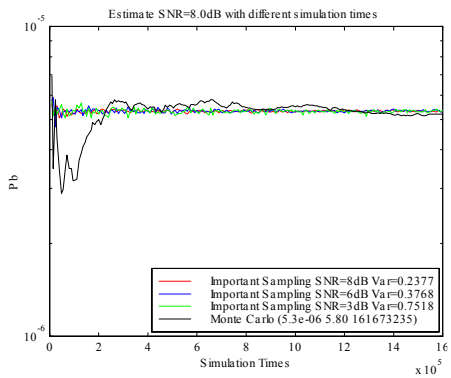


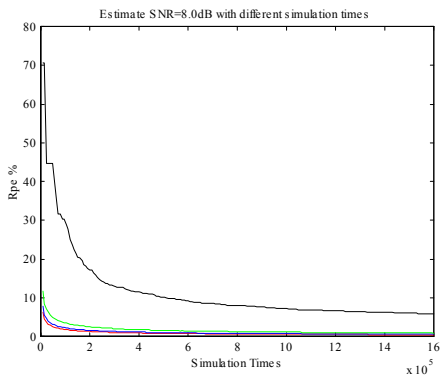
	MC	SNR=3.0	SNR=1.5
P_b	1.2e-03	1.1e-03	1.0e-03
Rpe %	7.74	17.4	17.2
Com	155713373	182182136	113213118
Ratio	1	1.17	0.7271

$$Ratio = \frac{Com^{IS}}{}$$

Simulation Results

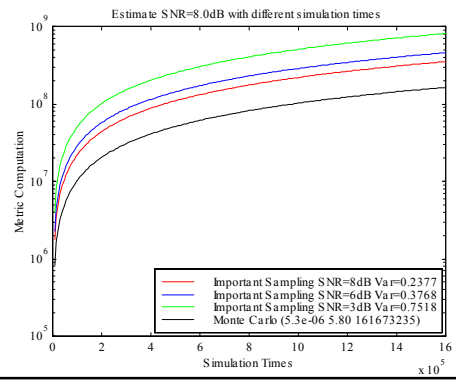
- Estimate the bit error probability of **SNR=8.0dB** by **the second Biased Nonstationary Channels**





Simulation Results

- Time-spent during estimating the bit error probability of **SNR=8.0dB** by **the second Nonstationary Channels**



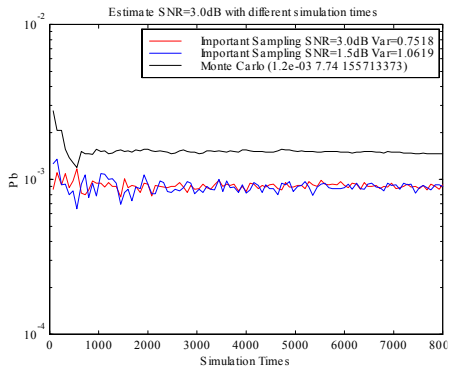
	MC	SNR=8.0	SNR=6.0
P_b	5.3e-06	5.2e-06	5.4e-06
Rpe %	5.80	5.78	5.77
Com	161673235	1947871	4107672
Ratio	1	0.01205	0.02541

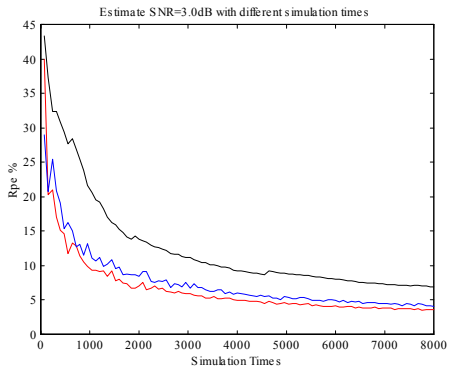
	SNR=3.0
P_b	5.2e-06
Rpe %	5.76
Com	19117489
Ratio	0.11825

$$Ratio = \frac{Com^{IS}}{}$$

Simulation Results

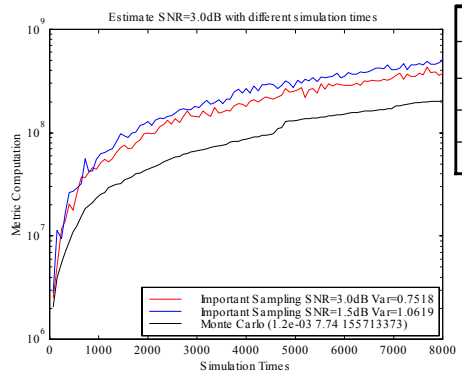
- Estimate the bit error probability of **SNR=3.0dB** by **the second Biased Nonstationary Channels**





Simulation Results

- Time-spent during estimating the bit error probability of **SNR=3.0dB** by **the second Nonstationary Channels**

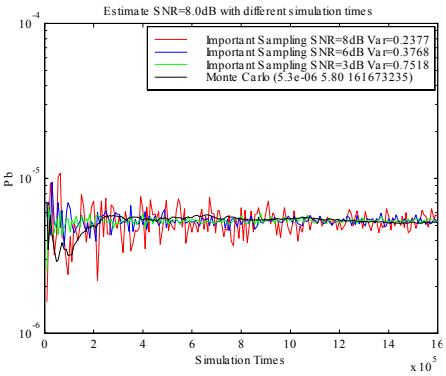


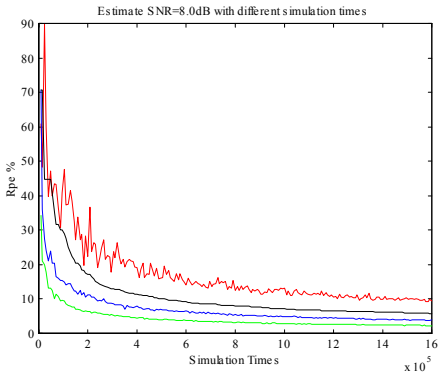
	MC	SNR=3.0	SNR=1.5
P_b	1.2e-03	1.0e-03	9.8e-04
Rpe %	7.74	7.79	7.75
Com	155713373	76220279	139841615
Ratio	1	0.4895	0.8981

$$Ratio = \frac{Com^{IS}}{}$$

Simulation Results

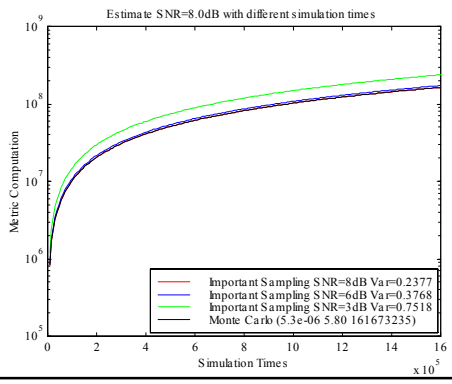
- Estimate the bit error probability of **<u>SNR=8.0dB</u>** by **<u>the third Biased Nonstationary Channels</u>**





Simulation Results

- Time-spent during estimating the bit error probability of **<u>SNR=8.0dB</u>** by **<u>the third Nonstationary Channels</u>**



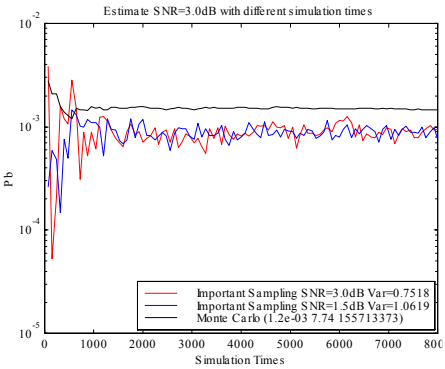
	MC	SNR=6.0	SNR=3.0
P_b	5.3e-06	5.3e-06	5.4e-06
Rpe %	5.80	5.83	5.85
Com	161673235	76910196	36824995
Ratio	1	0.4757	0.2278

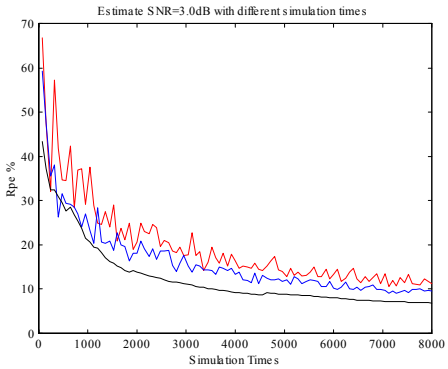
$$Ratio = \frac{Com^{IS}}{}$$

34

Simulation Results

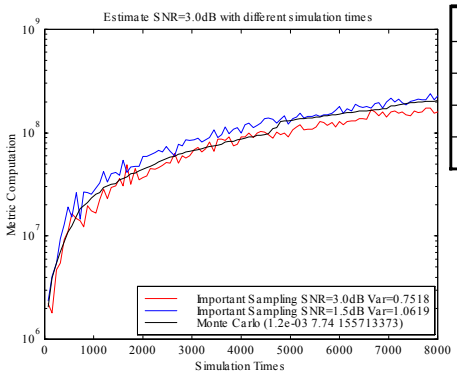
- Estimate the bit error probability of **SNR=3.0dB** by **the third Biased Nonstationary Channels**





Simulation Results

- Time-spent during estimating the bit error probability of **SNR=3.0dB** by **the third Nonstationary Channels**



	MC	SNR=3.0	SNR=1.5
P_b	1.2e-03	9.5e-04	9.8e-04
Rpe %	7.74	10.9	10.7
Com	155713373	171978318	148068555
Ratio	1	1.1045	0.9509

$$Ratio = \frac{Com^{IS}}{}$$

36

Simulation Results

- Estimate the bit error probability by the Best Biased Nonstationary Channel in order to achieve the same estimates of system performance

Monte Carlo Simulation for recording at 300 bit errors									
SNR(dB)	0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
Pb	6.5e-02	3.8e-02	2.4e-02	9.8e-03	4.8e-03	2.2e-03	1.2e-03	7.6e-04	4.5e-04
Rpe	17.2	17.2	16.7	14.2	13.4	9.83	7.74	6.87	6.26
Com	222276385	192710434	206555768	215889806	208319093	185857455	155713373	93004594	63063107

Importance Sampling Simulation for recording at the same RPE									
SNR(dB)	0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
Pb	1.2e-02	9.3e-03	4.2e-03	3.5e-03	2.8e-03	1.3e-03	1.0e-03	6.2e-04	4.3e-04
Rpe	17.3	17.2	16.3	14.3	13.3	9.76	7.79	6.89	6.26
Com	206742239	190742239	181950128	163045159	89587250	98307554	76220279	50410610	32072173
Gain	0.93011	0.98979	0.88088	0. 75522	0.43005	0.52894	0.48949	0.54202	0.50857

Simulation Results

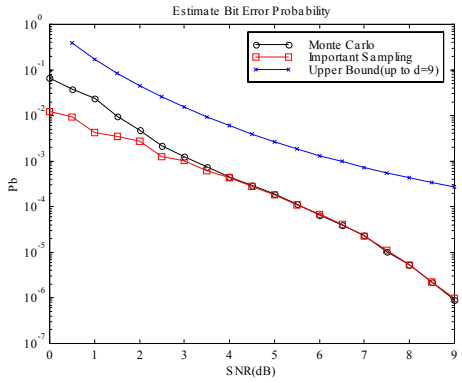
- Estimate the bit error probability by the Best Biased Nonstationary Channel in order to achieve the same estimates of system performance

Monte Carlo Simulation for recording at 300 bit errors									
4.5	5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0
2.9e-04	1.9e-04	1.1e-04	6.7e-05	3.9e-05	2.4e-05	1.0e-05	5.3e-06	2.2e-06	8.9e-07
6.22	6.03	5.92	5.86	5.84	5.81	5.78	5.80	5.76	5.76
47388940	35587019	33017961	33922090	39719359	49444037	94346705	161673235	358490311	856760117

Importance Sampling Simulation for recording at the same RPE									
4.5	5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0
2.8e-04	1.8e-04	1.1e-04	6.8e-05	4.0e-05	2.3e-05	1.1e-05	5.2e-06	2.2e-06	9.7e-07
6.22	6.09	5.96	5.80	5.84	5.81	5.81	5.78	5.77	5.79
18694896	13420903	8055117	5310740	3867645	2886998	2384527	1947871	1933822	1674667
0.39450	0.37713	0.24396	0.15656	0.09737	0.05839	0.02527	0.01205	0.00539	0.00195

Simulation Results

- Monte-Carlo VS Importance-Sampling



Simulation Results

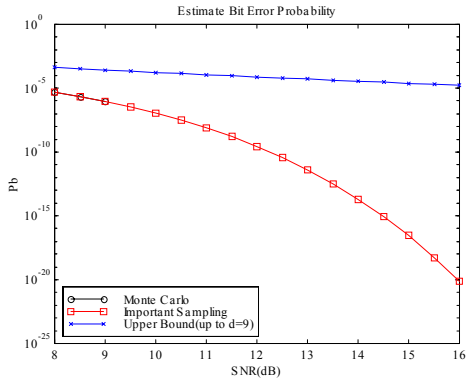
- Estimate the bit error probability by the Best Biased Nonstationary Channel in order to achieve the same estimates of system performance

Importance Sampling Simulation for recording at RPE=6%							
SNR(dB)	9.5	10.0	10.5	11.0	11.5	12.0	12.5
Pb	3.5e-07	1.1e-07	3.3e-08	7.9e-09	1.6e-09	2.7e-10	3.8e-11
Rpe	6.10	6.04	6.00	6.03	6.04	6.07	6.06
Com	1397725	1389150	1348576	1317392	1384507	1391025	1402515

Importance Sampling Simulation for recording at RPE=6%							
SNR(dB)	13.0	13.5	14.0	14.5	15.0	15.5	16.0
Pb	3.9e-12	3.3e-13	1.9e-14	8.5e-16	2.8e-17	5.3e-19	7.3e-21
Rpe	6.05	6.09	6.10	6.08	6.09	6.05	6.01
Com	1483599	1563825	1610578	1700387	1752832	1805472	1942095

Simulation Results

- Monte-Carlo VS Importance-Sampling



Conclusions

- There is no computational gain when we use the Importance Sampling Simulation by Biasd Stationary Channels. In fact, the noisier channel we use the more metric computations we pay, and it brings out the larger relative precision because of its larger variation than MC Simulation.

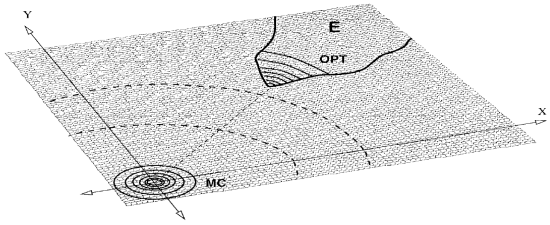
Conclusions

- The optimal distribution of the biased channel

$$f^*_{\vec{x}'}(\vec{y}\,|\,\vec{x}) = \frac{f(\vec{y}\,|\,\vec{x})I_{\vec{x}'}(\vec{y})}{P(\vec{x}'\,|\,\vec{x})}$$

can show why the second nonstationary channel causes the best performance among all channels.

E - Error region
MC - Monte Carlo density
OPT - Optimal density



Conclusions

- In this Simulation, the Importance Sampling can work well in high SNR, but less effectiveness in low SNR when we use the second biased channel. In fact, We gained 10-500 time-saving when we estimate bit error probability between SNR=6dB to 9dB, and gained 0-10 time-saving for SNR=0dB to 5.5dB.