A Variable Rate Source-Coding Theorem for Unstable Scalar Markov Processes

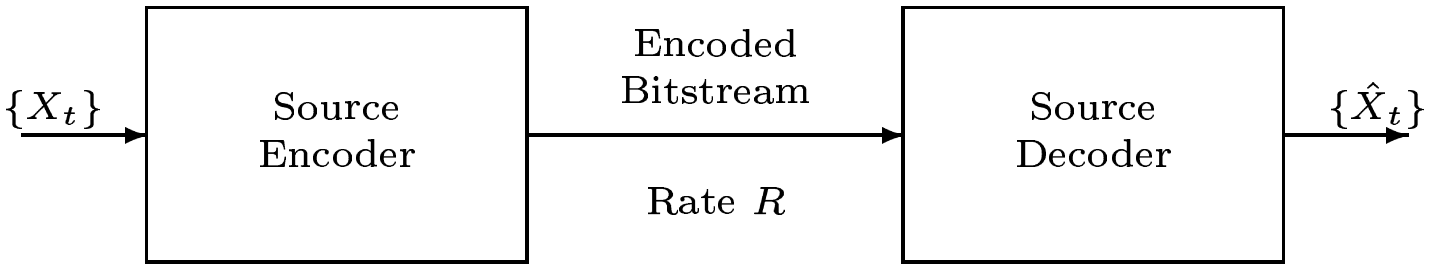
Anant Sahai
MIT LIDS and Enuvis, Inc.
sahai@alum.mit.edu

June 26, 2001

Outline

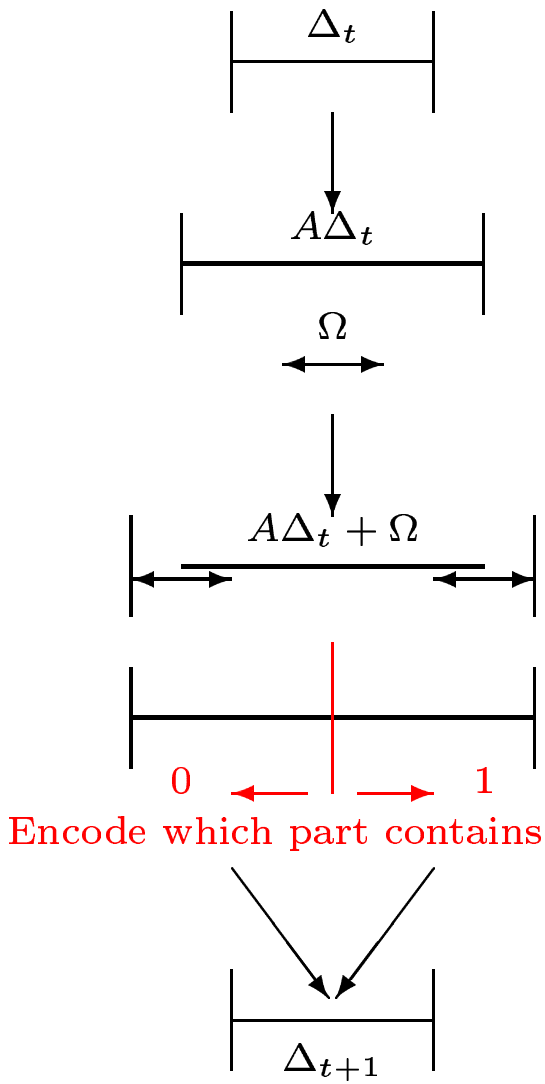
- Problem Statement
- Review of Causal Coding
- Distortion-Rate Function
- Source Coding Theorem
- Review of Berger's Wiener Process Proof
- Sketch of Our Proof

The Source Coding Problem



- Translate ongoing signal ($X_{t+1} = AX_t + W_t$, $A \geq 1$) to bits
- For a given rate R, we would like to minimize average $E[(X_t - \hat{X}_t)^2]$ even as $t \to \infty$
- Encoder and decoder need to work "online," but we may be willing to tolerate some finite delay between X_t and $\hat{X}_t$
- Assume that driving noise $\{W_t\}$ has bounded support so $W_t \in [-\frac{\Omega}{2}, \frac{\Omega}{2}]$ for convenience

Causal Source Code: Keep X_t In A Box



Window around $\hat{X}_t$ known to contain X_t

Window grows by factor of $A > 1$ due to the dynamics and by a constant due to the bounded driving noise W_t

giving rise to larger window of uncertainty for X_{t+1}

By sending n bits, we cut the window by a factor of 2^{-n}

giving a new window around the updated estimate $\hat{X}_{t+1}$

- If $R > \log_2 A$, we can keep the Δ_t bounded

What if we are willing to accept delay?

- Define Distortion-Rate function as limit of finite horizon problems:

$$D(R) = \liminf_{N\to\infty} \inf_{\substack{p(\hat{X}_1^N|X_1^N) \\ \frac{1}{N}I(X_1^N;\hat{X}_1^N)\le R}} E\left[\frac{1}{N}\sum_{t=1}^{N}(X_t-\hat{X}_t)^2\right]$$

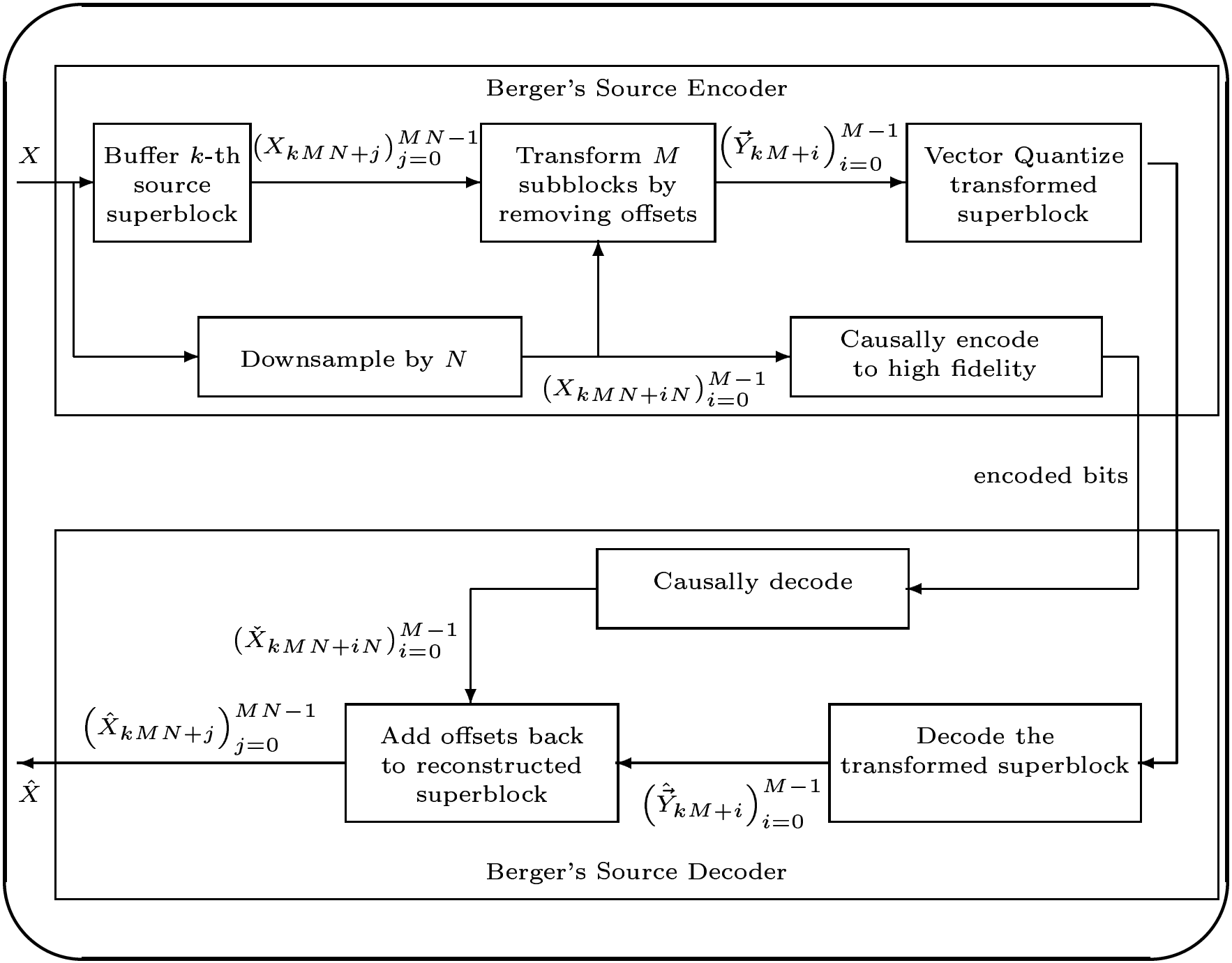
- Can clearly do no better than this, even if we knew the realization of the entire semi-infinite source stream before we started.
- Can we approach this bound for $A \geq 1$?
  - Gray solved finite horizon problem for $A \geq 1$ in 1970
  - Berger solved infinite horizon for $A = 1$ in 1970
  - Infinite horizon ("streaming") problem for $A > 1$ remained open till now

Source Coding Theorem

For the unstable Markov Source ($X_{t+1} = AX_t + W_t$, $A \geq 1$), there exist variable rate streaming codes that achieve average bit-rate and distortion arbitrarily close to $D(R)$ provided we are willing to tolerate enough end-to-end delay. This delay stays bounded for all time.

In his 1970 paper on codes for autoregressive processes, Gray proves a source coding theorem, but comments:

> It should be emphasized that when the source is non-stationary, the above theorem is not as powerful as one would like. Specifically, it does not show that one can code a long sequence by breaking it up into smaller blocks of length n and use the same code to encode each block. The theorem is strictly a "one-shot" theorem unless the source is stationary, simply because the blocks $[(k-1)n, kn]$ do not have the same distribution for unequal k when the source is not stationary.
>
> Berger has proved the stronger form of the positive coding theorem for the Wiener process by using a specific coding scheme (delta modulation) to track the starting point of each block.


Berger's Source Encoder
X
Buffer k-th source superblock
$(X_{kMN+j})_{j=0}^{MN-1}$
Transform M subblocks by removing offsets
$(\vec{Y}_{kM+i})_{i=0}^{M-1}$
Vector Quantize transformed superblock
Downsample by N
$(X_{kMN+iN})_{i=0}^{M-1}$
Causally encode to high fidelity
encoded bits
Causally decode
$(\check{X}_{kMN+iN})_{i=0}^{M-1}$
$(\hat{X}_{kMN+j})_{j=0}^{MN-1}$
Add offsets back to reconstructed superblock
$(\hat{\vec{Y}}_{kM+i})_{i=0}^{M-1}$
Decode the transformed superblock
$\hat{X}$
Berger's Source Decoder

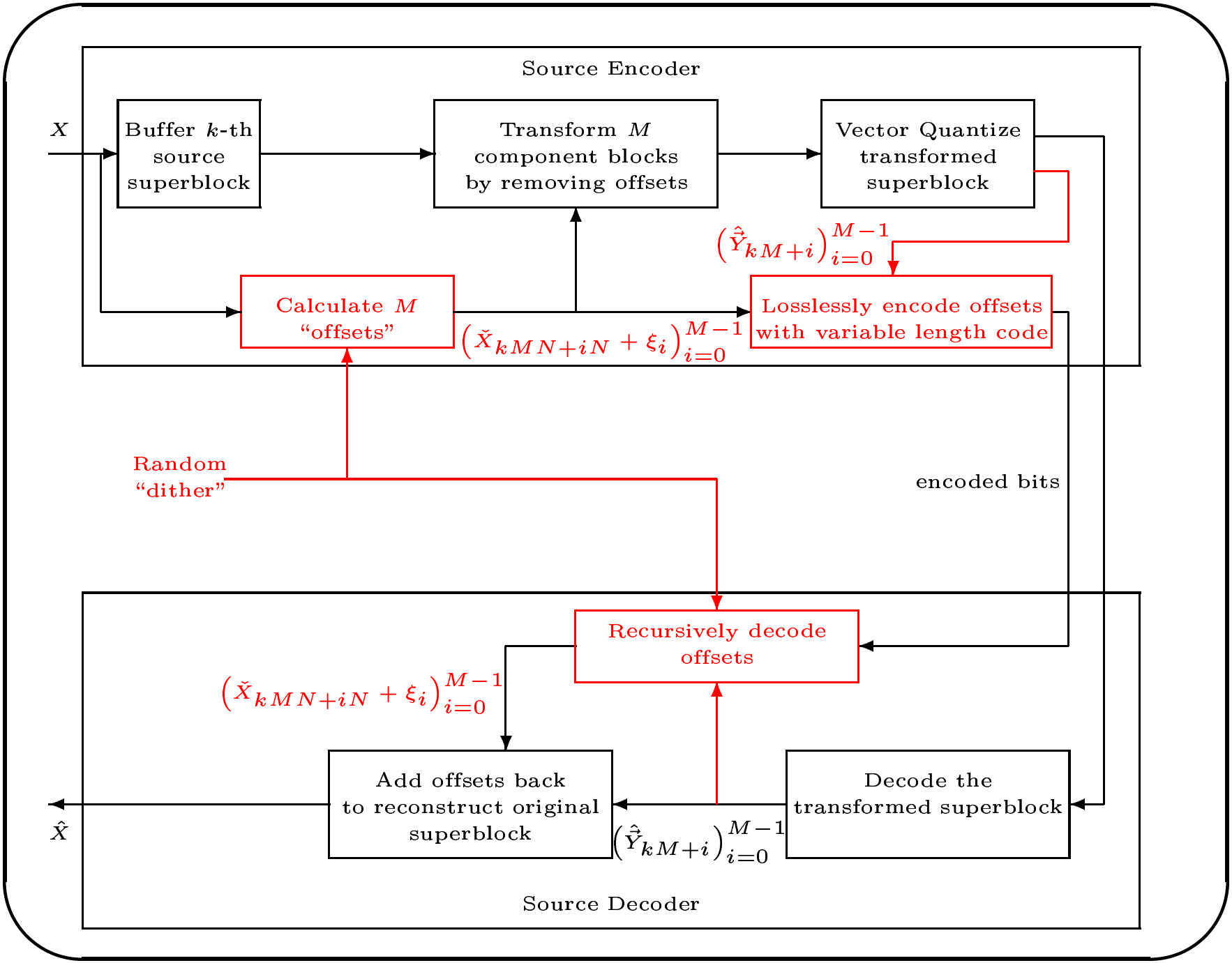
Why Berger's Strategy Does Not Work For $A > 1$

- $A = 1$
  - Independent increments process. Subtracting a downsampled stream results in i.i.d. subblocks.
  - The lower bound on rate is $\log_2 A^N = 0$. For large N, the cost of causally encoding the downsampled stream in parallel to a specified fidelity grows only as $\log_2 N$ which becomes negligible relative to N.
- $A > 1$
  - Not independent increments, but the underlying system is *time invariant* so we can still get i.i.d. subblocks.
  - The lower bound on rate is $\log_2 A^N = N \log_2 A$ for the downsampled stream. It never becomes negligible relative to N.

Augment the Strategy

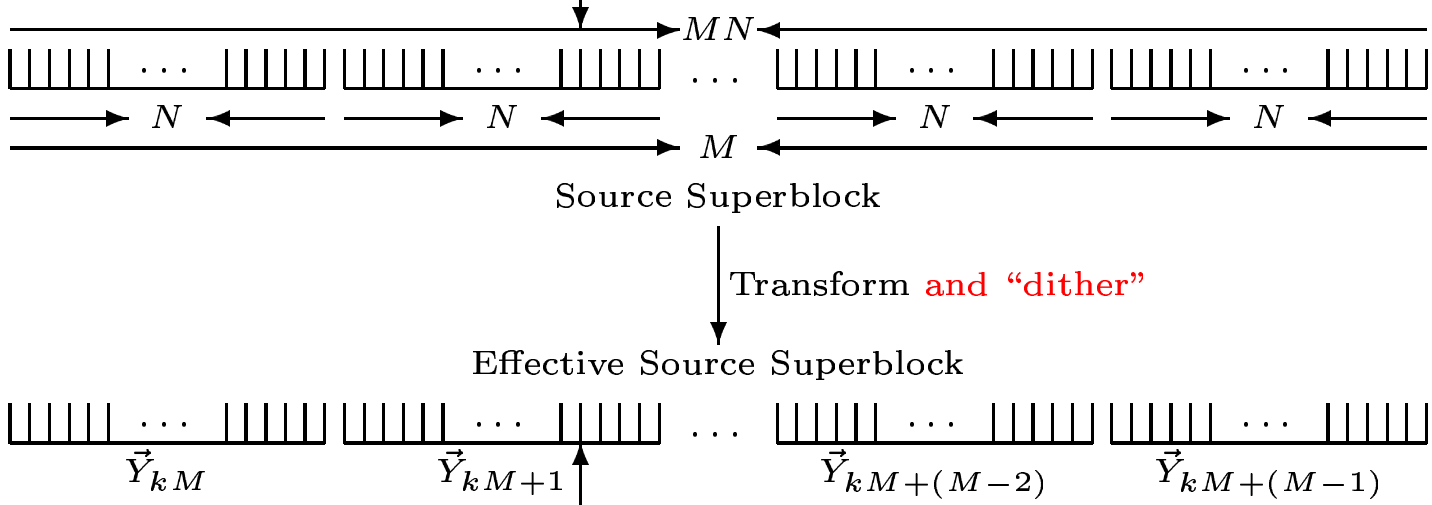
- Update the transformation to counteract the effect of A.
- Use random "dither" to get quantized offsets.
- Make the encoding of the "downsampled" offset depend on the encoding of the main data blocks. Predict the next offset position using the previous offset and the main data block.
- Use a variable length lossless code to encode the resulting integer-valued correction process.
- Take limits carefully.

Lab. Information & Decision Systems, MIT



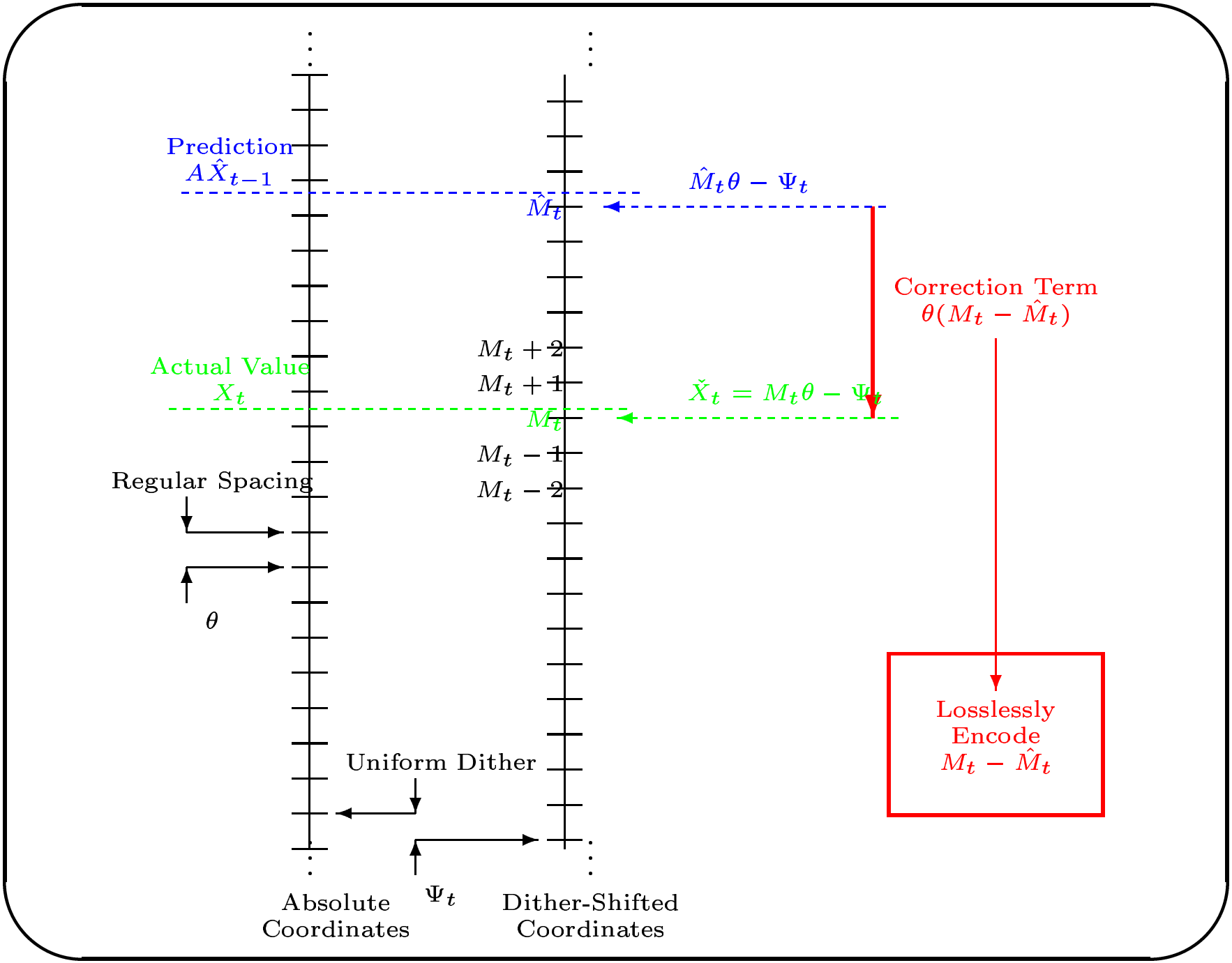
How To Transform The Superblock

$$X_{kMN+iN+j} = AX_{(kMN+iN+j)-1} + W_{(kMN+iN+j)-1}$$



$$\begin{aligned} Y_{kM+i,j} &= X_{kMN+iN+j} - A^j(\check{X}_{kMN+iN} + \xi_{kM+i}) \\ &= AY_{kM+i,(j-1)} + W_{(kMN+iN+j)-1} \end{aligned}$$

- Uses time invariance to make sub-blocks look alike
- Since $A > 1$, for large N it is very sensitive to errors in offsets $\check{X}_t$. Need to know them almost perfectly.


Prediction
$A\hat{X}_{t-1}$
$\hat{M}_t\theta - \Psi_t$
$\hat{M}_t$
Correction Term
$\theta(M_t - \hat{M}_t)$
$M_t + 2$
$M_t + 1$
Actual Value
X_t
$\check{X}_t = M_t\theta - \Psi_t$
M_t
$M_t - 1$
$M_t - 2$
Regular Spacing
θ
Losslessly
Encode
$M_t - \hat{M}_t$
Uniform Dither
Ψ_t
Absolute
Coordinates
Dither-Shifted
Coordinates

Why this works

- The transformed sub-blocks look like finite horizon versions of the original problem
- The sub-blocks look i.i.d. to the VQ, so it approaches $D(R)$
- Encoding the offset stream involves losslessly encoding the correction terms
  - Since the code is good, the predictions are close to the actual values and $E[(X_t - A\hat{X}_t)^2] \leq K$
  - By Chebychev, the integer correction term $(M_t - \hat{M}_t)$ has at most a power-law tail in its distribution
  - Encoding integers takes logarithmic bits and so the code-length distribution has at most an exponential tail
  - Thus it has a small expected rate which becomes negligible relative to N