DECENTRALIZED ESTIMATION IN COMMUNICATION NETWORKS WITH DELAY

by

Ali Özbek
S. B., Massachusetts Institute of Technology
(1980)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREES OF
MASTER OF SCIENCE
and
ELECTRICAL ENGINEER
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
January 1984

Signature of Author: ....
Department of Electrical Engineering and
Computer Science, January 20, 1984

Certified by: ....
Robert R. Tenney
Thesis Supervisor

Accepted by: ....
Arthur C. Smith
Chairman, Departmental Graduate Committee

DECENTRALIZED ESTIMATION IN COMMUNICATION NETWORKS WITH DELAY

by

Ali Özbek

Submitted to the Department of Electrical Engineering and Computer Science on January 20, 1984 in partial fulfillment of the requirements for the Degrees of Master of Science and Electrical Engineer.

ABSTRACT

Previous work on decentralized estimation has assumed unlimited capacity on the communication channels between the sensors and the estimators. In practice, this is not possible; increased traffic rate leads to network congestion and decreased performance. This thesis develops optimum reporting schedules for sensors, routing schemes for messages and data fusion algorithms for intermediate nodes.

Thesis Supervisor: Robert R. Tenney

Title: Assistant Professor of Electrical Engineering

ACKNOWLEDGEMENTS

I would like to thank my advisor, Bob Tenney, for his constant guidance and encouragement during the course of the research. This thesis was completed only because of his personal understanding and patience.

I am very grateful to Delphine Radcliffe for her wonderful work and long hours spent typing this thesis.

This research was conducted at the M.I.T. Laboratory for Information and Decision Systems, with partial support provided by the Office of Naval Research, under contract #N00014-77-C-0532.

DEDICATED TO MY PARENTS

Birgül and Hâdi Özbek

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

LIST OF FIGURES

GLOSSARY OF SYMBOLS

The meanings of the more important symbols are summarized here, together with references to the definitions of the others. The symbols are listed with respect to the chapters in which they appear.

Chapter 2

$\{A, B, C_i\}$	Realization of the linear dynamical system with state x(t), input w(t) and output (observation) y_i.
a	A constant, see (2.3).
c_i	(For i = 1, 2, 3) constants, see (2.41), (2.43).
$d_{ij}(f_{ij})$	Delay on the link (i, j) when the intermessage frequency is f_{ij}.
E	Expected value of.
$\{F, G, H\}$	Realization with state variable z(t) and related to the realization $\{A, B, C\}$ by the similarity transformation M.
f_{ij}	Intermessage frequency on link (i, j).
$\mathcal{H}(s)$	Input-output transfer function.
M	Invertible matrix representing the linear transformation between x(t) and z(t).
N	Number of sensor nodes in the network.
O(i)	Set of nodes #j for which the links (i, j) exist from node #i.
P(t)	Error covariance matrix associated with the optimal (minimum mean-square error) estimation of x(t).
$P_i(t)$	Local error covariance matrix at node #i.
$\bar{P}_i$	Steady-state value of $P_i(t)$.

$p(t)$	Mean-square estimation error of $x(t)$.
p_i	(For $i = 1, 2, 3$) Constants defined by (2.5), (2.9) and (2.10).
Q	Intensity of $w(t)$.
q	Intensity of $w(t)$, for the scalar case.
R_i	Intensity of $v_i(t)$.
r	Intensity of $v(t)$, for the scalar case.
r_e	Constant, defined by (2.16).
$S(t)$	Error covariance matrix associated with the optimal prediction of $x(t)$.
S_o	$S(t = 0)$.
s	Independent time variable.
$s_{ij}(t)$	$(i, j)^{th}$ entry of $S(t)$.
s_{ijo}	$(i, j)^{th}$ entry of S_o.
t	Independent time variable.
$V(t)$	Error covariance matrix associated with the optimal prediction of $z(t)$.
V_o	$V(t = 0)$.
$v_i(t)$	Observation noise vector at sensor node #i.
$w(t)$	Process noise vector.
$x(t)$	State variable vector of the linear dynamical system under observation.
$\hat{x}(t)$	Optimal (minimum mean-square error) estimate of $x(t)$.

$y_i(t)$	Measurement vector at sensor node #i.
$z(t)$	State variable vector related to x(t) by the linear transformation M.
α_i	Constants, see (2.30).
β	Constant, see (2.8).
β_e	Constant, see (2.17).
β_i	Constants, see (2.30).
γ_{ij}	Constants, see (2.32).
θ_i	Constants, see (2.30).
$\xi(t)$	Scalar time function, see (2.41).
ξ_o	$\xi(t = 0)$.
σ_{ij}	Constants, see (2.39).
(i, j)	Communications link directed from node #i to node #j.
#i	Sensor node #i.

<u>Chapter 3</u>

$\{A, B, C_i\}$	Same as in Chapter 2.
E	Expected value of.
$g_i(t)$	(For i, j = 1, 2) A correction term similar to $r_i(t)$, but one which takes into consideration that the observations of node #j $\neq$ i were not available since the arrival of the last message from node #j.
$h_i(t)$	A correction term similar to $r_i(t)$, but one which takes into consideration that the observations of node #i were not available since the arrival of the last message from node #i.

$P(t)$	Global error covariance matrix associated with $\hat{x}(t)$.
$P_i(t)$	Local error covariance matrix associated with $\hat{x}_i(t)$.
R_i	Same as in Chapter 2.
$r_i(t)$	A term needed to correct correlation of $\hat{x}_1(t)$ and $\hat{x}_2(t)$ due to $\hat{x}(0)$ and $w(t)$, whose statistical specifications are common knowledge to both sensor nodes.
Q	Same as in Chapter 2.
$S_i(t)$	Global error covariance matrix associated with $\tilde{x}_i(t)$.
s	Independent time variable.
t	Independent time variable.
t^j	Time instant the j^{th} message is sent from both sensor nodes to the destination when reporting times are synchronized.
t_i^j	Time instant the j^{th} message is sent from node #i to the destination when reporting times are not synchronized.
t_i^{j*}	See Fig. 3.2.
t_i^{j**}	See Fig. 3.2.
$v_i(t)$	Same as in Chapter 2.
$w(t)$	Same as in Chapter 2.
$x(t)$	Same as in Chapter 2.
$\hat{x}(t)$	Global optimal estimate of $x(t)$.
$\hat{x}_i(t)$	Local optimal estimate of $x(t)$ at node #i.
$\tilde{x}_i(t)$	(For $i, j = 1, 2$) Global optimal estimate of $x(t)$, but one which takes into consideration that the observations of node $\#j \neq i$ were not available since the arrival from node #j.

$\overline{x}_i(t)$	Local optimal estimate of x(t) at node #i, but one which takes into consideration that the observations of node #i were not available since the arrival of the last message from node #i.
$y_i(t)$	Same as in Chapter 2.
$Z_i(t)$	Local error covariance matrix associated with $\overline{x}_i(t)$.

Chapter 4

$\{A, B, C_i\}$	Same as in Chapter 2.
A, B, C	(In Sections 4.4.2 and 4.4.3) Sets of paths j from the sensor node to the destination node, with associated average intermessage periods T^j and corresponding delays d^j.
A^*	Set containing the same paths j as A, but with $T^j = T^{j^*}$.
a'	Constant, see (4.6).
a^i	The value of the argument of Φ_m for p^i. See (4.13).
a_{wc}	The value of the argument of Φ_m for p_{wc}. See (4.26).
B_V	$\{i \in B : i \in V\}$
$\overline{B}_V$	$\{i \in B : i \notin V\}$
b'	Constant, see (4.6).
b^i	The value of the second argument of η_m for p^i. See (4.13).
b_{wc}	The value of the second argument of η_m for p_{wc}. See (4.26).
C^j	Channel capacity of path j.
C_{ij}	Channel capacity of link (i, j).

$\underline{C}^{-1}$	Vector of C^{-1}'s.
$C^{i^{-1}}$	i^{th} element of $\underline{C}^{-1}$
c	(In Section 4.2) Unit of time: $c \triangleq T_1/k_1 = T_2/k_2$, where k_1 and k_2 are mutually prime integers.
D	(In Section 4.4) Effective (information) delay from the sensor node for the optimum adjustment of the departure times on the paths j to the destination, given constant values of T^j and d^j.
D^*	(In Section 4.4) Optimal value of D, minimized with respect to average departure frequencies and departure times on the paths to the destination.
D(A)	Term defined by (4.58).
$\bar{D}(A)$	(In Section 4.4) Effective (information) delay from the sensor node for the optimum adjustment of departure times on the paths which are elements of set A.
$\underline{D}(A - \{k\})$	Term defined by (4.104).
D_i	(In Section 4.4.4) Effective (information) delay from node #i to the destination at each local minimum of p_{wc}, reflecting optimal adjustment of the departure times on the links originating from node #i, for given average departure frequencies and corresponding communication delays.
$d_i(T_i)$	(In Section 4.2) Delay on the link from node #i to the destination node when the intermessage period is T_i.
$d_{ij}(T_{ij})$	Delay on link (i, j) when the intermessage period is T_{ij}.
$d^j(T^j)$	(In Section 4.4) Delay on path j from the sensor node to the destination when the reciprocal of the average intermessage frequency is T^j.

$\bar{d}_B$	(In Section 4.4) Weighted average of delays on path j with $j \in B$.
$\bar{d}_V$	(In Section 4.4) Effective communication delay from the sensor node to the destination.
f^j	Average intermessage frequency on path j.
f^{j^*}	Optimum value of f^j.
G	A set of sets of paths, defined by (4.83).
$H^k(T^1, \cdots, T^{k-1}, T^{k+1}, \cdots, T^L)$:	A hyperplane in $\mathbb{R}^{L-1}$, defined by (4.94).
$I_k^\#$	Set of indices, defined by (4.118), for the k^{th} iteration of the algorithm of Proposition 4.7.
Interval A	$[0, \Delta_{wc})$.
Interval B	$[\Delta_{wc}, c)$.
L	(In Section 4.4) Number of paths from the sensor node to the destination.
m_k	The first non-negative integer m such that (4.133) is satisfied, for the k^{th} iteration of the algorithm of Proposition 4.7.
O(i)	Same as in Chapter 2.
P(t)	Same as in Chapter 2.
$\bar{P}$	Steady-state value of P(t).
$P_m(t)$	(In Section 4.3) Error covariance matrix associated with the optimal estimation of x(t) with nodes #m, #m+1, $\cdots$, #N reporting.
p(t)	Same as in Chapter 2.

p^i	i^{th} peak in a certain period of p(t).
p'	A certain peak of p(t) given by (4.5).
$p^{\ell Ai}$	(For i = 1,2) Candidate peaks for the highest peak for $\Delta \in$ Interval A.
$p^{\ell Bi}$	(For i = 1,2) Candidate peaks for the highest peak for $\Delta \in$ Interval B.
p_{opt}	(In Section 4.2) Minimum value of the highest peak of p(t) with respect to Δ for given T_1, d_1, T_2 and d_2 values.
p_{wc}	Maximum value in time of p(t) for the worst possible timing relationship between the reporting times of the nodes.
Q	Same as in Chapter 2.
R_i	Same as in Chapter 2.
$\mathbb{R}^m$	m-dimensional Euclidean space.
S(t)	(In Section 4.3) Same as in Chapter 2.
$S_m(t_1, t_2)$	(In Section 4.2) Error covariance matrix resulting from prediction during the last t_2 seconds, after getting observations from node #m for t_1 seconds.
s_k^i	Scalar sequences defined by (4.120).
$\mathcal{T}$	(In Section 4.3) Spanning tree linking the sensor nodes to the destination node.
T_i	(In Section 4.2) The intermessage period from sensor node #i to the destination node.
T_i^*	$\underset{T_i > 0}{\operatorname{argmin}}\ T_i + d_i(T_i)$.

T_{ij}	Intermessage period on link (i, j).
T^*_{ij}	$\underset{T_{ij}>0}{\text{argmin}}\ T_{ij} + d_{ij}(T_{ij})$.
T^j	(In Section 4.4) The reciprocal of the average intermessage frequency on path j.
T^{j^*}	The optimal values of T^j which minimize D.
$\overline{T}_V$	(In Section 4.4) Effective period, or the inverse of the sum of the average departure frequencies on the paths from the sensor node to the destination node.
$\underline{T}$	Vector of T_{ij}'s.
$\underline{\tilde{T}}_k$	Vector of coordinates T^i_k with $i \in I^\#_k$, for the k^{th} iteration of the algorithm of Proposition 4.7.
$\underline{\overline{T}}_k$	Vector of coordinates T^i_k with $i \notin I^\#_k$, for the k^{th} iteration of the algorithm of Proposition 4.7.
T^i_k	i^{th} element of $\underline{T}$, for the k^{th} iteration of the algorithm of Proposition 4.7.
t	Independent time variable.
t_{wc}	Defined by $p_{wc} = p(t_{wc})$.
t^i_{dn}	Time of departure of the i^{th} message from sensor node #n.
t^i_{an}	Time of arrival of the i^{th} message from sensor node #n.
$\overline{t}^A_{ai}$	(For i = 1,2) Time duration defined by (4.37), (4.38).
$\overline{t}^B_{ai}$	(For i = 1,2) Defined similarly to $\overline{t}^A_{ai}$ for a particular $\Delta_B \in$ Interval B.
t^j_d	(In Section 4.4) Time of departure of the j^{th} message from the sensor node.

t_a^j	(In Section 4.4) Time of arrival of the j^{th} message at the destination node.
U_{ij}	Upper limit on T_{ij}, imposed for optimization purposes.
$\underline{U}$	Vector of U_{ij}'s.
U^i	i^{th} element of $\underline{U}$.
u'	Time instant defined by (4.7).
V	(In Section 4.4) Largest set of "useful paths' each of which contribute to the minimization of D.
$\bar{V}$	$\{j \in A : j \notin V\}$ in the proof of Proposition 4.5.
v^i	(In Section 4.4) i^{th} element of set V.
$v_i(t)$	Same as in Chapter 2.
W^n	(In Section 4.4) Sets of paths defined by (4.69), (4.70).
w(t)	Same as in Chapter 2.
x(t)	Same as in Chapter 2.
Y	Set of paths defined by (4.59).
$y_i(t)$	Same as in Chapter 2.
Z	A subset of $\{1, 2, \cdots, L\}$.
$\underline{Z}_m$	The sequence defined by (4.124).
α_k	β^{m_k}
β	A constant out of the interval (0, 1).
β_k	A constant out of the interval (0, 1), for the k^{th} iteration of the algorithm of Proposition 4.7.
β_m	Scalar sequences defined by (4.130).

γ_m	Scalar sequences defined by (4.129).
Δ	(Section 4.2) Phase difference between the reporting sequences of the two sensor nodes.
Δ^*	(Section 4.2) Value of Δ which minimizes the highest peak of p(t) for T_1^*, d_1^*, T_2^*, d_2^*.
Δ_{wc}	(Section 4.2) Value of Δ which maximizes the highest peak of p(t) for given T_1, d_1, T_2, d_2.
Δ_{sup}	Upper bound for Δ, given by (4.22).
Δ_A	A constant out of Interval A.
Δ_B	A constant out of Interval B.
$\delta(j)$	(In Section 4.2) Time difference between the departure of message #j from node #2 and the departure of the most recent message from node #1.
ε	A positive constant scalar.
ε_k^i	A scalar defined by (4.119).
ζ	A constant out of the interval $(0, \frac{1}{2})$.
$\eta_m(t_1, t_2)$	Trace $S_m(t_1, t_2)$.
Λ_k	A positive definite symmetric matrix, for the k^{th} iteration of the algorithm of Proposition 4.7.
$\lambda(t)$	(Section 4.3) Trace S(t).
μ^i	A positive constant scalar.
μ_k^i	Scalar sequences satisfying (4.121).
$\underline{\nu}_k$	Search direction for the k^{th} iteration of the algorithm of Proposition 4.7.
$\underline{\tilde{\nu}}_k$	Vector of coordinates ν_k^i with $i \in I_k^\#$.

$\underline{\bar{\nu}}_k$	Vector of coordinates ν_k^i with $i \notin I_k^{\#}$.
$\pi(T^k;\ T^1, T^2, \cdots, T^{k-1}, T^{k+1}, \cdots, T^L)$:	(L-1) dimensional curve defined by (4.93).
$\underline{\rho}_m$	Residual sequence defined by (4.128).
$\underline{\tilde{\sigma}}_k$	Vector with coordinates $\partial p_{wc}(\underline{T}_k)/\partial T^i$, with $i \in I_k^{\#}$.
$\underline{\bar{\sigma}}_k$	Vector with coordinates $\partial p_{wc}(\underline{T}_k)/\partial T^i$, with $i \notin I_k^{\#}$.
Φ	The empty set.
$\Phi_m(t)$	(In Section 4.2) trace $P_m(t)$.
$\psi_m(t)$	(In Section 4.3) trace $P_m(t)$.
$\tilde{\Omega}_k$	A diagonal positive definite matrix with elements $\partial^2 p_{wc}(\underline{T}_k)/(\partial T^i)^2$ along the diagonal.
$\bar{\Omega}_k$	Hessian matrix of p_{wc} with respect to the coordinates T^i, with $i \notin I_k^{\#}$.
$\underline{\omega}_m$	The conjugate direction sequence defined by (4.127).
(i, j)	Same as in Chapter 2.
#i	Same as in Chapter 2.
$1, 2, \cdots, L$	(In Section 4.4) Paths from the sensor node to the destination node.
$\bar{1}, \bar{2}, \cdots, \bar{M}$	(In Section 4.4) Elements of set $\bar{V}$.
$1^*, 2^*, \cdots, L^*$	(In Section 4.4) Paths from the sensor node to the destination node with $T^j = T^{j*}$, $d^j = d^{j*}$.

Chapter 5

$\{A, B, C\}$	Same as in Chapter 2.

D_i	(In Section 5.3) (For i = 1,2) Aggregate delay on link (i, 3).
d_i	(In Section 5.3) (For i = 1,2) Communication delay on link (i, 3).
$\tilde{d}$	(In Section 5.2) Random variable denoting the delay incurred by a message sent from the sensor node to the destination node.
$\tilde{d}_k$	(In Section 5.2) Random variable denoting the delay incurred by the message sent at $t = t_o + kT$.
E	Expected value of.
k	(In Section 5.2) Scalar factor defined by (5.8).
k_{ij}	Constants, see (5.12) and (5.13).
$m_d(T)$	Expected value of $\tilde{d}$ as a function of T.
P(t)	Error covariance matrix associated with the optimal prediction of x(t).
p(t)	Same as in Chapter 2.
p_n	Prob ("n-tuple" at t_o).
p^*	(In Section 5.3) Minimum mean-square error.
Q	Same as in Chapter 2.
q	Same as in Chapter 2.
R	Same as in Chapter 2.
r_i	Same as in Chapter 2.
r_{eq}	Constant defined by (5.14).
S	Steady-state error covariance matrix associated with the optimal estimation of x(t).

T	(In Section 5.2) Intermessage period from the sensor node to the destination node.
T_i	(In Section 5.3) (For i = 1,2) Intermessage period from sensor node #i to the destination node.
$v_i(t)$	Same as in Chapter 2.
w(t)	Same as in Chapter 2.
x(t)	Same as in Chapter 2.
$y_i(t)$	Same as in Chapter 2.
α_2	Constant, defined by (5.14).
β_2	Constant, defined by (5.14).
$\delta(t)$	(In Section 5.2) Random variable for the delay of information (message from other node or observations) from the time it arrives at an intermediate node to the time it arrives at the destination node.
(i, j)	Same as in Chapter 2.

$\forall$	For all.
$\exists$	There exist(s).
$\cup$	Union.
$\cap$	Intersection.
$\in$	Element of.
$\notin$	Not an element of.
$\subset$	Subset of.

CHAPTER 1

INTRODUCTION

Recently there has been a significant amount of work on the subject of decentralized estimation. The various approaches can be divided into two classes. The first class consists of methods which use the distributed structure of the problem in such a way as to achieve an overall estimator whose error corresponds to that of a fully centralized estimator [1], [2]. The second class of approaches consists of utilizing a fixed structure to achieve the best performance possible with this restricted structure [3], [4].

The proposed thesis research can be categorized into the second class of problems. It incorporates time delay constraints on information flow, a characteristic which was not addressed in the above work. While the research does not intend to extend the above work with added delay constraints, it aims to address certain decentralized estimation problems with an underlying delay formulation. The structure that is considered here is a network of processors which also have sensors for taking measurements. Various types of delays, such as queueing, processing and propagation between two nodes, are aggregated into a formulation which is a function of the traffic rate between the nodes.

In the network considered, state estimates are desired at a destination node. With no delays, getting observations from all sensors as often as possible would give the best performance, since the observation noise processes of the sensors are assumed to be independent. However, for a formulation where the delays are monotonically increasing functions of message traffic, sending observations at a high rate would cause large delays. Since there is statistical uncertainty in the evolution of the system under observations, delays cause a degration of estimation performance. Therefore, there seems to be an optimum rate of information sent from the sensors to the destination.

Also taking into account that some sensors provide more information about the states observed (with respect to a measure like the signal to noise ratio), regulating the amount of information sent from each node is one of several ancillary issues. Routing to minimize delays directly and data compression (by combining observations or local estimates) to reduce traffic, thus minimizing delay indirectly, are the other issues addressed. Problem formulations of routing and flow control in computer communication networks assume given demand statistics for each origin-destination pair [5]. However, in this research, the input rate at each node will be a control variable also. Thus this work can also be viewed as the development of a flow control scheme for one particular class of network problems, namely decentralized linear estimation.

CHAPTER 2

MODELS AND OBJECTIVE FUNCTIONS

2.1 Introduction

In this chapter we will present the components of the general problem formulation. This will include the specification of the network structure, the class of systems under observation, the observation model for the sensor nodes, the content of messages sent between the nodes, the models for the delays incurred by these messages on the communication links and, finally, the type of objective functions considered.

2.2 The Network Structure

The network is composed of N sensor nodes and a destination node, where the state estimate of the observed system is desired. Each node #i is directly connected to a set of neighboring nodes, O(i), through directed links (i, j), $j \in O(i)$. The links have distortionless communication channels with finite capacity.

2.3 The System and the Observations

The system under observation is modelled as a linear dynamical time-invariant stochastic system:

$$dx(t) = Ax(t)dt + Bdw(t). \tag{2.1}$$

Node #i makes observations:

$$dy_i(t) = C_i x(t)dt + dv_i(t). \tag{2.2}$$

Initial state $x(t_o)$ is uncorrelated with w and v_i. Also, w, v_1, v_2, $\cdots$, v_N are Brownian motion processes which are all uncorrelated. These assumptions have been adopted for the sake of simplicity; extensions are possible for the cases of correlated and colored noise inputs [6]. Wiener process w(t) has intensity Q, and $v_i(t)$ have intensity R_i.

The system and observation models are assumed to be time-invariant. (Alternatively, we might assume that these change sufficiently slowly so as not to interfere with the convergence of optimization algorithms discussed in this thesis.)

We also assume that $Q > 0$, $R_i > 0$, $i = 1, \cdots, N$; and that the pair $\{A, B\}$ is stabilizable and the pairs $\{A, C_i\}$ are detectable for $i = 1, \cdots, N$.

For the purposes of this thesis, we need to impose some further restrictions on the dynamical and statistical structure of the system and observation models. Before being more specific, we present some facts.

Consider the scalar linear stochastic dynamical system:

$$dx(t) = ax(t)dt + dw(t), \tag{2.3}$$

with the scalar observations

$$dy(t) = x(t)dt + dv(t). \tag{2.4}$$

w(t) and v(t) are independent zero-mean scalar Brownian motion processes, with intensities q and r, respectively, which are strictly positive. Also,

$$E[x(0)]^2 = p_o. \tag{2.5}$$

If the observations are processed by a filter which minimizes the mean-square estimation error at all times (e.g. a Kalman filter), then p(t), the mean-square estimation error at time t is given by the Riccati differential equation:

$$\dot{p}(t) = 2\,ap(t) + q - (1/r)\,p^2(t), \quad p(0) = p_o. \tag{2.6}$$

Since this equation is scalar, it can readily be solved, yielding [7]:

$$p(t) = p_1 - \frac{p_1 + p_2}{1 + [(p_o + p_2)/(p_1 - p_o)]e^{2\beta t}}, \tag{2.7}$$

where

$$\beta = \sqrt{a^2 + (q/r)}\ , \tag{2.8}$$

$$p_1 = r(\beta + a)\ , \tag{2.9}$$

$$p_2 = r(\beta - a)\ . \tag{2.10}$$

Note that we have $p(t) \to p_1$ as $t \to \infty$.

Lemma 2.1: If $p_1 > p_o$, then $\dot{p}(t) > 0$ for all $t > 0$; i.e. the mean-square estimation error is a monotonically increasing function of time.

Proof: From (2.7),

$$\dot{p}(t) = 2\beta \frac{p_1 + p_2}{\left\{1 + [(p_o + p_2)/(p_1 - p_o)]e^{2\beta t}\right\}^2} \cdot \left(\frac{p_o + p_2}{p_1 - p_o}\right)e^{2\beta t} \tag{2.11}$$

Since all terms are strictly positive, $\dot{p}(t) > 0$ for all $t > 0$.

Q.E.D.

Let us briefly describe an example situation where the above result is relevant. Assume that there are two sensors making observations:

$$dy_1(t) = x(t)dt + dv_1(t) \tag{2.12}$$

$$dy_2(t) = x(t)dt + dv_2(t) \tag{2.13}$$

of the system (2.3).

v_1 and v_2 are independent zero-mean Brownian motion processes with intensities r_1 and r_2, respectively. Assume that the sensors have been making observations for a long time, so that the optimal filter processing them is in the steady state. Then the steady-state mean-square estimation error, p_o, is given by the algebraic Riccati equation [7]:

$$0 = 2ap_o + q - \left(\frac{1}{r_1} + \frac{1}{r_2}\right)p_o^2 , \qquad (2.14)$$

with the solution

$$p_o = r_e(\beta_e + a) , \qquad (2.15)$$

where

$$r_e = \left(\frac{1}{r_1} + \frac{1}{r_2}\right)^{-1} , \qquad (2.16)$$

$$\beta_e = \sqrt{a^2 + (q/r_e)} . \qquad (2.17)$$

At time t = 0, assume that one of the sensors, say the second one, stops making observations. In this case the propagation in time of the mean-square error, p(t), is given by (2.7) - (2.10), with r replaced by r_1. As $t \to \infty$, p(t) approaches the steady-state value of p_1, given by (2.9). It is easy to show that $p_1 > p_o$. Now Lemma 2.1 indicates that the transition of p(t) from p_o to p_1 is monotonic.

Now consider the multidimensional linear stochastic dynamical system:

$$dz(t) = Fz(t)dt + Gdw(t) \qquad (2.18)$$

w(t) is a zero-mean Brownian motion process with intensity Q, which is positive definite. Also,

$$E[z(0)]^2 = V_o , \tag{2.19}$$

where V_o satisfies the algebraic matrix Riccati equation:

$$0 = FV_o + V_oF' + GQG' - V_oH' R^{-1} HV_o . \tag{2.20}$$

Hence we assume that there is an initial estimate of the state z at time t = 0. There are no observations for t > 0, so for optimal prediction which minimizes the mean-square prediction error at all times, the covariance matrix V(t) of the prediction error at time t is given by the linear matrix differential equation [7]:

$$\dot{V}(t) = FV(t) + V(t)F' + GQG', \quad V(0) = V_o. \tag{2.21}$$

Lemma 2.2: Unless matrix F is defective,* there exists an invertible linear transformation:

$$x(t) = Mz(t) \tag{2.22}$$

such that the error covariance matrix S(t) associated with the optimal prediction of x(t) is such that:

$$\frac{d}{dt} [\text{tr } S(t)] > 0 \qquad \text{for all } t \geq 0 , \tag{2.23}$$

i.e. the mean-square estimation error of x(t) is a monotonically increasing function of time.

Proof: From (2.18) and (2.22), x(t) satisfies the equation:

$$dx(t) = Ax(t)dt + Bdw(t) , \tag{2.24}$$

* A square matrix which does not have a complete set of eigenvectors is called a defective matrix [8].

where

$$A = MFM^{-1} , \tag{2.25}$$

$$B = MG . \tag{2.26}$$

Therefore (2.21) can be written as:

$$M\dot{V}(t)M' = AMV(t)M' + MV(t)M'A' + BQB' , \tag{2.27}$$

or as

$$\dot{S}(t) = AS(t) + S(t)A' + BQB' , \tag{2.28}$$

where

$$\begin{aligned} S(t) &= E\left\{[x(t) - \hat{x}(t)]\,[x(t) - \hat{x}(t)]'\right\} \\ &= E\left\{[Mz(t) - M\hat{z}(t)]\,[Mz(t) - M\hat{z}(t)]'\right\} \\ &= M\left(E\left\{[z(t) - \hat{z}(t)]\,[z(t) - \hat{z}(t)]'\right\}\right)M' \\ &= MV(t)M' . \end{aligned} \tag{2.29}$$

Now, unless matrix F is defective, matrix $A = MFM^{-1}$ can be put into the form shown in Eq. (2.30) (see next page) by an appropriate choice of the transform matrix M [9].

$$A = \begin{bmatrix} \theta_1 & & & & & & & & & & \\ & \theta_2 & & & & & & & & & \\ & & \ddots & & & & & & & & \\ & & & \theta_k & & & & & & & \\ & & & & \alpha_1 & -\beta_1 & & & & & \\ & & & & \beta_1 & \alpha_1 & & & & & \\ & & & & & & \alpha_2 & -\beta_2 & & & \\ & & & & & & \beta_2 & \alpha_2 & & & \\ & & & & & & & & \ddots & & \\ & & & & & & & & & \alpha_m & -\beta_m \\ & & & & & & & & & \beta_m & \alpha_m \end{bmatrix} \tag{2.30}$$

where $\theta_1, \theta_2, \cdots, \theta_k, \alpha_1 \pm j\beta_1, \alpha_2 \pm j\beta_2, \cdots, \alpha_m \pm j\beta_m$ are the eigenvalues of matrix F.

Defining the entries of S(t) and BQB' as:

$$[S(t)]_{i,j} = s_{ij}(t) , \tag{2.31}$$

$$[BQB']_{i,j} = \gamma_{ij} , \tag{2.32}$$

by (2.30), the components of (2.28) can be written as:

$$\dot{s}_{11}(t) = 2\theta_1 s_{11}(t) + \gamma_{11} ,$$

$$\vdots$$

$$\dot{s}_{kk}(t) = 2\theta_k s_{kk}(t) + \gamma_{kk} ,$$

$$\dot{s}_{(k+1)(k+1)}(t) + \dot{s}_{(k+2)(k+2)}(t) = 2\alpha_1 [s_{(k+1)(k+1)}(t) + s_{(k+2)(k+2)}(t)] + [\gamma_{(k+1)(k+1)} + \gamma_{(k+2)(k+2)}] ,$$

$$\vdots$$

$$\dot{s}_{(k+2m-1)(k+2m-1)}(t) + \dot{s}_{(k+2m)(k+2m)}(t) = 2\alpha_m [s_{(k+2m-1)(k+2m-1)}(t) + s_{(k+2m)(k+2m)}(t)] + [\gamma_{(k+2m-1)(k+2m-1)} + \gamma_{(k+2m)(k+2m)}] . \quad (2.33)$$

Defining:

$$S_o = S(0) = MV_oM' , \quad (2.34)$$

by (2.25), (2.20) can be written as:

$$0 = AMV_oM' + MV_oM'A + MGQG'M' - MV_oM'C'R^{-1}CMV_oM' , \quad (2.35)$$

or as:

$$0 = AS_o + S_oA' + BQB' - S_oC'R^{-1}CS_o , \quad (2.36)$$

where:

$$C = HM^{-1}. \quad (2.37)$$

Defining the entries of S_o and $S_o C' R^{-1} C S_o$ as:

$$[S_o]_{i,j} = s_{ijo} , \tag{2.38}$$

$$[S_o C' R^{-1} C S_o]_{i,j} = \sigma_{ij} , \tag{2.39}$$

the components of (2.36) can be written as:

$$0 = 2\theta_1 s_{11o} + \gamma_{11} - \sigma_{11} ,$$

$$\vdots$$

$$0 = 2\theta_k s_{kko} + \gamma_{11} - \sigma_{kk} ,$$

$$0 = 2\alpha_1 (s_{(k+1)(k+1)o} + s_{(k+2)(k+2)o}) + (\gamma_{(k+1)(k+1)} + \gamma_{(k+2)(k+2)}) - (\sigma_{(k+1)(k+1)} + \sigma_{(k+2)(k+2)}) ,$$

$$\vdots$$

$$0 = 2\alpha_m (s_{(k+2m-1)(k+2m-1)o} + s_{(k+2m)(k+2m)o}) + (\gamma_{(k+2m-1)(k+2m-1)} + \gamma_{(k+2m)(k+2m)}) - (\sigma_{(k+2m-1)(k+2m-1)} + \sigma_{(k+2m)(k+2m)}) \tag{2.40}$$

Equations in (2.33) and (2.40) all have the form:

$$\dot{\xi}(t) = c_1 \xi(t) + c_2 ; \; c_2 > 0 , \tag{2.41}$$

$$\xi(0) = \xi_o > 0 , \tag{2.42}$$

$$0 = c_1 \xi_o + c_2 - c_3 ; \; c_3 > 0 , \tag{2.43}$$

which has the solution:

$$\xi(t) = \xi_o e^{c_1 t} + \frac{c_2}{c_1}(e^{c_1 t} - 1), \quad t \geq 0. \tag{2.44}$$

Then:

$$\dot{\xi}(t) = (c_1 \xi_o + c_2) e^{c_1 t}$$

$$= c_3 e^{c_1 t} > 0, \quad t \geq 0 \tag{2.45}$$

by (2.43).

Hence:

$$\dot{s}_{ii}(t) > 0, \quad i = 1, 2, \cdots, k, \quad t \geq 0, \tag{2.46}$$

$$\dot{s}_{ii}(t) + \dot{s}_{(i+1)(i+1)}(t) > 0, \; i = k+1, k+3, \cdots, k+2m-1, \quad t \geq 0, \tag{2.47}$$

and

$$\frac{d}{dt} \operatorname{tr} S(t) = \frac{d}{dt} E\left\{[x(t) - \hat{x}(t)]' [x(t) - \hat{x}(t)]\right\}$$

$$= \sum_{i=1}^{k+2m} \dot{s}_{ii}(t) > 0 \quad \text{for} \quad t \geq 0. \tag{2.48}$$

Q.E.D.

Now we will bring some clarification and justification to the hypotheses of the above result.

Consider the system described by (2.1) and (2.2) for one sensor only, with $y_i = y$, $v_i = v$, $c_i = c$, $R_i = R$. Also consider the system resulting from the transformation (2.22):

$$dx(t) = Ax(t)dt + Bdw(t), \tag{2.49}$$

$$dy(t) = Cx(t)dt + dv(t), \tag{2.50}$$

where A, B and C are given by (2.25), (2.26) and (2.37).

It is easy to show that the systems (2.1) - (2.2) and (2.49) - (2.50) have the same input-output transfer function $\mathcal{H}(s)$, that is:

$$\mathcal{H}(s) = C(sI-A)^{-1} B + I + H(sI-F)^{-1} G + I \text{ for all } s \tag{2.51}$$

In other words, given the transfer function, a realization of the system in state-space form is unique, modulo a similarity transformation. Therefore we wish to argue that an invertible transformation of state variable coordinates is not a serious compromise of the generality of the formulation.

Assume that the sensor has been making observations for a long time, so that the optimal filter processing them is in the steady state. Then under the stabilizability and detectability assumptions for the pairs (F, G) and (F, H), which imply the same conditions for the pairs (A, B) and (A, C) [10], the steady-state error covariance matrix S_o associated with the optimal estimation of x(t) is given by the algebraic matrix Riccati equation (2.36) [6].

At time t = 0, assume that the sensor stops making observations. Then the propagation in time of the error covariance matrix S(t) associated with the optimal prediction of x(t) for t > 0 is given by (2.28), with the initial value of $S(0) = S_o$. Now Lemma 2.2 indicates that tr S(t); or the mean-square prediction error of x(t) increases monotonically with time.

In view of the preceding discussion, the worst-case optimization results presented in Chapter 4 are valid with the following restrictions imposed on the dynamical system under observation:

1. In Sections 4.2 and 4.3, the dynamical system is assumed to be one-dimensional.
2. In Sections 4.4.2 and 4.4.3, the system is assumed to be multidimensional with the structure indicated in (2.30).
3. In Section 4.4.4, there is no need to impose additional structure on the system; the A matrix can be any multi-dimensional square real matrix.

It should be noted that the conditions imposed on the system models in Sections 4.2, 4.3, 4.4.2 and 4.4.3 are sufficient to insure that the mean-square error is a monotonically increasing function of time for the optimization problem considered. However, they are not necessary; there exist more general systems which display monotonicity without satisfying the above conditions.

2.4 The Estimation Scheme and the Content of Messages

We assume that at each sensor node #i there is a local linear least-squares estimator which calculates the current state estimate $\hat{x}_i(t)$ given the previous local observations:

$$\hat{x}_i(t) = E\,(x(t) \mid y_i(s),\ t_o \le s \le t)\,. \tag{2.52}$$

The local error covariance matrix $P_i(t)$ associated with the optimal estimator is given by the matrix Riccati differential equation [6]:

$$\dot{P}_i(t) = AP_i(t) + P_i(t)A' + BQB' - P_i(t)C_i'R_i^{-1}C_iP_i(t)$$

$$P_i(t_o) \text{ given} \tag{2.53}$$

In this thesis, we will deal for the lim $t_o \to -\infty$ case; that is, we assume that the process has been under observation for a long time. In this case, due to our assumptions of the stabilizability of the pair (A, B) and the detectability of the pair (A, C_i), the solution of the Riccati equation (2.52) approaches a steady-state constant value $\overline{P}_i$, which is the unique nonnegative-definite symmetric solution of the albegraic matrix Riccati equation:

$$0 = A\overline{P}_i + \overline{P}_i A' + BQB' - \overline{P}_i C_i' R_i^{-1} C_i \overline{P}_i \tag{2.54}$$

The messages sent by the nodes at discrete times are assumed to contain all the information about the observations taken by the node since the last message was sent, plus all the information received from the other nodes during this interval. The content of messages for transfer of information with no loss of optimality (that is, reduced sufficient statistics which lead to the same estimation performance as would the actual set of observations) is discussed in Chapter 3. In short, in our model there is no loss of information due to the communications between the nodes; there is only delay of information.

2.5 Model for the Communication Delays

Message delays between nodes in a communications network in general exhibit a stochastic behavior. In this thesis, with the exception of Section 5.2, we will approximate this behavior with their average values, for the sake of mathematical tractability. In particular, except for Section 5.3, we assume that the delay on link (i, j) is a convex and monotonically increasing function of the average traffic rate on that link in messages/unit time, with a finite escape value C_{ij} which will be defined as the capacity of the link. A possible dependence of d_{ij}, the communications delay on f_{ij}, the average traffic rate is illustrated in Fig. 2.1.

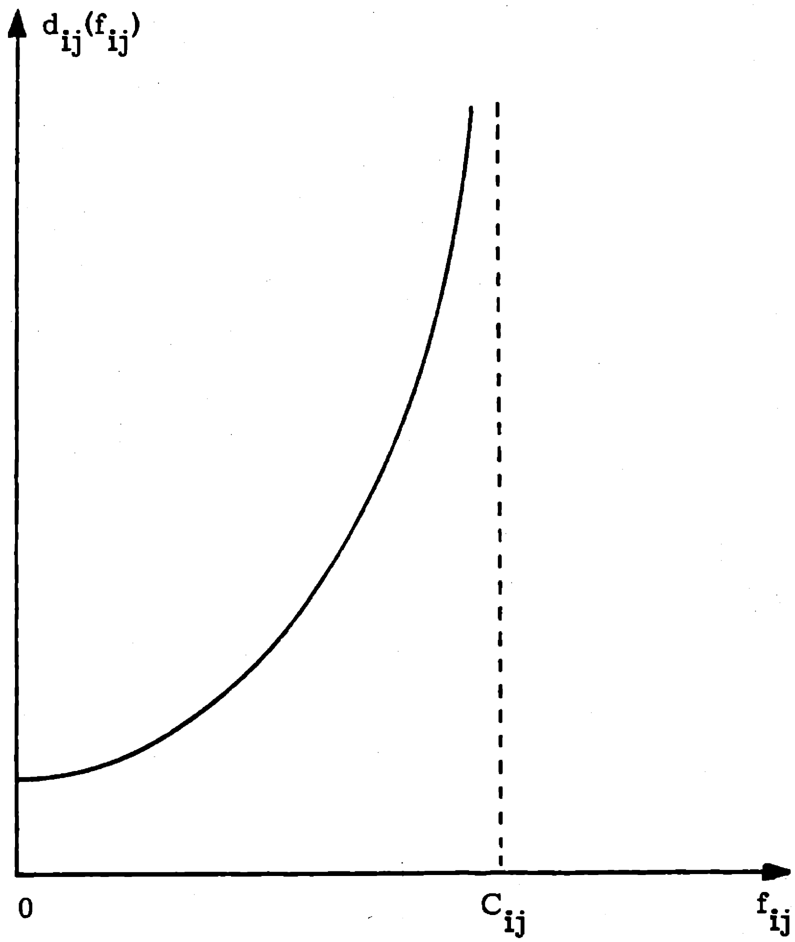

Fig. 2.1 Dependence of d_{ij} on f_{ij}.

2.6 The Objective Functions

The goal is to have the best mean-square estimate of the state of the system at the destination node. But since statistical information carried by messages from sensor nodes arrives at the destination node at discrete times, the mean-square estimation error is a time-varying quantity. For the purposes of optimization, it is desirable to try to express the performance in terms of a scalar. For the problem under consideration, the time-average and the maximum value of the mean-square estimation error at the destination are meaningful possibilities. Again for the sake of mathematical tractability, we will prefer the latter criterion in Chapter 4, where the main optimization results are presented.

CHAPTER 3

DECENTRALIZED ESTIMATION ISSUES

3.1 Introduction

Consider the system configuration shown in Fig. 3.1

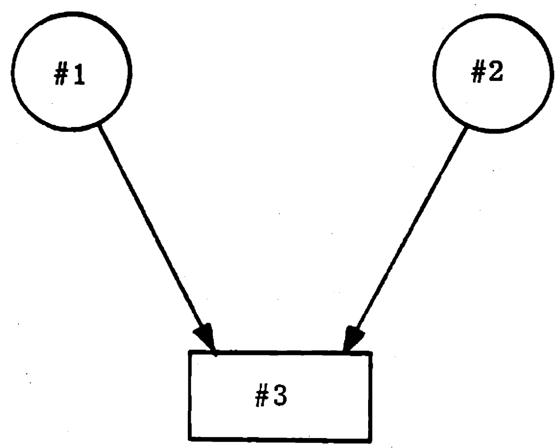

Fig. 3.1 System configuration for Chapter 3.

There are two sensor nodes, #1 and #2, and a destination node, #3. The sensor nodes observe independently a system driven by w, a Brownian process of intensity Q:

$$dx(t) = Ax(t)dt + Bdw(t) . \tag{3.1}$$

The measurements obtained by the nodes #1 and #2 are denoted as:

$$dy_i(t) = C_i x(t)dt + dv_i(t), \quad t \geq 0 \tag{3.2}$$

where v_1 and v_2 are independent Brownian motion processes such that:

$$E\left\{\begin{bmatrix} dv_1(t) \\ dv_2(t) \end{bmatrix} \begin{bmatrix} dv_1'(s) & dv_2'(s) \end{bmatrix}\right\} = \begin{bmatrix} R_1 & O \\ O & R_2 \end{bmatrix} dt. \tag{3.3}$$

The filtered minimum mean-square estimate of the state x(t) is desired at the destination node. We will denote it as:

$$\hat{x}(t) = E\left[x(t) \middle| \begin{array}{ll} y_1(s), & 0 \le s \le t \\ y_2(s), & 0 \le s \le t \end{array} \right] \tag{3.4}$$

We assume that $\hat{x}(0)$, and the corresponding error covariance matrix, P(0), are available.

There are two strategies to obtain $\hat{x}(t)$ at node #3 [2]:

(i) Centralized Approach: Nodes #1 and #2 send their measurements to node #3 and node #3 runs a continuous-time Kalman filter to obtain $\hat{x}(t)$.

(ii) Decentralized Approach: Nodes #1 and #2 perform local information processing and send sufficient statistics to node #3 which utilizes these to obtain $\hat{x}(t)$.

The clue to finding the correct approach lies in the amount of information that has to be transmitted between the nodes. For a realistic modelling of the communication links with finite channel capacity, it is not possible to transmit information continuously with finite delay. Rather, messages must be coded and transmitted starting at discrete points in time. For the estimation problem considered here, let t_i^j be the time the j^{th} message is sent from node #i to the destination. For optimum state estimation at the destination node with discrete-time communications, node #i must send at time t_i^j some information which will lead to the same estimation performance (in terms of the mean-square error at all points in time) as if it had sent all its observations since the last message: $\{y_i(t),\ t_i^{j-1} \le t \le t_i^j\}$. (This assumes no messages are lost during transmission. If messages may be lost, then information equivalent to

$\{y_i(t),\ 0 \le t \le t_i^j\}$ must be sent.) Since sending $\{y_i(t),\ t_i^{j-1} \le t \le t_i^j\}$ is not possible due to capacity limitations (or, assuming the observations are also made at discrete times much more frequently than messages are sent, sending all the observations is impractical in view of the large volume of data), doing more local information processing and sending shorter messages which give the same estimation performance is desired.

3.2 Local Information Processing for Decentralized Estimation with Asynchronous Reporting Times

An algorithm for optimal decentralized estimation with decentralized processing and finite length messages have been developed [2], [11] for synchronous reporting times, that is, for $t_1^j = t_2^j \triangleq t^j$. According to this algorithm, $\hat{x}(t^j)$ is obtained as follows.

At time t^j, the sensor nodes send $\hat{x}_i(t^j)$, i = 1, 2, the local m.s.e. estimates, and $r_i(t^j)$, i = 1, 2, a term needed to correct correlation of $\hat{x}_1$ and $\hat{x}_2$ due to $\hat{x}(0)$ and w(t), whose statistical specificiations are common knowledge to both sensor nodes. $\hat{x}_i(t^j)$ and $r_i(t^j)$ are obtained by

$$\dot{\hat{x}}_i(t) = [A - P_i(t)C_i'R_i^{-1}C_i]\hat{x}_i(t) + P_i(t)C_i'R_i^{-1}y_i(t),\ i = 1,2 \tag{3.5}$$

$$\hat{x}_i(0) = \hat{x}(0),\ i = 1,2 \tag{3.6}$$

$$\begin{aligned}\dot{r}_i(t) = {} & [A - P(t)(C_1'R_1^{-1}C_1 + C_2'R_2^{-1}C_2)]\,r_i(t) \\ & + [P(t)P_i^{-1}(t) - I]\,BQB'\,P_i^{-1}(t)\hat{x}_i(t),\ i = 1,2\end{aligned} \tag{3.7}$$

$$r_1(0) + r_2(0) = -\hat{x}(0) \tag{3.8}$$

$$\begin{aligned}\dot{P}(t) = {} & AP(t) + P(t)A' + BQB' - P(t)(C_1'R_1^{-1}C_1 \\ & + C_2'R_2^{-1}C_2)P(t),\ P(0)\ \text{given}\end{aligned} \tag{3.9}$$

$$\dot{P}_i(t) = AP_i(t) + P_i(t)A' + BQB' - P_i(t)C_i' R_i^{-1} C_i P_i(t), \quad i = 1,2 \tag{3.10}$$

$$P_i(0) = P(0), \quad i = 1,2 \tag{3.11}$$

Then:

$$\hat{x}(t^j) = r_1(t^j) + r_2(t^j) + P(t^j)[P_1^{-1}(t^j)\hat{x}_1(t^j) + P_2^{-1}(t^j)\hat{x}_2(t^j)] \tag{3.12}$$

Now we will describe the extension of the above solution to the case of asynchronous reporting times. Again let t_1^j and t_2^k be the j^{th} and the k^{th} reporting times of sensor nodes #1 and #2 respectively. Let $t_2^{k^*}$ be a particular reporting time of node #2. Other relevant times are defined below and in the timing diagram, Fig. 3.2.

$$t_1^{j^*} \triangleq \min_{t_1^j \geq t_2^{k^*}} t_1^j \tag{3.13}$$

$$t_1^{j^{**}} \triangleq \max_{t_1^j \leq t_2^{k^*+1}} t_1^j \tag{3.14}$$

We assume that there exist some t_1^j in the interval $[t_2^{k^*}, t_2^{k^*+1}]$. Otherwise, the solution can be applied by exchanging the roles of the nodes #1 and #2. We will specify the sufficient statistics that must be sent by the sensor nodes #1 and #2 and the necessary computations that must be carried out at the destination node #3 to find the optimal $\hat{x}(t)$ at node #3 during the interval $[t_1^{j^*}, t_2^{k^*+1}]$. This interval is the smallest period that captures all the characteristics of the solution.

Theorem 3.1: Sufficient statistics for asynchronous, decentralized linear estimation are given as follows:

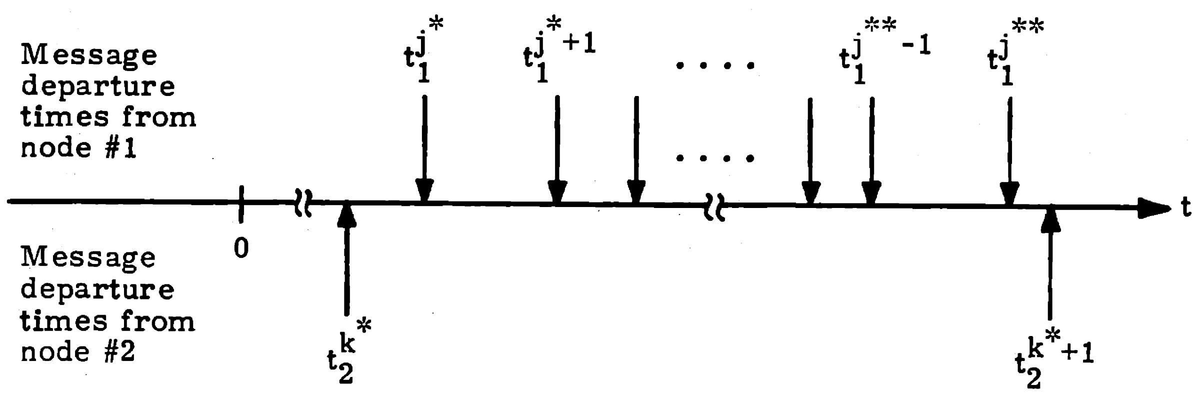

Fig. 3.2 Departure times of the messages from the sensor nodes.

I. Node #2 at $t_2^{k^*}$ sends:

1. $\hat{x}_2 (t_2^{k^*})$ from:

$$\dot{\hat{x}}_2(t) = [A - P_2(t)C_2' R_2^{-1} C_2]\hat{x}_2(t) + P_2(t)C_2' R_2^{-1} y_2(t) \tag{3.15}$$

$$\hat{x}_2(0) = \hat{x}(0) \tag{3.16}$$

2. $r_2 (t_2^{k^*})$ from:

$$\dot{r}_2(t) = [A - P(t)(C_1' R_1^{-1} C_1 + C_2' R_2^{-1} C_2)] r_2(t) + [P(t)P_2^{-1}(t) - I] BQB' P_2^{-1}(t) \hat{x}_2(t) \tag{3.17}$$

$$r_2(0) = 0 \tag{3.18}$$

$$\dot{P}(t) = AP(t) + P(t)A' + BQB' - P(t)(C_1'R_1^{-1} C_1 + C_2' R_2^{-1} C_2) P(t), \; P(0) \text{ given} \tag{3.19}$$

$$\dot{P}_2(t) = AP_2(t) + P_2(t)A' + BQB' - P_2(t)C_2' R_2^{-1} C_2P_2(t) \tag{3.20}$$

$$P_2(0) = P(0) \tag{3.21}$$

3. Additional statistic necessary for optimal computation of $\hat{x}(t)$ at the destination before $t_1^{j^*}$. This statistic would be named $g_2 (t_2^{k^*})$ and is the counterpart of $g_1 (t_1^{j^{**}})$ described in the following entries.

II. Node #1 at $t_1^{j^*}$ sends:

1. $\hat{x}_1 (t_2^{k^*})$ from:

$$\dot{\hat{x}}_1(t) = [A - P_1(t) C_1' R_1^{-1} C_1] \hat{x}_1(t) + P_1(t) C_1' R_1^{-1} y_1(t) \qquad (3.22)$$

$$\hat{x}_1(0) = \hat{x}(0) \qquad (3.23)$$

2. $\underline{r_1 (t_2^{k^*})}$ from:

$$\dot{r}_1(t) = [A - P(t)(C_1' R_1^{-1} C_1 + C_2' R_2^{-1} C_2)] r_1(t)$$
$$+ [P(t)P_1^{-1}(t) - I] BQB' P_1^{-1}(t) \hat{x}_1(t) \qquad (3.24)$$

$$r_1(0) = - \hat{x}(0) \qquad (3.25)$$

with P(t) from (3.19) and:

$$\dot{P}_1(t) = AP_1(t) + P_1(t)A' + BQB' - P_1(t)C_1R_1^{-1}C_1P_1(t) \qquad (3.26)$$

$$P_1(0) = P(0) \qquad (3.27)$$

3. $\underline{\hat{x}_1 (t_1^{j^*})}$ from (3.22), (3.23):

4. $\underline{g_1 (t_1^{j^*})}$ from:

$$\dot{g}_1(t) = [A - S_1(t) C_1' R_1^{-1} C_1] g_1(t)$$
$$+ [S_1(t)P_1^{-1}(t) - I] BQB' P_1^{-1}(t) \hat{x}_1(t) \qquad (3.28)$$

$$g_1(t_2^{k^*}) = 0 \qquad (3.29)$$

with $P_1(t)$ from (3.26), (3.27) and

$$\dot{S}_1(t) = AS_1(t) + S_1(t)A' + BQB' - S_1(t)C_1' R_1^{-1} C_1 S_1(t) \qquad (3.30)$$

$$S_1(t_2^{k^*}) = P(t_2^{k^*}) \tag{3.31}$$

with P(t) from (3.19).

III. Upon receiving messages sent at $t_2^{k^*}$ and $t_1^{j^*}$, <u>node #3</u> <u>computes:</u>

1. <u>$\hat{x}(t_2^{k^*})$</u> $\triangleq$ $E[x \mid y_1(s),\ 0 \le s \le t_2^{k^*};\ y_2(s),\ 0 \le s \le t_2^{k^*}]$

$$= r_1(t_2^{k^*}) + r_2(t_2^{k^*}) + P(t_2^{k^*})[P_1^{-1}(t_2^{k^*})\hat{x}_1(t_2^{k^*}) + P_2^{-1}(t_2^{k^*})\,\hat{x}_2(t_2^{k^*})] \tag{3.32}$$

2. <u>$\tilde{x}_1(t_1^{j^*})$</u> $\triangleq$ $E[x \mid y_1(s),\ 0 \le s \le t_1^{j^*};\ y_2(s),\ 0 \le s \le t_2^{k^*}]$

$$= g_1(t_1^{j^*}) + h_2(t_1^{j^*}) + S_1(t_1^{j^*})\,[P_1^{-1}(t_1^{j^*})\hat{x}_1(t_1^{j^*}) + Z_2^{-1}(t_1^{j^*})\bar{x}_2(t_1^{j^*})] \tag{3.33}$$

with P(t) from (3.19); $P_1(t)$ from (3.26), (3.27); $P_2(t)$ from (3.20), (3.21); $S_1(t)$ from (3.30), (3.31); $g_1(t)$ from (3.28), (3.29) and

$$\dot{h}_2(t) = [A - S_1(t)C_1'R_1^{-1}C_1]\,h_2(t) + [S_1(t)Z_2^{-1}(t) - I]\,BQB'\,Z_2^{-1}(t)\,\bar{x}_2(t) \tag{3.34}$$

$$h_2(t_2^{k^*}) = -\hat{x}(t_2^{k^*}) \tag{3.35}$$

$$\dot{\bar{x}}_2(t) = A\bar{x}_2(t) \tag{3.36}$$

$$\bar{x}_2(t_2^{k^*}) = \hat{x}_2(t_2^{k^*}) \tag{3.37}$$

$$\dot{Z}_2(t) = AZ_2(t) + Z_2(t)A' + BQB' \tag{3.38}$$

$$Z_2(t_2^{k^*}) = P_2(t_2^{k^*}) \tag{3.39}$$

IV. Node #1 at t_1^j, $j^* + 1 \leq j \leq j^{**} - 1$ sends:

1. $\hat{x}_1(t_1^j)$ from (3.22), (3.23).

2. $g_1(t_1^j)$ from (3.28), (3.29).

V. Upon receiving the message sent at t_1^j, $j^* + 1 \leq j \leq j^{**} - 1$, mode #3 computes:

$$\tilde{x}_1(t_1^j) \triangleq E[x \mid y_1(s),\ 0 \leq s \leq t_1^j;\ y_2(s),\ 0 \leq s \leq t_2^{k^*}], \tag{3.40}$$

which is given by (3.33) with all terms evaluated at $t = t_1^j$ and all other terms defined in Part III.

VI. Node #1 at $t_1^{j^{**}}$ sends:

1. $\hat{x}_1(t_1^{j^{**}})$ from (3.22), (3.23).

2. $g_1(t_1^{j^{**}})$ from (3.28), (3.29).

3. $r_1(t_1^{j^{**}})$ from (3.24), (3.25).

VII. Node #2 at $t_2^{k^*+1}$ sends:

1. $\hat{x}_2(t_1^{j^{**}})$ from (3.15), (3.16).

2. $r_2(t_1^{j^{**}})$ from (3.17), (3.18).

3. $\hat{x}_2(t_2^{k^*+1})$ from (3.15), (3.16).

4. $\underline{g_2(t_2^{k^*+1})}$ from:

$$\dot{g}_2(t) = [A - S_2(t)\, C_2'\, R_2^{-1}\, C_2]\, g_2(t) + [S_2(t) P_2^{-1}(t) - I]\, BQB'\, P_2^{-1}(t)\, \hat{x}_2(t) \quad (3.41)$$

$$g_2(t_1^{j^{**}}) = 0 \quad (3.42)$$

with $P_2(t)$ from (3.20), (3.21) and

$$\dot{S}_2(t) = AS_2(t) + S_2(t)A' + BQB' + S_2(t) C_2' R_2^{-1} C_2 S_2(t) \quad (3.43)$$

$$S_2(t_1^{j^{**}}) = P(t_1^{j^{**}}) \quad (3.44)$$

with $P(t)$ from (3.19).

Proof: To show that the mean-square error at node #3 during the interval $[t_1^{j^*}, t_2^{k^*+1}]$ by this method is the same as the case with full observation transmission, it suffices to show that they are equal at the departure times; since in between these times the same optimal prediction will be performed at the destination node.

In [2] and [11] it was shown that computing $\hat{x}(t)$ from $\hat{x}_i(t)$ and $r_i(t)$, as given in (3.5) through (3.12), is equivalent to computing it from $y_i(t)$. These equations are the basis for the solution presented here also. $\tilde{x}_1(t_1^{j^*})$, which is defined in (3.33), is computed as follows: $r_i(t)$, the correction terms, are defined by the differential equations (3.7) which are driven by $y_i(t)$, the observations. Thus they have to be calculated at the sensor nodes, except that they can also be calculated at the destination node when there are no observations available, provided the initial conditions can be obtained. The interval $[t_2^{k^*}, t_1^{j^*}]$ is such an interval when the observations $y_2(t)$ are not available at the destination node by any means. Consider the initial condition (3.8). It states that the sum of the initial values of the correction terms must be equal to the negative of the global state estimate. Node #2 at time $t_2^{k^*}$ sends $\hat{x}_2(t_2^{k^*})$ and $r_2(t_2^{k^*})$ and node #1 at time $t_1^{j^*}$ sends $\hat{x}_1(t_2^{k^*})$ and $r_1(t_2^{k^*})$; thus $\hat{x}(t_2^{k^*})$ is obtained at the destination. For the interval $[t_2^{k^*}, t_1^{j^*}]$, node #1 runs the differential equation for $g_1(t)$ (which is also a correction term like $r_1(t)$, but one which takes into consideration that the observations of node #2 are not available for this interval, hence $R_2 = \infty$ or $C_2 = 0$) with the initial condition of $g_1(t_2^{k^*}) = 0$, since the global estimate is not available there. The destination node runs the differential equations of the predicted local state estimate at node #2, $\bar{x}_2(t)$, using the initial condition $\hat{x}(t_2^{k^*})$ and the corresponding correction term $h_2(t)$, with the initial condition $h_2(t_2^{k^*}) = -\hat{x}(t_2^{k^*})$, making use of the global estimate already computed. As a result, utilizing the received values of $\hat{x}_1(t_1^{j^*})$, $g_1(t_1^{j^*})$ and the computed values of $\bar{x}_2(t_1^{j^*})$, $h_2(t_1^{j^*})$, the destination node #3 calculates $\tilde{x}_1(t_1^{j^*})$ as in (3.33), which is a variation of (3.12).

For the computation of $\tilde{x}_1(t_1^j), j^* + 1 \le j \le j^{**} - 1$, as defined in (3.40), node #1 needs only to send $\hat{x}_1(t_1^j)$ and $g_1(t_1^j)$, which is the correction term which takes into consideration the fact that observations of node #2 are not available for the period $[t_2^{k^*}, t_1^j]$. On receiving these statistics, node #3 computes $\tilde{x}_1(t_1^j)$ as described for $\tilde{x}_1(t_1^{j^*})$ above.

At $t = t_1^{j^{**}}$, node #1 sends $\hat{x}(t_1^{j^{**}})$ and $g_1(t_1^{j^{**}})$ for the calculation of $\tilde{x}_1(t_1^{j^{**}})$, and in addition sends $r_1(t_1^{j^{**}})$ which is necessary for the calculation of $t_2^{k^*+1}$. These statistics are counterparts of the statistics sent by node #2 at $t_2^{k^*}$. At $t = t_2^{k^*+1}$, node #2 sends the counterparts of the statistics sent by node #1 at t_1^{j*}. By symmetry, the roles of the two nodes are reversed, but the procedure is the same for the next interval.

Q.E.D.

As it will be apparent from an examination of the algorithm, the common knowledge required at the sensor nodes consists of the state-space stochastic model of the system, the local observation models (including noise intensities), and the reporting times for both sensors. It is possible to extend the above algorithm to the case of a network of sensors, with the common knowledge including the local observation models and reporting times of all nodes.

For the sake of simplicity, we have ignored the delays that are incurred by the messages. This simplification does not sacrifice any generality, and if the messages are marked by their departure times, their contents can be utilized retroactive to that time at the destination. The algorithm still leads to the same estimation performance to the one that would have been achieved had the messages been carrying all the previous local observations.

With the existence of at least one set of finite dimensional sufficient statistics established, we now turn to the primary topic of concern - how to regulate the flow of messages in the network to achieve optimal performance.

CHAPTER 4

WORST-CASE MINIMAX OPTIMIZATION IN THE STEADY STATE

4.1 Introduction

In this chapter we will consider the situation where the local estimation processes at the sensor nodes have reached the steady-state in the sense of Section 2.4. The objective will be to minimize the maximum value the mean-square error takes over time at the destination node. In Section 4.2 we will consider a specific system configuration with two sensor nodes and a destination node in order to illustrate the basic approach of this chapter. Optimization with the phase difference between the reporting times of the two sensors as a control variable as well as the frequencies of the reports will be contrasted with optimization with respect to the frequencies only. In the latter approach the optimization will be done for the worst-case phase difference that gives rise to the highest possible mean-square error at the destination node.

In Sections 4.3 and 4.4, we will study the extension of the worst-case minimax optimization approach to networks with arbitrary number of sensor nodes and a destination node. In Section 4.3, we will allow any given sensor node to send messages to only one other node. The resulting solution to the optimization problem will be compared in its simplicity with the solution to Section 4.4, in which the sensor nodes can send messages on more than a single path. The solution to the problem will require nonlinear programming.

4.2 Special Case With 2 Sensor Nodes

4.2.1 The Problem Description

Consider the network structure of Fig. 4.1.

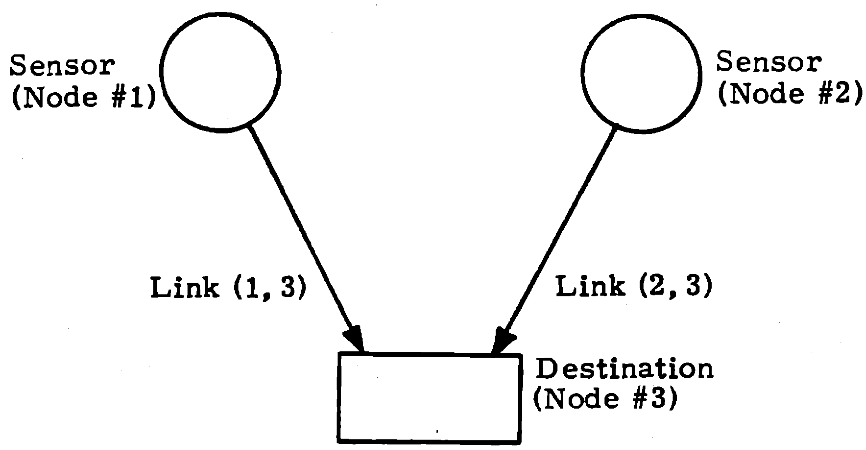

Fig. 4.1 System configuration for Section 4.2.

In Sections 4.2 and 4.3 we will consider a scalar dynamical system with scalar observations:

$$dx(t) = Ax(t)dt + Bdw(t) \tag{4.1}$$

$$dy_i(t) = C_i x(t)dt + dv_i(t) \tag{4.2}$$

$$i = 1, 2.$$

Here x(t) is the state and $y_i(t)$ are the observations. w(t) and $v_i(t)$ are uncorrelated zero-mean Wiener processes with intensities Q and R_i. In addition, the pair $\{A, B\}$ is stabilizable and the pairs $\{A, C_i\}$ are detectable.

The links have the delay models described in Chapter 2 with $d_1 = f_1(T_1)$ and $d_2 = f_2(T_2)$, where T_1 is the intermessage period on link (1, 3) and T_2 is the period on link (2, 3). The nodes make continuous measurements, run continuous-time Kalman filters, and send sufficient statistics, which summarize their observations since the last report., to the destination node every T_1 and T_2 seconds, respectively.

The objective is to minimize the maximum value over time of the mean-square error at the destination node. In Section 4.2.2 the control variables will be T_1, T_2, and Δ, the phase difference between the reporting sequences. In Section 4.2.3, the optimization will be done over T_1 and T_2 for the worst case Δ.

4.2.2 Optimization With Respect to T_1, T_2 and Δ

In this section we will consider minimizing the maximum peak value of $p(t) \triangleq$ mean-square estimation error at the destination mode. We will assume throughout this section that $T_1/T_2 = k_1/k_2$ for some integers k_1 and k_2 so that the waveform for p(t) is periodic. Since the set of rationals is dense in the set of reals, we lose nothing by this assumption.

The points in time at which the messages are sent from the two sensor nodes and received at the destination node are labeled as shown in Fig. 4.2.

For example, the j^{th} message from Node #2 leaves Node #2 at t^j_{d2} and arrives at the destination at t^j_{a2}. According to this notation, the phase difference Δ is defined as:

$$\Delta \triangleq \min_{t^j_{d2} \geq t^i_{d1}} \left[t^j_{d2} - t^i_{d1} \right] \quad \text{for all } i, j \tag{4.3}$$

As an aid for visualization, Fig. 4.3 is an example which shows the waveform of p(t) for the system model:

$$\begin{aligned} dx(t) &= dw(t), \\ dy_1(t) &= x(t)dt + dv_1(t), \\ dy_2(t) &= x(t)dt + dv_2(t), \end{aligned} \tag{4.4}$$

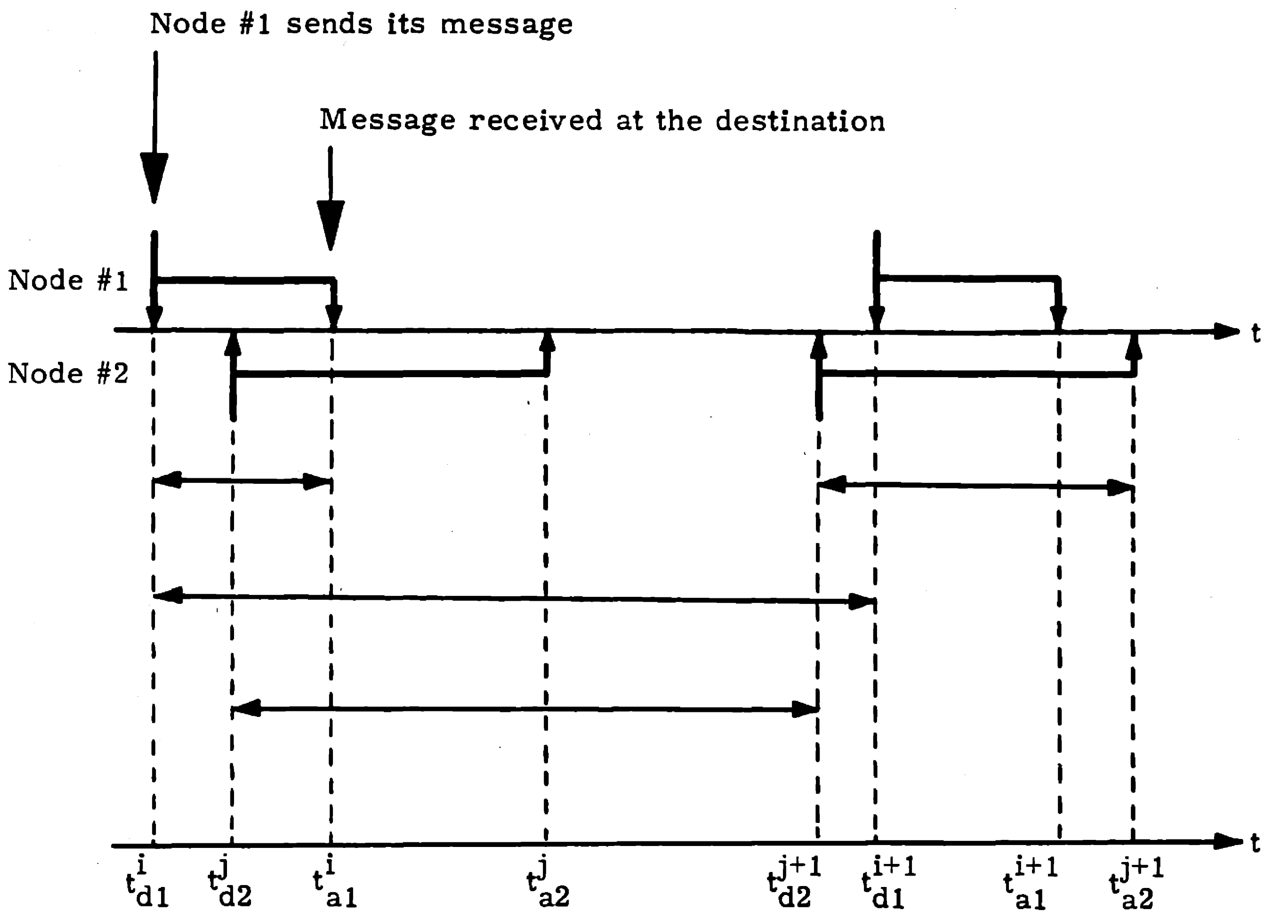

Fig. 4.2 Departure and arrival times of the messages from the sensor nodes.

where w, v_1 and v_2 are independent Brownian motion processes with intensities q, r_1 and r_2; and for particular values of T_1, d_1, T_2 and d_2.

If a certain peak p' occurs at t', then:

$$p' = p(t') = \Phi_m(a') + \eta_m(a', b') \tag{4.5}$$

where

$$a' = u' - \max_{t_{dn}^k \leq u'} t_{dn}^k \quad \text{for all } k$$

$$b' = t' - u' \tag{4.6}$$

$$u' = \max_{\substack{t_{d1}^k:\ t_{a1}^k \leq t' \\ t_{d2}^\ell:\ t_{a2}^\ell \leq t'}} \left[t_{d1}^k,\ t_{d2}^\ell\right] \quad \text{for all } k, \ell \tag{4.7}$$

m = 1, n = 2, if t_{d1}^k is max, in (4.7);
m = 2, n = 1, if t_{d2}^k is max, in (4.7).

Functions Φ_m and η_m are defined as follows:

$$\Phi_m(a') = \text{trace } P_m(a'), \ m = 1, 2 \tag{4.8}$$

$$\dot{P}_m(t) = AP_m(t) + P_m(t)A' + BQB' - P_m(t)C_m' R_m^{-1} C_m P_m(t)$$

$$P_m(0) = \overline{P} \tag{4.9}$$

$$0 = A\overline{P} + \overline{P}A' + BQB' - \overline{P}C_1' R_1^{-1} C_1 \overline{P} - \overline{P}C_2' R_2^{-1} C_2 \overline{P} \tag{4.10}$$

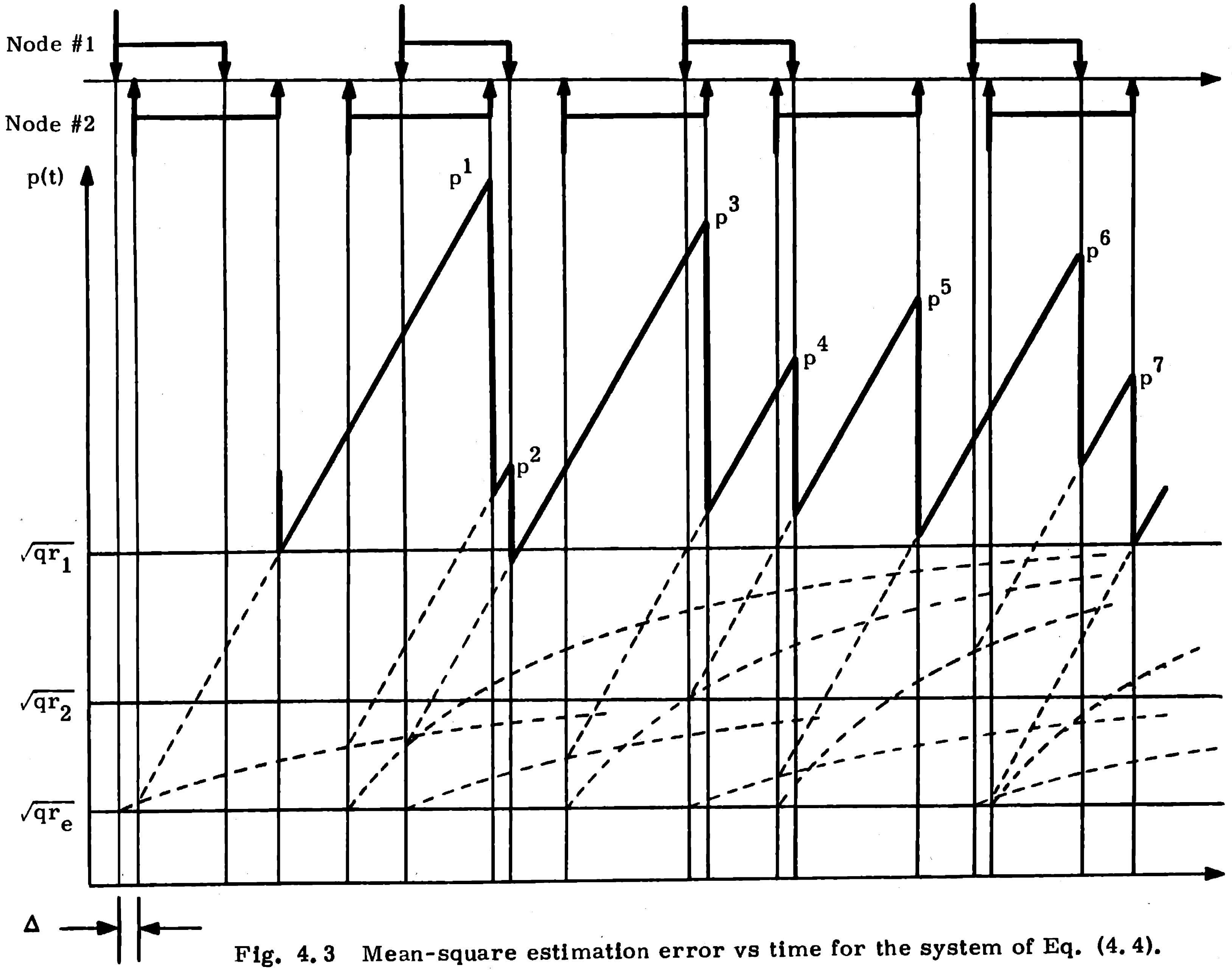

Fig. 4.3 Mean-square estimation error vs time for the system of Eq. (4.4).

$$\eta_m(a',b') = \text{trace } S_m(a',b') \tag{4.11}$$

$$\dot{S}_m(a',t) = AS_m(a',t) + S_m(a',t)A' + BQB'$$

$$S_m(a',0) = P_m(a') \tag{4.12}$$

Therefore, $\overline{P}$ is the steady-state error covariance matrix that results from observations by both sensor nodes. It satisfies the matrix algebraic Riccati equation (4.10). $P_m(a')$ is the error covariance matrix that results when only the observations for sensor node #m (m = 1,2) are available for the last a' seconds. It satisfies the Riccati differential matrix equation (4.9). $S_m(a',b')$ is the error covariance matrix resulting from prediction during the last b' seconds after getting observations from node #m for a' seconds. $S_m(a',t)$ satisfies the linear matrix equation (4.12).

In each period of p(t), there are $k_1 + k_2$ peaks, if $\frac{T_1}{k_1} = \frac{T_2}{k_2} \triangleq c$, where k_1 and k_2 are relatively prime integers. From (4.4) and (4.5) we can deduce that the peak values in the period denoted as $p^1, p^2, \cdots, p^{(k_1+k_2)}$ can be expressed in the form:

$$p^i = \Phi_m(a^i) + \eta_m(a^i, b^i), \quad m = 1,2, \quad i = 1,2,\cdots,k_1+k_2. \tag{4.13}$$

Here m, a^i and b^i satisfy one of the following 3 conditions:

<u>Type 1</u>:
$$m = \begin{cases} 1 \text{ if } T_1 + d_1 < T_2 + d_2 \\ 2 \text{ if } T_1 + d_1 > T_2 + d_2 \end{cases}$$

$$a^i \le |(T_1 + d_1) - (T_2 + d_2)|$$

$$b^i = \min(T_1 + d_1,\ T_2 + d_2) \tag{4.14}$$

Type 2: $m = 1$

$$a^i + b^i = T_2 + d_2$$

$$b^i \le T_1 + d_1 \tag{4.15}$$

Type 3: $m = 2$

$$a^i + b^i = T_1 + d_1$$

$$b^i \le T_2 + d_2 \tag{4.16}$$

Using the relationships (4.14) through (4.16) and the fact that Φ_m and η_m are monotonically increasing functions of t, with η_m increasing faster than Φ_m with respect to its second argument, we can see that reducing $T_1 + d_1$ and $T_2 + d_2$ results in a reduction of the values of the peaks in general. In particular, it can be shown that if p_{opt} is the value of the highest peak obtained by finding the optimum Δ for a given $(T_1 + d_1, T_2 + d_2)$ pair, then $p'_{opt} \le p_{opt}$ for a $(T'_1 + d'_1, T'_2 + d'_2)$ pair if $T'_1 + d'_1 \le T_1 + d_1$ and $T'_2 + d'_2 \le T_2 + d_2$. Hence the first step in the optimization procedure is to minimize $T_1 + d_1(T_1)$ and $T_2 + d_2(T_2)$ individually, again remembering that the delay on a link is only dependent on the traffic on that link. Therefore:

$$T_1^* \triangleq \underset{T_1 > 0}{\text{argmin}}\ T_1 + d_1(T_1) \tag{4.17}$$

$$T_2^* \triangleq \underset{T_2 > 0}{\text{argmin}}\ T_2 + d_2(T_2) \tag{4.18}$$

As mentioned above, T_1^* and T_2^* are assumed to satisfy:

$$\frac{T_1^*}{k_1} = \frac{T_2^*}{k_2} \triangleq c \qquad (4.19)$$

for some relatively prime integers k_1 and k_2.

Now, using the values T_1^*, $d_1(T_1^*)$, T_2^* and $d_2(T_2^*)$, we have to find Δ^* which minimizes the value of the highest peak. To this end, we will give the following facts.

Lemma 4.1: $0 \leq \Delta < c$. That is, we only have to carry out the search for Δ_{opt} in the interval $[0, c)$.

Proof: Define $\delta(j)$ as the time difference between the departure of message #j from node #2 and the departure of the most recent message from node #1:

$$\delta(j) \triangleq t_{d2}^j - \max_{\substack{i \\ t_{d2}^j \geq t_{d1}^i}} t_{d1}^i \qquad (4.20)$$

So, according to this terminology,

$$\Delta \triangleq \min_j \delta(j) \qquad (4.21)$$

Let us keep the periodic departure time sequence for node #1 fixed. As we shift the periodic departure time sequence for node #2 with respect to the first one, the value of Δ will increase until $t_{d1}^i = t_{d2}^j$ for some i, j when Δ becomes zero. Therefore the configuration resulting at the upper limit for Δ is the same as that of $\Delta = 0$, and the upper limit for Δ will be:

$$\Delta_{sup} = \min_{\substack{j \\ \delta(j) > 0}} \delta(j) \qquad (4.22)$$

for this configuration. Let us mark the time at which two departures coincide from the two nodes as $t = 0$. Due to the assumption that k_1 and

k_2 are relatively prime integers, during the next interval of p(t), there will be k_2 and k_1 departures from nodes #1 and #2, respectively. The times for these departures will be 0, T_1, $2T_1$, $\cdots$, k_2T_1 and 0, T_1, $2T_2$, $\cdots$, k_1T_2, or 0, k_1c, $2k_1c$, $\cdots$, k_2k_1c and 0, k_2c, $2k_2c$, $\cdots$, k_1k_2c for nodes #1 and #2. Let us label the departures from node #2 in this period with superscripts j = 1, 2, $\cdots$, k_1 so that

$$t_{d2}^{j} = jk_2c, \qquad j = 1, 2, \cdots, k_1. \tag{4.23}$$

Then:

$$\delta(j) = (m_jk_2 - n_jk_1)c = r_jc \tag{4.24}$$

where m_j, n_j and r_j are positive integers for j = 1, 2, $\cdots$, k_1. Now $r_i \neq r_j$ for $i \neq j$, $i, j \in \{1, 2, \cdots, k_1\}$, because otherwise it would mean that the time difference between the departure of a message from node #2 and the departure of the most recent message from node #1 are the same for two different messages from node #2 during the p(t) period $(0, k_1k_2c]$. This in turn would imply that there is a smaller period of p(t) within the period $(0, k_1k_2c]$ which contradicts the fact that k_1 and k_2 are mutually prime integers. By the definition of $\delta(j)$, $\delta(j) = r_jc < T_1 = k_1c$; therefore $r_j \in \{1, 2, \cdots, k_1 - 1\}$ for j = 1, 2, $\cdots$, k_1. Together with $r_i \neq r_j$ for $i \neq j$, which was shown above, this implies that $\exists\, j \in \{1, 2, \cdots, k_1\}$ such that $r_j = 1$ so that $\delta(j) = c$. This implies that $\Delta_{sup} = c$.

Q. E. D.

<u>Lemma 4.2</u>: The "worst case" highest peak occurs when $t_{a1}^{i} = t_{a2}^{j}$ for some i, j, as exemplified in Fig. 4.4. The corresponding phase difference, denoted by Δ_{wc}, is given by:

$$\Delta_{wc} = c\{(d_1 - d_2) - [d_1 - d_2]\} \tag{4.25}$$

where [$\cdot$] function gives the largest integer smaller or equal to its real argument.

Proof: Let us say that at a point in time t_{wc} the arrivals of messages from the two sensor nodes coincide, i.e. $t_{wc} = t^i_{a1} = t^j_{a2}$ for some i, j. Then the mean-square error at the destination node just before their arrival, $p(t^-_{wc})$, is given by:

$$p(t^-_{wc}) = \Phi_m(a_{wc}) + \eta_m(a_{wc}, b_{wc}) \tag{4.26}$$

where Φ_m and η_m are given by (4.8) through (4.12), and:

$$m = \begin{cases} 1 & \text{if } T_1 + d_1 < T_2 + d_2 \\ 2 & \text{if } T_2 + d_2 < T_1 + d_1 \end{cases} \tag{4.27}$$

$$a_{wc} = |(T_1 + d_1) - (T_2 + d_2)| \tag{4.28}$$

$$b_{wc} = \min(T_1 + d_1, T_2 + d_2), \tag{4.29}$$

so that

$$a_{wc} + b_{wc} = \max(T_1 + d_1, T_2 + d_2) \tag{4.30}$$

Relationships (4.14) through (4.16) show the range of arguments a^i and b^i for the three possible types of peaks of p(t). From these relationships it can be deduced that, for all peaks p^i,

$$a^i + b^i \leq \max(T_1 + d_1, T_2 + d_2), \tag{4.31}$$

$$b^i \leq \min(T_1 + d_1, T_2 + d_2). \tag{4.32}$$

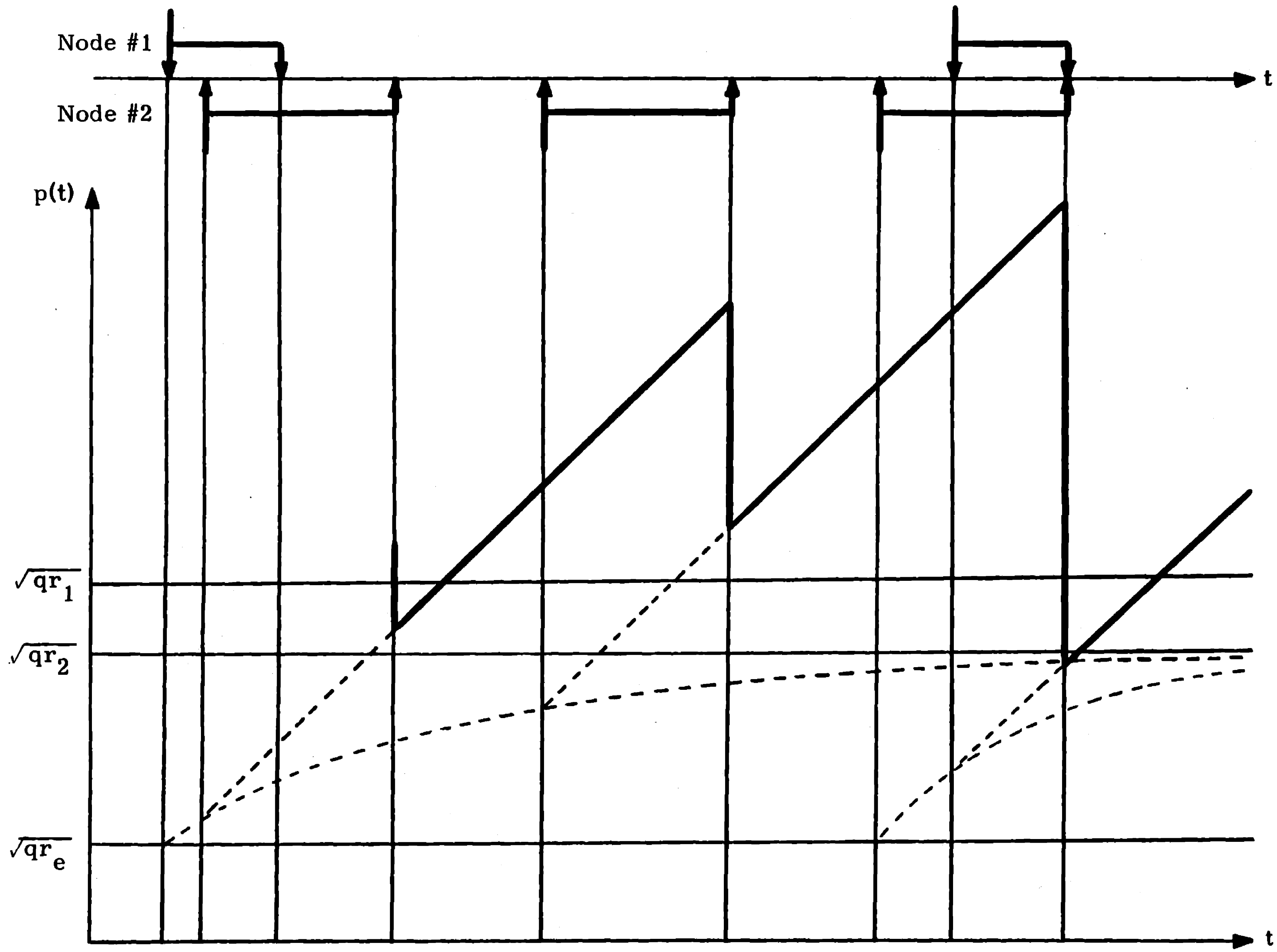

Fig. 4.4 Occurrence of the "worst-case" highest peak of the mean-square estimation error.

Thus the relationships (4.28) through (4.32) imply that b_{wc} is the maximum value that the second argument of the function η_m can take and that a_{wc} is the maximum value that the argument of the functions Φ_m can take given b_{wc}. Since Φ_m and η_m are monotonically increasing functions with η_m increasing faster than Φ_m with regard to its second argument, $p(\bar{t}_{wc}) = p_{wc}$.

The value of Δ_{wc} in (4.20) can be shown with arguments similar to those given in the proof of Lemma 4.1.

Q.E.D.

Lemma 4.3: Consider the following two intervals for Δ:

$$\text{Interval A:} \quad [0, \Delta_{wc}) \tag{4.33}$$

$$\text{Interval B:} \quad [\Delta_{wc}, c) \tag{4.34}$$

For a particular $\Delta_A \in$ Interval A, define:

$$p^{\ell_{A1}} = p(\bar{t}_{a1}{}^{A}) \tag{4.35}$$

$$p^{\ell_{A2}} = p(\bar{t}_{a2}{}^{A}) \tag{4.36}$$

where

$$\bar{t}_{a1}{}^{A} = \underset{t^i_{a1}:\ t^i_{a1} < t^j_{a2}}{\operatorname{argmin}} (t^j_{a2} - t^i_{a1}) \quad \forall\ i,j \tag{4.37}$$

$$\bar{t}_{a2}{}^{A} = \underset{t^j_{a2}:\ t^j_{a2} < t^i_{a1}}{\operatorname{argmin}} (t^i_{a1} - t^j_{a2}) \quad \forall\ i,j \tag{4.38}$$

and $\ell_{A1}, \ell_{A2} \in \{1, 2, \cdots, (k_1 + k_2)\}$.

Then,

$$\max_t p(t) = \begin{cases} p^{\ell_{A1}} \\ \text{or} \\ p^{\ell_{A2}} \end{cases} \tag{4.39}$$

for all values of $\Delta \varepsilon$ Interval A.

Defining $p^{\ell_{B1}}$ and $p^{\ell_{B2}}$ analogously for a particular $\Delta_B \varepsilon$ Interval B, we have:

$$\max_t p(t) = \begin{cases} p^{\ell_{B1}} \\ \text{or} \\ p^{\ell_{B2}} \end{cases} \tag{4.40}$$

for all values of $\Delta \varepsilon$ Interval B.

Proof: Assume $T_1 + d_1 > T_2 + d_2$. For $\Delta = \Delta_A$ in Interval A,

$$p^{\ell_{A1}} = \Phi_2(a^{\ell_{A1}}) + \eta_2(a^{\ell_{A1}}, b^{\ell_{A1}}), \tag{4.41}$$

with

$$a^{\ell_{A1}} + b^{\ell_{A1}} = T_1 + d_1. \tag{4.42}$$

Thus, $p^{\ell_{A1}}$ is a Type 3 peak as described by (4.16). From (4.5), (4.6), (4.35) and (4.37), $b^{\ell_{A1}}$ is the maximum value the second argument of η_2 can take at all Type 3 peaks for $\Delta = \Delta_A$. Since Φ_2 and η_2 are monotonically increasing functions with η_2 increasing faster than Φ_2 with respect to its second argument, $p^{\ell_{A1}}$ is the highest among Type 3 peaks for $\Delta = \Delta_A$.

This implies that $p^{\ell_{A1}}$ is the highest Type 3 peak for all values of Δ in Interval A. The values of the peaks vary continuously with respect to Δ in Interval A. Therefore, assuming that some peak p' is higher than $p^{\ell_{A1}}$ for some Δ' in Interval A, there must be Δ'' between Δ_A and Δ' where $p' = p^{\ell_{A1}}$. Then p' and $p^{\ell_{A1}}$ belong to different periods of p(t), with $p' = p^{\ell A1}$ for all Δ in Interval A.

Now, for $\Delta = \Delta_A$,

$$p^{\ell A2} = \Phi_2(a^{\ell A2}) + \eta_2(a^{\ell A2}, T_2 + d_2). \tag{4.43}$$

Thus, $p^{\ell A2}$ is a Type 1 peak as described by (4.14). From (4.5), (4.6), (4.36) and (4.38), $a^{\ell A2}$ is the maximum value the argument of Φ_2 can take at all Type 3 peaks for $\Delta = \Delta_A$. Since Φ_2 and η_2 are monotonically increasing functions with η_2 with respect to its first argument, $p^{\ell A2}$ is the highest among Type 1 peaks for $\Delta = \Delta_A$. Furthermore, it is clear from (4.15) that $p^{\ell A2}$ is higher than all Type 2 peaks which have $a^i + b^i = T_2 + d_2$, which is equal to the second argument of η_2 at $p^{\ell A2}$. Thus $p^{\ell A2}$ is the highest among Type 1 and Type 2 peaks for $\Delta = \Delta_A$.

Similarly as above, it can be shown that $p^{\ell A2}$ is the highest among Type 1 and Type 2 peaks for all values of Δ in Interval A. Hence (4.39) has been established.

For a particular $\Delta = \Delta_B$ in Interval B, it can be shown that no Type 1 peak exists, and that $p^{\ell B1}$ and $p^{\ell B2}$ are the highest among Type 3 and Type 2 peaks, respectively, and consequently that this is so for all values of Δ in Interval B, thus establishing (4.40).

For $T_1 + d_1 \leq T_2 + d_2$, the proof is similar.

Q.E.D.

The physical interpretation of this result is as follows. For Interval A, $p^{\ell A1}$ and $p^{\ell A2}$ are the two "candidates" for being the highest peak: so are $p^{\ell B1}$ and $p^{\ell B2}$ for Interval B. Now the stated result says that once we find the candidate peaks for a particular value of Δ in the interval, they stay as candidate peaks for all values of Δ inside that interval. Hence we only have to consider the values of these 2 peaks rather than $(k_1 + k_2)$ peaks.

Lemma 4.4: Either the optimal phase difference $\Delta = \Delta^*$ is zero, or at Δ^*, $p^{\ell_{A1}} = p^{\ell_{A2}}$ if $\Delta^* \in$ Interval A and $p^{\ell_{B1}} = p^{\ell_{B2}}$ if $\Delta^* \in$ Interval B.

Proof: From (4.5), (4.6), (4.35), (4.36), (4.41) and (4.42) it can be shown that $p^{\ell_{A1}}$ decreases monotonically as a function of Δ in Interval A and similarly that $p^{\ell_{A2}}$ increases monotonically with $p^{\ell_{A2}} = p_{wc} > p^{\ell_{A1}}$ at Δ_{wc}. Hence either $p^{\ell_{A2}} > p^{\ell_{A1}}$ for all $\Delta \in$ Interval A, or $p^{\ell_{A2}} = p^{\ell_{A1}}$ at a point in Interval A. Similar facts are true for Interval B. Hence the claim of the lemma follows from (4.39) and (4.40).

Q.E.D.

Now we can summarize the optimization procedure.

Proposition 4.1: Minimization of the maximum peak value of p(t) with respect to T_1, T_2 and Δ can be achieved by the following algorithm.

Step 1: Find T_1^* and T_2^* by (4.17) and (4.18).

Step 2: Find Intervals A and B by (4.19), (4.25), (4.32) and (4.34).

Step 3: Using $\Delta_A = \Delta_{wc}/2$ and $\Delta_B = (\Delta_{wc} + c)/2$, find the candidate peaks with indices ℓ_{A1}, ℓ_{A2}, ℓ_{B1} and ℓ_{B2}.

Step 4: In Interval A, find the phase difference Δ_A^* for which $p^{\ell_{A1}} = p^{\ell_{A2}} \triangleq p_A^*$. If $p^{\ell_{A2}} > p^{\ell_{A1}}$ everywhere in Interval A, then set $\Delta_A^* = 0$ and $p_A^* = p^{\ell_{A2}}$ for $\Delta = 0$.

Carry out the same procedure for Interval B also, and find Δ_B^* and p_B^*.

Step 5:

$$\Delta^* = \begin{cases} \Delta_A^* & \text{if} \quad p_A^* \le p_B^* \\ \Delta_B^* & \text{if} \quad p_A^* > p_B^* \end{cases} . \tag{4.44}$$

4.2.3 Worst-Case Δ Optimization with Respect to T_1 and T_2

Adopting a minimax approach and optimizing with respect to T_1 and T_2 for the phase difference Δ_{wc} which gives rise to the highest possible peak simplifies the optimization procedure significantly.

The situation when the worst case peak occurs was depicted in Fig. 4.4 and the corresponding Δ_{wc} was given by (4.25). The value of the worst-case peak can be written as:

$$p_{wc} = \Phi_m(a) + \eta_m(a,b)$$

where:

$$m = \begin{cases} 1 & \text{if } T_1 + d_1 < T_2 + d_2 \\ 2 & \text{if } T_1 + d_1 > T_2 + d_2 \end{cases}$$

$$a = |(T_1 + d_1) - (T_2 + d_2)|$$

$$b = \min\,[T_1 + d_1,\ T_2 + d_2]. \tag{4.45}$$

Now, η_m increases faster than Φ_m with respect to its second argument, and $a + b = \max\,[T_1 + d_1,\ T_2 + d_2]$ = fixed. Hence minimizing p_{wc} involves first minimizing b, then a. Thus min $p_{wc} \Rightarrow \min b = \min\,[T_1 + d_1,\ T_2 + d_2]$ first. Then $\min a = |(T_1 + d_1) - (T_2 + d_2)|$; hence min max $[T_1 + d_1,\ T_2 + d_2]$. Therefore we have to minimize $T_1 + d_1$ and $T_2 + d_2$ individually. Again we have assumed the independence of d_1 and d_2.

This result is summarized in the following.

Proposition 4.2: Minimization of the highest peak of p(t) for the worst-case phase difference Δ, i.e. minimization of $\max_{\Delta,t}$ p(t) with respect to T_1 and T_2 is equivalent to the minimization of $T_1 + d_1(T_1)$ with respect to T_1 and $T_2 + d_2(T_2)$ with respect to T_2.

4.3 A Restricted Routing Solution to the Worst-Case Minimax Optimization Problem in a Sensor Network

4.3.1 The Problem Description

In this section we are going to consider a network with N sensor nodes and a destination node. Any given node #i has a set of neighbors $\mathcal{O}(i)$ for which links (i, j), $j \in \mathcal{O}(i)$ exist. Delay on link (i, j) is assumed to be a function of the traffic rate on that link only: $d_{ij} = f_{ij}(T_{ij})$, where d_{ij} is the delay per message and T_{ij} is the intermessage period on link (i, j).

We will consider a restricted class of routing schemes in this section. Specifically, any given node will be allowed to send messages to only one of its neighbors, for all time; and these messages will be sent periodically. The general case is considered in Section 4.4.

There is assumed to be one destination node, and the objective is to minimize the maximum value over time, or the highest peak, of the mean-square state estimation error at the destination node. We will optimize for the worst-case timing relationship that can possibly exist between the reporting times of the nodes, when one set of messages for all the nodes arrives at the destination node simultaneously with the greatest total delay. Therefore, here the worst possible mean-square error with given T_{ij}, hence d_{ij} values is implied. This worst case, by extension of the result of Case (2) of Section 4.2.2, will occur when the messages arrive at node #j almost at the same time, but only slightly later than the departure time of a message from node #j, for all nodes j. In this case the messages which have just arrived have to wait at node #j for the interreporting period of this node.

For all schemes of routing, it is assumed that optimal data fusion is employed at the intermediate nodes, described in Chapter 3.

4.3.2 The Algorithm

Proposition 4.3: The algorithm for minimizing $p_{wc} = \max_t p(t)$ at the destination node for the worst-case timing relationship between the reporting times of the nodes is as follows.

1. Minimize $T_{ij} + d_{ij}(T_{ij})$ for all links (i, j). Let $T_{ij}^* \triangleq \operatorname{argmin} T_{ij} + d_{ij}(T_{ij})$.

2. Find the shortest path spanning tree $\mathcal{T}$, using $T_{ij}^* + d_{ij}(T_{ij}^*)$ as the "length" of link (i, j).

3. Let node #i send its messages on link (i, j), $(i, j) \in \mathcal{T}$, with a period of T_{ij}^*. Do this for all nodes of the network.

Proof: The worst-case highest peak will occur at time $t' = t_{a1}^i = t_{a2}^j = \cdots = t_{aN}^z$, for some i, j, $\cdots$, z, according to the timing notation of Section 4.2. For any algorithm which conforms to the restrictions we have posed, let node #i send its messages periodically with period T_{ij} on link (i, j), and let this message be routed over intermediate nodes #i, #j, #k, $\cdots$, #y, #z. Define:

$$D_i \triangleq (T_{ij} + d_{ij}) + (T_{jk} + d_{jk}) + \cdots (T_{yz} + d_{yz}). \tag{4.46}$$

Rename the nodes so that $D_i \geq D_{i+1}$ for $1 \leq i \leq N$. Then the value of the worst-case highest peak, p_{wc}, can be written as:

$$p_{wc} = \psi_N (D_{N-1} - D_N) + \lambda(D_N) \tag{4.47}$$

$$\psi_m(t) = \text{trace } P_m(t), \quad m = 2, \cdots, N \tag{4.48}$$

$$\dot{P}_m(t) = AP_m(t) + P_m(t)A' + BQB' - \sum_{k=m}^{N} P_m(t)\, C_k' \, R_k^{-1} \, C_k \, P_m(t) \qquad (4.49)$$

$$P_m(0) = \begin{cases} P_{m-1}\,(D_{m-2} - D_{m-1}) & m = 3, \cdots, N \\ \overline{P} & m = 2 \end{cases} \qquad (4.50)$$

$$0 = A\overline{P} + \overline{P}A' + BQB' - \sum_{k=1}^{N} \overline{P} C_k' \, R_k^{-1} \, C_k \overline{P} \qquad (4.51)$$

$$\lambda(D_N) = \text{trace } S(D_N) \qquad (4.52)$$

$$\dot{S}(t) = AS(t) + S(t)A' + BQB' \qquad (4.53)$$

$$S(0) = P_N\,(D_{N-1} - D_N) \qquad (4.54)$$

Now, $\lambda(t) > \psi_m(t)$ with $\lambda(0) = \psi_m(0)$ for all t, $m = 1, \cdots, N$ and $\psi_k(t) > \psi_\ell(t)$ with $\psi_k(0) = \psi_\ell(0)$ for all t, $k > \ell$. Since D_m's can be minimized independently (due to independence of delay from traffic on different links), minimizing ρ_{wc} requires minimizing first D_N, then $D_{N-1} - D_N$, hence D_{N-1}; then $D_{N-2} - D_{N-1}$, hence D_{N-2}, and so on. Therefore, all D_1's have to be minimized independently, which can be achieved by the algorithm as stated above.

Q.E.D.

4.4 A General Routing Approach to the Worst-Case Minimax Optimization Problem in a Sensor Network

4.4.1 Introduction

In this section, we will lift the restriction that a given node can send messages to only one other node. The nodes will be allowed to send messages over several different paths to the destination, utilizing some channel capacity which may have been left unused by the routing solution of the previous section. This way a node will be able to send information more frequently with the same effective delay or, alternatively, the same amount of information with less delay.

As explained in Section 2.3, in Sections 4.4.2 and 4.4.3, the A matrix in the dynamical system model is assumed to be multivariable with the structure indicated in (2.30). This will insure that the mean-square error is a monotonically increasing function of time for the optimization problems considered in these sections.

We have seen in the last section that the worst-case highest peak of the mean-square error occurs at the time instance when the messages from all the nodes happen to arrive simultaneously. This is because just before these messages all arrive, the destination node has not received any information from all the nodes for the maximum possible duration of time; specifically, for $T_i + d_i$ units from node #i, where d_i is the total delay to the destination. We will call this duration the "effective delay" from node #i, and denote it by D_i.

When we allow the messages to be routed over multiple paths, the effective delays on different paths will in general be different. If we label the departure and arrival times of the messages in the order they depart, as shown in Fig. 4.5 (for k = 3), then the effective delay from node #i will be:

$$D_i = \max_j \left(t_a^j - \max_{t_a^k < t_a^j} t_d^k \right) \quad \text{for all } j, k. \qquad (4.55)$$

Now, by rearranging the departure times on the paths while holding the average departure frequencies constant and also adjusting the

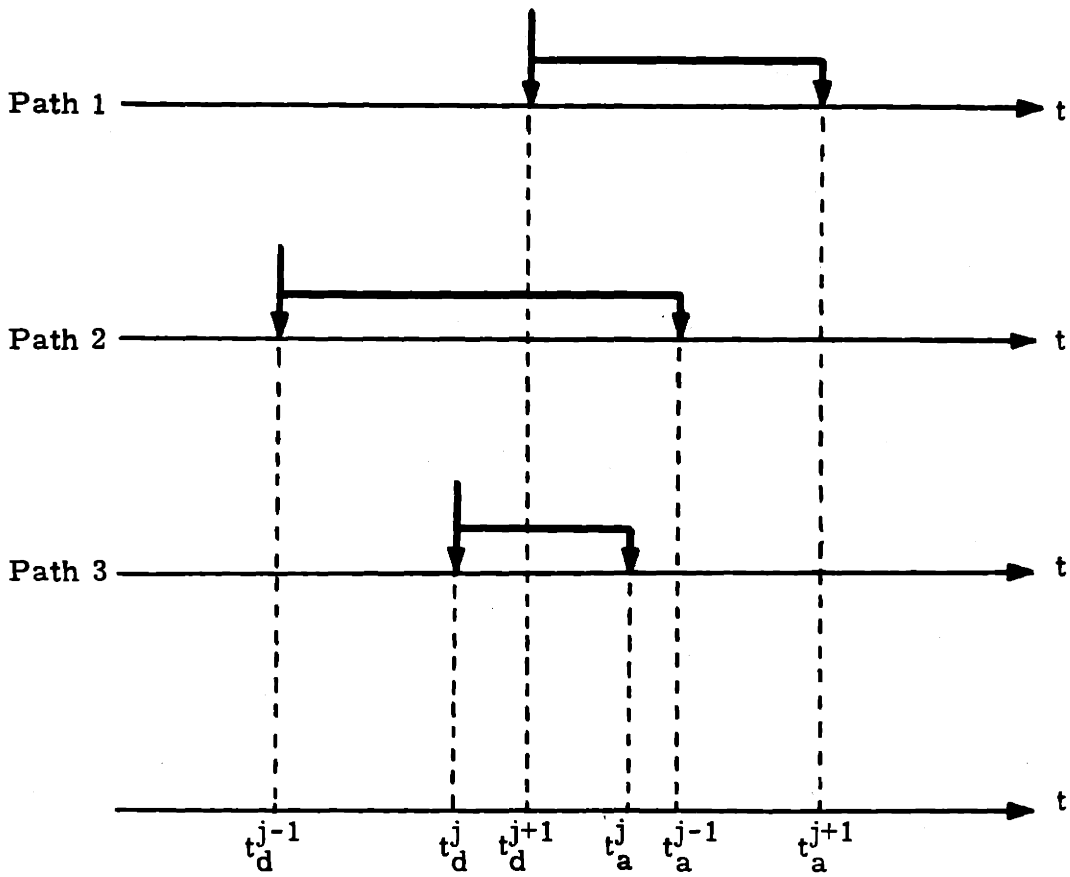

Fig. 4.5 Departure and arrival times of the messages on different paths from the sensor node to the destination node.

relative timings between the paths, we can reduce D_i. Obviously, the minimum will be achieved when all $(t_a^j - \max_{t_a^k < t_a^j})$ effective delays are equal. An example of what the mean-square error looks like for a simple example of a single node with two alternate paths on which the timings have been optimally adjusted is shown in Fig. 4.6. The observed system is assumed to be modelled as a scalar Wiener process.

In Section 4.4.2, we will derive the effective delay (D) for the optimum adjustment of the departure times on the paths from a single node, given fixed values of T^j and d^j associated with the paths. In Section 4.4.3, we will consider minimizing D with respect to T^j. In Section 4.4.4, we will address the problem of minimizing p_{wc}, the worst-case maximum peak value of the mean-square error for the general network case.

4.4.2 Effective Delay from a Single Node

Let us assume that there are L disjoint paths from a single node to the destination; average frequency of departures on path j is $(T^j)^{-1}$ and the total delay on that path is d^j. Let's also assume that there exist mutually prime integers $k^1, k^2, \cdots, k^L$ such that:

$$\frac{(T^1)^{-1}}{k^1} = \frac{(T^2)^{-1}}{k^2} = \cdots = \frac{(T^L)^{-1}}{k^L} \tag{4.56}$$

As explained in Section 4.2.2, this assumption is not an unreasonable one, since the set of rationals is dense in the set of reals.

Let us first state the main result of this section, then prove it.

Proposition 4.4: The effective delay, D, from a single node for the optimum adjustment of the departure times on disjoint paths $1, 2, \cdots, L$ having associated average departure frequencies $(T^j)^{-1}$ and

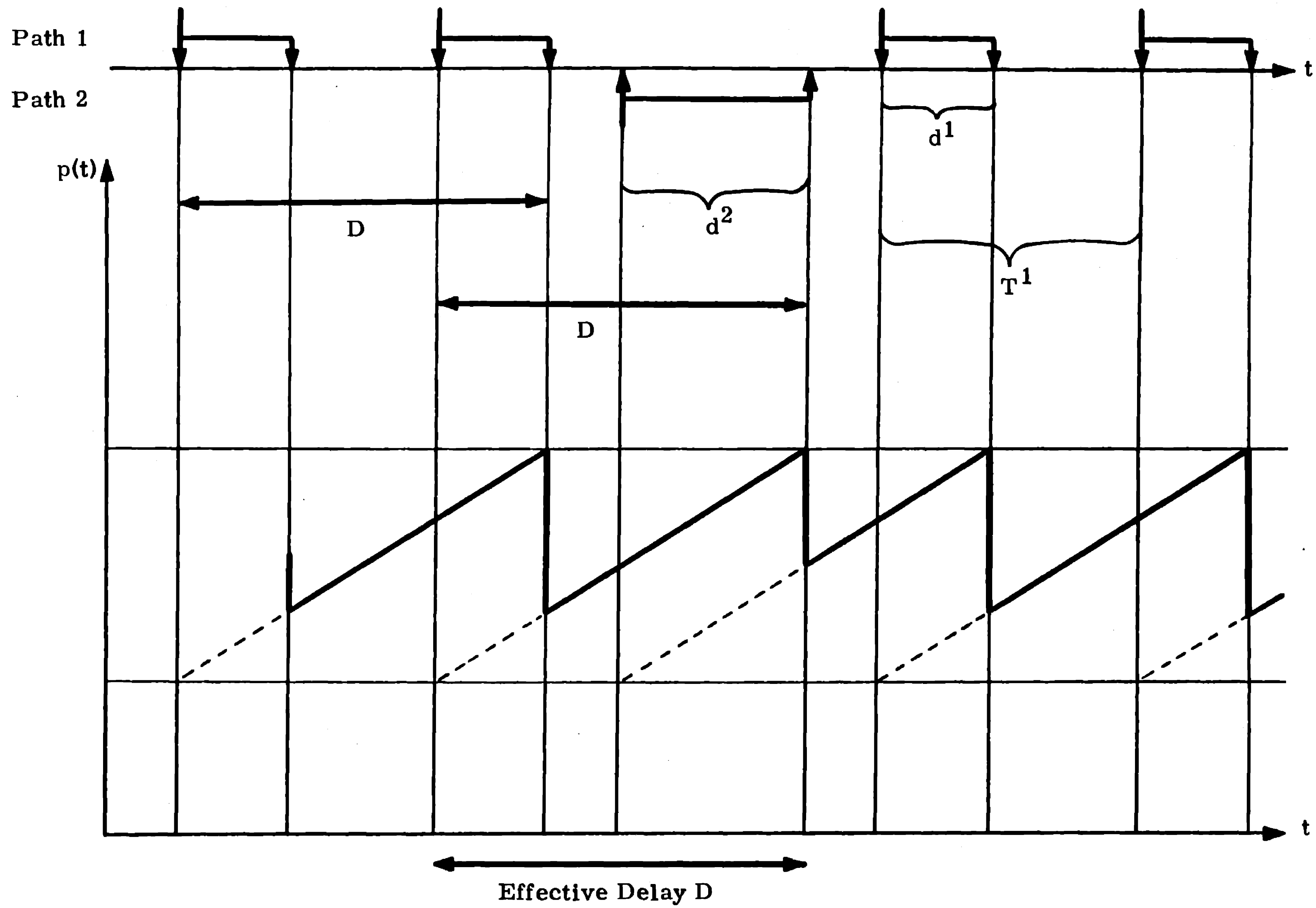

Fig. 4.6 Effective delay D from a single sensor node with two paths to the destination node.

delays d^j for path j, is given by:

$$D = D(V) \qquad (4.57)$$

where D(A) is defined on a set A of paths as:

$$D(A) \triangleq \left[\sum_{j \in A} (T^j)^{-1}\right]^{-1} \left[1 + \sum_{j \in A} \frac{d^j}{T^j}\right] \qquad (4.58)$$

The set V is defined as:

$$V = \bigcup Y, \text{ where sets Y have the property:}$$

$$Y = \left\{j \in \{1, 2, \cdots, L\} : d^j < D(Z) \ \forall \ Z \subset \{1, 2, \cdots, L\}\right\}.$$

$$(4.59)$$

Proposition 4.5: The set V of Proposition 4.4 can be efficiently constructed by the following algorithm, written in Pseudo-Algol:

```
        Begin
Step 1: A: = {1, 2, ..., L};
        V: = Φ;
Step 2: For j = 1 thru L, j ε A do
        If d^j < D (A - {j})
             then V: = V ∪ {j};
Step 3: If V = A
             then Stop;
             else Begin
                        A: = V;
                        V: = Φ;
                        Go to   Step 2
                  End;
End.
```

Before proving these two propositions, let us introduce some preliminary results.

Lemma 4.5: Let A and B be two disjoint sets of paths, and let each path j have associated T^j and d^j values. Then one and only one of the following three cases holds:

(a) $$\bar{d}_B < D(A \cup B) < D(A) \tag{4.60}$$

(b) $$\bar{d}_B > D(A \cup B) > D(A) \tag{4.61}$$

(c) $$\bar{d}_B = D(A \cup B) = D(A) \tag{4.62}$$

Here D(A) is defined on set A as in (4.58) and $\bar{d}_B$ is defined as:

$$\bar{d}_B \triangleq \sum_{j \in B} \frac{(T^j)^{-1}}{\sum_{i \in B} (T^i)^{-1}} d^j \tag{4.63}$$

Proof: By trichotomy, there are 3 distinct possibilities:

(a') $$D(A \cup B) < D(A) \tag{4.64}$$

(b') $$D(A \cup B) > D(A) \tag{4.65}$$

(c') $$D(A \cup B) = D(A) \tag{4.66}$$

We will prove that (a') is equivalent to (a). The other cases will follow identically with changes in inequality and equality signs throughout the proof.

The proof will be in two stages:

(i) $$D(A \cup B) < D(A) \Longleftrightarrow \bar{d}_B < D(A) \tag{4.67}$$

(ii) $$\bar{d}_B < D(A \cup B) \Longleftrightarrow \bar{d}_B < D(A) \tag{4.68}$$

Proof of Part (i): Let's assume that $A = \{1, 2, \cdots, M\}$ and $B = \{M+1, M+2), \cdots, N\}$.

$$D(A \cup B) < D(A)$$

$$\Leftrightarrow \left[\sum_{j=1}^{N} (T^j)^{-1}\right]^{-1} \left[1 + \sum_{j=1}^{N} \frac{d^j}{T^j}\right]$$

$$< \left[\sum_{j=1}^{M} (T^j)^{-1}\right]^{-1} \left[1 + \sum_{j=1}^{M} \frac{d^j}{T^j}\right]$$

$$\Leftrightarrow \left(\sum_{j=1}^{M} \frac{\prod_{i=1}^{M} T^i}{T^j}\right) \left(\sum_{j=M+1}^{N} \frac{\prod_{i=M+1}^{N} T^i}{T^j} d^j\right)$$

$$< \left[\left(\prod_{j=1}^{M} T^j\right)\left(\sum_{j=1}^{N} \frac{\prod_{i=1}^{N} T^j}{T^j}\right) - \left(\prod_{j=1}^{N} T^j\right)\left(\sum_{j=1}^{M} \frac{\prod_{i=1}^{M} T^i}{T^j}\right)\right]$$

$$\cdot \left[1 + \sum_{j=1}^{M} \frac{d^j}{T^j}\right]$$

$$\Leftrightarrow \left[\sum_{j=M+1}^{N} (T^j)^{-1}\right]^{-1} \sum_{j=M+1}^{N} \frac{d^j}{T^j}$$

$$< \left[\sum_{j=1}^{M} (T^j)^{-1}\right]^{-1} \left[1 + \sum_{j=1}^{M} \frac{d^j}{T^j}\right]$$

$$\Leftrightarrow \bar{d}_B < D(A)$$

Part (ii) can be proved in a similar way.

Q. E. D.

Proof of Proposition 4.4: Let us first introduce just one more variable for added clarity. Define $\bar{D}(A)$ as the effective delay from the sensor node for the optimum adjustment of departure times on the paths which are elements of set A. Thus the claim of the proposition is that $D \triangleq \bar{D}(\{1, 2, \cdots, L\}) = D(V)$.

Suppose we are given the path set V as $V \triangleq \{v^1, v^2, \cdots, v^M\}$, satisfying (4.59). Reconstruct V as follows:

$$W^1 = \{v^1\} \tag{4.69}$$

$$W^n = W^{n-1} \cup \{v^n\}, \quad 2 \le n \le M. \tag{4.70}$$

We will show that:

$$\bar{D}(W^n) = D(W^n), \quad 1 \le n \le M, \tag{4.71}$$

and that:

$$\bar{D}(W^n) < \bar{D}(W^{n-1}), \quad 2 \le n \le M. \tag{4.72}$$

Then we will show that:

$$D(V) \le D(V \cup \{j\}), \tag{4.73}$$

$$\bar{D}(V) = \bar{D}(V \cup \{j\}) \ \forall\, j \notin V,\ j \in \{1, 2, \cdots, L\}. \tag{4.74}$$

This will complete the proof.

Let us consider a single period of p(t), of length $k^1T^1 = k^2T^2 = \cdots = k^LT^L$. We will show (4.71) and (4.72) for W^1 and W^2. The other stages will be similar for $3 \le n \le M$. For $v^1 \in W^1$, the minimum effective delay occurs for the equal spacing of departure times, and has the value:

$$\bar{D}(W^1) = T^{v^1} + d^{v^1} = D(W^1). \tag{4.75}$$

This is well known from the previous sections.

By (4.56), there will be k^{v^1} messages on path v^1 and k^{v^2} messages on path v^2 in a period of p(t). Divide path v^2 into k^{v^2} separate paths, $\bar{1}, \bar{2}, \cdots \bar{k}^{v^2}$, with period $k^{v^2} T^{v^2}$ and delay d^{v^2}. Consider first the set $W^1 \cup \{\bar{1}\}$. By (4.59) and (4.75), $d^{v^2} < \bar{D}(W^1)$. From Fig. 4.7 it can be observed that in this case the effective delay can be reduced by readjustment of departure times on paths v^1 and $\bar{1}$. Therefore $\bar{D}(W^1 \cup \{\bar{1}\}) < \bar{D}(W^1)$. Now calculate $\bar{D}(W^1 \cup \{\bar{1}\})$. From Fig. 4.7(b), in the interval $[0, k^1T^1 + d^{v^1}]$ there are $k^{v^1} + 1$ overlapping subintervals of length $\bar{D}(W^1 \cup \{\bar{1}\})$. Thus writing the entire length in terms of these subintervals minus the overlap durations,

$$(k^{v^1} + 1)\bar{D} - (k^{v^1} - 1)d^{v^1} - d^{v^2} = k^{v^1}T^{v^1} + d^{v^1} \tag{4.76}$$

$$\Longrightarrow \bar{D}(W^1 \cup \{\bar{1}\}) = \frac{k^{v^1}}{k^{v^1}+1}(T^{v^1} + d^{v^1}) + \frac{1}{k^{v^1}+1} d^{v^2}$$

$$= \frac{1}{(T^{v^1})^{-1} + (k^{v^2}T^{v^2})^{-1}} \left(1 + \frac{d^{v^1}}{T^{v^1}} + \frac{d^{v^2}}{k^{v^2}T^{v^2}}\right)$$

$$= D(W^1 \cup \{\bar{1}\}). \tag{4.77}$$

Then, by Lemma 4.5:

$$d^{\bar{2}} = d^{v^2} < D(W^1 \cup \{\bar{1}\}) = \bar{D}(W^1 \cup \{\bar{1}\}) \tag{4.78}$$

and similarly as above:

$$\bar{D}(W^1 \cup \{\bar{1}, \bar{2}\}) < \bar{D}(W^1 \cup \{\bar{1}\}). \tag{4.79}$$

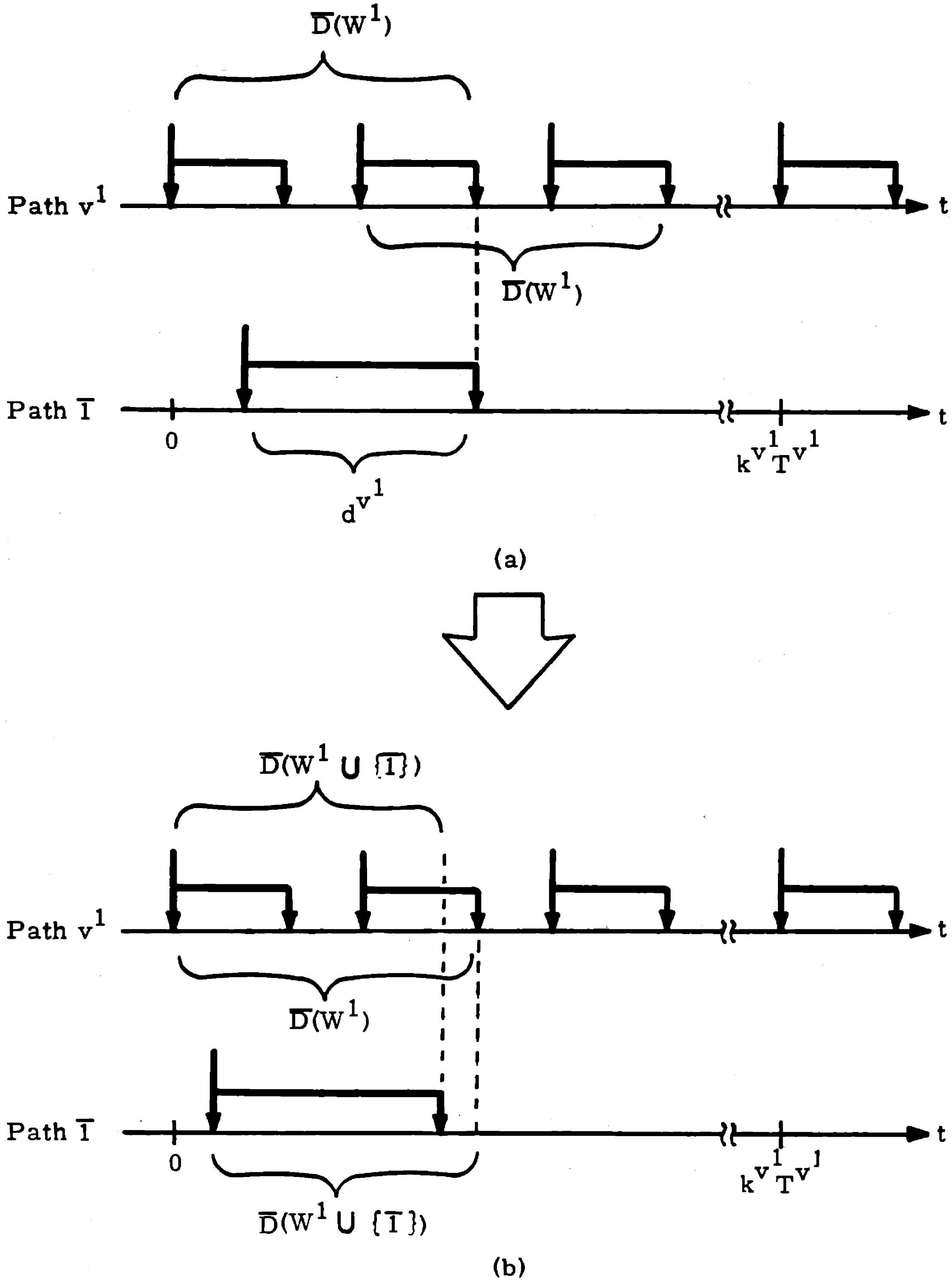

Fig. 4.7 Reduction of the effective delay by readjustment of departure times.

Repeating the procedure gives:

$$\overline{D}(W^2) = D(W^2) \quad \text{and} \quad \overline{D}(W^2) < D(W^1) \tag{4.80}$$

Therefore we have shown (4.71) and (4.72) for W^1 and W^2. For the other stages the proof is entirely similar. Given $d^{v^n} < D(W^{n-1})$, we can show every new message up to number k^{v^n} on path v^n reduces the effective delay. $\overline{D}(W^n)$ can also be calculated as in (4.76), subtracting the overlap durations from the overlapping intervals of $\overline{D}(W^n)$:

$$\left(\sum_{i=1}^{n} k^{v^i}\right)\overline{D}(W^n) - (k^{v^1} - 1)d^{v^1} - \sum_{i=2}^{n} k^{v^i} d^{v^i} = k^{v^1}T^{v^1} + d^{v^1} \tag{4.81}$$

$$\begin{aligned}
\Longrightarrow \overline{D}(W^n) &= \left(\sum_{i=1}^{n} k^{v^i}\right)^{-1}\left[k^{v^1}\left(T^{v^1} + d^{v^1}\right) + \sum_{i=2}^{n} k^{v^i} d^{v^i}\right] \\
&= \left[\sum_{i=1}^{n}\left(T^{v^i}\right)^{-1}\right]^{-1}\left[\left(T^{v^1}\right)^{-1}\left(T^{v^1} + d^{v^1}\right) + \sum_{i=2}\left(T^{v^i}\right)^{-1} d^{v^i}\right] \\
&= \left[\sum_{i=1}^{n}\left(T^{v^i}\right)^{-1}\right]^{-1}\left[1 + \sum_{i=1}^{n}\frac{d^{v^i}}{T^{v^i}}\right] \\
&= D(W^n)
\end{aligned} \tag{4.82}$$

Now we will prove (4.73) and (4.74), which can be interpreted as saying that set V is the largest set of "useful paths". Consider a path $j \notin V \Rightarrow \exists\, A \subset \{1, 2, \cdots, L\}$ such that $d^j \geq D(A)$. Define G as the set of such sets A:

$$G \triangleq \left\{A \subset \{1, 2, \cdots, L\} : d^j \geq D(A)\right\} \tag{4.83}$$

Let B be the set which satisfies:

$$D(B) = \min_{A \in G} D(A) \tag{4.84}$$

If B = V, then $d^j \leq D(V) \Longrightarrow D(V \cup \{j\} \leq D(V)$ from Lemma 4.5, giving (4.73).

If $B \neq V$, then write B as $B = B_v \cup \overline{B}_v$, where $B_v = \{i \in B : i \in V\}$, $\overline{B}_v = \{i \in B : i \notin V\}$. If $\overline{B}_v = \Phi$, $d^j \leq D(B) = D(B_v) \leq D(V)$ by the already proved (4.71) and (4.72). So in fact, B = V if $\overline{B}_v = \Phi$.

If $\overline{B}_v \neq \Phi$, then there exist two possibilities by Lemma 4.5:

$$\text{(i)} \quad \overline{d}_{\overline{B}_v} \geq D(B_v \cup \overline{B}_v) = D(B) \geq D(B_v) \tag{4.85}$$

$$\text{(ii)} \quad \overline{d}_{\overline{B}_v} < D(B) < D(B_v) \tag{4.86}$$

For case (i), $d^j \geq D(B) \geq D(B_v) \geq D(V)$.

For case (ii), $\overline{d}_{\overline{B}_v} < D(B) \Rightarrow \exists$ path $k \in \overline{B}_v$ such that $d^k < D(B)$. Since $k \notin V$, $\exists$ set C such that $d^k > D(C) \Longrightarrow D(B) > D(C)$, contradicting (4.84). Thus case (ii) is impossible.

Therefore we have proved that (4.73) for all cases. (4.74) follows directly from (4.73). $D(V) = \overline{D}(V)$ and $d^j \geq D(V)$; therefore $d^j \geq \overline{D}(V)$. The delay on path j is greater than the effective delay achieved by set V; therefore, inclusion of a message on path j in a period of p(t) cannot reduce the effective delay no matter at which point in the period it is placed. (4.74) states this fact. Q.E.D.

<u>Proof of Proposition 4.5</u>: This algorithm picks out the elements of $\{1, 2, \cdots, L\}$ which are not in V. We will show that at each iteration of the algorithm at least one such element is detected, unless all have been already detected.

At the n^{th} iteration, A is the set of paths which haven't been eliminated in the first n-1 iterations. If A = V, then $d^j < D(A - \{j\})$ $\forall j \in A$ will be indicated. Otherwise, write A as $A = V \cup \overline{V}$, where $\overline{V} = \{j \in A : j \notin V\}$. Let $\overline{V} = \{\overline{1}, \overline{2}, \cdots, \overline{M}\}$. We will show that

$\exists\ j \in \overline{V}$ such that $d^j < D(A - \{j\})$. If $d^j > D(A - \{j\})$ for some $j \in \{\bar{1}, \bar{2}, \cdots, \overline{M-1}\}$, we're done. If $d^j < D(A - \{j\})\ \forall\ j \in \{\bar{1}, \bar{2}, \ldots, \overline{M-1}\}$, we have to show that $d^{\overline{M}} > D(A - \{\overline{M}\})$. We have:

$$D(V \cup [\overline{V} - \{\bar{k}\}]) \;>\; d^{\bar{k}} \;\; \forall\ \bar{k} \in \{\bar{1}, \bar{2}, \cdots, \overline{M-1}\}, \tag{4.87}$$

Then from Lemma 4.5:

$$\bar{d}_{[\overline{V} - \{\bar{k}\}]} > D(V \cup [\overline{V} - \{\bar{k}\}]) > d^{\bar{k}} \;\; \forall\ \bar{k} \in \{\bar{1}, \bar{2}, \cdots, \overline{M-1}\}, \tag{4.88}$$

since $\bar{d}_{[\overline{V} - \{\bar{k}\}]} > D(V)$. Hence:

$$\exists\ j \in (\overline{V} - \{\bar{k}\}) \text{ such that } d^{\bar{j}} > d^{\bar{k}} \;\; \forall\ \bar{k} \in \{\bar{1}, \bar{2}, \cdots, \overline{M-1}\}, \tag{4.89}$$

which implies that $d^{\overline{M}} > d^{\bar{k}}\ \forall\ \bar{k} \in \{\bar{1}, \bar{2}, \cdots, \overline{M-1}\}$. Hence:

$$d^{\overline{M}} > \bar{d}_{\overline{V}} > D(V \cup \overline{V}) \;=\; D(A) > D(A - \{\overline{M}\}) \tag{4.90}$$

by using Lemma 4.5 twice.

Q.E.D.

4.4.3 Minimization of Effective Delay D from a Single Node with Respect to Average Departure Frequencies on Paths to Destination

Proposition 4.4 states that the effective delay D from a single node for the optimum adjustment of the departure times on the paths to the destination is given by:

$$D \;=\; \bar{T}_V + \bar{d}_V \tag{4.91}$$

where $\bar{T}_V$ is the inverse of the sum of the average departure frequencies on the paths, or it is the "effective period"; and $\bar{d}_V$ is the weighted average of the delays on the paths, or it is the "effective communication delay", not to be confused with the "effective (information) delay" D, which includes the holding of information at the originating node. The subscript V indicates that $\overline{T}$ and $\overline{d}$ are defined on a set of V of "useful paths". For fixed given values of T^j and d^j for path j, set V can be constructed by the algorithm given by Proposition 4.5.

In this section, the minimization of D with respect to T^j will be investigated. For an algorithm is used to minimize D = D(V) as in (4.57), at each iteration the set V has to be found by using the algorithm of Proposition 4.5, which would make the computational complexity very high.

Rather than minimizing a functional that depends on a set which may vary every iteration, we will attempt to use a functional which depends on the fixed path set $A \triangleq \{1, 2, \cdots, L\}$, the set of all disjoint paths to the destination. Specifically, minimize the functional D(A) = $\overline{T}_A + \overline{d}_A$. Then obviously the value of the functional D(A) will not be the same as if we were using D(V), but if the optimal values of both functionals are the same, this approach will be correct. The following proposition summarizes some properties of this optimization approach.

Proposition 4.6: D^*, the optimal value of the effective delay from a node, minimized with respect to average departure frequencies and departure times on the paths to the destination is given by:

$$D^* = D(A^*) \triangleq \min_{T^j} D(A) = D(V^*) \triangleq \min_{T^j} D(V) \qquad (4.92)$$

where D(A) is defined in (4.58), V in (4.59) and $A \triangleq \{1, 2, \cdots, L\}$.

Furthermore, considering $D(A) = D(T^1, T^2, \cdots, T^L)$ as a surface defined on $\mathbb{R}^L$, define the following curves:

$$\pi(T^k; T^1, T^2, \cdots, T^{k-1}, T^{k+1}, \cdots, T^L) \triangleq D(A) \cap H^k(T^1, \cdots, T^{k-1}, T^{k+1}, \cdots, T^L) \quad (4.93)$$

where

$$H^k = \{(x_1, x_2, \cdots, x_{k-1}, x_{k+1}, \cdots, x_L) \in \mathbb{R}^{L-1} : x_j = T^j, j = 1, 2, \cdots, k-1, k+1, L\} \quad (4.94)$$

is a hyperplane in $\mathbb{R}^{L-1}$. Then $\pi(T^k; T^1, T^2, \cdots, T^{k-1}, T^{k+1}, \cdots, T^L)$ is a function of T^k which has one stationary point and at most one inflection point for $k = 1, \cdots, L$ and for all $T^j \geq (C^j)^{-1}$, $j = 1, 2, \cdots, k-1, k+1, \cdots, L$, where C^j is the channel capacity associated with path j. Furthermore, D(A) is a convex function of T^j in the neighborhood of its minimum.

Proof: All the indices run from 1 to L except those noted otherwise. $d^{k'}$ and $d^{k''}$ denote first and second derivatives of d^k with respect to T^k.

$$\frac{d\pi}{dT^k} = \frac{\partial D(A)}{\partial T^k}$$

$$= \frac{\left(\prod_{i \neq k} T^i\right)}{\left(\sum_i \prod_{j \neq i} T^j\right)^2} \left[\left(\sum_i \prod_{j \neq i} T^j\right) d^{k'} + \left(\prod_{i \neq k} T^k\right)\left(1 + \sum_{i \neq k} \frac{d^i}{T^i}\right) - \left(\sum_{i \neq k} \prod_{j \neq i, k} T^j\right) d^k \right] \quad (4.95)$$

$$\frac{d^2\pi}{dT^{k^2}} = \frac{\partial^2 D(A)}{\partial T^{k^2}}$$

$$= \frac{\left(\prod_{i\neq k} T^i\right)}{\left(\sum_i \prod_{j\neq i} T^j\right)^3} \left\{ \left(\sum_i \prod_{j\neq i} T^j\right)^2 d^{k''} - 2\left(\sum_i \prod_{j\neq i} T^j\right) d^{k'} \right.$$

$$\left. -2\left[\left(\prod_{i\neq k} T^i\right)\left(1 + \sum_{i\neq k} \frac{d^i}{T^i}\right) - \left(\sum_{i\neq k} \prod_{j\neq i,k} T^j\right) d^k\right]\right\} \tag{4.96}$$

Let $T^{1^*}, T^{2^*}, \cdots, T^{L^*}$ be the optimum T^j values, i.e.:

$$\min D(A) \triangleq D(A^*) = D(T^{1^*}, T^{2^*}, \cdots, T^{L^*}). \tag{4.97}$$

To prove (4.92), we will show that:

$$T^{k^*} = \infty \;\forall\; k \in A^*, \quad k \notin V^*. \tag{4.98}$$

since

$$D(B \cup C) = D(B) \tag{4.99}$$

where

$$C = \{j : T^j = \infty\}, \tag{4.100}$$

(4.98) will yield (4.92).

From the proof of Proposition 4.5, $\exists\; k \notin V^*, k \in A^*$ such that $D(A^* - \{k\}) < d^k$. Consider $\pi(T^k; T^{1^*}, T^{2^*}, \cdots, T^{k-1^*}, T^{k+1^*}, \cdots, T^{L^*})$. From (4.95), we can write $d\pi/dT^k$ as:

$$\frac{d\pi}{dT^k} = \frac{\left(\prod_{i \neq k} T^i\right)\left(\sum_{i \neq k} \prod_{j \neq i, k} T^j\right)}{\left(\sum_i \prod_{j \neq i} T^j\right)^2}$$

$$\cdot \left\{\left[T^k + \left(\sum_{i \neq k} {T^i}^{-1}\right)^{-1}\right] d^{k'}\right.$$

$$\left. + D(A^* - \{k\}) - d^k\right\} \tag{4.101}$$

where the indices $i \in \{1^*, 2^*, \cdots, k-1^*, k, k+1^*, \cdots, L^*\}$. Now since $d^{k'} > 0 \ \forall \ T^k > (C^k)^{-1}$ from Chapter 2, $\frac{d\pi}{dT^k} < 0 \ \forall \ T^k > (C^k)^{-1}$. Therefore, $T^{k^*} = \infty$ and $D(A^*) = D(A^* - \{k^*\})$ from (4.99).

Now again by Proposition 4.5, $\exists \ m \notin V^*$, $m \in A^*$ such that $D(A^* - \{m\}) = D(A^* - \{k^*\} - \{m\}) < d^m$, and $T^{m^*} = \infty$ will follow similarly as above.

Therefore,

$$A^* = V^* \cup \{j : T^j = \infty\} \tag{4.102}$$

which gives (4.92) by (4.98).

Next consider the cross-section curve $\pi(T^k; T^1, T^2, \cdots, T^{k-1}, T^{k+1}, \cdots, T^L)$. At an inflection point of π:

$$\frac{d^2\pi}{dT^{k^2}} = 0$$

$$\Longrightarrow \sum_i \prod_{j \neq i} T^j$$

$$= \frac{d^{k'} \pm \sqrt{(d^{k'})^2 + 2d^{k''}\left[\left(\prod_{i \neq k} T^i\right)\left(1 + \sum_{i \neq k} \frac{d^i}{T^i}\right) - \left(\sum_{i \neq k} \prod_{j \neq i, k} T^j\right) d^k\right]}}{d^{k''}} \tag{4.103}$$

Since $d^{k'} < 0$ and $d^{k''} > 0$ for all T^k, and since the left hand side is positive, in order for (4.103) to have a finite solution, we must have:

$$\underline{D}(A - \{k\}) \triangleq \Big(\prod_{i \neq k} T^i\Big)\Big(1 + \sum_{i \neq k} \frac{d^i}{T^i}\Big) - \Big(\sum_{i \neq k} \prod_{j \neq i, k} T^j\Big) d^k > 0, \tag{4.104}$$

or

$$d^k < D(A - \{k\}) \tag{4.105}$$

at the inflection point. Furthermore, at the inflection point:

$$\frac{d\pi}{dT^k} = \Big(\prod_{i \neq k} T^i\Big)\left[\frac{\underline{D}(A - \{k\})}{\sum_i \prod_{j \neq i} T^j} + \frac{1}{1 + \sqrt{2\, \dfrac{d^{k''}}{(d^{k'})^2}\, \underline{D}(A - \{k\})}}\right] > 0. \tag{4.106}$$

We can also show that $d^2\pi/dT^{k^2}$ is positive to the left of the inflection point and negative to the right of it. Therefore we have two possible shapes for π as shown in Fig. 4.8.

To see that there is a neighborhood of D(A) which is convex, we can write the second order derivatives as:

$$\frac{\partial^2 D}{\partial T^{k^2}} = \frac{\Big(\prod_{i \neq k} T^i\Big) d^{k''} - 2\,\dfrac{\partial D}{\partial T^k}}{\sum_i \prod_{j \neq i} T^j} \tag{4.107}$$

$$\frac{\partial^2 D}{\partial T^m\, \partial T^n} = \frac{\Big(\prod_{i \neq m, n} T^i\Big)^2 \Big(T^m \dfrac{\partial D}{\partial T^m} + T^n \dfrac{\partial D}{\partial T^n}\Big)}{\sum_i \prod_{j \neq i} T^j} \tag{4.108}$$

At the optimum point, the first-order partial derivatives will be

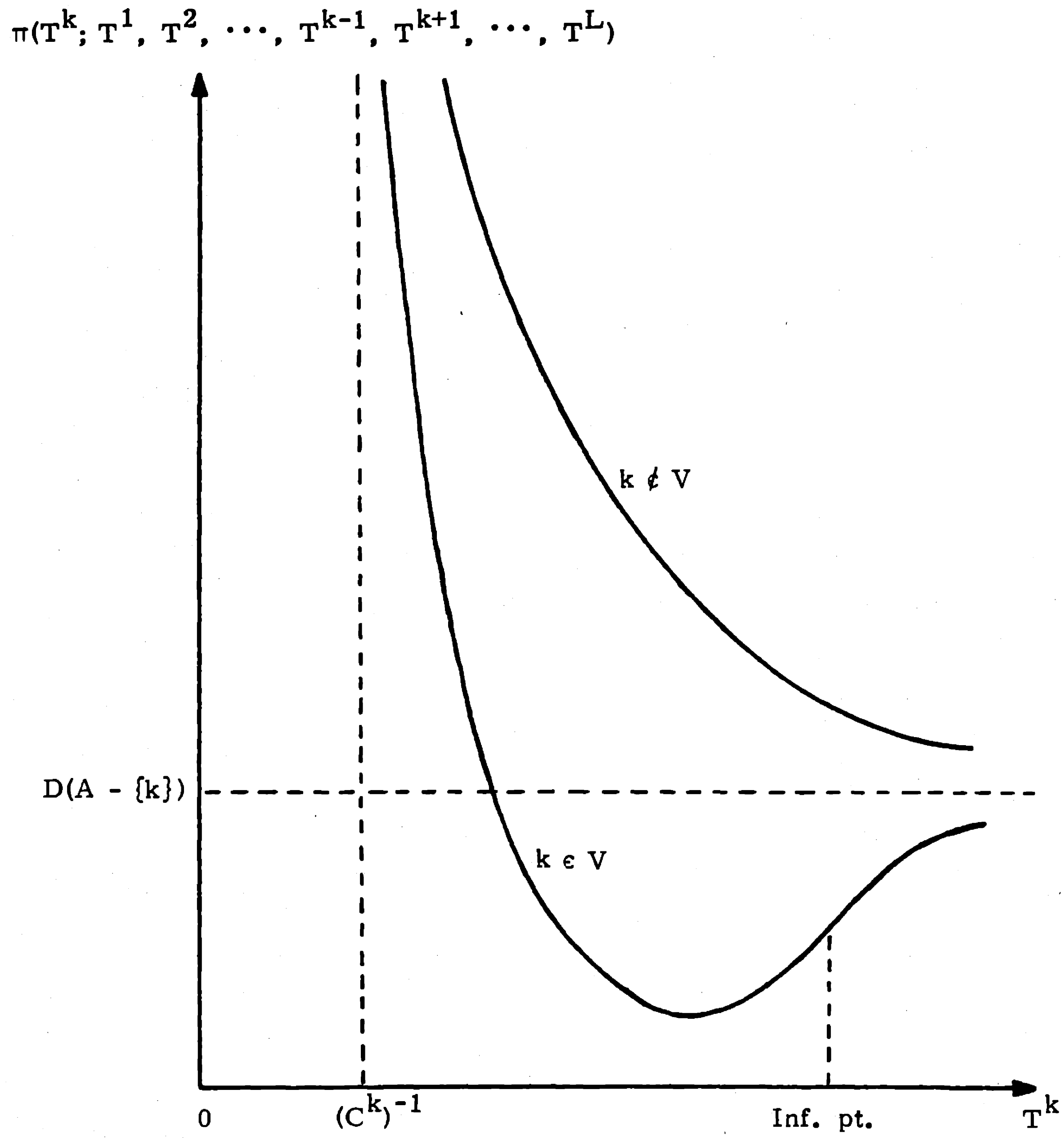

Fig. 4.8 Two possible shapes for the π curve as a function of T^k.

zero while $d^{k''} > 0$; therefore the Hessian matrix $\nabla^2 D(A)$ will be diagonal with positive elements; hence $\nabla^2 D(A) > 0$. Due to continuity of the first and second order partial derivatives of d^j, there will be a neighborhood of the optimum point where the Hessian will still be positive, making the functional convex.

Q.E.D.

For actual implementation of an optimization algorithm, writing D(A) in terms of frequencies as:

$$D(A) = \frac{1 + \sum_{i \in A} f^i d^i}{\sum_{i \in A} f^i} \qquad (4.109)$$

will be more appropriate, since $f^{k^*} = 0$ rather than $T^{k^*} = \infty$ for $k \notin V$. Our reason for working with T's was to interpret D(A) as an effective period plus an effective delay, since we have been working with terms like $T + d$ in the previous sections also.

For the formulation in terms of frequencies, the possible cross-section curves are shown in Fig. 4.9.

We want to make a final comment on the minimization of the effective delay D with respect to T^j. One might ask whether minimization of effective delays $T^j + d^j$ on paths j independently for all paths leads to the minimum effective delay from the node. The fact that this superposition does not hold is illustrated by the following example.

Assume that there are two disjoint paths to the destination, 1 and 2. Let $T^{1^*} \triangleq \min_{T^1} (T^1 + d^1(T^1))$ and $T^{2^*} \triangleq \min_{T^2} (T^2 + d^2(T^2))$. For the values $T^{1^*} = 1$, $d^1(T^{1^*}) = 6$; $T^{2^*} = 2$, $d^2(T^{2^*}) = 5$, assume that $d^2(T^2 = 6) = 2$. Then $D(T^1 = T^{1^*} = 1, T^2 = T^{2^*} = 2) = 6.33$, whereas $D(T^1 = T^{1^*} = 1, T^2 = 6) = 6.28$. Therefore $D(T^{1^*}, T^{2^*}, \cdots, T^{i^*}) \neq D^*$.

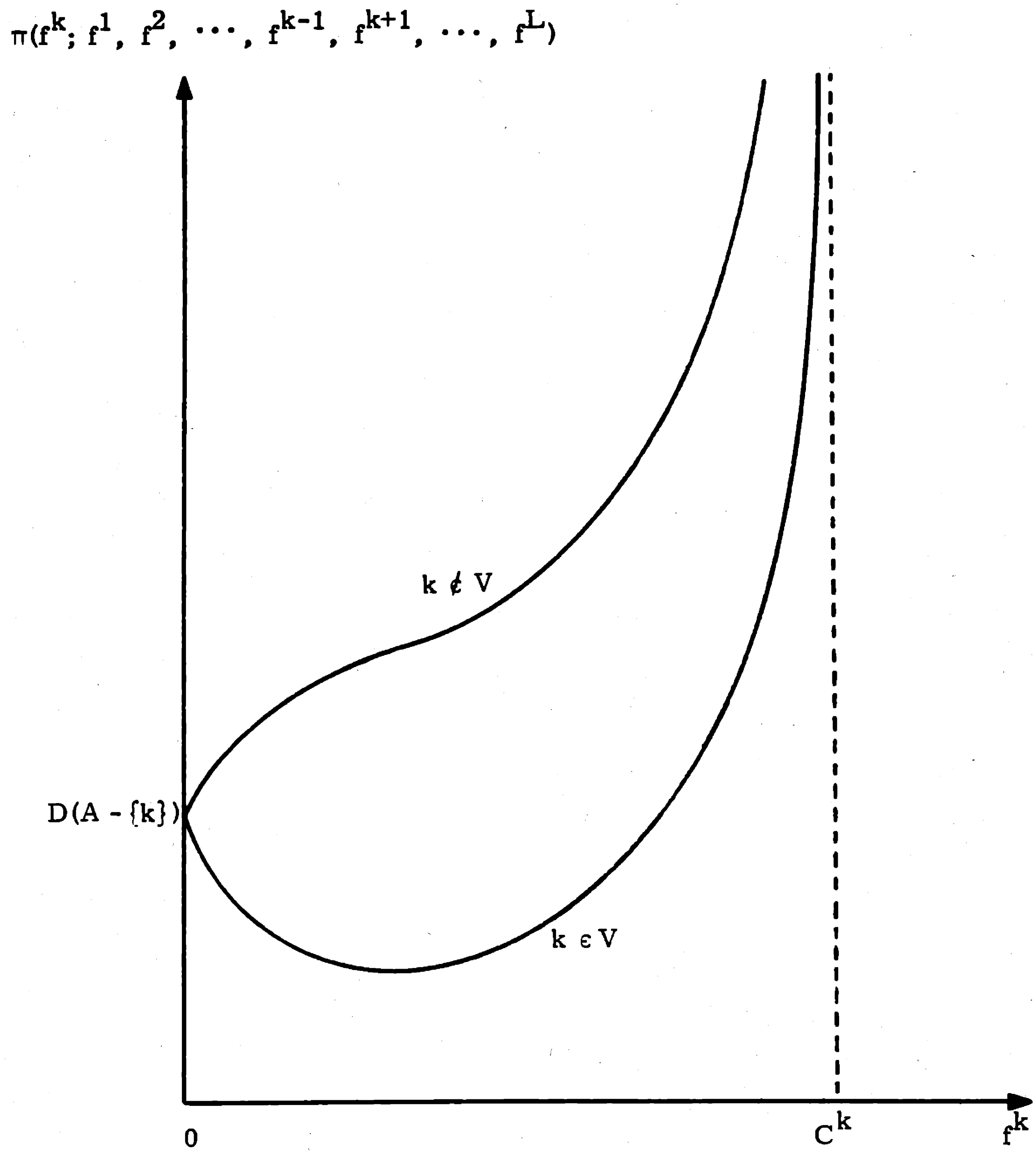

Fig. 4.9 Two possible shapes for the π curve as a function of f^k.

4.4.4 Minimization of Worst-Case Peak Value of the Mean-Square Error for the General Network

For a network with a single sensor node and several links to the destination, the worst-case peak value of the mean-square error is:

$$p_{wc} = \text{trace } P(D) \tag{4.110}$$

$$\dot{P}(t) = AP(t) + P(t)A' + BQB' \tag{4.111}$$

$$0 = AP(0) + P(0)A' + BQB - P(0)C'R^{-1}CP(0) \tag{4.112}$$

$$D = \max_{j} \left(t_a^j - \max_{t_a^k < t_a^j} t_d^k\right) \quad \forall \; j, k \tag{4.113}$$

The optimization problem is:

$$\min_{T^j} p_{wc} \tag{4.114}$$

where $(T^j)^{-1}$ is the average reporting frequency on link j.

Since tr P(D) is a monotonically increasing function of D_V, minimizing p_{wc} (due to the special structure of matrix A - see Section 2.3) with respect to T^j is equivalent to minimizing D with respect to T^j, which was discussed in the last section.

For the general network with N sensor nodes, p_{wc} is given by (4.47) through (4.54), where

$$D_i = \frac{1 + \sum\limits_{j \in \mathcal{O}(i)} \dfrac{d_{ij}(T_{ij}) + D_j}{T_{ij}}}{\sum\limits_{j \in \mathcal{O}(i)} \dfrac{1}{T_{ij}}} \tag{4.115}$$

$(T_{ij})^{-1}$ is the average departure frequency, $d_{ij}(T_{ij})$ is the corresponding delay on link (i, j). O(i) is the set of nodes for which link (i, j) exists. For the path delay d^j in (4.58), we have substituted $d_{ij}(T_{ij}) + D_j$, which represents the worst-case effective delay from node #i to the destination over link (i, j) and node #j. As we have shown in the last section, the above expression does not equal the effective delay from node #i at each iteration of an algorithm used to minimize p_{wc}. Rather, its value equals the effective delay at each local minimum of p_{wc}. Remember that the effective delay reflects the optimum adjustment of the departure times on the links originating from a sensor node, for given average departure frequencies and corresponding communication delays.

For the above formulation, D_i are not independent as was the case in Section 4.3; therefore they cannot be minimized independently. On the other hand, in this section we assume that matrix A can be any multivariable square real matrix, so p_{wc} is not a convex, concave or monotonic function of D_i in general, only a continuous function of them. The behavior of p_{wc} as a function of D_i depends entirely on the system and observation model with the statistical characterizations and, as such, it is decoupled from the behavior of D_i as a function of T_{ij}, which depends entirely on the communications model of the network. Therefore, it is not in general possible to arrive at conclusions about the nature of p_{wc} as a function of T_{ij}, i.e. whether it has a unique minimum, etc.

However, it can be shown that in a neighborhood of the local minima, p_{wc} is a convex function of T_{ij}. This follows from the fact that $\partial D_i/\partial T_{jk}$ is finite for all i, j, k (can be shown from the results of the last section) and that $\partial p_{wc}/\partial D_i$ is also finite, from (4.47) through (4.54) and assuming that the matrices P(0) and Q are finite. Therefore, in these convex neighborhoods of the local minima, we can use some nonlinear programming algorithms for the minimization of p_{wc}.

Before proceeding further though, we have to introduce a minor adjustment. In the last section we have shown that for paths $j \notin V$, $T^{j^*} = \infty$ or $f^{j^*} = 0$. Figure 4.8 shows that as $T^j \to \infty$, D is a convex function of T^j, but it is not possible to run a numerical algorithm when the optimum point is at the infinity. Alternatively, as Fig. 4.9 shows, as $f^j \to 0$, D is a concave function of f^j, rendering the formulation of D in terms of f_{ij}'s undesirable. One way to solve this problem is to keep the formulation in terms of T_{ij}'s, and to first find the links (i, j) for which $T^*_{ij} = \infty$, by using the algorithm of Proposition 4.5, and then execute the (bigger) algorithm for the remaining variables for each iteration of the (bigger) algorithm.

A second possible approach is less precise, but more convenient. For each link (i, j), we assign an upper limit on T_{ij}, call it U_{ij}, and say that the optimum value of T_{ij} is ∞ if the algorithm gives $T^*_{ij} = U_{ij}$. Useful values of U_{ij} may be taken based on physical considerations, but can be readjusted by checking with the algorithm of Proposition 4.5.

We now describe the algorithm, which is based on a class of algorithms introduced in [12] and [13].

<u>Proposition 4.7</u>: The optimization problem is:

$$\min_{\underline{C}^{-1} \le \underline{T} \le \underline{U}} \quad p_{wc}(\underline{T}) \tag{4.116}$$

where $\underline{T}$ is the vector of T_{ij}'s, and $\underline{C}^{-1}$ and $\underline{U}$ are the corresponding vectors of C^{-1}_{ij}'s and U_{ij}'s, respectively. p_{wc} is given by (4.47) through (4.54) and (4.115). For simplicity of notation, the elements of $\underline{T}$ will be denoted as T^1, T^2, $\cdots$, and similarly for $\underline{C}^{-1}$, $\underline{U}$. Subscripts will denote the iterations.

For a vector $\underline{Z}$ we denote by $[Z]^{\#}$ the vector with coordinates:

$$[Z]^{\#^i} = \begin{cases} U^i & \text{if } Z^i \geq U^i \\ Z^i & \text{if } C^{i^{-1}} < Z^i < U^i \\ C^{i^{-1}} & \text{if } Z^i \leq C^{i^{-1}} \end{cases} \tag{4.117}$$

The k^{th} iteration is as follows:

<u>Step 1</u>: Find the set:

$$I_k^{\#} = \left\{ i \,|\, C^{i^{-1}} \leq T_k^i \leq C^{i^{-1}} + \epsilon_k^i \text{ and } \frac{\partial p_{wc}(\underline{T}_k)}{\partial T^i} > 0 \right.$$

$$\left. \text{or } U^i - \epsilon_k^i \leq T_k^i \leq U^i \text{ and } \frac{\partial p_{wc}(\underline{T}_k)}{\partial T^i} < 0 \right\}, \tag{4.118}$$

where

$$\epsilon_k^i = \min \left\{ \epsilon, s_k^i \right\} \tag{4.119}$$

where $\epsilon > 0$ is a constant scalar and

$$s_k^i = \left| T_k^i - \left[T_k^i - \mu_k^i \frac{\partial p_{wc}(\underline{T}_k)}{\partial T^i} \right]^{\#} \right| \tag{4.120}$$

and μ_k^i are scalar sequencies such that

$$\mu_k^i \geq \mu^i > 0 \tag{4.121}$$

with $\mu^i > 0$ a constant scalar.

Step 2: Partition $\underline{T}_k$ as:

$$\underline{T}_k = \begin{bmatrix} \underline{\tilde{T}}_k \\ \underline{\bar{T}}_k \end{bmatrix} \tag{4.122}$$

where $\underline{\tilde{T}}_k$ is the vector of coordinates T_k^i with $i \in I_k^{\#}$ and $\underline{\bar{T}}_k$ is the vector of coordinates T_k^i with $i \notin I_k^{\#}$. Then a "search direction",

$$\underline{\nu}_k = \begin{bmatrix} \underline{\tilde{\nu}}_k \\ \underline{\bar{\nu}}_k \end{bmatrix} \tag{4.123}$$

is obtained by solving the systems of equations:

$$\tilde{\Omega}_k \underline{\tilde{\nu}}_k = - \underline{\tilde{\sigma}}_k \tag{4.124}$$

$$\bar{\Omega}_k \underline{\bar{\nu}}_k = - \underline{\bar{\sigma}}_k \tag{4.125}$$

where $\underline{\tilde{\sigma}}_k$ (or $\underline{\bar{\sigma}}_k$) is the vector with coordinates $\dfrac{\partial p_{wc}(\underline{T}_k)}{\partial T^i}$ with $i \in I_k^{\#}$ (respectively $i \notin I_k^{\#}$), $\tilde{\Omega}_k$ is a diagonal positive definite matrix with elements $\dfrac{\partial^2 p_{wc}(\underline{T}_k)}{(\partial T^i)^2}$ along the diagonal, and $\bar{\Omega}_k$ is a symmetric positive definite matrix which is equal to the Hessian of p_{wc} with respect to the coordinates T^i, $i \notin I_k^{\#}$.

Equation (4.125) may be computationally impractical to solve exactly, and an approximate solution by the following scaled version of the conjugate gradient method may be used.

Choose a positive definite symmetric matrix Λ_k and generate the sequence $\{\underline{Z}_m\}$ according to the iteration:

$$\underline{Z}_o = 0, \quad \underline{Z}_{m+1} = \underline{Z}_m + \gamma_m \underline{\omega}_m, \quad m = 0, 1, \cdots \tag{4.126}$$

where the conjugate direction sequence $\{\underline{\omega}_m\}$ is given by:

$$\underline{\omega}_o = -\Lambda_k \underline{r}_o, \quad \underline{\omega}_m = -\Lambda_k \underline{\rho}_m + \beta_m \underline{\omega}_{m-1}, \quad m = 1, 2, \cdots \tag{4.127}$$

the residual sequence $\{\underline{\rho}_m\}$ is defined by:

$$\underline{\rho}_m = \bar{\Omega}_k \underline{Z}_m + \underline{\bar{\sigma}}_k, \qquad m = 0, 1, \cdots \tag{4.128}$$

and the scalars γ_m and β_m are given by:

$$\gamma_m = \frac{\underline{\rho}_m' \Lambda_k \underline{\rho}_m}{\underline{\sigma}_m' \bar{\Omega}_k \underline{\sigma}_m}, \qquad m = 0, 1, \cdots \tag{4.129}$$

$$\beta_m = \frac{\underline{\rho}_m' \Lambda_k \underline{\rho}_m}{\underline{\rho}_{m-1}' \Lambda_k \underline{\rho}_{m-1}}, \qquad m = 1, 2, \cdots \tag{4.130}$$

Terminate at an iteration m if the residual $\underline{\rho}_m$ satisfies:

$$|\underline{\rho}_m| \leq \beta_k |\underline{\rho}_o| \tag{4.131}$$

where β_k is some scalar factor less than unity which may depend on the iteration index k.

Then use $\underline{\bar{\nu}}_k \cong \underline{Z}_m$.

<u>Step 3:</u> Then

$$\underline{T}_{k+1} = [\underline{T}_k + \alpha_k \underline{\nu}_k] \tag{4.132}$$

where $\alpha_k = \beta^{m_k}$, and m_k is the first non-negative integer m such that:

$$p_{wc}(\underline{T}_k) - p_{wc}[\underline{T}_k(\beta^m)]$$

$$\geq \zeta \left\{ -\beta^m \sum_{i \notin I_k^{\#}} \frac{\partial p_{wc}(\underline{T}_k)}{\partial T^i} \nu_k^i + \sum_{i \in I_k^{\#}} \frac{\partial p_{wc}(\underline{T}_k)}{\partial T^i} [T_k^i - T_k^i(\beta^m)] \right\} \quad (4.133)$$

where

$$\underline{T}_k(\alpha) \triangleq [T_k + \alpha \underline{\nu}_k]^{\#} \quad \forall \quad \alpha \geq 0 \quad (4.134)$$

and $\beta \in (0, 1)$, $\zeta \in (0, \frac{1}{2})$.

As proved in the papers mentioned above, this algorithm has a superlinear convergent rate, and with the conjugate directions approximate solution in step 2, it is very suitable for large-scale optimization problems such as the case at hand.

The iterations are centralized, and as such they suit the optimization problem under consideration. Calculation of the partial derivatives of p_{wc} requires knowledge of the values of all the variables in the network and therefore rules out decentralized iterations.

CHAPTER 5

EXTENSIONS

5.1 Introduction

In this chapter we will relax some of the restrictions of the model assumed in Chapter 4. However, rather than paralleling the analysis given there, we will point out some major differences in the type of solutions that result. In Chapter 4 we considered a network of sensors which have been taking measurements for a long time so that the filtering problem at each node has reached a steady state. The link delays were modelled as deterministic functions of traffic rates on the links and independent of the traffic on other links. In Section 5.2 we will consider some formulations based on probabilistic models for the delays. In Section 5.3 we will demonstrate the effect of interdependent delays with an example.

5.2 Formulations Based on Probabilistic Models for Delays

5.2.1 Introduction

In Chapter 4, we assumed the message delays on links to be deterministic functions of the traffic rates. Actually, as discussed in Chapter 2, the delays are random variables and we approximated their stochastic behavior with their mean values.

In this section, we will consider some implications of the probabilistic models, particularly the effect of the reversal of the order of messages from the same source. We will also discuss some relevant objective functionals for the probabilistic framework.

We will assume that on link (i, j) node #i sends messages every T seconds to node #j, and that the delays of different messages are independent random variables with the same distribution.

5.2.2 Order Reversal of Messages and Statistical Characterization of Delays

First let us consider the effect of the order reversal of messages. This situation happens when a certain message A arrives later than another message B which was sent after message A. Assuming that each message contains sufficient statistics for the entire observation history of the source node, in this case message A will be rendered useless, since message B carries all the information of message A plus more. Therefore the resulting effect will be as if message A was lost, or alternatively, as if the source skipped a reporting time.

Figure 5.1 illustrates the effect of an order reversal on the mean-square estimation error. The m.s.e. at the destination node is plotted for a one sensor - one destination network configuration. The observed system is modelled a Brownian motion process.

In this example the message sent at time T arrived after the one sent at time 2T; thus it was useless for the destination node. If a message arrives before all the messages that are sent later, we will say that that message has "arrived". Otherwise we will say that it is "lost". The occurrence of two consecutive lost messages will be called a "double" and the occurrence of n consecutive lost messages will be called an "n-tuple".

For a quantitative analysis of the order reversal effect, we will need the expected frequencies of occurrence of "n-tuples". This is equal to the probability that an "n-tuple" starts with a message which is sent at some t_o, long after the process has started. In other words, it is the probability that the message sent at t_o - T "arrives", the messages

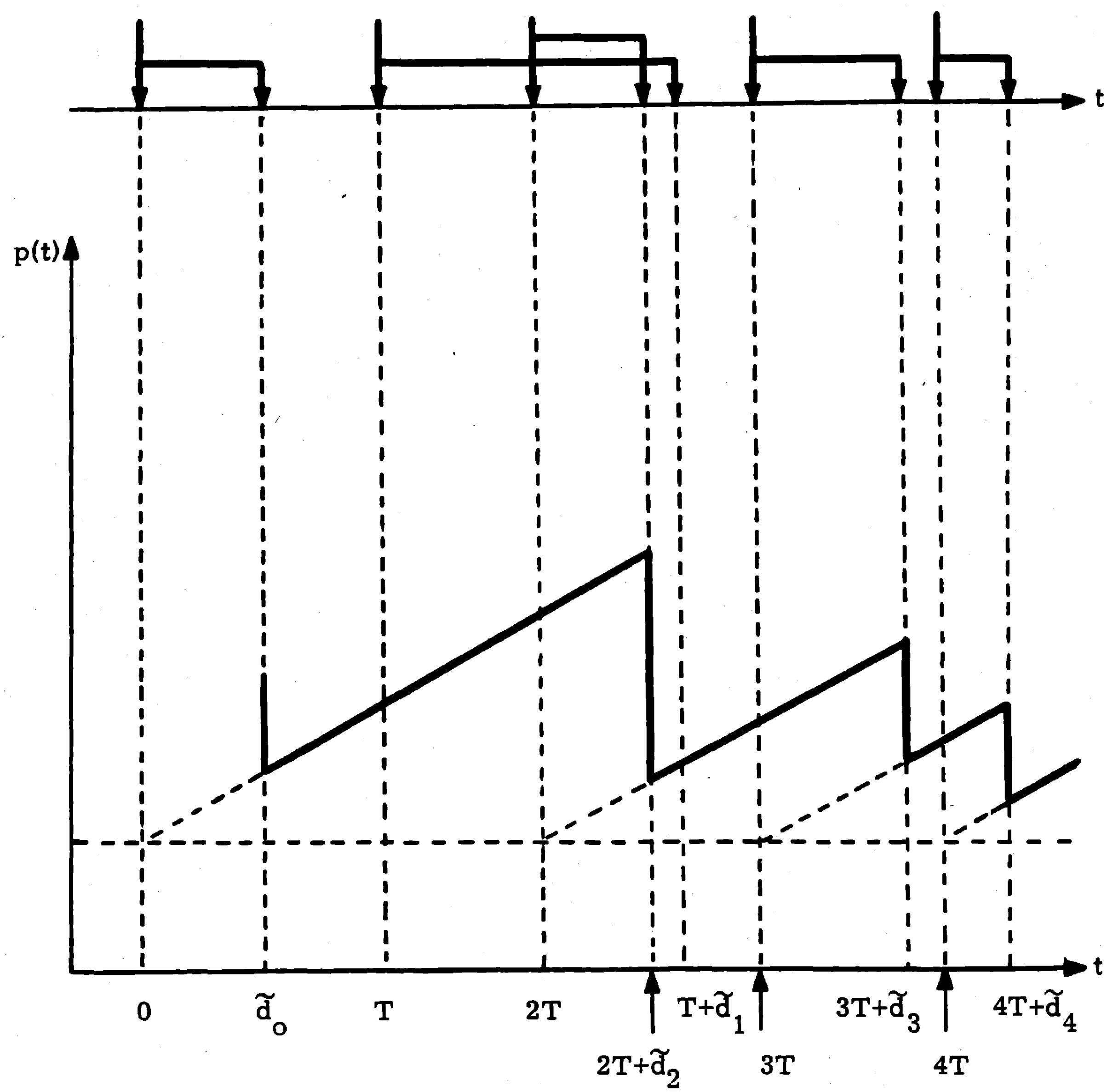

Fig. 5.1 Mean-square estimation error vs time for a single sensor configuration with stochastic delays and Brownian motion process under observation.

sent at t_o, $t_o + T$, $\cdots$, $t_o + (n-1)T$ are "lost", and the message sent at $t_o + nT$ "arrives" again. If we denote the delay occurred by the message sent at $t_o + kT$ by $\tilde{d}_k$, then the probability of this event is:

$$
\begin{aligned}
p_n \;&\triangleq\; \text{Prob ("n-tuple" at } t_o) \\
&= \text{Prob } [\tilde{d}_{-1} < \tilde{d}_k + (k+1)T \quad \forall\ k = 0, 1, 2, \cdots \\
&\quad \text{and } \exists\ k = 1, 2, \cdots \text{ s.t. } \tilde{d}_o > \tilde{d}_k + kT \\
&\quad \text{and } \exists\ k = 2, 3, \cdots \text{ s.t. } \tilde{d}_1 > \tilde{d}_k + (k-1)T \\
&\quad \vdots \\
&\quad \text{and } \exists\ k = n, n+1, \cdots \text{ s.t. } \tilde{d}_{n-1} > \tilde{d}_k + (k-n+1)T \\
&\quad \text{and } \tilde{d}_n < \tilde{d}_k + (k+1)T \quad \forall\ k = n+1, n+2, \cdots] \qquad (5.1)
\end{aligned}
$$

If the statistical characterization of delays is given by a probability density function, and if this p.d.f. takes on nonzero values in a finite interval only, the above probability can be calculated in terms of multiple integrals. Alternatively, rather than the p.d.f. for the delays, the expected frequencies of "n-tuples", p_n, be specified. In fact, these probabilities are much easier to estimate than the p.d.f. The only other statistic that we need for further analysis is the mean value of the delays.

5.2.3 Optimization Criteria

Worst-case optimization as in Chapter 4 is not as meaningful when the delays are viewed as random variables, especially if the p.d.f.'s for delays on some links have nonzero values over an infinite domain. Therefore the natural criteria are the ones which are based on mean values.

On possible formulation is:

$$\min_{T} E\,[p(t)] \tag{5.2}$$

Since the delays are random variables, the mean-square error, p(t), becomes a stochastic process. For a one-sensor node - one-destination node configuration, its expected value is:

$$E\,p(t) = \operatorname{tr} S + \sum_{n} p_n \cdot E\,\frac{1}{T}\int_{\tilde{d}_k}^{T+\tilde{d}_{k+n}} \operatorname{tr} P(t)\,dt \tag{5.3}$$

where

$$0 = AS + SA' + BQB' - SC'R^{-1}CS \tag{5.4}$$

$$\dot{P}(t) = AP(t) + P(t)A' + BQB' \tag{5.5}$$

$$P(0) = S \tag{5.6}$$

For the special case when the observed system is modelled as a Wiener process, the optimization problem has a particularly simple form:

$$\min_{T} E\,[p(t)] \Longrightarrow \min_{T} k\,\frac{T}{2} + m_d(T) \tag{5.7}$$

where

$$k = 1 + \sum_{n} (n^2 - n)p_n \tag{5.8}$$

and $m_d(T)$ is the mean value of the delay on the link between the two nodes. It is dependent on the reporting period T, because the probability distribution of the delay is a function of T.

Another possible formulation, which is simpler than (5.2) is to minimize the mean of the peaks of the p(t) waveform. Then the optimization problem for a 1 sensor node - 1 destination node configuration becomes:

$$\min_{T} \sum_{n} p_n \cdot E \, tr \, P\,(nT + \partial) \qquad (5.9)$$

where P(t) is given by (5.4) - (5.6).

For these formulations, it is not ready to find closed form expressions for the objective functions for a general multinode, multi-link network configuration. Instead, we propose to introduce another approximation in order to develop a mathematically more tractable objective function. According to our present model, the sensor nodes make continuous observations of a stochastic process and report their sufficient statistics to other nodes over communication links at discrete times. Therefore the information about an observation made at a particular point in time "waits" at this node until the next reporting time; plus it incurs a communication delay before reaching the node at the other end of the link. Likewise, a message originating at some node #i and routed through an intermediate node #j on its way to node #k has to wait at node #i until the next message leaves for node #k. Naturally, it too incurs a communication delay on link (j, k). If an observation is made or a message from another node is received at time t, let $\delta(t)$ denote the total time it takes for the information about it to reach the other node. Figure 5.2 illustrates $\delta(t)$ corresponding to the delay samples of Fig. 5.1.

The mean value of $\delta(t)$ can be calculated:

$$E\,[\delta(t)] = \left[1 + \sum_{n} (n^2 - n)\, p_n\right] \left[\frac{T}{2} + m_d(T)\right] \qquad (5.10)$$

Now the approximation proposed is to model the messages as being sent continuously rather than periodically and to assign a fixed delay $E[\delta(t)]$ to these messages. Obviously, $E[\delta(t)]$ will depend on the

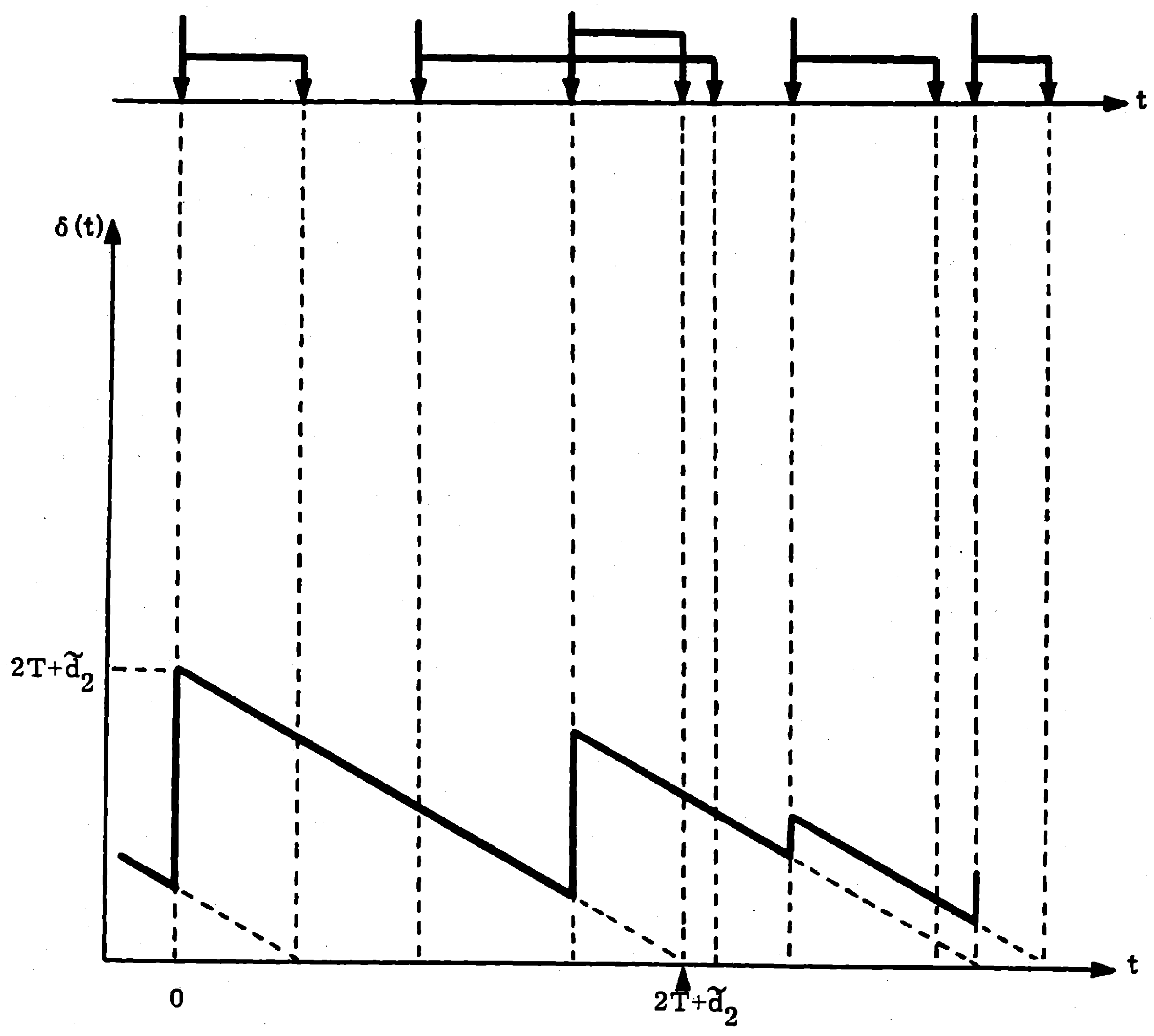

Fig. 5.2 Delay of observations due to waiting on the sensor node and communications, versus time.

actual p.d.f. for delays on that link and the actual reporting period T. Under this new model, the mean-square estimation error, p(t), will be constant in the steady state, and the natural objective function will be to minimize this. With this new model, all links are effectively assigned a "length", and the optimization procedures discussed in Chapter 4 are equally valid for this new set of lengths.

5.3 Interdependent Delays

We will present a simple numerical example to show that, when link delays are dependent on traffic on other links, sensors whose observations are too noisy may have to be shut off; or links whose traffic has a very adverse effect on the delays of other links may have to be left unused.

Consider the configuration of Fig. 5.3, where there are two sensor nodes and a destination node:

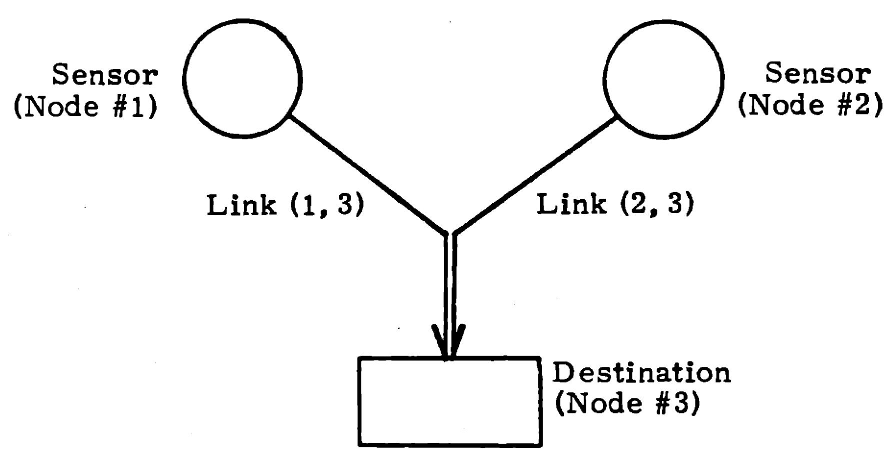

Fig. 5.3 System configuration for Section 5.3.

The sensors are assumed to be making continuous observations of a Wiener process; their observation noises are independent Brownian motion processes with intensities r_1 and r_2.

$$dx(t) = dw(t)$$
$$dy_i(t) = x(t)dt + dv_i(t), \quad i = 1, 2 \tag{5.11}$$

We assume that the sensors have been making observations for a long time, so that the local filters can be assumed to have reached the steady state.

The sensor nodes continuously send sufficient statistics to the destination node.

The delays on links (1, 3) and (2, 3) are D_1 and D_2. We adopt the following model for the delays:

$$D_1 = \frac{T_1}{2} + d_1, \quad d_1 = k_{10} + \frac{k_{11}}{T_1} + \frac{k_{12}}{T_2} \tag{5.12}$$

$$D_2 = \frac{T_2}{2} + d_2, \quad d_2 = k_{20} + \frac{k_{21}}{T_1} + \frac{k_{22}}{T_2} \tag{5.13}$$

This model can be interpreted as the continuous approximation to discrete-time communications using average effective delay of information. The delays include cross-coupling terms which reflect the effect of the traffic on one link on the delay of the other.

Then, if $D_1 > D_2$, the mean-square error of the state estimate at the destination node is:

$$p = \sqrt{qr_2}\ \frac{1 - \beta_2 e^{-2\alpha_2 (D_1 - D_2)}}{1 + \beta_2 e^{-2\alpha_2 (D_1 - D_2)}} + qD_2 ,$$

$$\alpha_2 = \sqrt{q/r_2}\ , \quad \beta_2 = \frac{\sqrt{qr_2} - \sqrt{qr_{eq}}}{\sqrt{qr_2} + \sqrt{qr_{eq}}}$$

$$r_{eq} = \frac{r_1 r_2}{r_1 + r_2} . \tag{5.14}$$

Let us assume the following parameters for the link delay functions:

$$k_{10} = 4, \quad k_{11} = 2, \quad k_{12} = 0.5$$

$$k_{20} = 1, \quad k_{21} = 0.1, \quad k_{22} = 1$$

For two different sets of noise intensities, we get the following optimal (starred) values for the periods, delays and the mean-square error:

Case 1:

$$q = 1, \quad r_1 = 40, \quad r_2 = 400$$

$$T_1^* = 2.10 \qquad D_1^* = 6.32$$

$$T_2^* = 1.58 \qquad D_2^* = 2.47$$

$$p^* = 11.72$$

Case 2:

$$q = 1, \quad r_1 = 16, \quad r_2 = 4$$

$$R_1^* = \infty \qquad D_1^* = \infty$$

$$T_2^* = 1.41 \qquad D_2^* = 1.41$$

$$p^* = 4.41$$

For the parameters of Case 2, it is optimal for Node #1 to send nothing at all.

CHAPTER 6

SUMMARY AND DIRECTIONS FOR FUTURE RESEARCH

6.1 Summary

In this thesis we considered decentralized linear estimation problems with data-traffic-dependent delay constraints arising from the assumed underlying communications network structure. We assumed that the system model under observation can be modelled as a linear multivariable system driven by Brownian motion; that the sensors in the estimation network make independent noisy observations; and that the system and observation models are such that the estimation problem is in a "steady state".

The delays on the communication links between the sensor nodes were considered to be deterministic, convex and monotonically increasing functions of the traffic rates in the network for much of the thesis.

The linear estimate of the state of the system under observation was desired at the so-called "destination node", and the objective was to minimize the highest value the mean-square estimation error could take over time at this node. For the special case of a network composed of two sensor nodes with direct connections to the destination node, an algorithm was described to minimize this performance criterion as a function of the frequencies of the periodic messages sent by the sensors to report the statistical content of their observations, and also as a function of the time or phase relationship between the message sequences of the two nodes.

When the network structure is generalized to allow an arbitrary number of sensor nodes and an arbitrary number of communication links between these nodes, messages are ordinarily routed over intermediate nodes on their way from a source node to the destination node. However, unlike the usual communication networks where the messages must be relayed to their destination intact, we have shown that the statistical content of the messages can be combined at the intermediate nodes without any loss of information, so that the traffic does not increase in the downstream links of the network due to messages originating in upstream nodes. Furthermore, sufficient statistics were found which permit fusion of statistical data from different sources with no loss of information without having to transmit the entire set of observations.

For the general network, a further complication arises over the two-sensor-node example. Routing, or message frequencies on alternate paths must be considered as an additional factor in optimization. To alleviate some of the complexity, a form of worst-case optimization policy was adopted by considering the situation where the phase relationships between the message sequences on the links lead to the highest possible mean-square error at the destination. Thus, the phase relationships between the message sequences were eliminated from the set of control variables.

We have pursued two types of worst-case optimization approaches for the general network. In both cases, we assumed that the underlying communications network is of wire-network type, where the message delays are dependent only on the message traffic on the link the message is travelling. In the first case, we allowed any node in the network to send messages to only one other node. Under this restriction, the optimization problem was particularly simple and resulted in a spanning-tree type of routing solution. In the second case, we allowed the nodes to route their messages to the destination along multiple paths.

Previously unused channel capacity was thus utilized for better performance. Since the message delays on separate paths from a sensor node to the destination node are in general different, adjustment of the precise departure times on the paths for average departure frequencies and corresponding delays on these paths was another issue to be taken into consideration.

For the general multi-path case, first the optimization problem for a one-sensor-node network with many alternate paths to the destination was analyzed. The correct objective function was derived, minimizing which gives the optimal departure rates on each path, taking into consideration the optimal adjustment of departure times. It was shown that the objective function is convex in the neighborhood of the optimum point; and evidence was presented that the only stationary point is the optimum.

For the multi-path case with arbitrary number of nodes, it was argued that in general there are a number of stationary points. A superlinearly convergent nonlinear programming algorithm was described for optimization near the stationary points.

Until this point the message delays were treated as deterministic functions of the traffic rate on the particular link the message is travelling. In the last chapter, we studied some implications of relaxing this assumption for the cases of random message delays and interdependent delays.

For the random delay case, an approximate problem formulation based on average delay of information was developed. For the interdependent delay case, it was shown by way of a numerical example that sometimes it is best for a sensor node to refrain from sending any messages, since the adverse effect its messages have on the delays on the other links outweighs the contribution of the information contained in those messages.

6.2 Suggestions for Future Research

This work may be extended for the case of multi-destination nodes. A reasonable objective function may be the weighted sum of the worst-case maximum values over time at these nodes. An extra complication arising in this case is that multiple copies of the same information may now have to be transmitted from the sensor nodes.

REFERENCES

[1] J. L. Speyer, "Computation and Transmission Requirements for a Decentralized LQG Control Problem", IEEE Trans. on Auto. Control, Vol. AC.24, pp. 266-269, April 1979.

[2] B. C. Levy, D. A. Castañon, G. C. Verghese, A. S. Willsky, "A Scattering Framework for Decentralized Estimation Problems", MIT/LIDS Paper 1075, March 1981.

[3] E. C. Tacker and C. W. Sanders, "Decentralized Structures for State Estimation in Large Scale Systems", Large Scale Systems, Vol. 1, No. 1, February 1980.

[4] D. A. Castañon, "Decentralized Estimation of Linear Gaussian Systems", MIT/LIDS Paper 1167, 1981.

[5] R. G. Gallager and S. J. Golestaani, "Flow Control and Routing Algorithms for Data Networks", Proc. 5th Int. Conf. on Comp. Comm., October 1980.

[6] H. Kwakernaak, R. Sivan, "Linear Optimal Control Systems", Wiley-Interscience, 1972.

[7] A. E. Bryson, Y.-C. Ho, "Applied Optimal Control", Hemisphere, 1975.

[8] B. Noble, J. W. Daniel, "Applied Linear Algebra", Prentice-Hall, 1977.

[9] R. Bellman, "Introduction to Matrix Analysis", McGraw-Hill, 1970.

[10] T. Kailath, "Linear Systems", Prentice-Hall, 1980.

[11] A. S. Willsky, M. Bello, D. A. Castañon, B. C. Levy, G. Verghese, "Combining and Updating of Local Estimates and Regional Maps Along Sets of One-Dimensional Tracks", MIT/LIDS Paper 955, June 1981.

[12] D. P. Bertsekas, "Projected Newton Methods for Optimization Problems with Simple Constraints", SIAM Journal on Control and Optimization, March 1982.

[13] D. P. Bertsekas and E. M. Gafni, "Projected Newton Methods and Optimization of Multicommodity Flows", MIT/LIDS Paper 1140, August 1981.