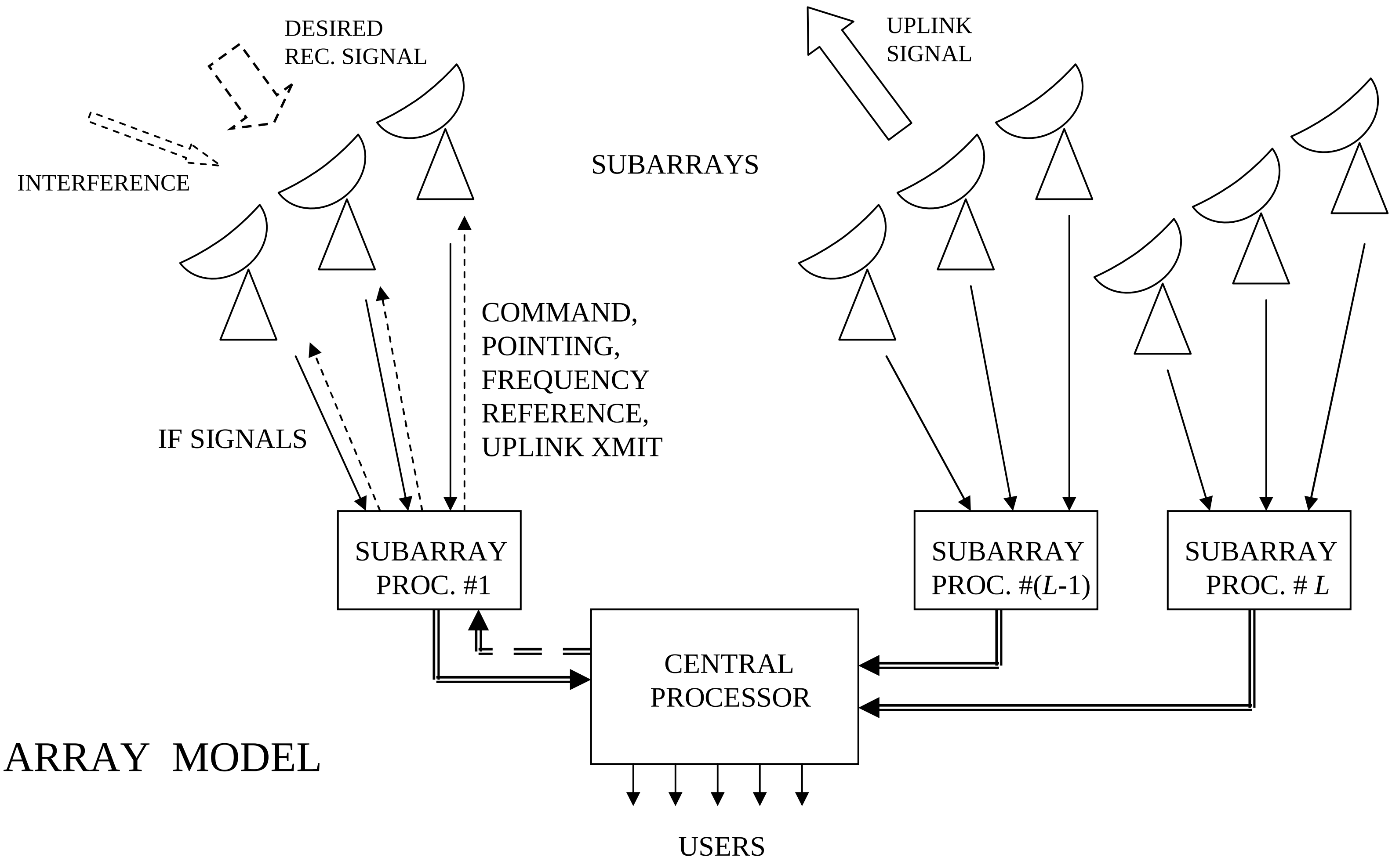
LARGE ARRAY SIGNAL PROCESSING FOR DSN APPLICATIONS: PART I

VIC VILNROTTER

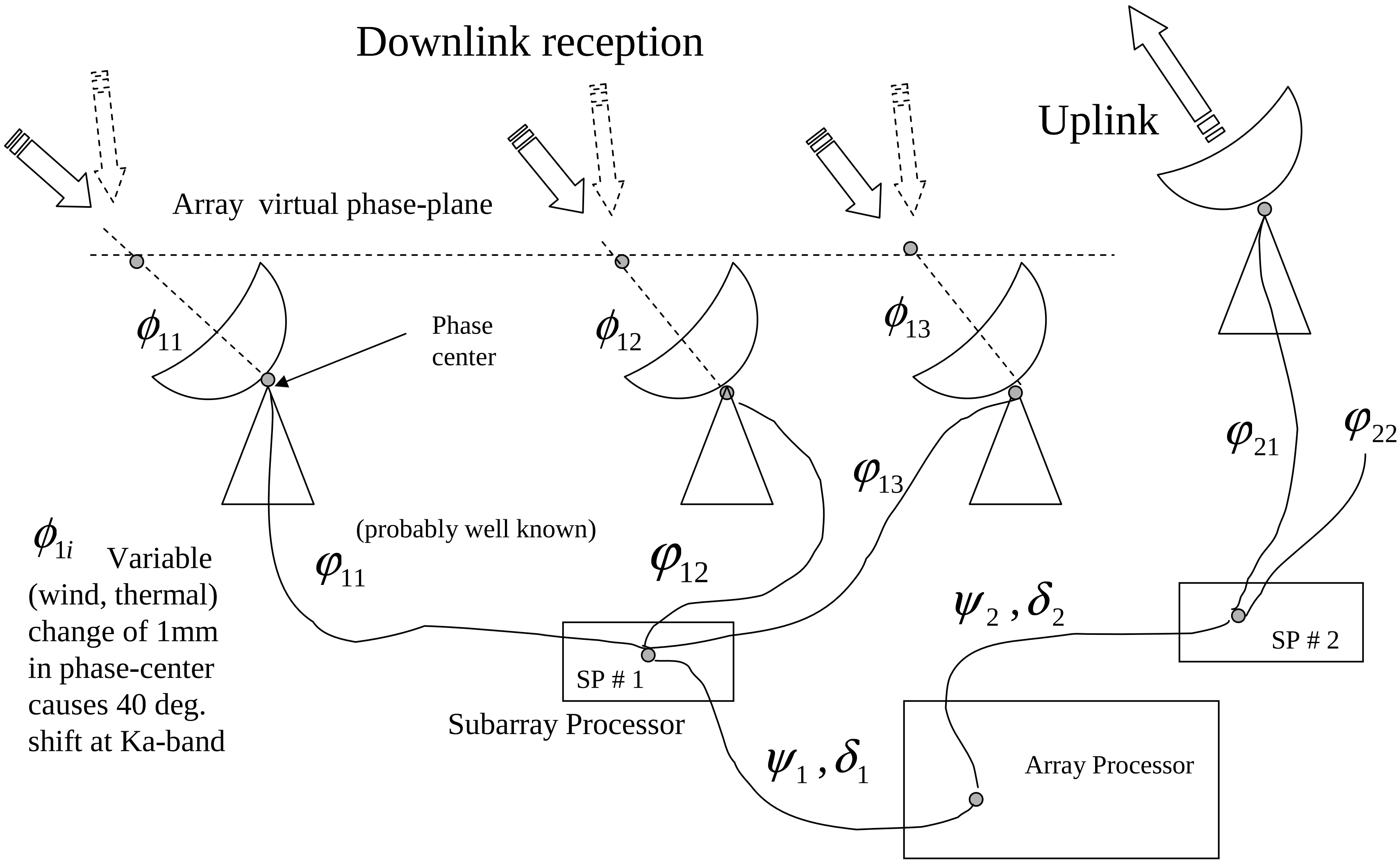
MAY 21, 2002

LARGE ARRAY SIGNAL PROCESSING

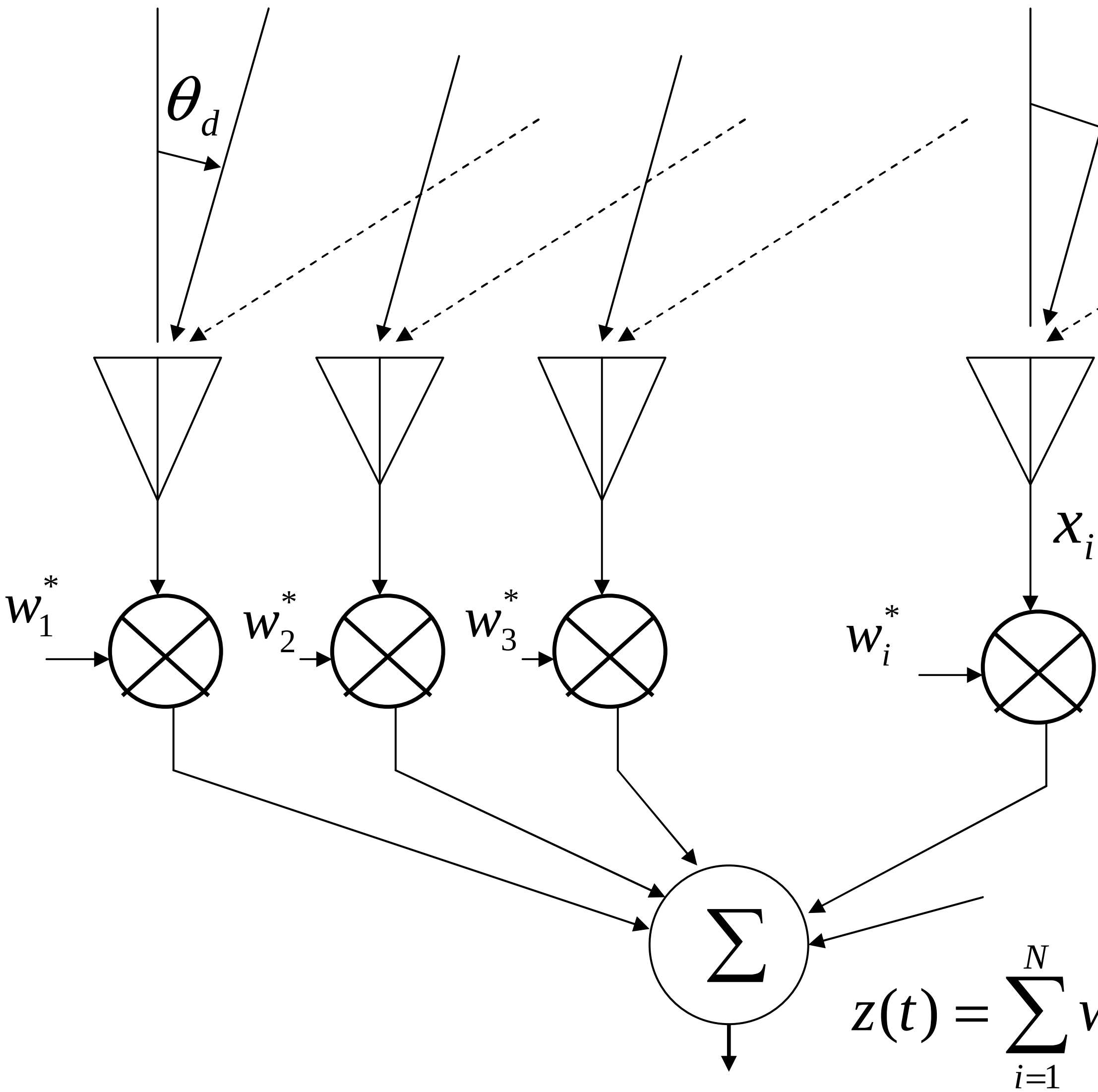


ARRAY MODEL

LARGE ARRAY SIGNAL PROCESSING



LARGE ARRAY SIGNAL PROCESSING



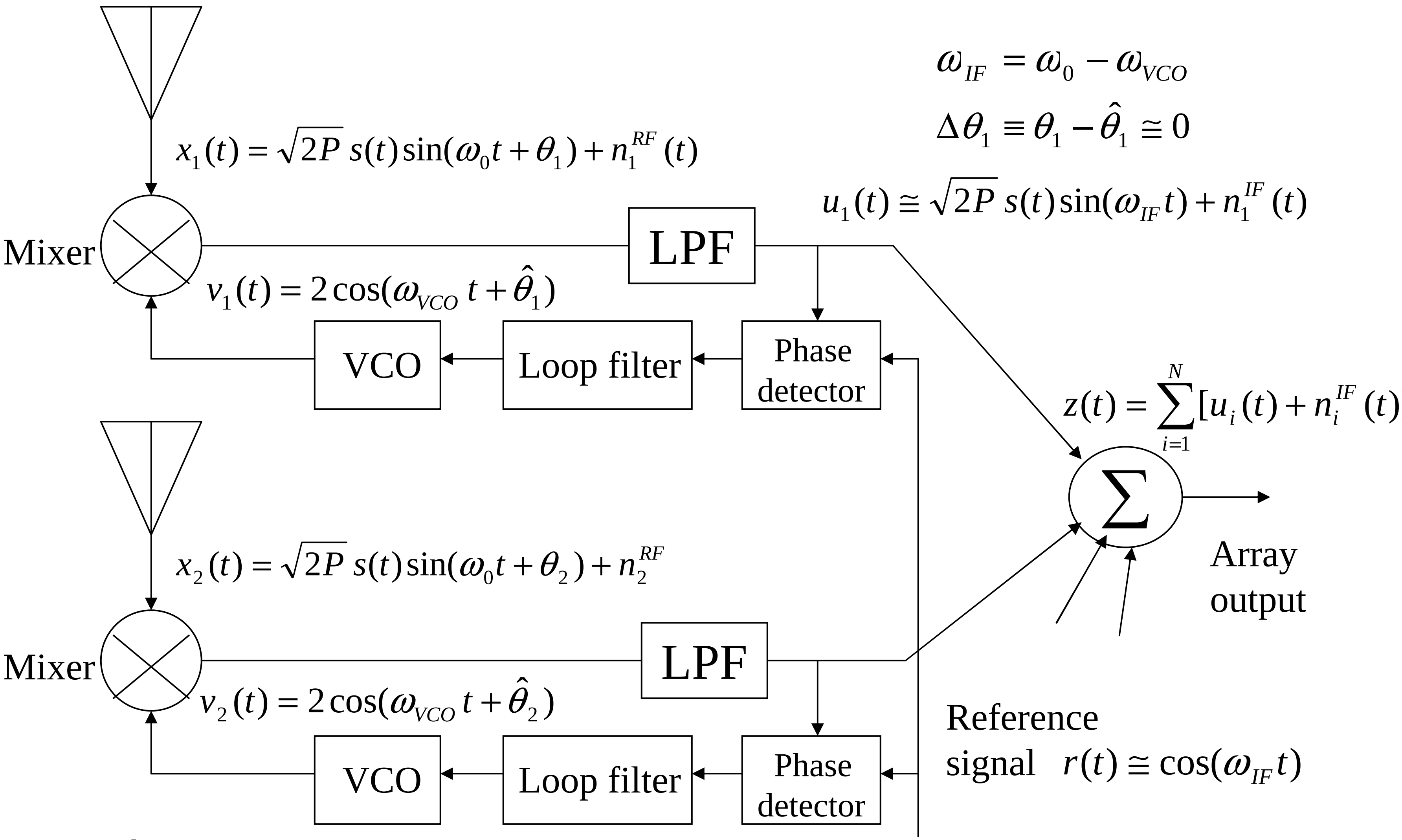
$$x_i(t) = d_i(t) + I_i(t) + n_i(t)$$

Reed, Mallett, Brennan (RMB) notation: $w_i^* x_i$

$$z(t) = \sum_{i=1}^{N} w_i^* \, x_i(t)$$

LARGE ARRAY SIGNAL PROCESSING

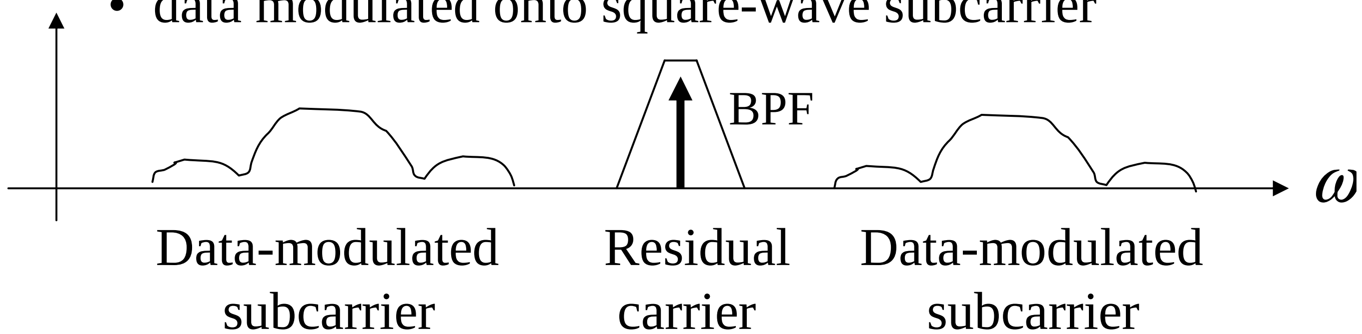
THE PHASE-LOCK LOOP ARRAY



LARGE ARRAY SIGNAL PROCESSING

- Reference Signal Requirements
  - must be **correlated** with the desired signal
  - must be **uncorrelated** with interference

- EXAMPLE: NASA Deep-Space Modulation Format
  - residual carrier usually present
  - data modulated onto square-wave subcarrier

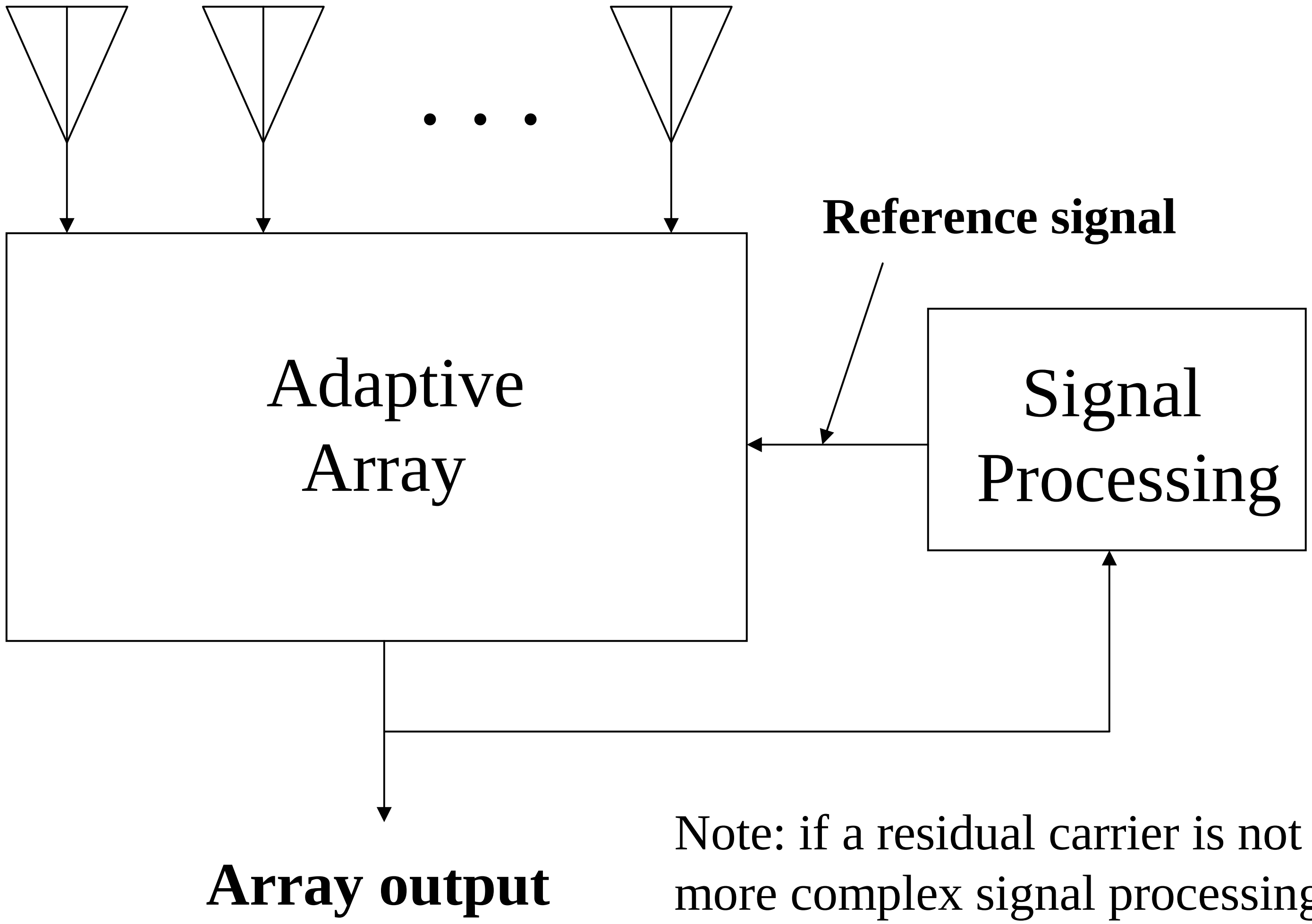


- *PLL* Array Output *SNR* (perfect reference):

$$SNR \equiv \frac{\left(N\sqrt{2P}s(t)\sin(\omega_{IF}(t))\right)^2}{N\sigma^2} = N\frac{P}{\sigma^2}$$

LARGE ARRAY SIGNAL PROCESSING

REFERENCE SIGNAL GENERATION



Note: if a residual carrier is not available, more complex signal processing often yields a useful reference signal.

LARGE ARRAY SIGNAL PROCESSING

COMPLEX VECTOR FORMULATION OF THE **LARGE ARRAY SIGNAL PROCESSING** PROBLEM

$$\mathbf{X} = [x_1(t), x_2(t), \cdots, x_N(t)]^T; \quad \mathbf{W} = [w_1, w_2, \cdots, w_N]^T$$

$$\mathbf{X} = \mathbf{X}_d + \mathbf{X}_I + \mathbf{X}_n$$

$$z(t) = \sum_{i=1}^{N} w_i^* \, x_i(t) = \mathbf{W}^{*T}\mathbf{X} \equiv z_d(t) + z_I(t) + z_n(t)$$

$$P_d = E\,|\,z_d(t)\,|^2; \quad P_I = E\,|\,z_I(t)\,|^2; \quad P_n = E\,|\,z_n(t)\,|^2$$

$$SINR = \frac{P_d}{P_I + P_n} = \frac{P_d}{P_u}; \quad \text{want } \mathbf{W}_{opt} \text{ that maximizes } SINR$$

LARGE ARRAY SIGNAL PROCESSING

NARROWBAND ASSUMPTION:

$$\mathbf{X}_d = s(t)\,\mathbf{U}_d \quad \text{signal times "source direction vector"}$$

$$= A_d(t)\{\exp[j(\psi_d + \varphi_d(t))]\}\,[1, e^{j\theta_{d2}}, \cdots, e^{j\theta_{dN}}]^T$$

DESIRED ARRAY OUTPUT SIGNAL AND POWER:

$$y_d(t) = \mathbf{W}^{*T}\mathbf{X}_d = s(t)\,\mathbf{W}^{*T}\,\mathbf{U}_d$$

$$P_d = E\,|\,s(t)|^2\,|\,\mathbf{W}^{*T}\,\mathbf{U}_d\,|^2 = E\,|\,s(t)|^2\,\mathbf{W}^{*T}\mathbf{U}_d\mathbf{U}_d^{*T}\mathbf{W}$$

$$= \mathbf{W}^{*T}E[\,s(t)\,\mathbf{U}_d\,s^*(t)\mathbf{U}_d^{*T}\,]\mathbf{W}$$

$$= \mathbf{W}^{*T}\,E[\mathbf{X}_d\,\mathbf{X}_d^{*T}\,]\mathbf{W}$$

$$= \mathbf{W}^{*T}\,\mathbf{\Phi}_d\,\mathbf{W}$$

LARGE ARRAY SIGNAL PROCESSING

THE "UNDESIRED" COMPONENTS:

$$\mathbf{\Phi}_u = \mathbf{\Phi}_I + \mathbf{\Phi}_n \equiv E[\mathbf{X}_I \mathbf{X}_I^{*T}] + E[\mathbf{X}_n \mathbf{X}_n^{*T}] = E[\mathbf{X}_u \mathbf{X}_u^{*T}]$$

$$P_u = P_I + P_n = \mathbf{W}^{*T} \mathbf{\Phi}_I \mathbf{W} + \mathbf{W}^{*T} \mathbf{\Phi}_n \mathbf{W} = \mathbf{W}^{*T} \mathbf{\Phi}_u \mathbf{W}$$

"SIGNAL TO INTERFERENCE PLUS NOISE RATIO"

$$SINR = \frac{P_d}{P_u} = \frac{\mathbf{W}^{*T} \mathbf{\Phi}_d \mathbf{W}}{\mathbf{W}^{*T} \mathbf{\Phi}_u \mathbf{W}} = E\,|\,s(t)\,|^2 \,\frac{|\,\mathbf{W}^{*T} \mathbf{U}_d\,|^2}{\mathbf{W}^{*T} \mathbf{\Phi}_u \mathbf{W}}$$

LARGE ARRAY SIGNAL PROCESSING

MAXIMIZATION OF ***SINR*** (known look direction)

EXAMPLE 1: $\mathbf{\Phi}_I = \mathbf{0}, \quad \mathbf{\Phi}_u = \mathbf{\Phi}_n = \sigma^2 \mathbf{I}, \quad \mathbf{\Phi}_u^{-1} = \frac{1}{\sigma^2}\mathbf{I}$

$$SNR = E\,|\,s(t)\,|^2 \, \frac{|\,\mathbf{W}^{*T}\mathbf{U}_d\,|^2}{\mathbf{W}^{*T}\mathbf{\Phi}_u\,\mathbf{W}} = E\,|\,s(t)\,|^2 \, \frac{\left|\sum_{i=1}^{N} w_i^* \, u_{d,i}\right|^2}{\sigma^2 \sum_{i=1}^{N} |\,w_i\,|^2} \qquad (1)$$

Schwarz inequality: $\left|\sum_{i=1}^{N} w_i^* \, u_{d,\,i}\right|^2 \le \sum_{i=1}^{N} |\,w_i\,|^2 \, \sum_{i=1}^{N} |\,u_{d,\,i}\,|^2$

with equality iff $w_i = c\,u_i$.

LARGE ARRAY SIGNAL PROCESSING

The optimum weights are $w_{opt,\,i} = u_i$. Letting $c = 1/\sigma^2$, the optimum weight vector can be expressed as

$$\mathbf{W}_{opt} = \mathbf{\Phi}_u^{-1}\mathbf{U}_d$$

and yields $$SNR = \frac{E\,|\,s(t)\,|^2}{\sigma^2}\sum_{i=1}^{N}|\,u_i\,|^2 = N\,\frac{P}{\sigma^2}$$

- Note that with **optimum weights**, the array output ***SNR*** is N times the elemental ***SNR***, as with a *PLL* Array
- Need to determine “source direction” separately
  - once source direction is determined, the optimum weights are also known

LARGE ARRAY SIGNAL PROCESSING

EXAMPLE 2: $\mathbf{\Phi}_I = \mathbf{0}, \quad \mathbf{\Phi}_u = \mathbf{\Phi}_n = diag[\sigma_1^2, \sigma_2^2, \cdots, \sigma_N^2]$

$$\mathbf{\Phi}_u = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_N^2 \end{bmatrix}; \quad \mathbf{\Phi}_u^{-1} = diag[\sigma_1^{-2}, \sigma_2^{-2}, \cdots, \sigma_N^{-2}]$$

$$SNR = E\,|\,s(t)\,|^2 \frac{|\,\mathbf{W}^{*T}\mathbf{U}_d\,|^2}{\mathbf{W}^{*T}\mathbf{\Phi}_u\,\mathbf{W}} = E\,|\,s(t)\,|^2 \frac{\left|\sum_{i=1}^{N} w_i^*\,u_{d,i}\right|^2}{\sum_{i=1}^{N} \sigma_i^2\,|\,w_i\,|^2} \qquad (2)$$

LARGE ARRAY SIGNAL PROCESSING

First rewrite inner product, then apply Schwarz inequality:

$$\left| \sum_{i=1}^{N} w_i^* \, u_{d,i} \right|^2 = \left| \sum_{i=1}^{N} \sigma_i w_i^* \frac{u_{d,i}}{\sigma_i} \right|^2 \le \sum_{i=1}^{N} \sigma_i^2 \, | w_i |^2 \sum_{i=1}^{N} \frac{| u_{d,i} |^2}{\sigma_i^2}$$

$$\frac{SINR}{E\,| s(t) |^2} = \frac{\left| \sum_{i=1}^{N} \sigma_i w_i^* \frac{u_{d,i}}{\sigma_i} \right|^2}{\sum_{i=1}^{N} \sigma_i^2 \, | w_i |^2} \le \frac{\sum_{i=1}^{N} \sigma_i^2 \, | w_i |^2 \sum_{i=1}^{N} \frac{| u_{d,i} |^2}{\sigma_i^2}}{\sum_{i=1}^{N} \sigma_i^2 \, | w_i |^2} = \sum_{i=1}^{N} \frac{| u_{d,i} |^2}{\sigma_i^2} \qquad (3)$$

with equality iff $\sigma_i w_i = c \dfrac{u_{d,i}}{\sigma_i}; \quad \rightarrow \quad w_{opt,\,i} = \dfrac{u_{d,i}}{\sigma_i^2}$

LARGE ARRAY SIGNAL PROCESSING

The optimum weight vector can again be expressed as

$$\boxed{\mathbf{W}_{opt} = \mathbf{\Phi}_u^{-1}\mathbf{U}_d} \qquad (4)$$

***SINR* of combined array output:**

From (3), the *SINR* of the array output with optimum weights is:

$$SINR = \frac{|\,\mathbf{W}_{opt}^{*T}\,\mathbf{U}_d\,|^2}{\mathbf{W}_{opt}^{*T}\,\mathbf{\Phi}_u\,\mathbf{W}_{opt}} = \sum_{i=1}^{N}\frac{E\,|\,s(t)\,|^2}{\sigma_i^2} = \sum_{i=1}^{N}\frac{P}{\sigma_i^2}$$

The maximum value of the ***SINR*** is achieved by the optimum weights $\mathbf{W}_{opt}$. When these weights are applied, the ***SINR*** of the output is equal to the sum of elemental ***SINR***-s.

LARGE ARRAY SIGNAL PROCESSING

EXAMPLE 3: the general case: $\boldsymbol{\Phi}_u = \boldsymbol{\Phi}_I + \boldsymbol{\Phi}_n$ **(Applebaum)**

The covariance matrix of the general undesired component is Hermitian, therefore it can be diagonalized by a unitary transformation. Let $\mathbf{A}$ be a unitary matrix, so that

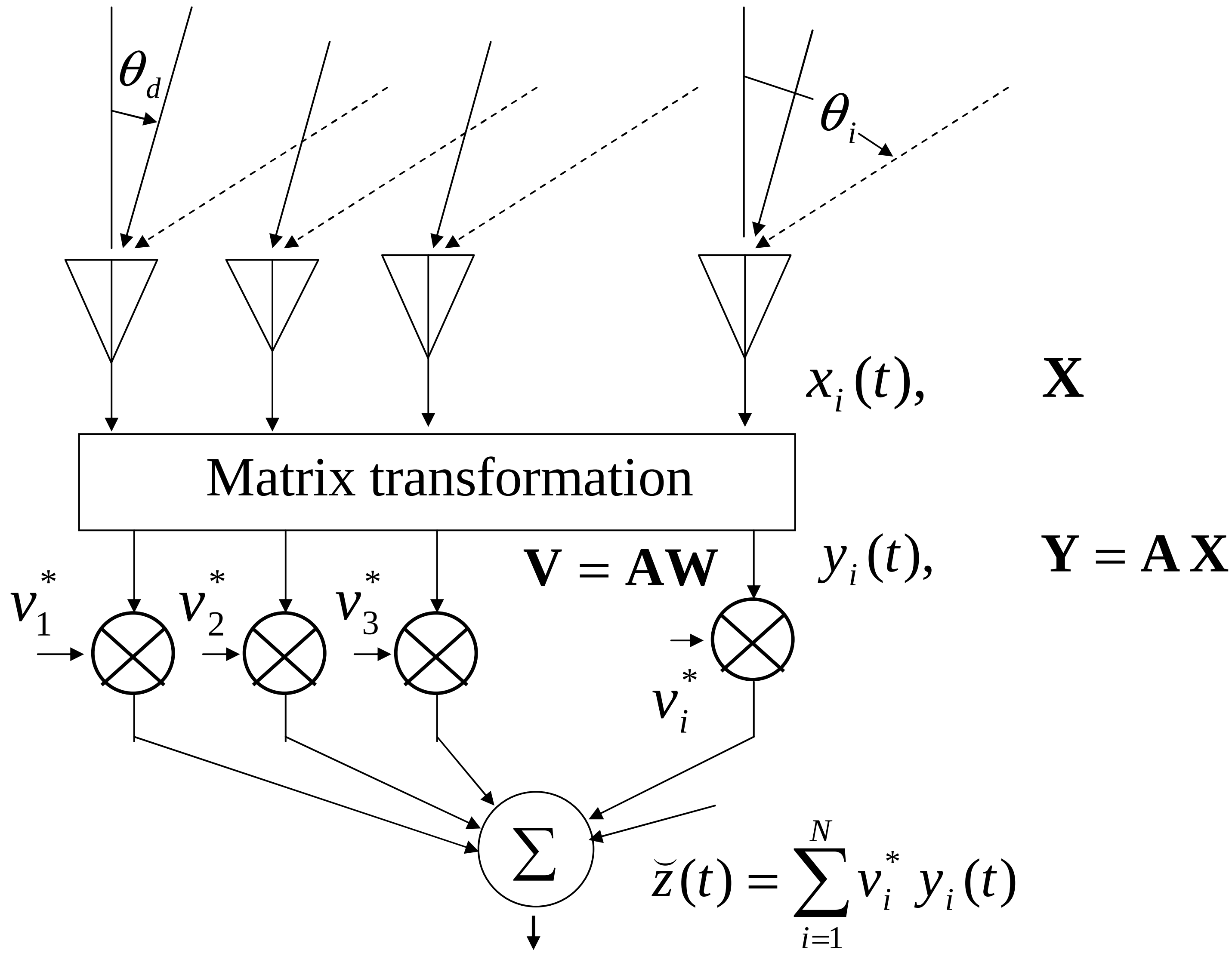
$$\mathbf{A}^{*T} = \mathbf{A}^{-1}; \qquad \mathbf{A}^{*T}\mathbf{A} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$$

and define the "transformed" vectors as

$$\mathbf{Y}_d = \mathbf{A}\,\mathbf{X}_d\,, \quad \mathbf{Y}_I = \mathbf{A}\,\mathbf{X}_I\,, \quad \mathbf{Y}_n = \mathbf{A}\,\mathbf{X}_n$$

These "transformed" vectors represent the original input vectors in a new, "rotated" coordinate system, and lead to a minor conceptual modification to the system block diagram.

LARGE ARRAY SIGNAL PROCESSING



LARGE ARRAY SIGNAL PROCESSING

Let the covariance matrix of the signals in the rotated coordinate system be designated by Ψ_u, defined as

$$\mathbf{\Psi}_u = E\left[\mathbf{Y}_u \mathbf{Y}_u^{*T}\right] = E\left[\mathbf{A}\mathbf{X}_u (\mathbf{A}\mathbf{X}_u)^{*T}\right] = E\left[\mathbf{A}\mathbf{X}_u \mathbf{X}_u^{*T}\mathbf{A}^{*T}\right]$$
$$= \mathbf{A}E\left[\mathbf{X}_u \mathbf{X}_u^{*T}\right]\mathbf{A}^{*T} = \mathbf{A}\mathbf{\Phi}_u\mathbf{A}^{*T}$$

If the columns of $\mathbf{A}$ correspond to the eigenvectors of $\mathbf{\Phi}_u$, then the transformation diagonalizes $\mathbf{\Phi}_u$, with the eigenvalues of $\mathbf{\Phi}_u$ occupying the diagonal:

$$\mathbf{\Psi}_u = \mathrm{diag}[\lambda_1, \lambda_2, \cdots, \lambda_N] = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_N \end{bmatrix}$$

LARGE ARRAY SIGNAL PROCESSING

Maximization of *SINR* for the general case:

The solution for the case of a diagonal covariance matrix has already been solved in Example 2: the optimum weights are proportional to the ratio of source-direction vector (in the rotated coordinates) to the total noise power. Defining the "rotated source-direction" as $\mathbf{Q} = \mathbf{A}\mathbf{U}_d$, the optimum weights can be obtained from equation (4) by inspection:

$$\mathbf{V}_{opt} = \mathbf{\Psi}_u^{-1}\mathbf{Q}_d$$

But $\mathbf{V} = \mathbf{A}\mathbf{W}$ implies that $\mathbf{W}_{opt} = \mathbf{A}^{-1}\mathbf{V}_{opt}$, so we can write

$$\begin{aligned}\mathbf{W}_{opt} &= \mathbf{A}^{-1}[\mathbf{\Psi}_u^{-1}\mathbf{A}\mathbf{U}_d] = [\mathbf{A}^{-1}\mathbf{\Psi}_u^{-1}\mathbf{A}]\mathbf{U}_d \\ &= [\mathbf{A}^{-1}\mathbf{\Psi}_u\mathbf{A}]^{-1}\mathbf{U}_d;\ \text{using}\,[\mathbf{ABC}]^{-1} = [\mathbf{C}^{-1}\mathbf{B}^{-1}\mathbf{A}^{-1}] \qquad (5)\end{aligned}$$

LARGE ARRAY SIGNAL PROCESSING

Since $\mathbf{\Psi}_u = \mathbf{A}\,\mathbf{\Phi}_u\,\mathbf{A}^{*T}$, it follows that

$$\mathbf{A}^{-1}\,\mathbf{\Psi}_u\mathbf{A} = \mathbf{A}^{-1}[\mathbf{A}\,\mathbf{\Phi}_u\,\mathbf{A}^{*T}]\mathbf{A} = \mathbf{\Phi}_u$$

Substituting into (5) yields the optimum weight vector that maximizes ***SINR*** for the general case:

$$\mathbf{W}_{opt} = [\mathbf{A}^{-1}\mathbf{\Psi}_u\mathbf{A}]^{-1}\mathbf{U}_d = \mathbf{\Phi}_u^{-1}\,\mathbf{U}_d$$

Since a constant scale factor, μ, applied to the weight vector does not change the ***SINR***, we can also write:

$$\boxed{\mathbf{W}_{opt} = \mu\,\mathbf{\Phi}_u^{-1}\,\mathbf{U}_d}$$

LARGE ARRAY SIGNAL PROCESSING

PROCESSORS FAMILIAR FROM THE LITERATURE:

1. **Conventional Beamformer:** $\mathbf{W}_{opt} = (\text{constant})\mathbf{U}_d$

2. **NAME (noise-alone matrix inverse):**

$$\mathbf{W}_{opt} = \mathbf{\Phi}_u^{-1}\mathbf{U}_d$$

3. **SPNAMI (signal-plus-noise matrix inverse):**

$$\mathbf{W}_{opt} = \mathbf{\Phi}^{-1}\mathbf{U}_d$$

Both NAME and SPNAMI achieve the same *SINR*

LARGE ARRAY SIGNAL PROCESSING

THE EQUIVALENCE OF USING $\mathbf{\Phi}_u^{-1}\,\mathbf{U}_d^*$ **OR** $\mathbf{\Phi}^{-1}\,\mathbf{U}_d^*$ **TO MAXIMIZE *SINR*** (Applebaum-Compton notation: $w_i\ x_i$)

Recall that $\mathbf{\Phi} = \mathbf{\Phi}_d + \mathbf{\Phi}_u = E\,|\,s(t)\,|^2\,\mathbf{U}_d^*\,\mathbf{U}_d^T + \mathbf{\Phi}_u$

The inverse of $\mathbf{\Phi}$ can be calculated with the help of the following **Matrix Inversion Lemma:** if **B** is a nonsingular $N \times N$ matrix, **Z** is an $N \times 1$ column vector, and β is a scalar, then the inverse of $\mathbf{Q} = \mathbf{B} - \beta\,\mathbf{Z}^*\mathbf{Z}^T$ is given by

$$\mathbf{Q}^{-1} = \mathbf{B}^{-1} - \alpha\,\mathbf{B}^{-1}\mathbf{Z}^*\mathbf{Z}^T\mathbf{B}^{-1}$$

where $\alpha^{-1} + \beta^{-1} = \mathbf{Z}^T\mathbf{B}^{-1}\mathbf{Z}^*$

LARGE ARRAY SIGNAL PROCESSING

Applying this lemma to $\mathbf{\Phi}$, we find its inverse as

$$\mathbf{\Phi}^{-1} = \mathbf{\Phi}_u^{-1} - \alpha \mathbf{\Phi}_u^{-1} \mathbf{U}_d^* \mathbf{U}_d^T \mathbf{\Phi}_u^{-1}$$

Evaluating α and substituting, after some algebra we get

$$\mathbf{\Phi}^{-1} \mathbf{U}_d^* = (\text{constant}) \ \mathbf{\Phi}_u^{-1} \mathbf{U}_d^*$$

Since multiplying the weight vector by a constant does not affect the ***SINR***, these two weight vectors produce identical ***SINR***.

LARGE ARRAY SIGNAL PROCESSING

Eigenvector Approach from linear algebra:

Recall that the ***SINR*** can be expressed as the ratio of two "quadratic forms":

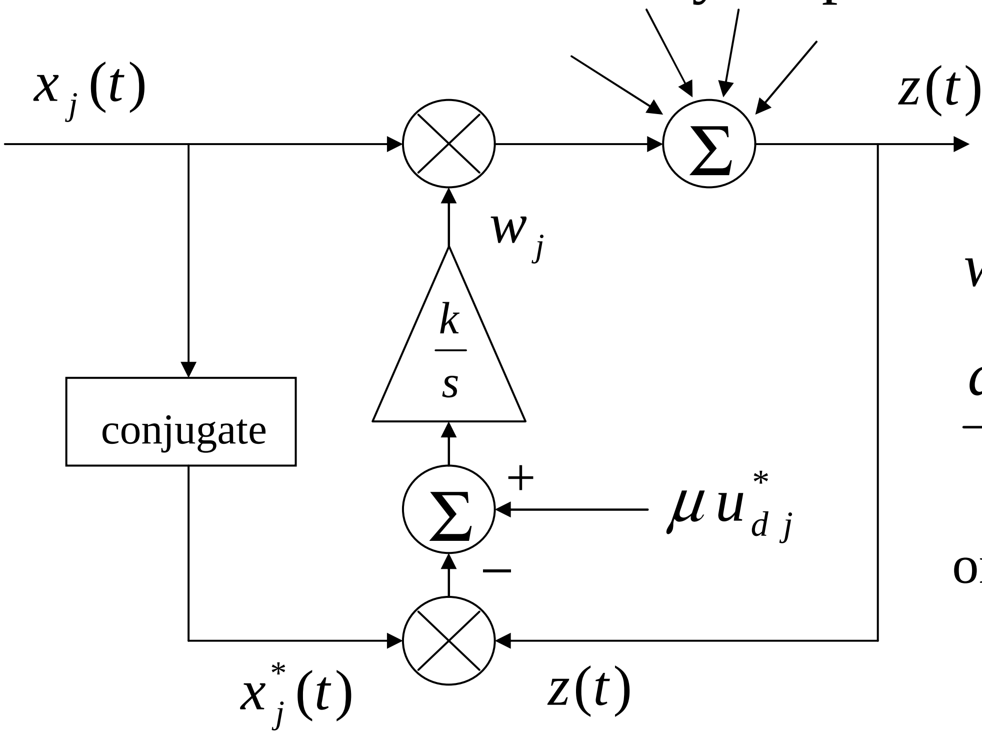
$$SINR = \frac{P_d}{P_u} = \frac{\mathbf{W}^{*T}\mathbf{\Phi}_d\,\mathbf{W}}{\mathbf{W}^{*T}\mathbf{\Phi}_u\,\mathbf{W}}$$

The ratio of two quadratic forms attains its maximum value when **W** is the eigenvector associated with the largest eigenvalue of $[\mathbf{\Phi}_u^{-1}\mathbf{\Phi}_d]$. This approach will be detailed and demonstrated in Part II.

LARGE ARRAY SIGNAL PROCESSING

CLOSED-LOOP ESTIMATION OF OPTIMUM WEIGHTS:

1. THE (MODIFIED) **APPLEBAUM** LOOP: consider a single branch of the array, with weights determined as a real-time correlation of the array output with each elemental signal



(AC notation)

$$w_j(t) = k\int_{-\infty}^{t}\left[\mu u^*_{dj} - x^*_j(\tau) z(\tau)\right] d\tau$$

$$\frac{dw_j(t)}{dt} = k\left[\mu u^*_{dj} - x^*_j(t) z(t)\right]$$

or, in vector form:

$$\frac{d\,\mathbf{W}}{dt} = k\left[\mu\,\mathbf{U}^*_d - \mathbf{X}^*\, z(t)\right]$$

Need the desired signal direction in advance.

LARGE ARRAY SIGNAL PROCESSING

From previous slide: $$\frac{d\,\mathbf{W}}{dt} = k\left[\mu\,\mathbf{U}_d^* - \mathbf{X}^*\,z(t)\right] \qquad (6)$$

Recalling that $z(t) = \mathbf{X}^T\mathbf{W}$ and substituting into (6), yields

$$\frac{d\,\mathbf{W}}{dt} = k\left[\mu\,\mathbf{U}_d^* - \mathbf{X}^*\mathbf{X}^T\mathbf{W}\right], \qquad \frac{d\,\mathbf{W}}{dt} + k\,\mathbf{X}^*\mathbf{X}^T\mathbf{W} = k\mu\,\mathbf{U}_d^*$$

Using the approximation $\hat{\mathbf{\Phi}} \equiv \mathbf{X}^*\mathbf{X}^T \rightarrow E\left(\mathbf{X}^*\mathbf{X}^T\right)$, we get

$$\frac{d\,\mathbf{W}}{dt} + k\,\hat{\mathbf{\Phi}}\mathbf{W} \cong k\mu\,\mathbf{U}_d^* \qquad (7)$$

In the steady-state $d\,\mathbf{W}/dt = 0$, yielding $\hat{\mathbf{\Phi}}\mathbf{W}_{ss} = \mu\,\mathbf{U}_d^*$, or

$$\boxed{\mathbf{W}_{ss} = \mu\,\hat{\mathbf{\Phi}}^{-1}\,\mathbf{U}_d^* \cong \mathbf{W}_{opt}}$$

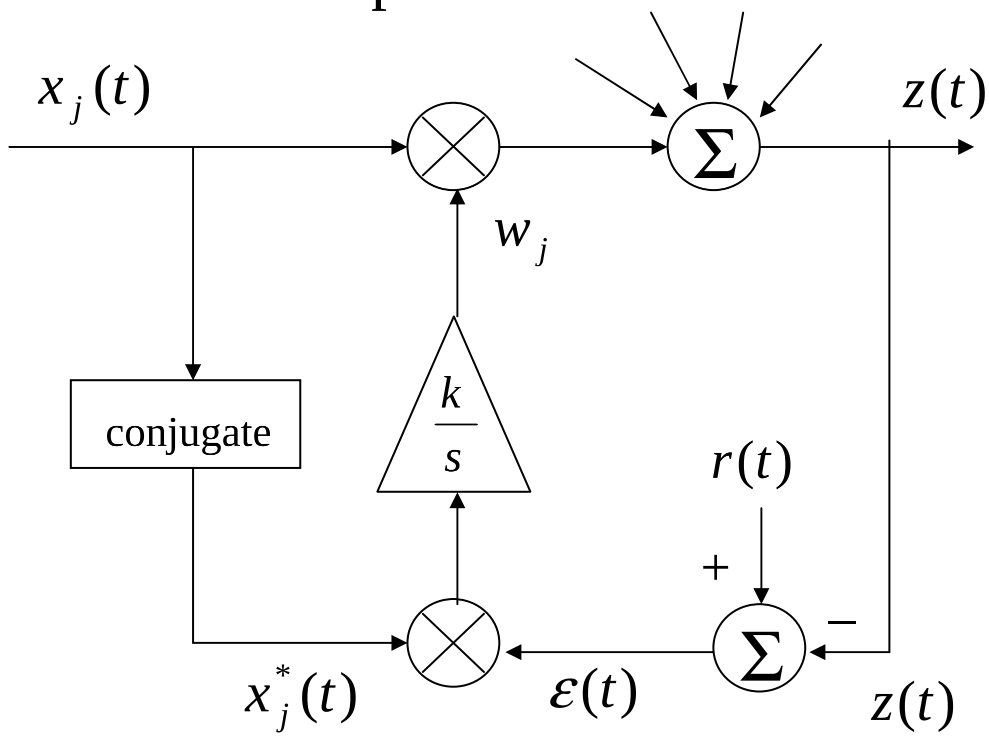
LARGE ARRAY SIGNAL PROCESSING

2. THE **LMS** LOOP: if the differential equation describing the **APPLEBAUM** loop is modified slightly, we obtain the differential equation for the LMS loop:

$$\frac{d\,\mathbf{W}}{dt} + k\,\hat{\mathbf{\Phi}}\mathbf{W} \cong k\,\hat{\mathbf{P}} \qquad (8)$$

$$\mathbf{P} = E[\mathbf{X}^* r(t)] \cong \mathbf{X}^* r(t) \equiv \hat{\mathbf{P}}$$

$$\varepsilon(t) = r(t) - z(t)$$



Needs reference signal, but not the desired signal direction.

LARGE ARRAY SIGNAL PROCESSING

RELATIONSHIP BETWEEN **APPLEBAUM** AND **LMS** ARRAYS

Weights of the **Applebaum** array satisfy the differential equation

$$\frac{d\,\mathbf{W}}{dt} + k\,\hat{\mathbf{\Phi}}\mathbf{W} \cong k\mu\;\mathbf{U}_d^*$$

Weights of the **LMS** array satisfy the differential equation

$$\frac{d\,\mathbf{W}}{dt} + k\,\hat{\mathbf{\Phi}}\mathbf{W} \cong k\,\hat{\mathbf{P}}. \qquad \text{But } \mathbf{P} = E[\mathbf{X}^* r(t)] = E[s^*(t) r(t)]\mathbf{U}_d^*$$

It follows that if $\mu\;\mathbf{U}_d^* = \hat{\mathbf{P}}$, the **LMS** and **Applebaum** arrays will perform identically, both maximizing ***SINR***. However, the LMS array does not need to know the source direction to track the signal.

LARGE ARRAY SIGNAL PROCESSING

3. THE DISCRETE VERSION OF THE **LMS** LOOP:

Starting with $\frac{d\,\mathbf{W}}{dt} + k\,\hat{\mathbf{\Phi}}\mathbf{W} \cong k\,\hat{\mathbf{P}}$ and substituting for the estimates, we get

$$\frac{d\,\mathbf{W}}{dt} \cong k[\hat{\mathbf{P}} - \hat{\mathbf{\Phi}}\mathbf{W}] = k[\,\mathbf{X}^{*} r(t) - \mathbf{X}^{*}\mathbf{X}^{T}\mathbf{W}]$$

$$= k\,\mathbf{X}^{*}[r(t) - z(t)] = k\,\mathbf{X}^{*}\varepsilon(t)$$

which means that each component satisfies

$$\frac{d\,w_i(t)}{dt} \cong k\,x_i^{*}(t)\,\varepsilon(t) \qquad (9)$$

LARGE ARRAY SIGNAL PROCESSING

Next, approximate the derivative with the difference

$$\frac{dw_i}{dt} \cong \frac{w_i(n+1) - w_i(n)}{\Delta t}$$

where $w_i(n)$ is a sample of the *i*-th weight at time $t_n = n\Delta t$.

Rewriting (9) in terms of the difference yields

$$w_i(n+1) - w_i(n) = \gamma x_i^*(n)\,\varepsilon(n); \quad \gamma \equiv k\Delta t$$

which can be put into a form known as the "**LMS algorithm**"

$$\boxed{w_i(n+1) = w_i(n) + \gamma x_i^*(n)\,\varepsilon(n)}$$

where $\varepsilon(n) = r(n) - z(n)$.

LARGE ARRAY SIGNAL PROCESSING

SOME PROPERTIES OF THE **LMS** ALGORITHM:

$$\boxed{w_i(n+1) = w_i(n) + \gamma x_i^*(n)\,\varepsilon(n)}$$

- Needs a reference signal (correlated with the received signal)
- Does not need to know the source direction
- Complexity per update: order *N* (for *N* antennas)
- Magnitude of updates diminish as output signal “approaches” the reference signal (error approaches zero)

LARGE ARRAY SIGNAL PROCESSING

THE "CONSTANT MODULUS ALGORITHM" (**CMA**)

Recall the form of the discrete **LMS** algorithm derived before:

$$w_i(n+1) = w_i(n) + \gamma x_i^*(n)\,\varepsilon(n)$$

If we let $\varepsilon(n) = \left(|\,s(n)\,|^2 - s_0^2\right) s(n)$, the resulting algorithm is known as the **CMA**:

$$\boxed{w_i(n+1) = w_i(n) + \gamma\left(|\,s(n)\,|^2 - s_0^2\right) x_i^*(n)\,s(n)}$$

LARGE ARRAY SIGNAL PROCESSING

SOME PROPERTIES OF THE **CMA**:

$$w_i(n+1) = w_i(n) + \gamma\left(|\,s(n)\,|^2 - s_0^2\right)x_i^*(n)\,s(n)$$

- The **CMA** does **not need** either a **source direction** or a **reference signal**, only the “target” power of the desired source
- Any change in source power is attributed to interference, which the CMA attempts to cancel
- Order *N* complexity (multiplies per update)

LARGE ARRAY SIGNAL PROCESSING

SUMMARY OF REPRESENTATIVE "ORDER N" ALGORITHMS FOR DSN APPLICATIONS

- **LMS ALGORITHM:**
  - Needs a **reference signal** (filtered residual carrier, or other correlated reference derived via signal processing)
  - Adaptively maximizes ***SINR*** (nulls interference)
- **CMA:**
  - Needs estimate of desired **signal power** only
  - Adaptively maximizes ***SINR*** (nulls interference)

OPEN ISSUES: convergence rate under "realistic" DSN spacecraft tracking conditions

LARGE ARRAY SIGNAL PROCESSING

PART II: REAL-TIME DEMONSTRATIONS

LMS ALGORITHM (F. Pollara):

- Real-time convergence from initial weight vector to optimum, with and without noise
- Demonstration of gradient descent (min. of error surface)

NORMALIZED CMA (M. Srinivasan):

- New algorithm, needs estimate of **average signal power**
- Can phase up array with “noise-like” signals from quasars

EFFICIENT EIGENVECTOR ALGORTHM (C. Lee):

- Based on matrix theory result on maximization of ratio of two quadratic forms
- Efficient, iterative implementation