International Institute for
Applied Systems Analysis
IIASA www.iiasa.ac.at

Hierarchical Control Systems: An Introduction

Findeisen, W.

IIASA Professional Paper
April 1978

PP-78-1

HIERARCHICAL CONTROL SYSTEMS
AN INTRODUCTION

W. Findeisen

April 1978

ABSTRACT

The purpose of this paper is to describe the main concepts, ideas and operating principles of hierarchical control systems. The mathematical treatment is rather elementary; the emphasis of the paper is on motivation for using hierarchical control structures as opposed to centralized control. The paper starts with a discussion of multilayer control hierarchies, i.e. hierarchies where either the functions or the time horizons of the subsequent layers of control are different. Some attention has been paid, in this part, to the question of structural choices such as designation of control variables and selection of the time horizons. Next part of the paper treats decomposition and coordination in steady-state control: direct coordination, penalty function coordination and price coordination are discussed. The focus is on model-reality differences, that is on finding structures and operating principles that would be relatively insensitive to disturbances. The last part of the paper gives a brief presentation of the broad and still developing area of dynamic multilevel control. It was possible, within the restricted space, to show the three main structural principles of this kind of control and to provide for a comparison of their properties. A list of selected references is enclosed with the paper.

This paper is, in a sense, a forerunner of the book "Coordination and Control in Hierarchical Systems," by W. Findeisen, and co-authors, to appear in 1979 at J. Wiley, London, as a volume in the IIASA International Series. The results contained in the paper, as well as those in the above mentioned book, were obtained over a rather long research period. A partial support of this work by NSF Grant GF-37298 to the Institute of Automatic Control of the Technical University of Warsaw and to the Center for Control Sciences, University of Minnesota, is gratefully acknowledged.

TABLE OF CONTENTS

1. Introduction

The control of complex systems may be structured in the hierarchical way for several reasons. Some of them are the following:

- the limited decision making capability of an individual is extended by the hierarchy in a firm or organization ;
- subsystems (parts of the complex system) may be far apart and have limited communication with one another;
- there is a cost, delay or distortion in transmitting information;
- there exists a local autonomy of decision in the subsystems and their privacy of information (e.g. in the economical system).

In this paper we intend to present the basic principles and features of hierarchical control structures, in a possibly simple manner. Let us note that from the point of view of general principles it is, to a certain degree irrelevant whether we discuss a multilevel arrangement of computerized decisions, or a hierarchy of human decision makers, under the assumption that human decisions will be based on the same rational grounds. In particular , to both would apply the structural principles and several features of the coordination methods, e.g. the danger of violating the constraints, consequences of setting non-feasible demands, etc.

It shall be stressed that the paper is concerned with the *control* of systems, which means that the following is essential:

- we assume the system under control to be in operation and to be influenced by disturbances;

- current information about the system behavior or about the disturbances is available and can be used to improve the control decisions.

These two features make this study differ from studies of the problems of planning, scheduling, etc., where the only data we can use to determine a control or a policy come from an a priori model.

2. Hierarchical control concepts

A "complex system" will be an arrangement of some elements (*subsystems*) interconnected between their outputs and inputs,as it happens for example in an industrial plant. If we describe the interconnections by a matrix H we obtain a scheme as in Figure 1. The matrix H reflects the structure of the system. Each row in this matrix is associated with a single input of a subsystem. The elements in the row are zeros except for one place, where a "1" tells to what single output the given input is connected.

We are now interested in control of systems like Figure 1 by use of some special structures, referred to as "hierarchical". There are two fundamental and by now classical ideas in hierarchical control:

(i) the multilayer concept (Lefkowitz 1965), where the action of determining control for an object (plant) is split into algorithms (called "layers") acting at different time intervals;

(ii) the multilevel concept (Mesarović et al., 1965-1970) where the goal of control of an interconnected, complex system is divided into local goals and accordingly coordinated.

The *multilayer concept* is best depicted by Figure 2, where we envisage the task of determining control m as being split into:

Follow-up Control, causing *controlled variables* c to be equal to their desired values c_d,

Optimization, or an algorithm to determine optimal values of c_d, assuming some fixed parameters β of the plant and/or environment,

Adaptation, with the aim of setting optimal values of β.

The vector of parameters β may be treated more generally as determining also the structure of the algorithm performed at the lower layer and may be divided into several parts which would be adjusted at different time intervals: Thus, we might speak about having several adaptation layers.

The most essential feature of the structure in Figure 2 is that the layers intervene at different and increasing time intervals and that each of them is using some feedback or environment information. The latter is shown in the figure by dotted lines.

The application of structures like Figure 2 is usually associated with control of industrial processes, e.g. chemical reactors, furnaces, etc. It is not exclusive of other applications. For example the same philosophy underlies the case where the higher level of authority prescribes certain goals to be followed, but does not go into the detailed decisions necessary to actually follow the goals. Since it is the responsibility of the higher level to chose the *optimal* goals - the lower level may not even know the criterion of optimality.

The philosophy of a system like Figure 2 is clear and almost obvious: it is to implement control m, which cannot be strictly optimal (due to discrete as opposed to continuous interventions of the higher layers, which are thus unable to follow the strictly optimal continuous time pattern), but may possibly be obtained

in a cheaper manner. The clue must, therefore, be the tradeoff between loss of optimality and the computational and informational cost of control. A problem of that kind is most sound technically and also most difficult to formalize in a way permitting effective solutions.

The multilayer concept can also be related to a control system where the dynamic optimization horizon has been divided, as illustrated in Figure 3. The following two features are now essential:

- each of the layers is considering a different time horizon; highest layer has the longest horizon;
- the "model" used at each layer or the degree to which details of the problem are considered is also different: the least detailed consideration is done at the top layer.

Control structures of the kind presented in Figure 3 have been most widely applied in practice, for example in industrial or other organizations, in production scheduling and control, etc. These applications seem to be rather ahead of formal theory, which in this case - as it also was for Figure 2 - fails to supply explicit methods to design such systems. For example, we would like to determine how many layers to form, what horizon to consider at each layer, how simple the models may be, etc. Except for some rather academic examples, these questions can be answered only on the case by case basis.

The *multilevel concept* in hierarchical control systems has been derived from decomposition and coordination methods developed for mathematical programming. We should especially note

the difference between:

(a) decomposition applied to the solution of optimization problems, where we operate with mathematical models only and the goal of decomposition is to save computational effort,

(b) multilevel approach to on-line control, where the following features are important:

- the system is disturbed and the models are inadequate,
- reasonable measurements are available,
- no vital constraints can be violated,
- computing time is limited.

The "Mathematical Programming" decomposition can be applied directly only as an open-loop control (as a rule - with model adaptation) as shown in Figure 4. But here in fact any method of solving the optimization problem can be used and the results achieved will be all the same - all depending on model accuracy. Nevertheless, the study and development of decomposition methods in programming is highly desirable even from the point of view of control. The open-loop structures like Figure 4 should not be dismissed, since they offer advantages of inherent stability and fast operation. Structuring the optimization algorithm in Figure 4 as a multi-level one may also be desirable for the reasons of software (computational economy) as well as hardware (multi-computer arrangement) considerations. Nevertheless, in the rest of the paper we shall be paying much more attention to those multilevel structures of control where feedback information from the real system is used to improve control decisions. Figure 5 illustrates what we mean.

It is essential to see in Figure 5 that we have *local decision units* and a *coordinator*, whose aim it is to influence the local decision units in such a way as to achieve the overall goal. All these units will use mathematical models of the systems elements, but they may also use actual observations.

If we now look at the hierarchical systems in the whole (compare Figures 2,3 and 5) we see that they have one feature in common: the *decision making* has been *divided*. Moreover, it has been divided in a way leading to hierarchical dependence. This means, that *there exist several decision units in the structure, but only a part of them have access to the control variables of the process. The others are at a higher level of the hierarchy - they may define the tasks and coordinate the lower level units, but they do not override their decisions.*

We should say a few words about *why* the decision making should be divided and why we should have a hierarchy, as opposed to parallel decision units.

Some of the more general reasons were mentioned at the beginning. Let us add, that in industrial control applications the trend towards hierarchical control can also be associated with the technology of control computers.

Namely, the advent of microprocessors makes control computers so cheap and handy that they may be introduced almost at every place in the process, where previously the so-called analog controllers had been used. The information processing capabilities of the microprocessors are much more than needed to replace the analog controllers and they may easily be

assigned an appropriate part of the higher layer control functions, e.g. optimization.

All the above speaks for *decentralization* but it does not say yet why should we have *coordination* of the decentralized decision units. The general answer would be that in several cases the performance of a controlled system with a *purely decentralized* control structure may be unsatisfactory, if its internal interconnections are intensive.

Some of the other reasons for using hierarchical rather than centralized structures of control are:

- the desire to increase the overall system reliability ("robustness": will the system survive if one of the control units breaks down),
- the possibility that the system as a whole will be less sensitive to disturbance inputs, if the local units can be made to respond faster and more adequately than a more remote central decision unit.

The tasks of the *theory of hierarchical control systems* may be twofold: we may be interested in the *design* of such systems for industrial or organizational applications, or we may want to know how an existing hierarchical control system *behaves*. The second case applies to economic systems, for example. The focus of the two cases differs very much, as do the permissible simplifications and assumptions that can be made in the investigation.

For example, in relation to the multilevel system of Figure 5, if we want to *design* such a system, we would have to deal with questions like:

- what kind of coordination instruments should the coordinator be allowed to use and how will his decisions enter into the local decision processes?
- how much feedback information should be made available to the coordinator and to the local decision units?
- what procedures (algorithms) shall be used at each level, respectively, in determining the coordinating decisions and the control decisions (control actions) to be applied to the real system?
- how will the whole of the structure perform when disturbances appear?
- what will be the impact of distortion of information transmitted between the levels? etc. etc.

In an existing system some of the above questions were answered, when the system was designed and put into operation. However, we are often interested in modifying and improving an existing system, and the same system design problems will come up again.

3. Multilayer systems

3.1 *Temporal multilayer hierarchy*

Let us discuss the two principal varieties of multilayer systems in some more detail, starting with the *temporal multilayer hierarchy*.

One of the most essential features of a dynamic optimization problem is that, for the control or decision to be taken and applied at the current time t, we consider the future behavior of the system. We deal with the *optimization horizon*. As mentioned (see Fig.3), the optimization horizon can be divided, which results in a specific hierarchical system.

Let us exemplify the operation of such a hierarchy by reference to control of a water supply system with retention reservoirs. The top layer would determine, at time zero, the optimal state trajectory of water resource up to a final time, e.g. equal one year. This would be a long horizon planning and the model simplification mentioned before could consist in dropping the medium-size and small reservoirs, or lumping them into a single equivalent capacity. The model would be low-order, having only a few state variables (the larger water retentions). We can see on this example why it is necessary to consider the future when the present decision is being made and we deal with a dynamical system: the amount of water which we have in the retention at any time t may be used right away, or left for the next week, or left for the next month, etc., etc. Note that the outflow rate which we command today will have an influence on the retention state at any future t.

It might be good to note the difference between control of a dynamic system and control of a static time-varying system. In the latter case nothing is being accumulated or stored and the present control decision does not influence the future. An example might be the situation when we consider supplying water to a user who has a time-varying demand, but no storage facility of any kind.

The long horizon solution does supply the state trajectory for the whole year, therefore also for the first month, but this solution is not detailed enough: the states of medium size and small reservoirs are not specified. The intermediate layer would now be acting, computing - at time zero - the more detailed state trajectory for the month.

From this trajectory we could derive the optimization problem for the first day of system operation. Here, in the lowest layer, an all-detailed model must be considered, since we have to specify for each individual reservoir what is to be done, for example what should be the actual outflow rate. We consider each reservoir in detail, but we have here the advantage of considering a short horizon.

Let us now describe this hierarchy more formally.

Assume the water system problem was

$$\text{maximize} \quad \int_{t_0}^{t_f} f_0^1(x^1(t), m^1(t), z^1(t))\,dt \,,$$

and the system is described by state equation

$$\dot{x}^1(t) = f^1(x^1(t), m^1(t), z^1(t)) \,.$$

In those expressions x^1 stands for the vector of state variables, m^1 for vector of manipulated variables (control variables), z^1 for vector of disturbances (the exogenous inputs). The state $x^1(t_0)$ is given and $x^1(t_f)$ is free or specified as the required water reserve at $t = t_f$.

Let us divide this problem between three layers.

(i) Top layer (long horizon)

$$\text{maximize} \int_{t_0}^{t_f} f_0^3(x^3(t), m^3(t), z^3(t))dt$$

with

$\dot{x}^3(t) = f^3(x^3(t), m^3(t), z^3(t))$, $x^3(t_0)$ given, $x^3(t_f)$ free or specified like in the above.

Here, x^3 is the simplified (aggregated) state vector, m^3 is simplified control vector, z^3 is simplified or equivalent disturbance.

Solution to long-horizon problem determines, among other things, state $\hat{x}^3(t_f')$ i.e., the state to be obtained at time t_f' (this could be one month in the water system example). This state is a target condition for the problem considered at the layer next down the hierarchy.

(ii) Intermediate layer (medium horizon)

$$\text{maximize} \int_{t_0}^{t_f'} f_0^2(x^2(t), m^2(t), z^2(t))dt$$

with

$$\dot{x}^2(t) = f^2(x^2(t), m^2(t), z^2(t)),\ x^2(t_0) \text{ given},\ x^2(t_f') \text{ given by } \hat{x}^3(t_f') \ .$$

The final state requirement cannot be introduced directly because vector x^2 has a lower dimension than x^3, according to the principle of increasing the number of details in the model as we step down the hierarchy. We must introduce a function γ^2 and require

$$\gamma^2(x^2(t_f')) = \hat{x}^3(t_f') \ .$$

Function γ^2 is related to model simplifications (aggregation of state as we go upwards) and should be determined together with those simplifications.

Solution to the intermediate layer problem determines among other things the value of $\hat{x}^2(x_f'')$ i.e., the state to be obtained at $t = t_f''$ (this could be one day in the water system example).

(iii) The lowest layer (short horizon)

$$\text{maximize} \int_{t_0}^{t_f''} f_0^1(x^1(t), m^1(t), z^1(t))dt$$

with

$$\dot{x}^1(t) = f^1(x^1(t), m^1(t), z^1(t)), x^1(t_0) \text{ given},\ x^1(t_f'') \text{ given by } \gamma^1(x^1(t_f'')) = \hat{x}^2(t_f'') .$$

We drop explanation of the details of this problem since they are similar to those of previous problems.

Note only that the functions $f_0^1(\cdot)$ used here are the same as in the original problem (this means "full" model), but the time horizon is considerably shorter. The lowest layer solution determines the control actions $\hat{m}^1$ to be taken in the real system.

Consult Fig. 6 for a sketch of the three layers and their linkages.

Please note that if no model simplifications were used the multilayer structure would make little sense. If we used the full model at the top layer, we would have determined the trajectory $\hat{x}^1$ and the control actions $\hat{m}^1$ right there, and moreover not only for the interval (t_0, t_f'') but for the whole horizon (t_o, t_f). The lower layers would only repeat the same calculations.

Let us now introduce feedback, trying to use the actual system operation to improve control. One of the possibilities would be to use the really obtained $x^1(t_f'')$ as the initial condition for the intermediate layer problem. This means that at time t_f'' (one day in the example) we re-solve the intermediate layer problem (ii) using as initial condition:

$$x^2(t_f'') = \gamma^1(x^1(t_f'')) \ .$$

After the second day, i.e., at $t = 2t_f''$ we would use

$$x^2(2t_f'') = \gamma^1(x^1(2t_f'')) \ ,$$

and so on.

This way of using feedback is often referred to as "repetitive optimization", because the computational (open-loop) solution will be repeated many times in course of the control system operation.

The same feedback principle could be used to link feedback information up to the higher layers, with a decreased repetition rate. We shall refer to this concept of feedback when dealing with dynamic coordination in multilevel systems.

Consider what would be obtained if we used no feedback in form of really achieved states. The system would be a multilayer structure but its performance might be unnecessarily deteriorated. Note that without any updating the case would correspond to calculation of the targets for all days of the year being done at time zero, thus depending entirely on the accuracy of the model and prediction of environment behavior. The prediction itself calls for repetition of the optimization calculation at appropriate intervals. Dropping the feedback would be a waste of available information.

Needless to say that feedback would be redundant in the case where the model used at lowest layer would exactly describe the reality, inclusive of all disturbances - but this is not likely to happen.

An example of existing multilayer hierarchy is shown in Fig. 7, based on a state-of-the-art report on integrated control in steel industries (IIASA CP-76-13). We can see there how the time horizon gets shorter when we step down from long-range corporate planning to process control. It is also obvious that the problems considered at the top do not encompass the details.

On the contrary, at the bottom level each piece of steel must receive individual consideration, because the final action (manipulation) must be specified here.

It is a proper time now to ask the question if the top level model can really be an aggregated one and how aggregated it can be. A qualitative answer is as follows: the details of the present state have little influence on the distant future, and also: the prediction of details for distant future makes no sense, because it cannot be reliable. Quantitative answers are possible only for specific cases.

The multilayer hierarchy of Fig.3,6 or 7 made use of different optimization horizons; it may be appropriate to say a few words about the choice of horizon in a control problem.

Roughly speaking, we may distinguish two kinds of dynamic optimization problems:

(i) problems where the time horizon is implied by the problem itself,

(ii) problems where the choice has to be made by the problem solver.

Examples of the first variety are: a ship's cruise from harbor A to B, spaceship flight to the moon, one batch in an oxygen steel making converter.

Examples of the second kind could be: operation of an electric power system, a continuous production process, operation of a shipping company, operation of steel making shop.

For the problems of the second kind it is necessary to choose an optimization horizon. We are going to show, in a rather qualitative way, how this choice depends upon two

principal factors: dynamics of the system and characteristics of the disturbance.

Assume we have first chosen a fairly long time horizon t_f and formulated a problem

$$\text{maximize } Q = \int_{t_0}^{t_f} f_0(x(t),m(t),z(t))dt \ ,$$

for a system described by

$$\dot{x}(t) = f(x(t),m(t),z(t))$$

with $x(t_0)$ known and $x(t_f)$ free.

Because of the disturbance z this is a stochastic optimization problem and we should speak about maximizing expected value of Q, for example. Let us drop this accurate but rarely feasible approach and assume that we convert the problem into a deterministic one by taking $\bar{z}$, a predicted value of z, as if it was a known input. Assume we have got the solution: state trajectory $\hat{x}$ and control $\hat{m}$ for the interval (t_o,t_f).

Fig. 8 shows what is expected to result in terms of a predicted $\bar{z}$ and of the solution $\hat{x}$. There seem to be two crucial points here. First, a predicted $\bar{z}$ will start from the actually known value $z(t_0)$ and always end up in a shape which is either constant or periodic. This is because when the "correlation time" elapses the initial value $z(t_0)$ has no influence on the estimated value of the disturbance and what we get as $\bar{z}$ must be the mean value or a function with periodic properties. Secondly, if (t_0,t_f) is large enough (say one year for an industrial plant) we expect that in a period far from $t = t_0$

the initial state $x(t_0)$ has no influence any more on the optimal values $x(t)$. If we are still long before $t = t_f$, the final conditions have no influence either.

Thus what we expect is that the *calculated* at $t = t_0$ optimal trajectory $\hat{x}$ will exhibit a quasi-steady state interval (t_1,t_2) where $\hat{x}$ depends only on $\bar{z}$. But since $\bar{z}$ is going to be either constant or periodic, $\hat{x}$ will also do so (a more thorough discussion of it can be found elsewhere (Findeisen 1974).

The above has been a qualitative consideration, but it allows us to explain why practically we would be allowed to consider only (t_0,t_1) as the optimization horizon for our problem. Note that if we decide to use this short horizon we must formulate our problem as one with given final state:

$$\text{maximize } Q = \int_{t_0}^{t_1} f_0(x(t),m(t),\bar{z}(t))dt$$

for a system described by

$$\dot{x}(t) = f(x(t),m(t),\bar{z}(t))$$

with $x(t_0)$ known and $x(t_1)$ given as $\hat{x}(t_1)$ from Fig. 8.

The next clue is that the solution $\hat{x}$ got from this problem and the control $\hat{m}$ are correct only for a short portion of (t_0,t_1) due to the fact that real z will not follow the prediction $\bar{z}$. Thus we have to repeat the solution after some interval δ much shorter than (t_1-t_0), using the new initial values $z(t_o+\delta)$ and $x(t_o+\delta)$. The horizon should now reach to $t_1+\delta$. We have a *floating horizon* or *shifted horizon* control scheme.

It is relatively easy to verify our reasoning by a linear-quadratic problem study, by simulation or by just imagining how some real systems operate.

If we want a conclusion to be stated very briefly we can say: "the optimization horizon is long enough if it permits to take a proper control decision at $t = t_0$".

3.2 *Functional multilayer hierarchy. Stabilization and optimization layers*

The Introduction has explained very briefly (see Fig.2) what we intend to achieve by a functional multilayer hierarchy: a reduction in the frequency and hence in the effort of making control decisions.

Let us discuss the division of control between the first two layers: stabilization(direct control, follow-up control) and optimization, see Fig.2.

Assume that for a dynamic system described by

$$\dot{x}(t) = f(x(t), m(t), z(t)) \ ,$$

we have made a choice as to what variables of the plant should become the controlled variables, see Fig.9. We do it by setting up some functions $h(\cdot)$, relating $c(t)$ to the values of $x(t), m(t)$ at the same time instant

$$c(t) = h(x(t), m(t)) \ .$$

We will assume that c are directly measured (observed).

Functions $h(\cdot)$ would be identities $c \triangleq x$ if we chose the state vector itself as controlled variables - but this choice may be neither possible nor desired and a more general form expressed by function $h(\cdot)$ is appropriate.

The direct control layer (Fig. 9) will have the task of providing a follow-up of the controlled variables c with respect to their set-points (desired values) c_d:

DIRECT CONTROL LAYER: provide for $c = c_d$.

The optimization layer has to impose c_d which would maximize the performance index of the controlled system ("plant" in the industrial context):

OPTIMIZATION LAYER: determine c_d such as to maximize Q .

Note that Q has to be performance assigned to the operation of the controlled system itself, for example the chemical reactor's yield, with no consideration yet of the controllers or control structure. In other words Q is performance measure which we should know from the "user" of the system.

The question is how to choose the controlled variables c, that is how to structure the functions h(•). It is all too easy to say that the choice should be such as to bring no deterioration of the control result achieved in the two-layer system as compared to a direct optimization. It should be

$$\max_{c_d} Q = \max_{m} Q \quad ,$$

where the number on the left is plant performance achieved with the two-layer system of Fig.9 and the number on the right is the maximum achievable performance of the plant itself, since it involves directly the manipulated inputs that are available.

In order to get some more constructive indications let us require that a setting of c_d should uniquely determine both state x and control m which will result in the system of Fig.9 when a c_d is imposed. Since we are interested in getting optimal values x,m let us demand the following property:

$$c = \hat{c}_d \Rightarrow x = \hat{x},\ m = \hat{m} \quad .$$

A trivial solution and a wrong choice of controlled variables could be $c \triangleq m$. Imposing $m = \hat{m}$ on the plant would certainly do the job. It is a poor choice, however, because the state x that results from an applied m depends also on the initial condition $x(t_0)$ - the optimizer which sets c_d would have to know $x(t_0)$.

A trivial example explains the pitfall. Assume we made a two-layer system to control a liquid tank using two flow controllers as in Fig. 10. We delegate to the optimizer the task of determining the optimal flows, F_{1d} and F_{2d}. The optimizer would have no idea of what level x will be established in the tank, unless it memorized $x(t_0)$ and all the past actions. We can see it better while thinking of a steady-state: if the optimizer would impose *correct* steady-state optimal values $F_{1d} = F_{2d} = \hat{F}_d$, it still *would not* determine the steady level x which will result in the tank.

Let us therefore require that the choice of c should free the optimizer from the necessity to know the initial condition:

$$c(t) = c_d(t) \Rightarrow x(t) = \hat{x}(t), m(t) = \hat{m}(t), \forall t \geq t_1 > t_0$$

and the implications shall hold for any $x(t_0)$.

An example of what we aim at may be best given by considering that we want a steady-state $x(t) = x = \text{const}$ to be obtained in the system, while the system is subjected to a constant, although unknown disturbance $z(t) = z$. In that case also m and $c = c_d$ will not be time-varying. The state equations of the plant reduce to

$$f_j(x,m,z) = 0, \qquad j = 1,\ldots,\dim x \ , \tag{i}$$

due to the fact that $\dot{x}(t) = 0$, and if we add the equation which is set up by our choice of the controlled variables

$$h_i(x,m) = c_i, \qquad i = 1,\ldots,\dim c \ , \tag{ii}$$

we have a set of equations (i) and (ii) for which we desire that x,m as the dependent variables be uniquely determined by c. But we also want (i) and (ii) to be a non-contradictory set of equations; their number should not exceed the number of dependent variables x,m and thus we arrive at the requirement that dim c = dim m: the number of controlled variables should be equal to the number of manipulated inputs. Then,from the implicit function theorem, it is sufficient for the uniqueness of x,m that f_j, h_i are continuously differentiable, and

$$\det \begin{bmatrix} \dfrac{\partial f_j}{\partial x_k} & \dfrac{\partial f_j}{\partial m_k} \\ \dfrac{\partial h_i}{\partial x_k} & \dfrac{\partial h_i}{\partial m_k} \end{bmatrix} \neq 0$$

We leave it to the reader to verify that the system of Figure 10 does not comply with the above demand.

We should warn the reader of a possible misinterpretation of our argument. We have shown the conditions under which steady-state x,m resulting in the control system will be single-valued functions of c, but these functions may still contain z as a parameter. In other words, we did not say that a certain value of c will enforce the value of x,m in the plant, irrespectively of the disturbance. If, for example, we are interested in enforcing the value of state, we could choose $c \triangleq x$. But note that this may be not entirely feasible if we have too few manipulated inputs (remember that dim c = dim m).

The structure of Figure 9 can of course also be thought of as operating when the plant state x is time-varying. Then we should write, instead of (i) and (ii):

$$\dot{x}_j(t) = f_j((t),m(t),z(t)), \qquad j = 1,,,,,\text{dim } x \ , \qquad \text{(ia)}$$

$$h_i(x(t),m(t)) = c_i(t), \qquad j = 1,...,\text{dim } c \ . \qquad \text{(iia)}$$

The value of state at time t, that is x(t), will still depend upon the enforced $c(t) = c_d(t)$, but the dependence involves also $\dot{x}(t)$. This means that in order to obtain a certain state x(t) we must take into account the initial state $x(t_o)$, disturbance input over the interval $[t_o,t]$, $z_{[t_o,t]}$, and appropriately shape the control decision $c_{d[t_o,t]}$.

If we want to enforce the value of state x(t) in spite of the disturbances and without dependence on the initial state, we must investigate the *follow-up controllability:* is it possible, using the input m, to cause state x to follow a desired trajectory x_d?

Assume the follow-up has been achieved, that is $x(t) = x_d(t)$, $\dot{x}(t) = \dot{x}_d(t)$, $\forall t$. Then the state equations give

$$f_j(x_d(t), m(t), z(t)) - \dot{x}_d(t) = 0, \quad j = 1, \ldots, \dim x \quad , \qquad \text{(iii)}$$

We should note the meaning of (iii). Disturbance z is varying in time and its value z(t) is random. If (iii) has to hold we have to adjust m(t) so as to offset the influence of z(t). This must of course require certain properties of the functions $f_j(\cdot)$ and we also expect to have enough manipulated inputs. The requirements will be met if the set of equations (iii) will define m(t) as single-valued functions of z(t). The conditions for this are that $f_j(\cdot)$ are continuously differentiable and moreover that,

$$\operatorname{rank} \left[\frac{\partial f_j}{\partial m_k} \right] \geq \dim x \quad .$$

This implies $\dim m \geq \dim x$. We should note that the actual value m(t), as required by the disturbance z(t), should never lie on the boundary of the constraint set of manipulated inputs. Physically it means that we must always have the possibility to adjust m(t) up or down in order to offset the influence of the random disturbance. The actual value of this required reserve or margin depends on the range of possible disturbances. Any control practitioner knows this as an obvious thing.

Remember that we have set a requirement related to controllability, that is to the properties of the plant itself.

Controllability does not say how to generate control m such that $x = x_d$, it tells only that this control exists. If we decide to build a feedback control system as shown in Fig. 9 we have to choose the controlled variables c in an appropriate way. For the dynamic follow-up to be enforced by the condition $c = c_d$, the choice would have to be $c \hat{=} x$, that is the state variables themselves (as opposed to $c = h(x,m)$ which was all right for steady-state uniqueness of x).

The choice of controlled variables has been till now discussed from the point of view of the "uniqueness" property: how to choose c in such a way that when $c = c_d$ will be enforced, some well-defined values x,m will result in the plant. We have done this for the plant described by ordinary differential equations. An extension of this consideration to distributed parameter plants with lumped manipulated inputs is possible.

We turn now to the more spectacular aspect of choosing the controlled variables: can we choose them in a way permitting to reduce or to entirely avoid the on-line optimization effort, that is to eliminate the optimization layer in Fig. 9, leaving only the follow-up control?

To make the argument easier let us consider steady-state optimization.

For a plant

$$f_j(x,m,z) = 0, \qquad j = 1,\ldots,\dim x$$

we are given the task

$$\text{maximize } Q = f_0(x,m,z)$$

subject to inequality constraints

$$g_i(x,m) \leq b_i, \qquad i = 1,...$$

Assume the solution is $(\hat{x},\hat{m})$. At point $(\hat{x},\hat{m})$ some of the inequality constraints become equalities (active constraints), and other inequalities are irrelevant. Thus at $(\hat{x},\hat{m})$ we have a system of equations:

$$f_j(\hat{x},\hat{m},z) = 0, \qquad j = 1,...,\dim x$$

$$g_i(\hat{x},\hat{m}) = b_i, \qquad i = 1,..., k \leq \dim m \ .$$

If it happens that k = dim m then the rule is simple: choose the controlled variables as follows:

$$h_i(\cdot) = g_i(\cdot), \qquad i = 1,..., \dim m \ ,$$

$$\hat{c}_{di} = b_i \ .$$

This simply says that you put the controllers "on guard" that the plant variables (x,m) are kept to the appropriate border lines of the constraint set.

Note two things:

(i) we have assumed $g_i(x,m)$ and not $g_i(x,m,z)$, i.e., the disturbance did not affect boundaries of the constraint set;

(ii) we have assumed k = dim m (the number of active constraints equal to the number of controls), and we also failed to consider that even in such a case the solution $(\hat{x},\hat{m})$ may lie in different "corners" of the constraint set for different z.

Even under these assumptions, however, the case makes sense in many practical applications, since solutions to constrained optimization problems tend to lie on the boundaries.

For example, the yield of a continuous-flow stirred-tank chemical reactor would increase with the volume contained in the tank. This volume is obviously constrained by tank capacity, therefore, the control system design would result in implementing a level controller and in setting the desired value of the level at the full capacity. The level controller would perform all the current control, by adjusting inflow or outflow to keep the level. No on-line optimization is necessary.

We have mentioned already in the Introduction that the approach we have taken by letting the "direct controller" make current control decisions and providing for an upper level to set a rule or goal to which the direct control has to keep, has more than only industrial applications. It is also clear that a rule or goal does not have to be changed as often as the current decisions and hence a two-layer structure makes sense.

If the solution $(\hat{x},\hat{m})$ fails to lie on the boundary of the constraint set, or the number of active constraints $k < \dim m$, we may still look to structure the functions $h_i(\cdot)$ in such a way as to make the optimal value c_d independent of disturbances z.

The way to consider this may be as follows. We have solutions $\hat{m} = \hat{m}(z)$ and $\hat{x} = \hat{x}(z)$. Put them into the functions $h_j(\cdot)$ for $j = k + 1,\ldots, \dim m$:

$$h_j(\hat{x},\hat{m}) = h_j(\hat{x}(z),\hat{m}(z)), \qquad j = k + 1,\ldots, \dim m \quad .$$

By an appropriate choice of $h_j(\cdot)$ we may succeed in getting

$$\frac{\partial h_j}{\partial z} = 0, \qquad j = k + 1, \ldots$$

in the envisaged range of disturbances z.

We turn now to a more elaborate example of building-up a two-layer system.

3.3 *Example of two-layer control*

Consider a stirred-tank continuous-flow reactor presented in Fig. 11. Some material B inflows at rate F_B and has temperature T_B, material A inflows with F_A and T_A, mixing and reaction $A \rightarrow B$ takes place in the vessel, resulting in a concentration C_A. Heat input H is needed for temperature T to be obtained in the reactor. Outflow F_D carries the mixture of A and B out of the vessel. We want to provide a control structure that would optimize the operation of this reactor, having F_A and H as manipulated inputs. Let us do it in some orderly steps.

(i) Describe the plant

There will be three state variables and state equations:

$$\dot{W} = f_1(\cdot) = F_A + F_B - F_D$$

$$\dot{C}_A = f_2(\cdot)$$

$$\dot{T} = f_3(\cdot) \ .$$

We drop the detailed structure of the functions $f_2(\cdot)$, $f_3(\cdot)$ because it is not important for the example.

(ii) Formulate optimization problem

Assume we want to maximize production less the cost of heating:

$$\text{maximize } Q = \frac{1}{t_2-t_1} \int_{t_1}^{t_2} [(1-C_A)F_D - \psi(T)]dt \quad ,$$

where $\psi(T)$ expresses the cost of reaching temperature T.

There will be inequality constraints

$$W \leq W_m, \quad C_A \leq C_{Am}, \quad T \leq T_m \quad ,$$

and we also have to consider the state equations and initial and final conditions.

If there are reasons to assume that the optimal operation of the reactor is steady-state, x = const, then the plant equations reduce to

$$f_1(\cdot) = F_A + F_B - F_D = 0$$

$$f_2(\cdot) = 0$$

$$f_3(\cdot) = 0$$

and the optimization goal would be

$$\text{maximize } Q = (1 - C_A)F_D - \psi(T) \quad .$$

(iii) Solve optimization problem

Assume the optimization problem has been solved and the results are (the problem has really been solved for a full example):

$$\hat{W} = W_M$$

$$\hat{C}_A = C_{Am} \quad \text{if } z \varepsilon Z_1,\ \hat{C}_A = \phi_1(z) < C_{Am} \text{ otherwise}$$

$$\hat{T} = \phi_2(z) < T_m \quad \text{if } z \varepsilon Z_1,\ \hat{T} = T_m \text{ otherwise}$$

$$\hat{F}_A = \phi_3(z)$$

$$\hat{H} = \phi_4(z)$$

where z stands for disturbance vector (F_B, F_D, T_A, T_B) and Z_1 is a certain set in z-space, that is a certain range of disturbance values.

(iv) Examine the solution and choose control structure

Let us make a wrong step and choose as controlled variables the flows F_A, H. We would then: fail to get a uniquely determined steady-state volume W in the tank (a check on determinant condition would show it) and also the optimizer which sets the desired F_{Ad}, H_d would have to know disturbance vector z and functions $\phi_3(\cdot), \phi_4(\cdot)$. Note that this would involve an accurate knowledge of the state equations of the plant.

Inspection of optimization solution reveals volume W as a first-choice candidate to become controlled variable. The optimal W is W_m under all circumstances, no on-line optimization will be required, and no knowledge of plant state equations.

The second choice (we shall have two controlled variables since we have two manipulated inputs) could be either concentration C_A or temperature T.

Let us consult Fig.12 for a discussion. We have displayed there the feasible set in (W, C_A) plane and shown where the optimal solution lies in the two cases, that is when $z \varepsilon Z_1$ (point 1)

and in the other case (point 2). Note that solution is in a corner of the constraint set, but unfortunately not in the same corner for all z. Consider that you may:

- take C_A as a controlled variable and ask the optimizer to watch disturbances z and perform the following

  $C_{Ad} = C_{Am}$ for $z \in Z_1$

  $C_{Ad} = \phi_1(z)$ otherwise

  whereby a knowledge of the function $\phi_1(\cdot)$ is required,
- or take C_A as a controlled variable when $z \in Z_1$ and then set $C_{Ad} = C_{Am}$, whereby for $z \notin Z_1$ you would switch to T as controlled variable with a setting $T_d = T_m$. In this case the second-layer control would consist in performing the switching, that is, in detecting if $z \in Z_1$. This may be easier to do than to know the function $\phi_1(\cdot)$ which was required in the first alternative.

3.4 *The relevance of steady-state optimization*

Steady-state optimization, following the structure of Fig.9 is a quite common practice. It might be worthwhile to consider when it is really appropriate. If we exclude the cases where the *exact* solution for the optimal state is $\hat{x}$ = const, we may think of the remaining cases in the following way.

Let (a) in Fig. 13 be the optimal trajectory of a plant over optimization horizon (t_0, t_1).

Assume we control the plant by a two-layer system, have x as controlled variables, and choose to change desired value x_d at intervals T, being a small fraction of (t_0, t_1). Then (b) is

the plot of $x_d(t)$. Note we have thus decided to be non-optimal because x_d should be shaped like (a), and not be a step-wise changing function. Note also that the step values of x_d would have to be calculated from a dynamic (although discrete) optimization problem.

Now let us look at the way in which the real x will follow the step-wise changing x_d in the direct control system, compare Figure 9. In case (c), Fig.13, x almost immediately follows x_d. In case (d) the dynamics are apparently slow and the following of x_d cannot be assumed.

It is only in case (c) of Fig. 13 that we may be allowed to assume that state x is *practically constant* over periods T, thus permitting to set $\dot{x} = 0$ into the state equations and calculate the step value of x_d from a steady-state optimization problem.

The question is when will case (c) occur. By no means are we free to choose the interval T at will. We must relate it to the optimization horizon (t_0,t_1). Interval T would be a suitable fraction of this (1/10 or 1/50 for example). And here is the qualitative answer to the main question: if (t_0,t_1) has resulted from slow disturbances acting on a fast system, case (c) may take place, that is we may be allowed to calculate a step of x_d under steady-state assumption.

The importance of the possibility to replace the original dynamic optimization problem by an almost equivalent static optimization done in the two-layer system cannot be overemphasized. The reason is of computational nature: dynamic problems need much more effort to solve and for many life-size control tasks,

for example for a chemical plant, may be practically unsolvable, in the time being available. On the other hand, the operation of many plants is close to steady-state and the optimization of set-points done by static optimization is quite close to the desired result.

We devote in this paper a considerable space to steady-state on-line optimization structures. It is the more justified that the procedures for static optimization are principally different from those suitable for dynamic control, if feedback from the process is being used.

3.5 *Remarks on adaptation layer*

Let us come back to Fig.2. We have presented there an "adaptation layer" and assigned to it the task of readjusting some parameters β which influence the setting of the value of c_d. Assume this setting is done by means of a fixed function $k(\cdot)$:

$$c_d = k(\beta, z)$$

where z stands for the disturbance acting on the plant. We assume at this point, that it is measured and thus it can enter the function $k(\cdot)$.

We may of course assume existence of the strictly optimal value of c_d, referred to as $\hat{c}_d(z)$. With $\hat{c}_d(z)$ we would get a top value of performance denoted by $Q(\hat{c}_d(z))$. It represents the full plant possibilities.

Optimal values of β in the optimizer's algorithm could be found by solving the problem

$$\underset{\beta}{\text{minimize}} \quad E_z ||c_d(z) - k(\beta, z)|| \quad .$$

We drop discussion of this formulation because we should rather assume that the optimizer has only a restricted information about z, denoted z* (it could for example be samples of z taken at some intervals). This leads to $c_d = k(\beta,z^*)$ and the parameter adjustment problem should now be

$$\underset{\beta}{\text{minimize}} \quad {}_{z,z^*}E \; [Q(\hat{c}_d(z))-Q(k(\beta,z^*))]$$

which means that the choice of β should aim at minimizing the loss of performance with respect to full plant possibilities. An indirect and not equivalent way, but which may be easier to perform would be

$$\underset{\beta}{\text{minimize}} \quad {}_{z,z^*}E||c_d(z)-k(\beta,z^*)|| \quad .$$

Note that we would not be able to get $\beta = \hat{\beta}$ such that $E||\cdot||$ would be zero, since the basis for $k(\beta,\cdot)$ is z* and not z. It means that, with the best possible parameters, the control is inferior to a fully optimal one, the reason being the restricted information.

Our formulations till now apply to adjusting parameters β once, and keeping them constant thereafter for some period of time (it is over this period that the expectations $E||\cdot||$ should be taken).

In some practical adaptive systems we try to obtain the values of parameters of the plant, and thus also the values of β, by some kind of on-line identification procedure. We may refer to it as "on-line parameter estimation". A limit case may be of interest where we would assume that β are estimated continuously. Let us consider what this limit case could supply.

Note that for each z, an optimal value $\hat{\beta}(z)$ maximizing the performance exists and means a perfect control. We must assume, however, that we do not have $\hat{\beta}(z)$ but an estimated value of it, $\tilde{\beta}(z)$. With $\tilde{\beta}(z)$ our optimizing control would be

$$c_d = k(\tilde{\beta}(z), z^*)$$

where we assumed, realistically, that not all z is directly measured and only z* is available as current information.

The application of this control gives a loss of optimality which amounts to

$$E_{z,z^*} [Q(\hat{c}_d(z)) - Q(k(\tilde{\beta}(z), z^*))] \quad .$$

This value could be discussed with respect to the quality of estimating β, insufficiency of disturbance information z*, etc. In other words, it measures the overall efficiency of adaptation.

4. Decomposition and coordination in steady-state control

In this section we shall consider the multilevel control structures shown by Fig.5 in some more detail. One of the points of this and of the next section will be to indicate the practical difference between steady-state and dynamic control structures.

4.1 *Steady-state multilevel control and direct coordination*

Let us first describe the complex system of Fig. 1 more carefully.

Denote for the subsystem i : x_i the state vector, m_i manipulated input, z_i disturbance, u_i input from other subsystems, y_i output connected to other subsystems. The subsystem state equation will then be

$$x_i(t) = \phi_{i[t_0,t]}(x_i(t_0), m_{i[t_0,t]}, u_{i[t_0,t]}, z_{i[t_0,t]}) \quad . \tag{1}$$

For the use of this section we assume (1) to be in the particular form of ordinary differential equation

$$\dot{x}_i(t) = f_i(x_i(t),\ m_i(t),\ u_i(t),\ z_i(t)) \quad . \tag{1'}$$

The output y_i will be related to (x_i, m_i, u_i, z_i) by output equation

$$y_i(t) = g_i(x_i(t), m_i(t), u_i(t), z_i(t)) \quad . \tag{2}$$

Now assume that the first-layer or direct controls are added to the subsystem such that the following is enforced (see the previous Section for this idea)

$$c_i(t) = h_i(x_i(t), m_i(t), u_i(t)) = c_{di}(t) \quad . \tag{3}$$

Assume we are in steady-state, $x_i(t)=0,\forall t$, and the functions $h_i(\cdot)$ have been chosen properly so as to ensure uniqueness of the state x_{si} and manipulated output $m_i(t)$ in response to the imposed $c_i(t)$ and $u_i(t)$, with $z_i(t)$ as a parameter. Then (1') becomes

$$f_i(x_{si},m_i(t),u_i(t),z_i(t)) = 0 \qquad (4)$$

and (4) along with (3) provides for $x_{si},m_i(t)$ to be functions of $c_i(t)$. Therefore (2) becomes the following input-output dependence:

$$y_i(t) = F_i(c_i(t),u_i(t),z_i(t)) \ . \qquad (5)$$

Eqn. (5) is a relation between the instantaneous values. We have obtained it by assuming the system to be in steady-state, $x(t) = x_s = \text{const}$. In the steady-state the system ceases to be a dynamical one, because there is no change in accumulations.

We can consider the state to be time-varying; then (5) can be true only under the assumption that the actual state x is always enforced, that is, it follows the desired state trajectory x_{di}. As mentioned in Section 2.2 this is possible if the subsystem complies with the follow-up controllability condition and if $h_i(\cdot)$ is chosen for example such that $c_i \triangleq x_i$.

In the general case of time-varying state we would have to put into (2) the formula (1) for $x_i(t)$, which makes y(t) dependent upon initial state $x_i(t_0)$ and upon the inputs over interval $[t_0,t]$, that is upon $m_{i[t_0,t]},u_{i[t_0,t]},z_{i[t_0,t]}$. The Existence of an appropriate equation (3) allows to eliminate $m_{i[t_0,t]}$ in favor of $c_{i[t_0,t]}$ and thus we become, instead of (5)

$$y_i(t) = F'_{i[t_0,t]}(x_i(t_0), c_{i[t_0,t]}, u_{i[t_0,t]}, z_{i[t_0,t]}) \quad . \qquad (5')$$

The input-output relation in the form (5') is not very convenient for notational reasons. We may tacitly assume the initial state to be known, or we can treat $x_i(t_0)$ as part of the disturbance z_i. If we, additionally, use notation y_i, c_i, u_i, z_i to express time functions (as opposed to their values $y_i(t)$, etc.), then (5') becomes

$$y_i = F_i(c_i, u_i, z_i) \quad . \qquad (5")$$

The important difference with respect to (5) is that (5") denotes a mapping between time functions (describes a dynamical system).

When the subsystem is in steady state, (5) will hold. Its practical meaning is that "the dynamics of the subsystem are suppressed" and that is why we have a static input-output relation. We usually write (5) in abbreviated form, dropping the argument t and sometimes also the disturbance input:

$$y_i = F_i(c_i, u_i), \quad i \in \overline{1,N} \quad . \qquad (6)$$

Note that the form of (6) is similar to (5") and the notation does not indicate whether we describe a static or a dynamic system. This is rather convenient for considerations of general nature, but may also be misleading as the difference tends to be overlooked.

Right below we are going to speak about steady-state and we consider y_i, c_i, u_i to stand for $y_i(t), c_i(t), u_i(t)$.

The interconnections in the system are described by

$$u_i = H_i y, \qquad \text{so that } u = Hy \tag{7}$$

where H_i is part of matrix H.

We assume a "resource constraint" is imposed on the system as a whole

$$\sum_{1}^{N} r_i(c_i,u_i) \leq r_o \tag{8}$$

and also that some local constraints restricting (c_i,u_i) may exist

$$(c_i,u_i) \in CU_i, \quad i \in \overline{1,N} \quad . \tag{9}$$

We further assume that a local performance index (local objective function) is associated with the subsystem

$$Q_i(c_i,u_i), \quad i \in \overline{1,N} \tag{10}$$

whereby a global system performance is also defined and it is

$$Q = \psi(Q_1,Q_2,\ldots,Q_N) \quad . \tag{11}$$

The function ψ is assumed to be strictly order-preserving.

Note that (10) and (11) may result from two practical cases. It might be that there were some local decision makers already in existence and we decided to set up an overall Q to provide for some harmony in their actions. But it also might be that we had overall Q first and we then decided to distribute the decision making among the lower level units.

We are now ready to define the goal of the coordination level: it has to ensure that the overall constraints would be preserved and the overall performance would be extremized.

Coordination will be done by influencing decision making in the local units (and not by overriding control decisions already made).

We start with presenting coordination by *direct method*.

The simplest way to present direct coordination (also called primal or parametric coordination) is to assume that the coordinator would prescribe the outputs y_i, demanding an equality $y_i = y_{di}$. If a resource constraint (8) is present, coordinator would also allocate a value r_{di} to each local problem.

A local decision problem would become

maximize $Q_i(c_i,u_i)$

subject to

$$u_i = H_i y_d$$

$$F_i(c_i,u_i) = y_{di}$$

$$(c_i,u_i) \in CU_i$$

$$r_i(c_i,u_i) \leq r_{di} \quad .$$

When this problem is solved, results depend upon (y_d,r_{di}). Note they depend on the whole y_d, not on y_{di} only, since we had $u_i = H_i y_d$. We denote the results as $\hat{c}_i(y_d,r_{di})$ and $Q_i(\hat{c}_i(y_d,r_{di}),H_i y_d) \triangleq \hat{Q}_i(y_d,r_{di})$.

The *coordination instruments* (y_d,r_d) have to be adjusted to an optimum by solving the problem

$$\underset{(y_d, r_d)}{\text{maximize}}\ Q = \psi(\hat{Q}_1(y_d, r_{d1}), \ldots, \hat{Q}_N(y_d, r_{dN}))$$

subject to

$$r_{d1} + r_{d2} + \ldots + r_{dN} \leq r_o \ .$$

The main difficulty of the method lies in the fact that a local problem may have no solution for some (y_d, r_d) because of its inequality constraints (an output value may be not achievable and the allocated resources inadequate). Therefore the values (y_d, r_d) set by the coordinator must be such as to keep the local problems feasible, $(y_d, r_d) \in YR$, where YR is the feasible set.

The set YR cannot be easily determined, because it implicitly depends on local constraints.

Moreover, the boundaries of set YR may be affected by the disturbances, since these boundaries are related to local constraints and to the element equations. This has the implication that the "coordinator" would have to keep his decisions (y_d, r_d) in a "safe" region of YR, where "safe" would relate to the worst case of system uncertainties. Apart from the difficulty to define the safe region we of course realize that the worst case approach may give the result that the "safe region" is very small or even empty.

Before trying to find a remedy to this situation we shall make some additional remark on the direct method of coordination; namely, this method may entirely fail to be applicable if the number and role of local controls are inadequate.

We note that by prescribing the outputs we also preset the inputs and hence in the local subsystem equation we have only c_i as a free variable:

$$F_i(c_i, H_i y_d, z_i) = y_{di} \quad .$$

Strictly speaking we should consider the interconnected system in the whole, where we have

$$F(c,u,z) = y$$

and with $y = y_d$, $u = Hy$ this gives

$$F(c, Hy_d, z) = y_d \quad .$$

The above equality is to be enforced. This means that c must be available such that a certain system of equations, which we denote as

$$K(c,z) = y_d$$

could be satisfied by adjusting c (the control decision) for any y_d, z in their range envisaged.

The question would be: do we have an adequate number of control variables c_j, j = 1,..., dim c and are they appropriately placed in the system equations?

Let us clarify the implications by an example. Remember the chemical reactor of Fig. 11. The output vector y would in this case be (F_D, C_A, T) since the outflow from the reactor is

characterized by flow rate (F_D), composition (uniquely expressed by C_A) and temperature (T). We have only two manipulated variables F_A, H and hence two controlled variables, say W and C_A. Therefore, dim c = 2 while dim y = 3. We should be unable to prescribe an arbitrary value for the output vector. Indeed, the steady-state equation y = K(c,z) of the reactor inclusive of direct controls would be in scalar notation

$$F_D = z_1$$

$$C_A = C_{Ad}$$

$$T = K_3(W_d, C_{Ad}, z)$$

where z_1 stands for the flow rate demanded (imposed) by the receiving end of the pipe, and z for the whole vector of disturbances. By choosing W_D, C_{Ad} we would be able to steer the output C_A and T, but not F_D. Note that our control influence on the output T is rather complicated and the actual T depends also on disturbances. Nevertheless we can influence it by adjusting W_d, which means that we have "adequate c" for the purpose.

The question of local controls is vital for the direct method. We should, however, consider that in practical cases where this hierarchical structure would be applied, the number of local controls will always exceed the number of outputs which are being prescribed. Otherwise we might doubt if it makes sense to apply the structure: the coordinator could make all the c_i decisions directly.

Let us now come back to the problem posed by the ignorance of the feasible set at the coordination level. A solution is subject of the next subsection.

4.2 *Penalty functions in direct coordination*

We can propose an iterative procedure to be used at the coordination level such that the feasible set YR would not have to be known. The main idea is to use *penalty functions* in the local problems while imposing there the coordinator's demands. If we use penalty function for the matching of the output, the local problem will get the form:

$$\text{maximize } Q_i' = Q_i(c_i,u_i) - K_i(y_i-y_{di})$$

with the substitutions

$$u_i = H_i y_d$$

$$y_i = F_i(c_i,u_i)$$

and subject to constraints

$$(c_i,u_i) \in CU_i$$

$$r_i(c_i,u_i) \leq r_{di} \quad .$$

As can be seen we used penalty function to enforce the condition $y_i = y_{di}$. The resource constraint could also be dealt with by a penalty term, if necessary. Also the substitution $u_i = H_i y_d$ may be, if needed, replaced by penalty term. Interaction input u_i would then become a free decision variable in the local problem.

The result of using penalty formulation is that solution to the local problem would exist even for a non-feasible y_{di}. The demand on the output would simply not be met.

We must now establish a mechanism to let the coordinator know that he is demanding something impossible. We let his optimization become:

$$\underset{y_d, r_d}{\text{maximize}}\ \psi[\hat{Q}_1(y_d, r_{d1}) - K_1(\hat{y}_1 - y_{d1})), \ldots, (\hat{Q}_N(y_d, r_{dN}) - K_N(\hat{y}_N - y_{dN}))]$$

where the clue is that we introduce local performances less the penalty terms. Hence, the coordination iterations will try to adjust y_d so as to reduce the values of penalty terms, whereby the local problems do the same on their part, by influencing $\hat{y}_i$.

It can be shown, under relatively unrestrictive conditions that when the iterations reach their limit where penalty terms vanish, the values y_d obtained there are both feasible and strictly optimal.

Moreover, gradient procedures can be used at the coordination level, while in the pure form of direct method the subsystem results $\hat{Q}_i(y_d, r_{di})$ are, in general, non-differentiable.

4.3 *A mechanistic system or a human decision making hierarchy?*

The reader of the previous text may get confused as to what do our considerations really apply. Let us clarify it as follows:

(i) In the first place, we can obviously think of coordination used in off-line, model-based solving of a set of local problems. This would be "decomposition and coordination in mathematical programming" and it is quite appropriate there to discuss, for example, whether gradient procedures can be used or not.

Should we apply the solution of optimization problem, that is the finally obtained control values $\hat{c}_i$ to a real system, feasibility of the result with respect to the real system (differing from the models) must be considered. The problem of "generating feasible controls" will arise. From the control point of view we would have an open-loop structure.

(ii) In the second place, we can consider the coordination level as acting on local decision makers who control the real system elements and try to comply with the coordinator's demands. Here we may not even know what is the local decision making process. Let us look at this situation by assuming that the coordinator works by iteration; at each step of the iterative procedure the local decision makers "do their best" with respect to the real system inputs. Would we know the algorithm which the local decision maker is using, a discussion of time-behavior of the system from one coordination step to another could be done. Let us only state that this behavior may be unstable due to many separate decision makers acting on the same system. If the system is stable and a steady-state is achieved, the coordinator may make his next step, trying to improve the value of his performance function (whether in the penalty form or without it). Note that in the case where no penalty terms are used the direct coordination can in principle be achieved in one step: the coordinator sets values (y_d, r_d) which should optimize the system according to his best knowledge (i.e. according to the model of the system) and then the

local decision makers do their job by achieving $y_i = y_{di}$ and complying with the resources constraint. It is in this case, however, that y_{di} should be feasible for the real system, otherwise the expectations of the coordinator may not become reality.

If the coordinator's demands are feasible for the real system (for instance because he knows exactly the constraints, or he has decided to move in the "safe region" only), then the iterations of the direct method have the property that the demands are feasible in every step of the iterative procedure. Hence, the direct method is sometimes referred to as "feasible method". As opposed to it, the direct-penalty coordination is using non-feasible demands in course of the iterations. When the local decision maker is trying to comply with a non-feasible demand, his output may violate the constraints related to the input of another subsystem.

(iii) We can also consider a mechanistic decision making hierarchy of control, where we implement certain formal algorithms of decision making at the local level, as well as at the coordination level. It could be an open-loop control structure but this may not be a satisfactory and ultimate solution. The performance of control can be improved by using feedback information; the human decision makers postulated above in (ii) were using such information implicitly. Now we would have to say very explicitly what kind of current information is available and how it is being used in the

formal algorithms. For example we can assume that the real subsystem outputs y_{*i} are measured. Then we can consider them to be used in essentially two ways: in the local algorithm and in the coordination algorithm. The second possibility has been quite satisfactorily explored and is discussed to some extent below. Using this kind of feedback, we are able to obtain coordination algorithms which

- end in a point non-violating the real system constraints (provided they are of the form $(c_i,u_i) \in CU_i$ and $y \in Y$),
- provide for a value of overall performance which is superior to the result of open-loop control.

4.4 *A more comprehensive example*

A typical area of application of steady-state optimal control are the continuous chemical processes.

Let us present how the multilevel approach could be applied to control of an ammonia plant.

(i) Description of the process

Fig. 14 displays the principal parts of the plant. The first is *methane conversion*, where H_2 is gained from the methane and N_2 from atmospheric air, water steam being added to care for stochiometric balance. The second is *conversion* of carbon oxide, where CO is turned to CO_2 (CO could not be removed directly). Then we have *decarbonization*, where CO_2 is removed from the gas stream. At this point there should be no CO or CO_2 present in the gas stream - the rests of them are neutralized by turning them back into methane in the *methanization* part of the plant. The reason for doing it is that CO

and CO_2 are toxic to the catalyst used in the synthesis reactor. The *synthesis reactor* is the last essential part of the plant - here the mixture $3H_2 + N_2$ reacts to $2NH_3$ at high pressure and high temperature. A cooled liquid (essentially pure ammonia) F_a leaves the plant. The characteristic feature of the ammonia synthesis process is that the synthesis reactor works with a recycle, whereby its input flow consists of both the fresh gas and of the recycled gas - the latter with NH_3 removed (transferred to the liquid F_a). The fresh gas, however, contains not only H_2, N_2 but also some "inerts", i.e. components not reacting in the process. They would mainly be argon from the atmospheric air and CH_4 due to the methanization process used for removing the rest CO and CO_2. Inerts are no harm but they would cycle in the synthesis reactor loop endlessly; as new inerts continuously flow in with the fresh gas we would end in a considerable increase of inerts in the loop gas, leaving no space for the useful H_2, N_2. Inerts have to be removed. There is, however, no practical way to remove them selectively and the inert level is kept down by a very simple measure: part of the loop gas is being blown out into the atmosphere as the so-called purge, F_p.

(ii) The optimization problem

Assume we aim at maximizing the steady-state production rate Q of ammonia (in kg/hr). We have

$$(A) \quad Q = F_a - F_a \sum_j r_j$$

where r_j is solubility of j-th component of the circulating gas in liquid ammonia.

In order to get variables of other parts of the plant involved in the expression for Q let us write two mass balance equations.

Overall mass balance of the synthesis loop will be:

$$(B) \quad F_a + F_p = F_s$$

where F_s is the fresh gas inflow.

Mass balance of the inerts in the synthesis loop will be:

$$(C) \quad F_a r_{in} + F_p y_{pi} = F_s y_{si}$$

where r_{in} is solubility of inerts in liquid ammonia, y_{pi} is concentration of inerts in purge gas, y_{si} the same for fresh gas.

The use of (B) and (C) allows to arrive at

$$(D) \quad Q = F_s\left(1 - \frac{y_{si}-r_{in}}{y_{pi}-r_{in}}\right)\left(1 - \sum_j r_j\right) .$$

At this state we assume from physical and chemical knowledge: r_j, r_{in} do not depend on any plant variables, and $y_{pi} > y_{si} > r_{in}$. Under these circumstances we can see that Q is maximized when F_s is maximized, y_{si} is minimized and y_{pi} is maximized (please look at the physical meaning). We thus would have

$$Q = \psi(Q_1,Q_2,Q_3) = bF_s\left(1 - \frac{y_{si}-a}{y_{pi}-a}\right)$$

where a, b are constants. Note ψ is in this case a strictly order-preserving function.

There could be three local problems: maximize F_s, minimize y_{si}, maximize y_{pi}.

Since the local problems are of course interconnected, a coordination will be needed to provide for max Q and preserving all constraints at the same time. In an actual study performed it was assumed that F_s will be given. It was, however, found reasonable to replace y_{si} by two local performance indices, both to be minimized:

$$Q_1 \triangleq y^1_{CH_4} + y^1_{CO}, \quad Q_2 \triangleq y^2_{CH_4}$$

and to form three subsystems as shown in Fig. 15. They have the performance indices Q_1, Q_2 and $Q_3 \triangleq y_{pi}$, respectively.

We denoted by $y^1_{CH_4}$ the concentration of CH_4 at the output of the first subsystem. This CH_4 directly contributes to the inert content in the gas F_s, therefore it makes sense to minimize it right away. The same applies to CO content here, because CO will not be removed in decarbonization. The performance index Q_2 for the second subsystem is CH_4 concentration in the fresh gas stream F_s. This CH_4 involves result of methanization, which had to be done on CO_2. Local control can decrease this CH_4 by improving decarbonization, i.e. by decreasing the rest CO_2 content. Operation of the second subsystem is subject to the constraint that methanization is always complete, i.e. no CO_2 or CO can be left in the stream.

In the third subsystem we have to maximize $Q_3 = y_{pi}$, the concentration of inerts in the purge gas. This means of course that possibly little H_2, N_2 is lost, because in the balance all incoming inerts must be let out:

$$F_p y_{pi} = \text{const}$$

Note that we could replace the goal "maximize y_{pi}" by the equivalent "minimize F_p".

(iii) Coordination variables and coordination method

For the non-additive function ψ in

$$Q = \psi(Q_1, \ldots, Q_N)$$

we have to use coordination by direct method (the price coordination, described further on, could not be used here). Let us look at the possible coordination variables. In principle they should be all the subsystem outputs (or inputs). The coordinator would prescribe their values and thus separate the subproblems one from another.

Here a serious failure of the approach was encountered. Examination of the real plant has shown that there are many feed-forward and recycle linkages between parts of the system, not only in the main stream. This was due to the plant design where the linkages serve to utilize the heat energy generated in the plant and thus make the plant self-supporting in this respect.

The main links are shown in Fig.16. The failure of approach consisted in the fact that to describe a crosscut through all links would take about 40 variables; these would have to be decision variables in the coordination problem. But all parts of plant together had only 22 control variables to be adjusted (the set points of 22 different controllers). Hence we would replace a 22-variable problem by a 40-variable problem at the

coordination level plus a need to solve the local problems also. The two-level problem was more complex and expensive than the direct one.

An insight into quantitative properties of the problem and into the actual operating experience has permitted to propose an approximate solution. Only 5 out of 40 variables were found to be "essential" and were consequently chosen as coordination variables:

v_1 - gas (CH_4) inflow to the process,

v_2 - steam inflow,

v_3 - gas pressure in the gas preparation section,

v_4, v_5 - two principal heat steam flows .

The other variables were found to be either directly related to the five, or were assumed to be constant and needing no adjustment by the coordinator, or their values were almost irrelevant for the plant optimization.

Note, for example, that the coordinator would not have to prescribe the air inflow to the process. If he sets gas and steam, the amount of air is automatically dictated by the required N_2 to H_2 ratio.

The ammonia process has indicated an important topic for hierarchical control studies: *subcoordination* that is the use of less coordination variables than would be required for a strict solution.

4.5 *Subcoordination*

Let us very briefly present the problem of subcoordination for the case of the direct coordination method. The main point

is that the coordinator would prescribe the output y by using a vector v instead of y_d, where dim v < dim y. There may be two principal ways of using v in coordination.

One way of using v could be to set up a fixed matrix R and specify for the local problems:

$y_d = Rv$, that is $y_{di} = R_i v$ for each subsystem.

Note that if we knew our system accurately, we could set an adequate matrix $R = \hat{R}$ and a value $v = \hat{v}$, obtaining $y_d = \hat{y}_d$ (the strictly optimal value), whatever the dimension of v. This makes little sense, however; model vs. reality difference must be assumed to make the investigation meaningful.

Another way of using v could be to set a fixed function $\gamma(\cdot)$ and require from the local problems to comply with

$\gamma(y) = v$, that is $\gamma_i(y_i) = v_i$ for each subsystem.

This makes more sense intuitively, since we are granting the subproblems their freedom except for the fulfilment of the demands specified in v. For example, we demand a total production but do not specify the individual items. However, in this case the subproblems are not entirely separated and analysis of such a system is much more difficult.

Subcoordination approach is also possible in the price method. We will see it in the next paragraphs.

4.6 *Coordination by the use of prices; interaction balance method*

Let us recall the description of the system and of the control problem, as was given by (6) ... (10) in section 4.1, that

is, recall the subsystem equations, system interconnection equation, resource constraint, local constraints, and local performance indices.

Note that even before we define the global performance index of the system we can define the task of coordination, which can be to influence the local decision makers in such a way that system constraints will be preserved.

Price coordination consists in authorizing the coordinator to prescribe prices on inputs, outputs and resources and then permitting the local decision makers to define their own choices of the values of these variables. The system is *coordinated* when the local choices cause the interconnection equation (7) to be satisfied and the global constraint (8) to be non-violated. The prices which effect this state of the system can be termed *equilibrium prices*, since satisfaction of (7) means an equilibrium of the inputs and outputs.

The equilibrium prices bring about overall system optimum if the *global performance index* is a sum of local ones

$$Q = \sum_{i=1}^{N} Q_i \quad , \tag{12}$$

It is worth remembering, that direct and penalty function coordination methods presented before allowed a more general form of global performance, see (11).

The discussion of price coordination which will now follow omits the resource constraint (8), focusing on interconnections (7).

We will discuss the so-called *Interaction balance method (IBM)*. In this case the *local problems* i.e. problems associated with the individual subsystems can be formulated as follows (assuming $Q_i(c_i,u_i)$ has to be minimized):

$$\text{minimize } Q_{i\ \text{mod}} = Q_i(c_i,u_i) + \langle\lambda_i,u_i\rangle - \langle\mu_i,F_i(c_i,u_i)\rangle \quad (13)$$

subject to

$$(c_i,u_i) \in CU_i$$

with the results $\hat{c}_i(\lambda)$, $\hat{u}_i(\lambda)$, $\hat{y}_i(\lambda) = F_i(\hat{c}_i(\lambda),\hat{u}_i(\lambda))$.

If (13) is related to a finite-dimensional problem (as is the case in steady-state optimization), then the scalar product

$\langle\lambda_i,u_i\rangle$ means $\sum_{j=1}^{\dim u_i} \lambda_{ij}u_{ij}$, and $\langle\mu_i,(F_i(c_i,u_i)\rangle$ means

$$\sum_{j=1}^{\dim y_i} \mu_{ij}F_{ij}(c_i,u_i) \quad .$$

In the problem (13) we assumed coordination to be effected by a *price vector* λ, composed of prices on inputs in the whole system. Hence λ_i are prices on interaction input u_i; the prices μ_i on output y_i are defined as well by virtue of (7), namely

$$\mu_i = \sum_{j=1}^{N} H_{ji}^T\lambda_j \quad .$$

It is therefore right to say that the results of (13) are dependent exclusively on vector λ.

The *interaction balance* or *equilibrium* prices $\hat{\lambda}$ will be defined such as to provide for

$$\hat{u}(\hat{\lambda}) - H\hat{y}(\hat{\lambda}) = 0 \qquad (14)$$

where $\hat{y}(\lambda) = F(\hat{c}(\lambda), \hat{u}(\lambda))$.

Providing for the condition (14) to be satisfied is the task of the coordinator. In the classical economics this could be assigned to a "tatônnement" procedure at the stock exchange: a person outside the negotiating parties would vary the price λ, watch the responses $\hat{y}(\lambda)$ and $u(\lambda)$, and stop the procedure at $\lambda = \hat{\lambda}$.

Several questions can now be raised, for example:

- existence conditions for $\hat{\lambda}$, that is for the equilibrium price;
- system optimality with control $\hat{c}(\hat{\lambda})$;
- procedures to obtain $\hat{\lambda}$.

The answers are based upon discussion of the Lagrangian function of the global problem. After the local minimizations (13) have been performed, this Lagrangian is

$$\phi(\lambda) = \sum_{i=1}^{N} Q_i(\hat{c}_i(\lambda), \hat{u}_i(\lambda)) + \langle \lambda, \hat{u}(\lambda) - HF(\hat{c}(\lambda), \hat{u}(\lambda)) \rangle$$

and it is required that it has a maximum at $\lambda = \hat{\lambda}$:

$$\phi(\hat{\lambda}) = \max_{\lambda} \phi(\lambda) \quad .$$

If $\hat{\lambda}$ so defined exists, its further use to determine optimal control is practically restricted to the case where $(\hat{c},\hat{u})$, the mathematical solutions are single-valued functions of λ. This requirement appears to be vital for applications. Unfortunately we know sufficient conditions only: $(\hat{c},\hat{u})$ are single-valued if the functions $Q_i(\cdot)$ are strictly convex and the mappings $F_i(\cdot)$ are affine (linear). With $\lambda = \hat{\lambda}$ the unique solutions $\hat{c}(\lambda),\hat{u}(\lambda)$ are optimal.

It may be appropriate to indicate that the requirement of uniqueness of $(\hat{c},\hat{u})$ in response to a change in λ has a simple interpretation: since the prices λ aim at providing a match of the outputs to the inputs of other subsystems, they should have a well-defined influence.

In many real-life problems the uniqueness of response can be predicted by physical considerations for systems far from being linear (remember that we fail to know necessary conditions, while the sufficient ones are too severe to be of much practical use).

It is quite easy to show an example where the uniqueness of response will fail to appear. If λ would be price imposed by the coordinator on some product and $\hat{y}(\lambda)$ the optimal amount produced by a subsystem according to its own local optimization, the output $\hat{y}(\lambda)$ will not be well-defined in the particular case where the unit production cost would be equal to λ. Note that there would be no local gain or local loss associated with the size of production y.

Let us now turn back to the main stream of our considerations. What procedures could be used at the coordination level

in the search for λ? It can be shown [25] that if $Q_i(\cdot)$ are continuous and $F_i(\cdot)$ are continuous, then gradient procedures for λ can be used, provided we find a way to deal with the points where the $(\hat{c},\hat{u})$ are not unique and where the gradient is not defined (subgradients can be considered there). In those regions of λ-space where $(\hat{c},\hat{u})$ are unique, the following formula holds for the (weak) derivative of $\phi(\lambda)$

$$\nabla\phi(\lambda) = \hat{u}(\lambda) - HF(\hat{c}(\lambda),\hat{u}(\lambda)) \quad . \tag{15}$$

Note that this is exactly the input-output difference (the *discoordination* in the system, and it has to be brought to zero.

The second derivative, $\nabla^2\phi(\lambda)$, does not exist in the general case.

Let us mention that the interaction balance method (IBM) described so far can be applied to both static and dynamic problems, because we are dealing with models only. In particular, the search for λ is based on the difference, $\hat{u}(\lambda)-H\hat{y}(\lambda)$. It is, therefore, a computational concept rather than a control structure. In a system which is already in operation the interconnection equation is satisfied all the time, for any control c. We could never see if λ is correct. We could, therefore, use the described concept for open-loop control only. It means that we would first compute and then apply the computed $\hat{c}(\hat{\lambda})$ to the real system; the result will of course strongly depend on the accuracy of the models.

Let us now come back for a while to the resource constraint (8):

$$r_1(c_1,u_1) + \ldots + r_n(c_N,u_N) \leq r_o \quad .$$

This additive form of global constraint can be incorporated in the price coordination scheme by using an additional price vector η (the resource price) and adding to each local problem a value $\langle \eta, r_i(c_i,u_i) \rangle$, so that the local objective function becomes:

$$Q_{i\ mod} = Q_i(c_i,u_i) + \langle \lambda_i,u_i \rangle - \langle \mu_i,F_i(c_i,u_i) \rangle + \qquad (16)$$

$$+ \langle \eta,r_i(c_i,u_i) \rangle \quad .$$

By varying η the coordinator would change the resource requirements of the local problems so as to satisfy the overall constraint.

In the mathematical programming terminology, η would be a Kuhn-Tucker multiplier.

The next paragraphs will show some other ideas of price coordination, where feedback from the operating system will be used to improve control.

4.7 *Price coordination in steady-state with feedback to coordinator (the IBMF method)*

In this section we shall consider the optimization problem to be in the finite-dimensional space, i.e. to be a problem of non-linear programming. In terms of control it means control of steady-state in a complex system. We remember from Section 2.4 that steady-state control is an appropriate technique if the optimal state trajectory of a dynamic system is slow enough to assume that the value of state vector x is at any time related to control only, the state derivative $\dot{x}$ being so small as to be neglected.

The mappings F_i, Q_i are now functions in finite-dimensional space. We have therefore the following model-based global problem:

$$\text{minimize } Q = \sum_{i=1}^{N} Q_i(c_i,u_i)$$

subject to

$$y_i = F_i(c_i,u_i), \qquad i \in \overline{1,N}$$

$$u = H\, y$$

$$(c_i,u_i) \in CU_i, \qquad i \in \overline{1,N} \quad .$$

We have dropped the resource constraint for simplicity. A solution to the model-based problem yields *model-based control* $\hat{c}$. We intend now to pay considerable attention to the difference between model and reality, let us therefore formulate the following *real problem* :

$$\text{minimize } Q = \sum_{i=1}^{N} Q_i(c_i,u_i)$$

subject to

$$y_i = F_{*i}(c_i,u_i), \qquad i \in \overline{1,N}$$

$$u = H\, y$$

$$(c_i,u_i) \in CU_i, \qquad i \in \overline{1,N} \quad .$$

We should notice that the only difference between model and reality is herewith assumed to exist in the subsystem equations, that is the functions $F_{*i}(\cdot)$ are different from the model ones $F_i(\cdot)$. We shall indicate in the sequel some effective way to fight the consequences of this difference.

It must be stressed, however, that differences may exist also in the performance function and in the constraint set. For example, if a performance function is explicitly $Q_i(c_i,u_i,y_i)$ then it will reduce to some $Q_i(c_i,u_i)$ by using the subsystem equation, but this makes it model-based. The real $Q_{*i}(c_i,u_i)$ would be different from $Q_i(c_i,u_i)$. A similar reason may lead to the set CU_{*i} being different from CU_i.

Solution to the real problem will be termed *real-optimal control* $\hat{c}_*$. It is not obtainable by definition since reality is not known. We can only look for a structure which would yield a control that would be better than the purely model-based $\hat{c}$, but in principle what we will achieve is bound to be inferior to $\hat{c}_*$.

One of the possible structures is price coordination with feedback to the coordinator. It is shown schematically by Fig. 17.

The *local problems* are exactly the same as in the open-loop interaction balance method, that is we have for each $i \in \overline{1,N}$:

$$\text{minimize } Q_i(c_i,u_i) + \langle \lambda_i, u_i \rangle - \langle \mu_i, F_i(c_i,u_i) \rangle$$

subject to

$$(c_i,u_i) \in CU_i \quad .$$

The controls $\hat{c}_i(\lambda)$ determined by solving this problem (computationally) for the current value of λ are applied to the real system, resulting in some u_* and y_*. The coordination concept consists in the following upper-level problem:

$$\text{find } \lambda = \tilde{\lambda} \text{ such that } \hat{u}(\tilde{\lambda}) - u_*(\hat{c}(\tilde{\lambda})) = 0 \ . \qquad (17)$$

Condition (17) is an equality of model-based optimal input $\hat{u}(\lambda)$ and of the input u_*, measured in the real system and caused by control $\hat{c}(\lambda)$. Providing for this equality is the basic concept of "interaction balance method with feedback" (IBMF).

The properties of control based on condition (17) have been studied quite extensively, see [12]. The usual questions of existence of $\tilde{\lambda}$, system optimality with control $\hat{c}(\tilde{\lambda})$ and procedures to obtain $\tilde{\lambda}$ have been discussed and answers have been formulated. The essence of these answers is in principle as follows.

Solution $\tilde{\lambda}$ exists, if solution $\hat{\lambda}$ of the open-loop interaction balance method (IBM) exists for all s-shifted systems

$$u = H\,F(c,u) + s$$

where $s \in S$, and S is the set of all possible values of the model-reality difference

$$H\,F_*(c,u) - H\,F(c,u) = s$$

with $(c,u) \in CU = CU_1 \times \ldots \times CU_N$.

When the models do not differ from reality, $\hat{c}(\tilde{\lambda})$ is strictly optimal control and $\tilde{\lambda}$ equals equilibrium prices $\hat{\lambda}$ which would be obtained by solving the problem by the interaction balance method of the previous paragraph. When models differ from reality, the control based on (17) is in the first approximation always non-inferior to the one based on open-loop value $\hat{\lambda}$. In

the particular case where

$$F_{*i}(c_i,u_i) = F_i(c_i,u_i) + \beta_i \qquad i \in \overline{1,N}$$

that is the model-reality difference of the subsystems consist in a shift, the control based on (17) is strictly real-optimal. The open-loop would of course in this case be much inferior.

A most important feature of control based upon (17) is its property to keep to the constraints in the real system. Note that the real control c_* equals model-based $\hat{c}$ for any λ, because the result $\hat{c}(\lambda)$ is applied to the system. For $\lambda = \tilde{\lambda}$ we also have $u_* = \hat{u}$. Since the model-based solution will keep $(\hat{c}_i,\hat{u}_i) \in CU_i$, $i = \overline{1,N}$ the same will be kept in the real system, but only at $\lambda = \tilde{\lambda}$. Note that the open-loop control $\hat{c}(\hat{\lambda})$ may violate the constraints in the real system, because at $\lambda = \hat{\lambda}$ it will in general be $u_* \neq \hat{u}$.

The control based on $\lambda = \tilde{\lambda}$ does not violate the constraints $(c_i,u_i) \in CU_i$ if the real constraint sets equal the model ones $CU_{*i} = CU_i$, $i \in \overline{1,N}$. There exists also a modified method (MIBMF) where the case $CU_{*i} \neq CU_i$ is covered by appropriate use of feedback information, see [12].

As far as the procedures to find $\tilde{\lambda}$ are concerned, iterations have to be done at a rate acceptable by the real system, i.e. permitting new values u_* to establish themselves after a change of λ. Unfortunately, the expression

$$R_*(\lambda) = \hat{u}(\lambda) - u_*(\hat{c}(\lambda)) \tag{18}$$

which has to be brought to zero is not a derivative of any function, as it was in the case of interaction balance method. The

value $\tilde{\lambda}$ has to be found by equation-solving methods, aiming at $R_*(\lambda) = 0$. It should be stressed that if there are inequality constraints in the local problems, $R_*(\lambda)$ will in general be non-differentiable. Suitable numerical methods to find $\tilde{\lambda}$ have been proposed [12][31].

We are now able to justify discussion of steady-state control here as opposed to more general problem formulation in the previous paragraph. The reason is the practical field of application of coordination principle (17): it must be iteratively done on the real system. This can be performed in steady-state optimization, but not in a dynamical one. The only exception would be iterative optimization of batch or cyclic processes, the iteration in time-function space being performed from one batch to another. For that particular case all considerations can be appropriately generalized.

Let us add an example to explain what the on-line price coordination really means. Consider the electric power system and its customers. The amount of power that is being produced is matched to the current load. How can we tell whether the price on electrical energy is correct since there is no demand-supply difference? The on-line price adjustment proposed in this section applies to this problem: the price is considered to be correct when the production-load balance of the power which has actually established itself in the real system (u_*) is equal to the model-based optimal value ($\hat{u}$). The difference would be used to generate price modification.

4.8 *Decentralized control with price coordination (feedback to local decision units)*

The structure of Fig.17, although proved to be effective and superior to open-loop model-based control, may be criticised; the information about real system u_* is made available to the coordinator only. The local problems base on models and calculate their imaginative $\hat{u}$ for each λ, "knowing" that reality is different. The scheme of Fig. 17 is therefore a structure suitable for a mechanistic control system, but does not reflect the situation which would be established if the local problems were confined to decision makers with more freedom of choice.

We can expect that the local decision maker would tend to use the real value u_{*i} in his problem, that is that he would perform

$$\text{minimize } Q_i(c_i,u_i) + \langle \lambda_i, u_{*i} \rangle - \langle \mu_i, F_i(c_i,u_{*i}) \rangle \qquad (19)$$

subject to

$$(c_i,u_{*i}) \in CU_i \quad .$$

Schematically this is presented in Fig. 18 as feeding u_{*i} to the corresponding local problem. Even with fixed λ the control exercised by local decision makers on the system as a whole remains to some extent coordinated, since the value of λ will influence the control decisions. However, since u_{*i} are used locally, we may call the structure of Fig. 18 *decentralized*.

A problem for itself is system stability or the convergence of iterations made by local optimizers while trying to achieve their goals. It is obvious that all the iteration loops in the

system are interdependent, since an u_{*i} will depend on the decisions $c = (c_1,\ldots,c_N)$ in the previous stage, that is on the decisions of all decision units.

If the iterations converge, some steady-state values $\hat{c}(\lambda)$, $\hat{u}_*(\lambda)$ and $\hat{y}_*(\lambda)$ will be obtained for the given price vector λ.

It may be predicted that if this λ would happen to be $\tilde{\lambda}$ from the previous paragraph, the result of decentralized control would also be the same as in the previous structure. This does not say that we should aim at it, since the results obtained with $\tilde{\lambda}$ are not real-optimal and a better value of λ may exist.

We should look for some way to iterate on prices λ in the system of Fig. 18. A possibility might be

$$\text{minimize } Q = \sum_{i=1}^{N} Q_i(\hat{c}_i(\lambda),\hat{u}_{*i}(\lambda)) \qquad (20)$$

which simply means to find a price λ such that the overall result of local controls be optimized.

Two properties of the problem seem predictable. If the models are adequate, and all iterations converge, they will converge to the strict overall optimum for the system. If the models differ from reality, then the constraints $(c_i,u_i) \in CU_i$ will be secured (like in the structure in Fig.17), but the overall result will be suboptimal. This suboptimality is due to the fact that in performing the local optimizations we continue to have an inadequate (model-based) value of the output y_i.

5. Dynamic multilevel control

The structures of on-line dynamic control using decomposition of the control problem differ from those applicable to steady-state. The differences lie in the use of feedback from the system in operation. In steady-state control we could use feedback in the form of measured inputs or outputs of the system elements and provide for an extremum of a current or "instantaneous" performance index, as described above. The dynamic optimization needs considering at time t the future behavior of the system, that is to consider an "optimization horizon". Since the future behavior depends on both the initial state and the control input that follows it, we cannot determine the optimal control unless we know the present state of the system. It means that if we wish to have a control structure with feedback, this feedback must contain information on the state x(t).

There are three principal ways in which local dynamic control problems can be formed and, subsequently, coordinated by an appropriate supremal problem. They are the following:

- *dynamic price coordination*, where time-varying prices on the inputs and outputs are imposed by the coordinator, along with the target states to be achieved by each subsystem over the local optimization horizon;
- *structure based on state-feedback concept*, where the local decision making is reduced to a static (instantaneous) feedback decision rule, and the coordinator supplies signals which serve either to modify the local decisions, or to modify the local decision rules, so as to account for the performance of the system as a whole;

- *structures using conjugate variables*, where the local decision making is a kind of static (instantaneous) optimization, and the optimal dynamic policy is secured by a vector of prices on the trend of the subsystem state (i.e. by the vector of conjugate variables) imposed on the subsystems and readjusted by the coordinator.

In this section we shall briefly discuss these alternatives. We will particularly expose the "dynamic" features.

5.1 *Dynamic Price Coordination*

Assume the global control problem of the interconnected system to be as follows:

$$\text{minimize } Q = \sum_{i=1}^{N} \int_0^{t_f} q_{oi}(x_i(t), m_i(t), u_i(t))dt \qquad (21)$$

subject to

$$\dot{x}_i(t) = f_i(x_i(t), m_i(t), u_i(t)), \quad i \in \overline{1,N} \text{ (state equations)}$$

$$y_i(t) = g_i(x_i(t), m_i(t), u_i(t)), \quad i \in \overline{1,N} \text{ (output equations)}$$

$$u(t) = Hy(t) \quad \text{(interconnections)}$$

with $x(0)$ given, $x(t_f)$ free or specified.

Decomposition

Consider that in solving the problem we incorporate the interconnection equation into the following Lagrangian:

$$L = \sum_{i=1}^{N} \int_0^{t_f} q_{oi}(x_i(t), m_i(t), u_i(t))dt + \int_0^{t_f} \langle \lambda(t), u(t)-Hy(t) \rangle dt$$

where $\langle \lambda(t), u(t)-Hy(t) \rangle$ means $\sum_{j=1}^{\dim u} \lambda_j(t)(u(T)-Hy(t))_j$.

Assume the solution to the global problem using this Lagrangian has been found and it has provided for

$\hat{x}_i$, i = 1,...,N - optimal state trajectories

$\hat{m}_i$, i = 1,...,N - optimal controls

$\hat{u}_i$, i = 1,...,N - optimal inputs

$\hat{y}_i$, i = 1,...,N - optimal outputs

$\hat{\lambda}$ - solving value of Lagrangian multipliers.

Note that now the Lagrangian can be split into additive parts, thus allowing to form a kind of local problems:

$$\text{minimize } Q_i = \int_0^{t_f} [q_{oi}(x_i(t), m_i(t), u_i(t)) + \langle \hat{\lambda}_i(t), u_i(t) \rangle - \langle \hat{\mu}_i(t), y_i(t) \rangle]dt \tag{22}$$

where

$$y_i(t) = g_i(x_i(t), m_i(t), u_i(t))$$

and optimization is subject to

$$\dot{x}_i(t) = f_i(x_i(t), m_i(t), u_i(t))$$

where $x_i(0)$ is given and $x_i(t_f)$ is free or specified as in the original problem.

In the local problem the price vector $\hat{\lambda}_i$ is an appropriate part of $\hat{\lambda}$ and $\hat{\mu}_i$ is also given by $\hat{\lambda}$ as

$$\hat{\mu}_i = \sum_{j=1}^{N} H_{ji}^T \hat{\lambda}_j .$$

Notice that we have put optimal value of price vector λ into the local problems, which means that we have solved the global problem before. Thanks to it the solutions of local problems will be strictly optimal. There is little sense, however, in solving the local problems if the global was solved before, because the global solution would provide not only $\hat{\lambda}$ but also $\hat{x},\hat{m}$ for the whole system.

Short horizon and feedback at local level

To make the thing practical let us try to shorten the local horizons and to use feedback in the local problems. If we shorten the horizon from t_f to t_f', the local problem (22) becomes

$$\text{minimize } Q_i = \int_0^{t_f'} [q_{oi}(x_i(t),m_i(t),u_i(t)) + \\ + \langle\hat{\lambda}_i(t),u_i(t)\rangle - \langle\hat{\mu}_i(t),y_i(t)\rangle]dt \tag{23}$$

with $x_i(0)$ given as before, but the target state taken from the global long-horizon solution, $x_i(t_f') = \hat{x}_i(t_f')$. Here we might remind the reader on the discussion of multilayer hierarchies with the divided time horizon, discussed in Section 2.1 (see Fig.7).

For the local problem (23) we must of course supply the price vectors $\hat{\lambda}_i$, $\hat{\mu}_i$. It may be reasonable to use also $\hat{u}_i$ from the global solution, that is the "predicted" input value.

The short horizon formulation (23) will pay-off if we will have to repeat the solving of (23) many times as opposed to solving the global problem once only. Consult now Figure 19 where the principle of the proposed control structure is presented.

Feedback at the local level consists in solving the short-horizon local problems at some intervals $T_1 < t_f'$ and using the actual value of measured state $x_{*i}(kT_1)$ as new initial value for each repetition of the optimization problem.

This brings a new quality; we now have a truly on-line control structure and can expect, in appropriate cases, to get results better than those dependent on the models only.

The operation of the structure is more exactly as follows: at $t = 0$ we solve the problem max Q_i for the horizon $[0,t_f']$ with $x_i(0)$, then we apply control $\hat{m}_i$ to the real system for an interval $[0,T_1]$, at $t = T_1$ we again solve max Q_i for horizon $[T_1,t_f']$ with initial state $x_i(T_1) = x_{*i}(T_1)$ as measured, then we apply control $\hat{m}_i$ to the real system for the interval $[T_1,2T_1]$, etc.

We now have a practical gain from both decomposition and shortening the horizon. The local problems, which have to be repeated at intervals T_1, are low-dimension and short-horizon.

We should mention disturbances which act on the real system and were not yet shown explicitly in the formulations. Disturbance prediction would be used while solving (21) and (23),

that is the global and the local problems. And it is indeed because of the disturbances which in reality will differ from their prediction that we are inclined to use feedback structure of Figure 19.

Feedback at coordination level

The feedback introduced so far cannot compensate for the errors done by the coordination level in setting the prices $\hat{\lambda}$. Another repetitive feedback can be introduced to overcome this shortage, for example bringing to the coordinator actual value x_{*i} at time t_f', $2t_f'$,... and asking the global problem to be resolved for each new initial value. This principle of control is also indicated in Figure 19.

We should very well note that feeding back the actual values of state achieved makes sense if the models used in computation differ from reality, for example because of disturbances. Otherwise the actual state is exactly equal to what the models have predicted and the feedback information is irrelevant.

A doubt may exist whether the feedback to the coordinator makes sense, because the lower level problems have to achieve $x_i(t_f') = \hat{x}_i(t_f')$ as their goal and already use feedback to secure it. It should be remembered, however, that the model-based target value $\hat{x}_i(t_f')$ is not optimal for the real system and asking the local decision making to achieve exactly $x_{*i}(t_f') = \hat{x}_i(t_f')$ may be not advisable or even not feasible.

The coincidence of feedback to coordination level with times t_f', $2t_f'$ is not essential. It might be advisable to use this feedback and perform the re-computation of the global problem prior to time t_f', that is more often.

Static elements

In a practical case it may happen that some of system elements can be approximately considered as *static*, that is non-dynamical. It can be explained as follows.

The length of the global problem horizon t_f has to be matched to the slowest system element dynamics and the slowest of the disturbances. The shortened horizon t_f' for the local problems would in fact result from considering repetitive optimization at the coordination level, for example as 1/10 of t_f. It may then happen that the dynamics of a particular system element are fast enough to be neglected in its local optimization problem within the horizon t_f'. This means, in other words, that if we would take $\hat{m}_i, \hat{u}_i$ from the global optimization solution, the optimal state solution $\hat{x}_i$ follows these with negligible effect of element dynamics.

To make this assumption more formal let us consider that the system element has been supplied with first-layer follow-up controls of some appropriately chosen controlled variables c_i, see section 2.2. We are then allowed to assume that c_i determines both x_i and m_i of the original element and the optimization problem becomes

$$\text{minimize } Q_i = \int_0^{t_f'} [q'_{oi}(c_i(t), u_i(t)) + \langle \hat{\lambda}_i(t), u_i(t) \rangle - \langle \hat{\mu}_i(t), y_i(t) \rangle] dt \qquad (24)$$

where $q'_{oi}(\cdot)$ is a reformulation of the function q_{oi} due to substituting c_i in place of x_i, m_i.

Note well that although (24) will not be a dynamic problem its results will be time functions. In particular $\hat{c}_i$ will be time-varying control. This is due to time-varying prices $\hat{\lambda}_i, \hat{\mu}_i$.

Let us repeat the essential assumption under which the dynamical local problem (23) reduces to the static problem (24): the dynamic optimal solutions $\hat{m}_i, \hat{u}_i, \hat{x}_i$ were assumed to be slow.

The use of simplified models

In the described structure of on-line dynamic coordination we have made no use till now of the possibility of having a simplified model in the global problem, which is being solved at the coordination level at times 0, t_f', 2 t_f', etc.

The global problem may be simplified for at least two reasons: the solution of the full problem may be too expensive to be done, and the data on the real system, in particular prediction of disturbances, may be too inaccurate to justify a computation based on the exact model.

Simplification may concern dimension of state vector (introduce aggregated x^c instead of x), dimension of control vector (m^c instead of m) and dimensions of inputs and outputs ($u^c = H^c y^c$ instead of $u = Hy$).

The global problem Lagrangian will now be

$$L = \sum_{i=1}^{N} \int_0^{t_f} q_{oi}^c \, (x_i^c(t), m_i^c(t), u_i^c(t))dt + \int_0^{t_f} \langle \lambda^c(t), u^c(t) - H^c y^c(t) \rangle \, dt \quad . \tag{25}$$

The simplified solution will yield optimal state trajectory $\hat{x}^c = (x_1^c, x_2^c, \ldots, x_N^c)$ and optimal price function λ^c. The

linking of those values to the local problems cannot be done directly, because the local problems consider full vectors x_i, u_i and y_i.

We have to change the previous requirement $x_i(t_f') = \hat{x}_i(t_f')$ to a new one

$$\gamma_i[\hat{x}_i(t_f')] = x_i^c(t_f')$$

which incidentally is a more flexible constraint, and we also have to generate a full price vector $\hat{\lambda}$:

$$\hat{\lambda} = R\hat{\lambda}^c$$

where R is an appropriate "price proportion matrix". The prices composing the aggregated λ^c may be termed "group prices".

We should note that functions γ_i and matrix R have to be appropriately chosen. The choice may be made by model considerations, but even with the best possible choice optimality of overall solution will be affected, except for some special cases.

System interconnection through storage elements

The system interconnections considered till now were stiff, that is an output was assumed to be connected to an input in a permanent way. We may consider also another type of interconnection, a "soft" constraint of integral type:

$$\int_{kt_b}^{(k+1)t_b} (u_{ij}(t) - y_{lr}(t))dt = 0$$

which corresponds to taking input u_{ij} from a store, with some output y_{lr} connected to the same store and causing its filling.

Asking for integral over $[kt_b,(k+1)t_b]$ to be zero means that supply and drain have to be in balance over each balancing period t_b.

A store may be supplied by several outputs and drained by more than one subsystem input. There may also be many stores, for example for different products. If we assume the same balancing period for all of them the integral constraint becomes

$$\int_{kt_b}^{(t+1)t_b} (\bar{H}_1 u_w(t) - \bar{H}_2 y_w(t) dt = 0$$

where u_w, y_w are parts of u, y connected to the stores (the stiffly interconnected parts will be termed u_s, y_s).

Matrices $\bar{H}_1, \bar{H}_2$ show the way by which u_w, y_w are connected to various stores. The number of stores is of course dim $\bar{H}_1 y_w$ = dim $\bar{H}_2 u_w$. A state vector w of the inventories can also be introduced

$$w(kt_b + t) = w(kt_b) + \int_{kt_b}^{kt_b + t} (\bar{H}_1 u_w(t) - H_2 y_w(t) dt \qquad (26)$$

With both stiff and soft interconnections present in the system, the global problem Lagrangian becomes

$$L = \sum_{i=1}^{N} \int_0^{t_f} q_{oi}(x_i(t),m_i(t),u_i(t))dt + \int_0^{t_f} <\lambda(t),u_s(t)-Hy_s(t)>dt +$$

$$+ \sum_{k=0}^{k=\frac{t_f}{t_b}-1} <\eta^k, \int_{kt_b}^{(k+1)t_b} (\bar{H}_1 u_w(t)-\bar{H}_2 y_w(t))dt > \qquad (27)$$

and we of course continue to consider

$$\dot{x}_i(t) = f_i(x_i(t),m_i(t),u_i(t)), \qquad i = 1,\ldots,N$$

$$y_i(t) = g_i(x_i(t),m_i(t),u_i(t)) \qquad i = 1,\ldots,N$$

$$x_i(0) \text{ given, } x_i(t_f) \text{ free or specified, } i \in \overline{1,N} \ .$$

In comparison with the previous Lagrangian a new term has now appeared, reflecting the new constraint. Note that prices η^k associated with the integral constraint are constant over periods t_b. Note also, that if t_b will tend to zero, the integral constraint gets similar to the stiff one and the stepwise changing $\hat{\eta}$ will change continuously, like $\hat{\lambda}$ does.

With two kinds of interconnections the local problems also change correspondingly and they become

$$\text{minimize } Q_i = \int_0^{t_f} [q_{oi}(x_i(t),m_i(t),u_i(t))+<\hat{\lambda}_i(t),u_{si}(t)> - \qquad (28)$$

$$-<\hat{\mu}_i(t),y_{si}(t)>]dt + \sum_{k=0}^{k=\frac{t_f}{t_b}-1} <\hat{\eta}^k, \int_0^{t_b} (H_{1i}u_{wi}(t)-H_{2i}y_{wi}(t))dt>$$

where $y_{si}(t) = g_{si}(x_i(t),m_i(t),u_i(t)),y_{wi}(t) = g_{wi}(x_i(t),m_i(t),u_i(t))$ and optimization is subject to

$$\dot{x}_i(t) = f_i(x_i(t),m_i(t),u_i(t))$$

$x_i(0)$ given, $x_i(t_f)$ free or specified.

A new quality has appeared in problem (28) in comparison with (23): the inputs u_{wi} taken from the stores are now free control variables and can be shaped by the local decision maker, who previously had only m_i in his hand. The local decisions will be under the influence of prices $\hat{\lambda}$ and $\hat{\eta}=(\eta^0,\eta^1,\ldots)$, where both $\hat{\lambda}$ and $\hat{\eta}$ have to be set by the solution of the global problem.

The local problem (28) has no practical importance yet; it will make sense when we introduce local feedback and shorten the horizon, like it was in the previous stiff-interconnection case.

We shall omit the details and show it only as a control scheme (see Figure 20).

Thinking about how to improve action of the coordinator we made previously a proposal to feed actual $x_*(t_f')$ to his level. We have now additional state variables, the inventories w. If the price $\hat{\eta}^k$ is wrong, the stores will not balance over $[kt_b,(k+1)t_b]$. It is almost obvious that we can catch-up by influencing the price for the next period $\hat{\eta}^{k+1}$ and that we should condition the change on the difference $w[(k+1)t_b]- w_*[(k+1)t_b]$, where $w_*(\cdot)$ is a value measured in the real system. This kind of feedback is also shown in Figure 20.

Conclusion on dynamic price coordination

It has been shown that time-varying prices are a possible coordination instrument which can be used in a multilevel structure of on-line control. They must, however, be accompanied by prescribing also the target states.

The local problems may be formulated as short-horizon and each of them has low dimension. The coordination level must solve the global problem for full horizon in order to generate the optimal prices and the target states for the local problems. It is expected that a simplified global model may be used in appropriate cases.

The price coordination structure applies to systems with stiff interconnections and also to systems with interconnections through storage elements.

The operation of the structure depends on the possibility of numerical solution of optimization problems.

Analytical solutions of the dynamic problems involved are not needed, therefore we are by no means restricted to linear-quadratic systems.

5.2 *Multilevel control based upon state-feedback concept*

The literature on optimal control has paid considerable attention to the structure where the control at time t, that is m(t), would be determined as a given function of current state x(t). Comprehensive solutions exist in this area for the linear system and quadratic performance case, where the feedback function proved to be linear, that is, we have

$$\hat{m}(t) = R(t)\ x(t)$$

where R(t) is in general a time-varying matrix.

Trying to apply this approach to the complex system we might implement for each local problem

$$\hat{m}_i(t) = R_{ii}(t)\ x_i(t) \qquad (29)$$

where R_{ii} is one of the diagonal blocks of the matrix R.

The result of such local controls, although all state of the system is measured and used, is not optimal. Note that for $\hat{m}_i(t)$ we would rather have to use

$$\hat{m}_i(t) = R_i(t)x(t)$$

that is we should make $\hat{m}_i(t)$ dependent on the whole state x(t).

We can compensate for the error committed in (29) by adding a suitably computed correction signal

$$\hat{m}_i(t) = R_{ii}(t)x_i(t) + \hat{v}_i(t)\ . \qquad (30)$$

The exact way to get $\hat{v}_i(t)$ would be to generate it continuously basing upon the whole x(t). This would, however, be equivalent to implementing state feedback for the whole system directly, with no advantage in having separated the local problems.

From the local problem point of view, adding $\hat{v}_i(t)$ as in (30) means, in fact, overriding the local decision. In particular, dim v_i = dim m_i.

Exactness has to be sacrificed. With this in mind we may propose various solutions, for example (see Figure 21).

(i) $\hat{v}_i$ will be generated at t = 0 for the whole optimization horizon t_f (open-loop compensation);

(ii) $\hat{v}_i$ will be generated at $t = 0$ as before but will be recomputed at $t = t_f' < t_f$, using actual $x(t_f')$, etc. (repetitive compensation);

(iii) $\hat{v}_i$ will not be generated at all, but we implement instead in the local problems

$$\hat{m}_i(t) = \tilde{R}_{ii}(t)x_i(t) \tag{31}$$

where R_{ii} is adjusted so as to approach optimality. This structure may be referred to as decentralized control. We could think of re-adjusting $\tilde{R}_{ii}$ at some time intervals, which could be looked upon as adaptation. This adaptation would present a way of on-line coordination of the local decisions.

It may be worthwhile to mention that local decision making based upon (29),(30) or (31) makes more sense for a mechanistic implementation than for a hierarchy of human operators, where the previous approach based on "maximization of local performance subject to imposed prices" seems to be more adequate, to what really happens in the system.

We should also remember that the feedback gain solutions to optimization problems are available for a restricted class of these problems only.

5.3 *Structures using conjugate variables*

It is conceivable to base on-line dynamic control upon maximization of the current value of the Hamiltonian, thus making a direct use of the Maximum Principle.

For the complex system optimization problem, described as (21) at the beginning of this section, the Hamiltonian would be

$$\mathcal{H} = -\sum_{i=1}^{N} q_{oi}(x_i(t),m_i(t),u_i(t)) + \langle \psi(t), f(x(t),m(t),u(t)) \rangle . \tag{32}$$

The interconnection equation

$$u(t) - Hy(t) = u(t) - Hg(x(t),m(t),u(t)) = 0$$

provides for u(t) to be a function of (x(t),m(t)) in the interconnected system

$$u(t) = \phi(x(t),m(t)) \quad .$$

Therefore

$$\mathcal{H} = - \sum_{i=1}^{N} q_{oi}(x_i(t),m_i(t),\phi_i(x(t),m(t))) + \\ + \langle \psi(t), f(x(t),m(t),\phi(x(t),m(t))) \rangle \qquad (33)$$

Assume the global problem has been solved (model-based) using this Hamiltonian and hence the optimal trajectories of conjugate variables ψ are known.

We are going to use the values of $\hat{\psi}$ in local problems.

First let us note that having $\hat{\psi}$ we could re-determine optimal control by performing at the current time t

$$\text{maximize } \mathcal{H} = - \sum_{i=1}^{N} q_{oi}(x_i(t),m_i(t),\phi_i(x_i(x(t),m(t))) + \\ + \langle \hat{\psi}(t), f(x(t),m(t),\phi(x(t),m(t))) \rangle \qquad (34)$$

where the problem is an "instantaneous maximization" and needs no consideration of final state and future disturbances. This information was of course used while solving the global problem and determining $\hat{\psi}$ for the whole time horizon.

For the (34) to be performed we need the actual value of state x. We could obtain it by simulating system behavior

starting from the time t_1 when initial condition $x(t_1)$ was given, that is by using equation

$$\dot{x}(t) = f(x(t),m(t),\phi(x(t),m(t)))$$

with $x(t_1)$ given and $m = \hat{m}$ known for $[t_1,t]$ from the previous solutions of (34).

We could also know $x(t)$ by measuring it in the real system (note that a discussion of model-reality differences would be necessary).

Problem (34) is static optimization, not a dynamic one. We would now like to divide it into subproblems. It can be done if we come back to treating $u(t)-Hy(t) = 0$ as a side condition and solve (34) by using the Lagrangian

$$L = -\sum_{i=1}^{N} q_{oi}(x_i(t),m_i(t),u_i(t)) + \langle\hat{\psi}(t),f(x(t),m(t),u(t))\rangle + \\ + \langle\lambda(t),u(t) - Hy(t)\rangle \qquad (35)$$

where $y(t) = g(x(t),m(t),u(t))$.

Before we get any further with this Lagrangian and its decomposition let us note the difference with respect to dynamic price coordination presented before. We have had there

$$L = \int_0^{t_f} \sum_{i=1}^{N} q_{oi}(x_i(t),m_i(t),u_i(t))dt + \int_0^{t_f} \langle\lambda(t),u(t)-Hy(t)\rangle dt$$

subject to

$$\dot{x}_i(t) = f_i(x_i(t),m_i(t),u_i(t)), \qquad i \in \overline{1,N}$$

It was a dynamic problem.

In the present case there are no integrals in $L(\cdot)$ and the dynamics are taken care of by the values of conjugate variables $\hat{\psi}$. The differential equations of the system are needed only to compute the current value of x in our new, "instantaneous" Lagrangian. No future disturbances are to be known, no optimization horizon considered - all these are imbedded in $\hat{\psi}$.

Assume we have solved problem (35), using system model i.e., by computation and we have the current optimal value of price $\hat{\lambda}$, that is $\hat{\lambda}(t)$. We can then form the following static local problems to be solved at time t

$$\text{maximize } L_i = -q_{oi}(x_i(t), m_i(t), u_i(t) + \langle \hat{\psi}_i(t), f_i(x_i(t), m_i(t), u_i(t)) \rangle + \langle \hat{\lambda}_i(t), u_i(t) \rangle - \langle \hat{\mu}_i(t), y_i(t) \rangle \quad . \tag{36}$$

These goals could be used in a structure of decentralized control, see Figure 22. The local decision makers are asked here to maximize $L_i(\cdot)$ in a model-based fashion and to apply control $\hat{m}_i(t)$ to the system elements. Current value $x_i(t)$ is needed in performing the task. The coordination level would supply $\hat{\psi}_i(t)$ and the prices $\hat{\lambda}_i(t), \hat{\mu}_i(t)$ for the local problem. They would be different for each t.

Note that there is no hill-climbing search on the system itself.

Figure 22 would first imply that the local model-based problems are solved immediately with no lag or delay. We can therefore assume, conceptually, that the local decision making is nothing else but implementation of a state feedback loop, relating control $\hat{m}_i(t)$ to the measured $x_i(t)$.

If analytical solution of (36) is not the case we have to implement a numerical algorithm of optimization and some time will be needed to perform it. An appropriate discrete version of our control would have to be considered, but we drop this formulation.

Now let us think about feedback to the coordinator. We might decide to let him know the state of the system at some time intervals t_f', that is $x(kt_f')$. On this he could base his solution $\hat{\psi}$ for all $t > kt_f'$ and also the prices $\hat{\lambda}$ for the next interval $[kt_f', (k+1)t_f']$. This policy would be very similar to what was proposed in the "dynamic price coordination".

It might be worthwhile to make again some comparisons between dynamic price coordination and the structure using both prices and conjugate variables.

In the "maximum principle" structure the local problems are static. The local goals are slightly less natural, as they involve $< \hat{\psi}_i, \dot{x}_i(t) >$ that is the "worth of the trend". This would be difficult to explain economically and hence difficult to implement in a human decision making hierarchy. As the problem is static, no target state is prescribed.

Note that both these cases avoid to prescribe a state trajectory. It is felt that in the dynamic control this kind of direct coordination would be difficult to perform if model-reality differences are assumed.

5.4 *A comparison of the dynamical structures*

We have shown three main possibilities to structure a dynamic multilevel control system, using feedback from the real system in the course of its operation. We do not think it

possible at this stage to evaluate all advantages and drawbacks of the alternatives. It may be easily predicted that if the mathematical models used do not differ from reality, all structures would give the same result, the fully optimal control. The clue is what will happen if models are inadequate. Quantitative indications are essentially missing in this area, although efforts are being made and some results are available [11],[13].

Another feature of the structures concerns their use in a human decision making hierarchy. In that case it is quite essential what will be the local decision problem, confined to the individual decision maker. He may feel uncomfortable, for example, if asked to implement only a feedback decision rule (as it happens in the "state feedback" structure), or to account for the worth of the trend $\langle\hat{\psi}_i(t),\dot{x}_i(t)\rangle$ in his own calculations, as it is required in the structure using conjugate variables, see Table 1.

Table 1. Comparison of dynamic coordination structures.

SYSTEM TYPE	COORDINATOR	LOCAL PROBLEMS	LOCAL GOALS
DYNAMIC PRICE COORDINATION	solves global problem, sets prices $\hat{\lambda}$ and targets $\hat{x}_i$	dynamic optimization	maximize performance, achieve target state
STATE-FEEDBACK CONCEPT	solves global problem, supplies compensation signal $\hat{v}_i$	state feedback decision rule	no goal
USING CONJUGATE VARIABLES	solves global problem, sets prices $\hat{\lambda}$ and conjugate variables $\hat{\psi}_i$	static optimization	maximize performance inclusive of $\langle\hat{\psi}_i(t),\dot{x}_i(t)\rangle$

6. Conclusions

Hierarchical control systems, as a concept, are relatively simple and almost self-explanatory. They exist in many applications, ranging from industrial process control, through production management to economic and other systems [10],[17],[23], [30],[33]. Some of these systems may involve human decision makers only, other may be hierarchies of control computers, or mixed systems. The hierarchical control theory is developing quite rapidly; its goals may be defined as :

- to explain behavior of the existing systems, for example find out the reasons for some phenomena which occur;
- to help designing new system structures, for example determining what decisions are to be made at each level, what coordination instruments are to be used, etc;
- to guide the implementation of computer-based decision making in the system.

In the first two cases a qualitative theory may be sufficient, whereby the models or the description of the actual system do not have to be very precise. The available hierarchical control theory seems to be quite relevant for this kind of applications, and can help in drawing conclusions as well as in making system design decisions.

The third case calls for having relatively exact models of the system to be controlled (although suitable feedback structures relax the requirements) and calls also for having appropriate decision making algorithms, which would have to be programmed into the control computers. The existing theory and above all the existing experience are rather scarce in this area.

References

[1] Bailey, F.N., and K. Malinowski (1977). Problems in the design of multilayer-multiechelon control structures. Proceedings 4th IFAC Symposium on Multivariable Technological Systems, Fredericton (Canada), pp. 31-38.

[2] Brdys, M. (1975). Methods of feasible control generation for complex systems, Bull. Pol. Acad. of Sci., Vol.23.

[3] Chong, C.Y., and M. Athans (1975). "On the periodic coordination of linear stochastic systems". Proceedings 6th IFAC Congress, Pt. IIIA, Boston, Mass.

[4] Davison, E.J. (1977). "Recent results on decentralized control of large scale multivariable systems". Proceedings 4th IFAC International Symposium on Multivariable Technological Systems, Fredericton (Canada), pp. 1-10.

[5] Donoghue, J.F., and I. Lefkowitz (1972). "Economic tradeoffs associated with a multilayer control strategy for a class of static systems". IEEE Trans. on AC, Vol.AC-17, No.1, pp.7-15.

[6] Findeisen, W. (1974). Multilevel Control Systems. Warszawa, PWN (in Polish; German translation: Hierarchische Steuerungssysteme, Berlin, Verlag Technik 1977).

[7] Findeisen, W. (1976). "Lectures on hierarchical control systems". Report, Center for Control Sciences, University of Minnesota, Minneapolis.

[8] Findeisen, W. (1977). "Multilevel Structures for On-line Dynamic Control, Ricerche di Automatica, Vol. 8, No.1.

[9] Findeisen W., and I.Lefkowitz (1969). "Design and applications of multilayer control". Proceedings IV IFAC Congress, Warsaw.

[10] Findeisen, W., J. Pułaczewski and A. Manitius (1970). "Multilevel optimization and dynamic coordination of mass flows in a beet sugar plant", Automatica, Vol.6, No.2, pp. 581-589.

[11] Findeisen W., and K. Malinowski (1978). "Two level control and coordination for dynamical systems". Proceedings VII IFAC Congress, Helsinki.

[12] Findeisen, W., et al. (1978). "On-line Hierarchical Control for Steady-State Systems". IEEE Trans. on Autom. Control, Special Issue on Decentralized Control and Large-Scale Systems, April 1978.

[13] Findeisen, W., F.N. Bailey, M.Brdyś, K.Malinowski, P.Tatjewski and A.Woźniak. Control and Coordination in Hierarchical Systems, IIASA International Series, J. Wiley, London, to appear in 1979.

[14] Foord, A.G. (1974). "On-line optimization of a petrochemical complex, Ph.D. Thesis, University of Cambridge.

[15] Gutenbaum, J. (1974). "The synthesis of direct control regulator in systems with static optimization". Proceedings 2nd Polish-Italian Conf. on Applications of Systems Theory, Pugnochiuso (Italy).

[16] Hakkala, L., and H. Blomberg (1976). "On-line coordination under uncertainty of weakly interacting dynamical systems", Automatica, Vol. 12, pp. 185-193.

[17] Heescher, A., K.Reinisch and R. Schmitt (1975). "On multilevel optimization of nonconvex static problems - application to water distribution of a river system". Proceedings VI IFAC Congress, Boston.

[18] Himmelblau, D.M., ed.(1973). "Decomposition of Large-Scale Problems", Amsterdam, North Holland.

[19] Kulikowski, R. (1970). Control in Large-Scale Systems, WNT, Warszawa,(in Polish).

[20] Lasdon, L.S., (1970). Optimization Theory for Large Systems. London, Mcmillan.

[21] Lefkowitz, I. (1966). "Multilevel approach applied to control system design", Trans. ASME, Vol. 88, No.2.

[22] Lefkowitz, I. (1975). "Systems control of chemical and related process systems". Proceedings VI IFAC Congress, Boston.

[23] Lefkowitz, I., and A. Cheliustkin, eds.(1976). Integrated Systems Control in the Steel Industry, IIASA CP-76-13, Laxenburg.

[24] Malinowski, K. (1975). "Properties of two balance methods of coordination". Bulletin of the Polish Academy of Science, Ser. of Technical Science, Vol.23, No.9.

[25] Malinowski, K. (1976). "Lectures on hierarchical optimization and control", Report, Center for Control Sciences, University of Minnesota, Minneapolis.

[26] Mesarović, M.D., D.Macko and Y. Takahara (1970). Theory of Hierarchical, Multilevel Systems, New York, Academic Press.

[27] Milkiewicz, F. (1977). "Multihorizon - multilevel operative production and maintenance control". IFAC-IFORS-IIASA Workshop on Systems Analysis Application to Complex Programs, Bielsko-Biała (Poland).

[28] Pearson, J.D. (1971). "Dynamic decomposition techniques", in Optimization Methods for Large-Scale Problems (D.A. Wismer-ed.), Chapter 4, McGraw-Hill.

[29] Piervozwanski, A.A. (1975). Mathematical Models in Production Planning and Control, Nauka, Moscow, (in Russian).

[30] Pliskin, L.G. (1975). Continuous Production Control, Energy, Moscow, (in Russian).

[31] Ruszczynski, A. (1976). "Convergence conditions for the interaction balance algorithm based on an approximate mathematical model", Control and Cybernetics, Vol.5, No.4.

[32] Sandell, N.R., P. Varaiya and M. Athans (1976). "A survey of decentralized control methods for large-scale systems", Proceedings IFAC Symposium on Large-Scale Systems Theory and Applications, Udine (Italy).

[33] Siljak, D.D. (1976). "Competitive economic systems: stability, decomposition and aggregation". IEEE Trans. on Aut.Contr., Vol.AC-21, pp. 149-160.

[34] Siljak, D.D., and M.K. Sundareshan (1976). " A multilevel optimization of large-scale dynamic systems". IEEE Trans. on AC, Vol.AC-21, pp.79-84.

[35] Siljak, D.D., and M.B. Vukčević (1976). "Decentralization, stabilization and estimation of large-scale systems", IEEE Trans. on AC, Vol.AC-21, pp.363-366.

[36] Singh, M.G. (1977). Dynamical Hierarchical Control, Amsterdam, North Holland.

[37] Singh, M.G., S.A.W. Drew and J.F. Coales (1975). "Comparisons of practical hierarchical control methods for interconnected dynamical systems", Automatica, Vol.11, pp.331-350.

[38] Singh,M.G., M.F.Hassan and A.Titli (1976). "Multilevel Feedback Control for Interconnected Dynamical Systems using the Prediction Principle", IEEE Trans. Syst.Man. Cybern., Vol. SMC-6, pp.233-239.

[39] Smith, N.J., and A.P. Sage (1973). "An introduction to hierarchical systems theory". Computers and Electrical Engineering, Vol. 1, pp.55-71.

[40] Smith, N.J., and A.P. Sage (1973). "A sequential method for system identification in hierarchical structure". Automatica, Vol.9, pp.667-688.

[41] Stoilov, E.(1977). "Augmented Lagrangian method for two-level static optimization", Arch.Aut. i Telem. Vol.22, pp.210-237 (in Polish).

[42] Tamura, H. (1975). "Decentralised optimization for distributed-lag models of discrete systems". Automatica, Vol.11, pp. 593-602.

[43] Tatjewski, P. (1977). "Dual methods of multilevel optimization". Bull. Pol.Acad.Sci., Vol.25, pp.247-254.

[44] Tatjewski, P., and A. Woźniak (1977). "Multilevel steady-state control based on direct approach". Proceedings, IFAC-IFORS-IIASA Workshop on Systems Analysis Applications to Complex Programs, Bielsko-Biała (Poland).

[45] Titli, A. (1972). "Contribution a l'étude des structures de commande hierarchisées en vue de l'optimization de processus complexes". Ph.D.Thesis, Université Paul Sabatier, Toulouse, 1972. Also available in book form, Dunod, Paris.

[46] Tsuji, K., and I. Lefkowitz (1975). "On the determination of an on-demand policy for a multilayer control system". IEEE Trans. on AC Vol.AC-20, pp.464-472.

[47] Vatel, I.A., and N.N.Moiseev (1977). "On the modelling of economic mechanisms". Ekonomika i Matematicheskie Metody, Vol. 13, No.1 (in Russian).

[48] Wilson, I.D. (1977). "The design of hierarchical control systems by decomposition of the overall control problem". Proceedings, IFAC-IFORS-IIASA Workshop on Systems Analysis Applications to Complex Programs, Bielsko-Biała (Poland).

[49] Wismer, D.A., Ed.(1971). Optimization Methods for Large-Scale Problems, McGraw-Hill.

[50] Woźniak, A. (1976). "Parametric method of coordination using feedback from the real process", Proceedings, IFAC Symposium on Large Scale Systems Theory and Applications, Udine (Italy).

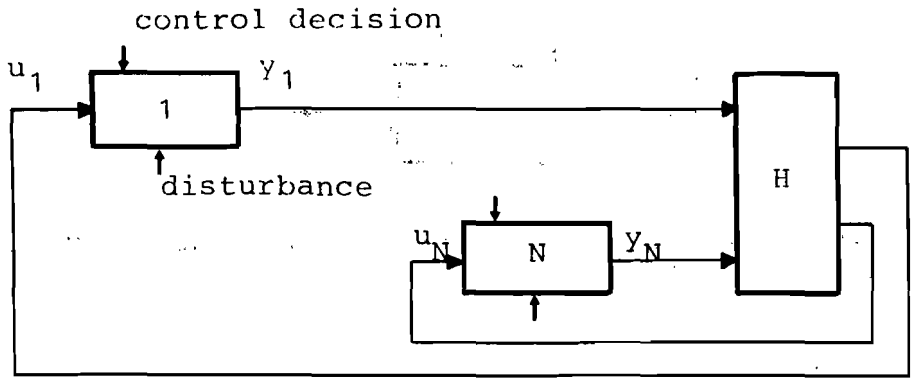


Fig.1 Schematic presentation of a complex system

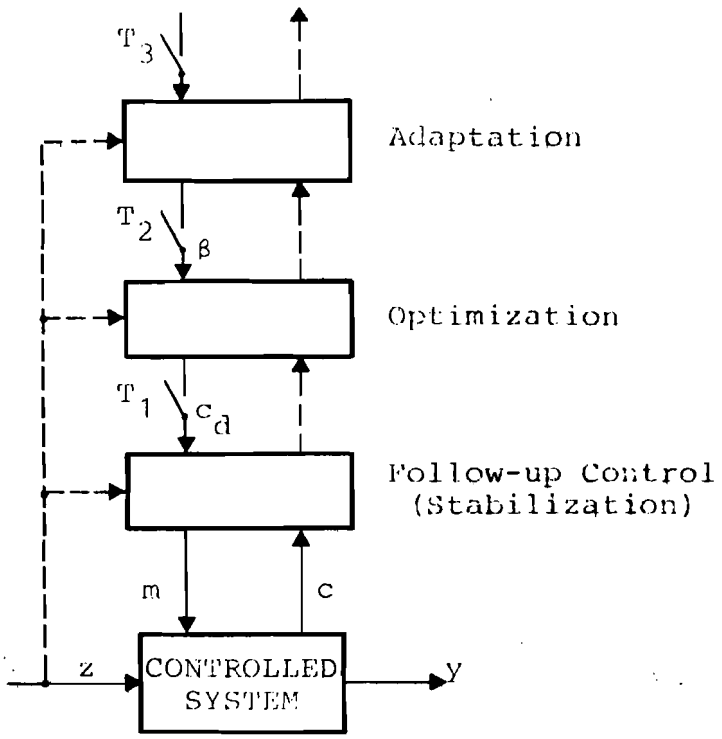


Fig.2 Multilayer control - a "functional" hierarchy .

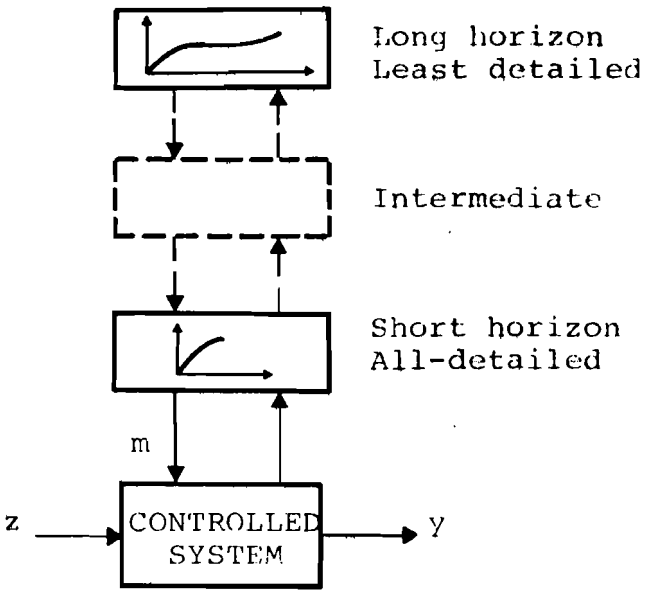


Fig.3 Multilayer system formed by differing the time horizons - a "temporal" hierarchy .

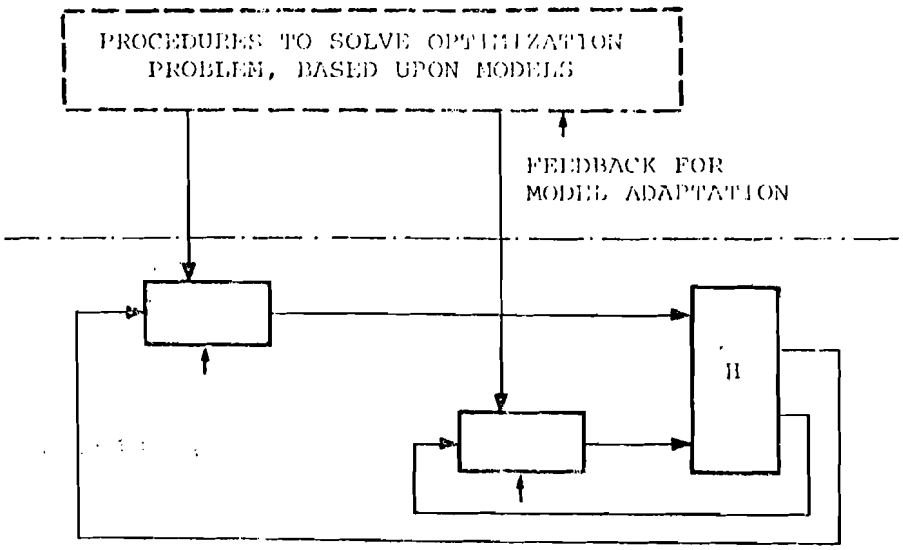


Fig.4 Open-loop control of a complex system.

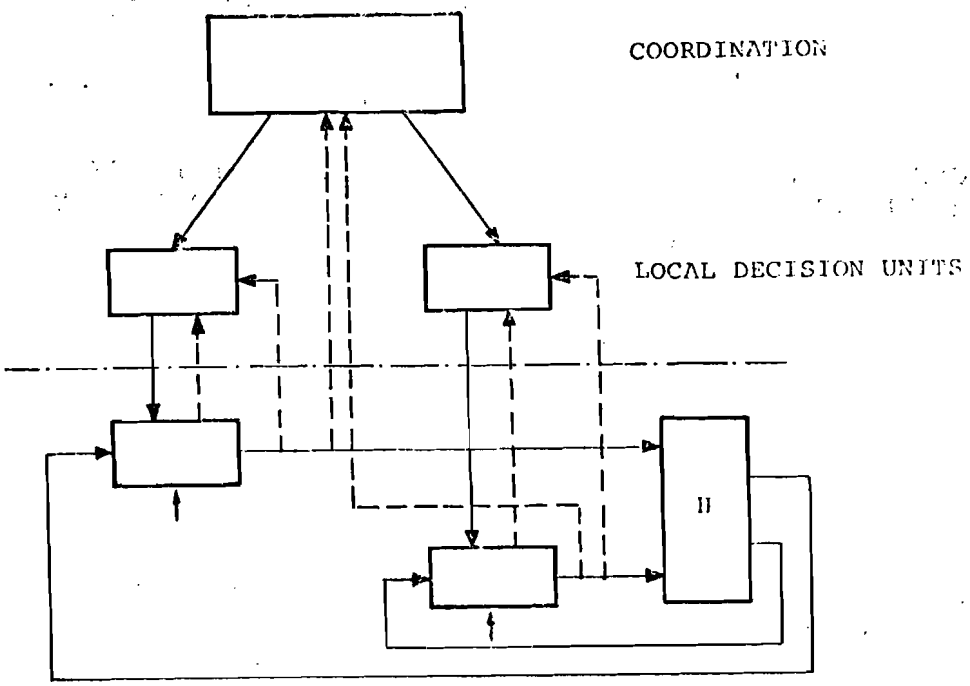


Fig.5 Multilevel control of a system. Dotted lines for possible feedbacks.

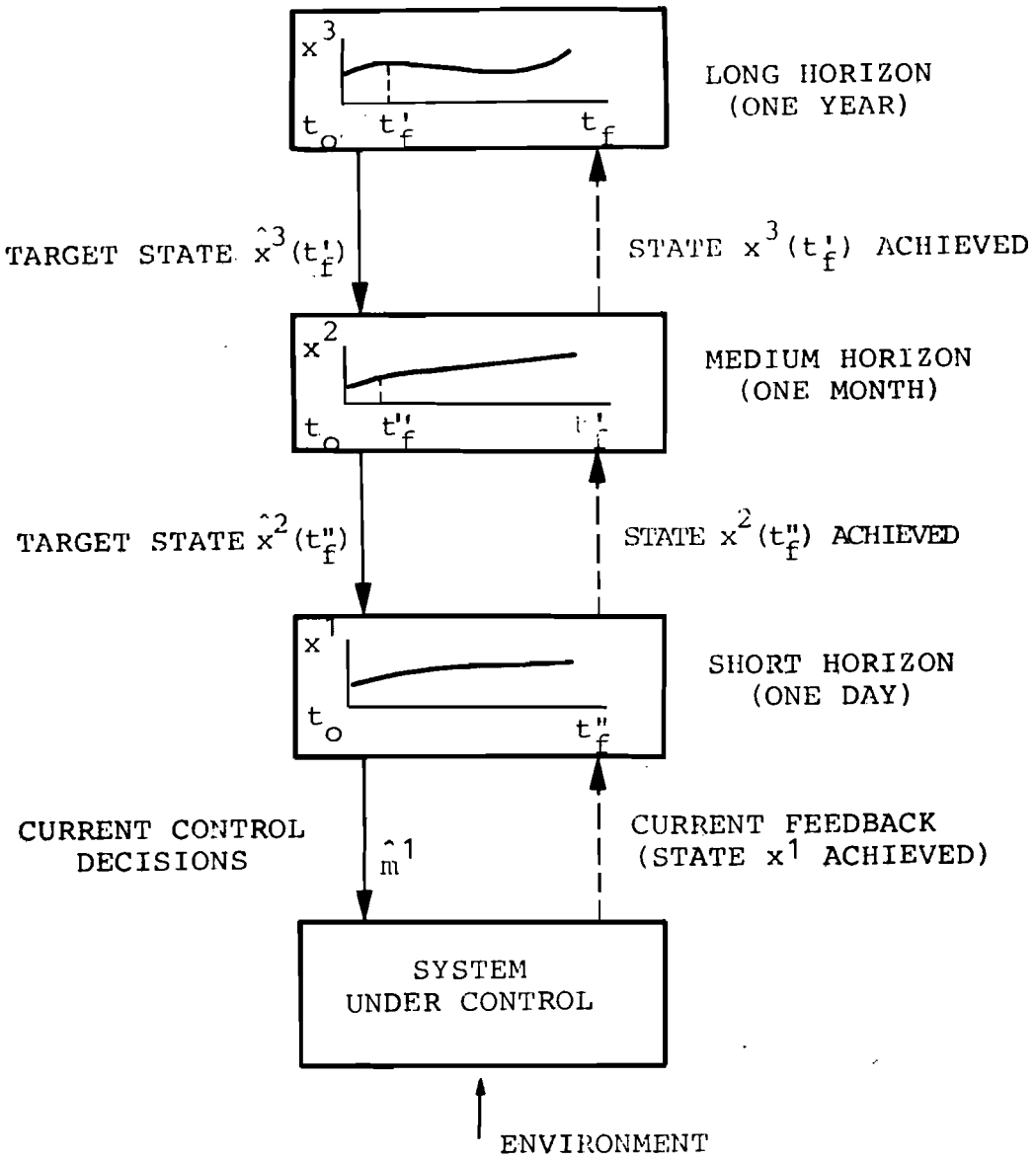


Fig.6 Multilayer concept applied to multi-horizon dynamic control.

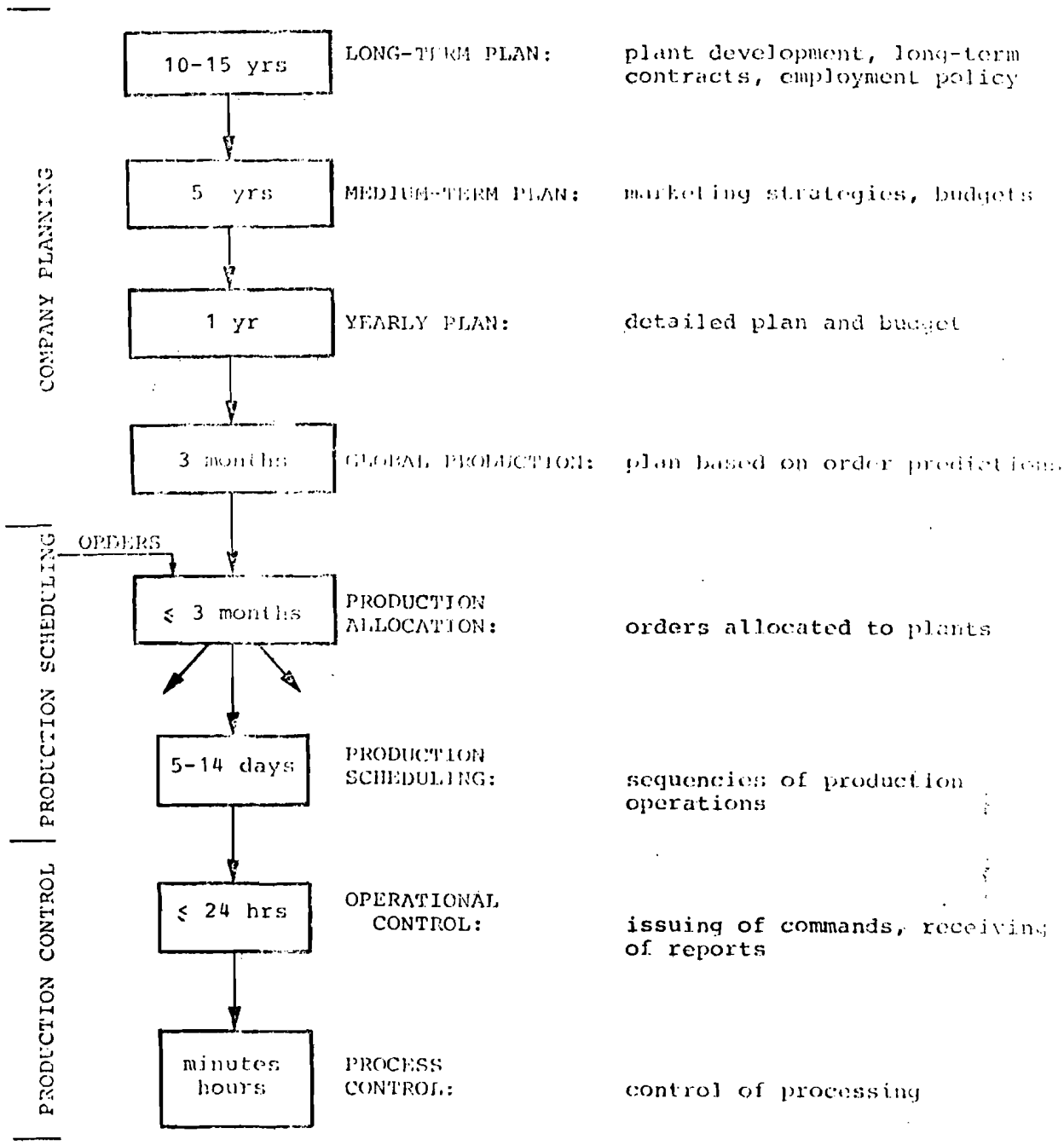


Fig. 7 Hierarchy of models and time horizons in a steel company.

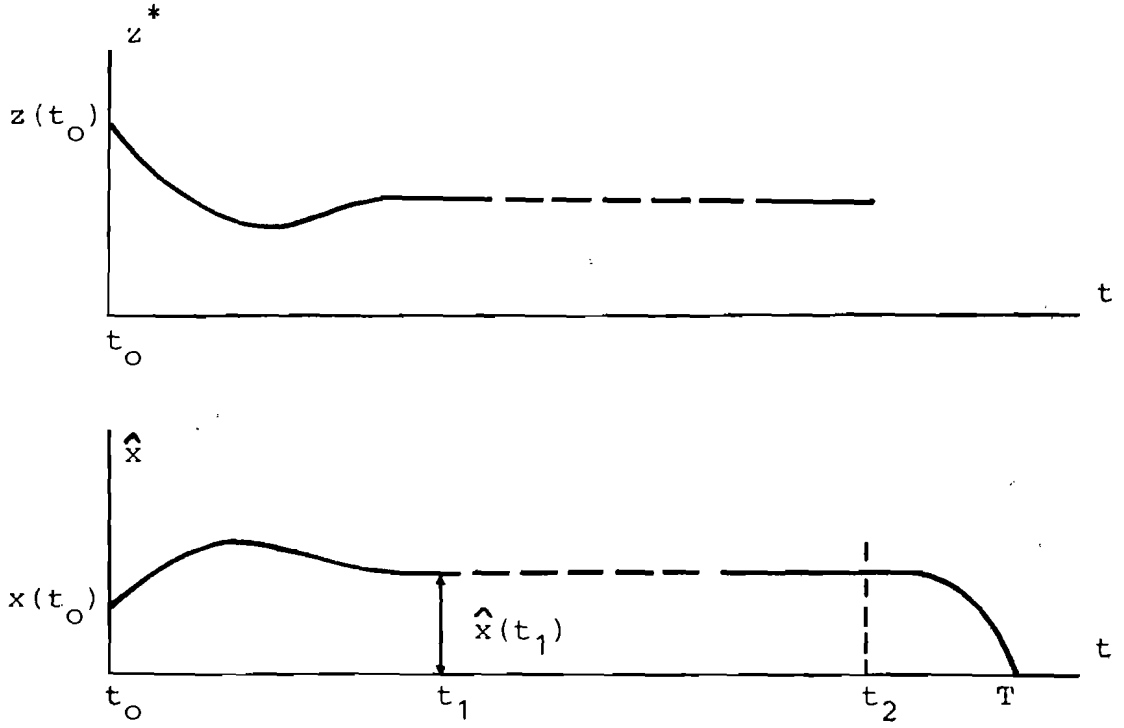


Fig.8 Illustration of optimization horizon.

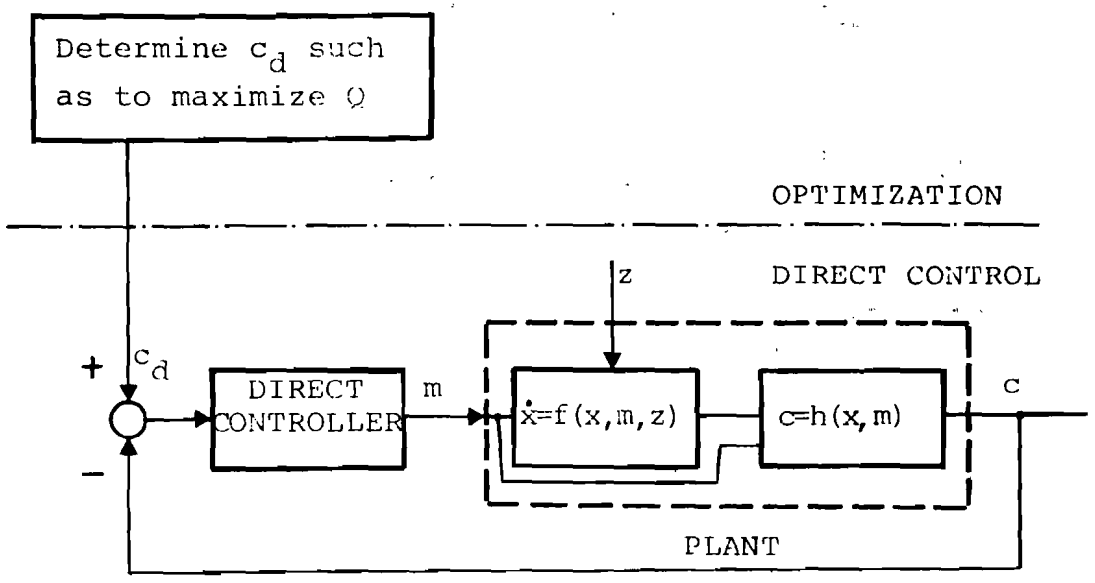


Fig.9 A two-layer system .

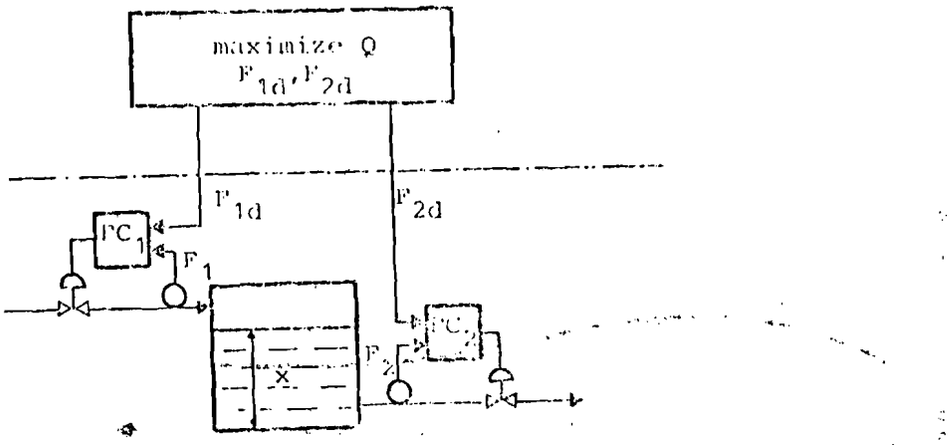


Fig.10 Explaining a poor choice of controlled variables.

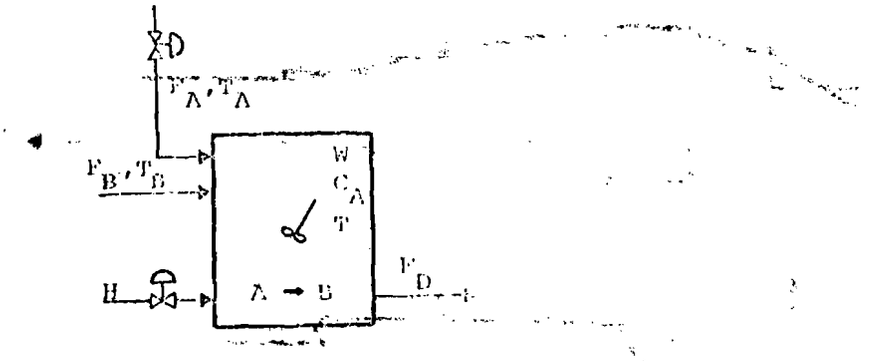


Fig.11 A stirred-tank reactor.

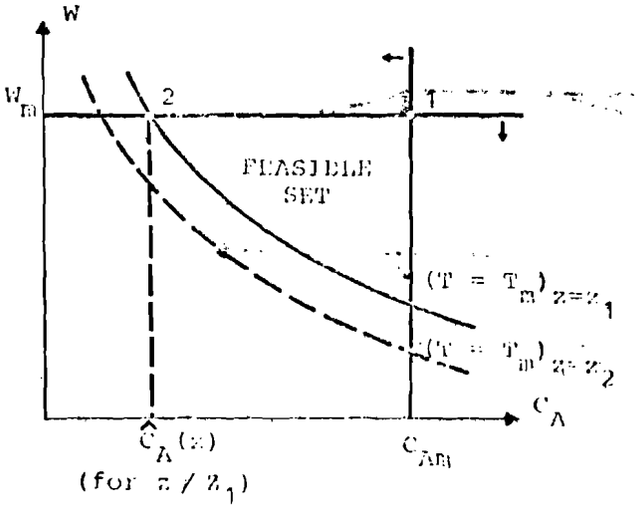


Fig.12 Position of solutions 1,2 in the feasible set for the exampl
Note that $T = T_m$ line moves in (W, C_A) plane with disturbance

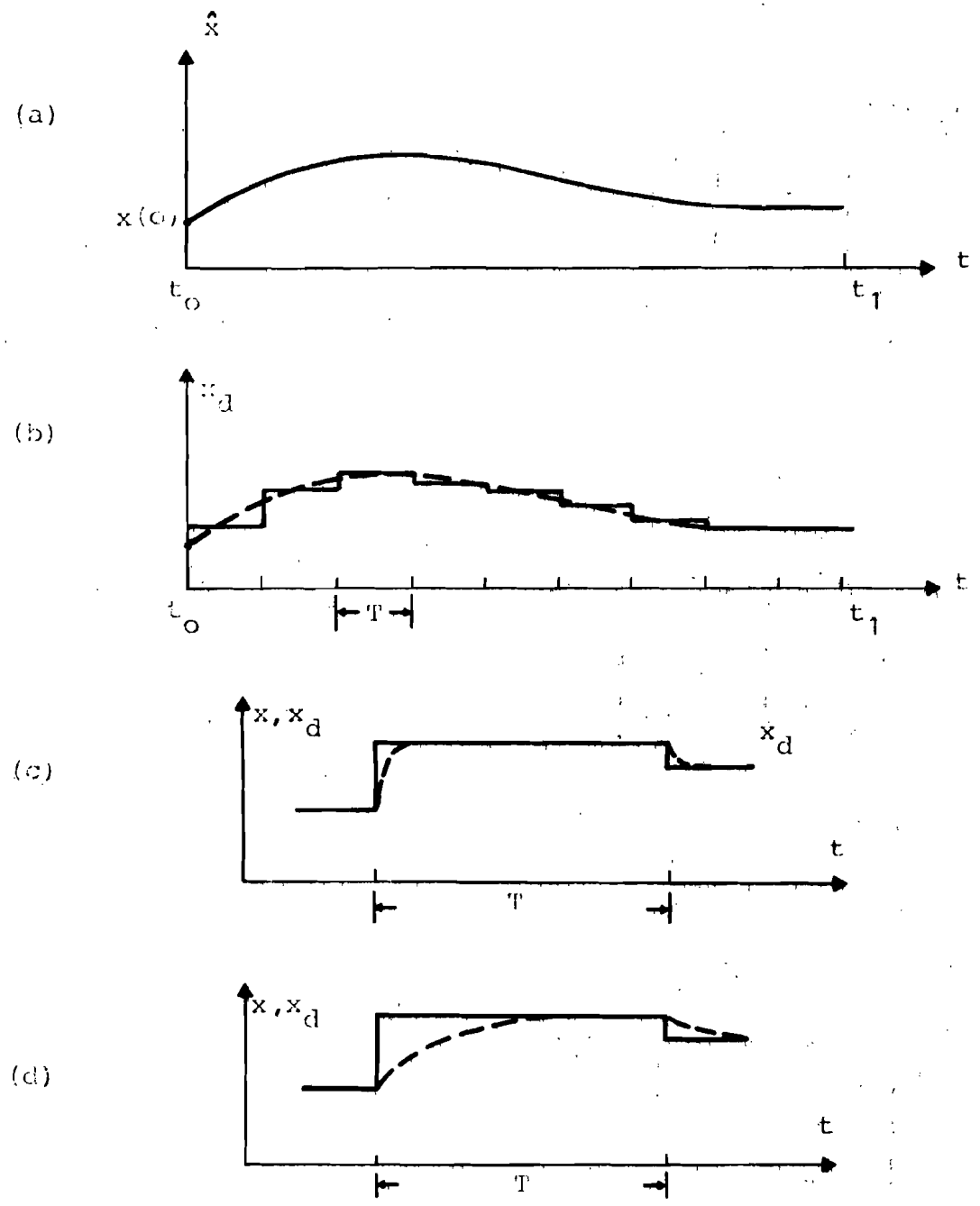


Fig.1 Explaining when is steady-state optimization appropriate.

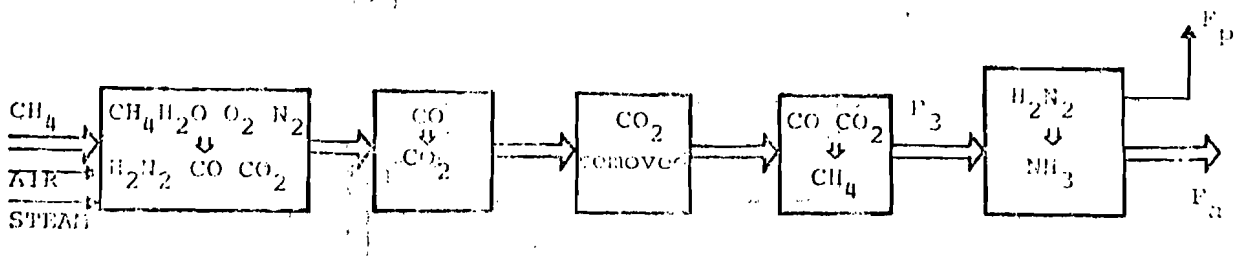


Fig.14 Principal parts of an ammonia plant.

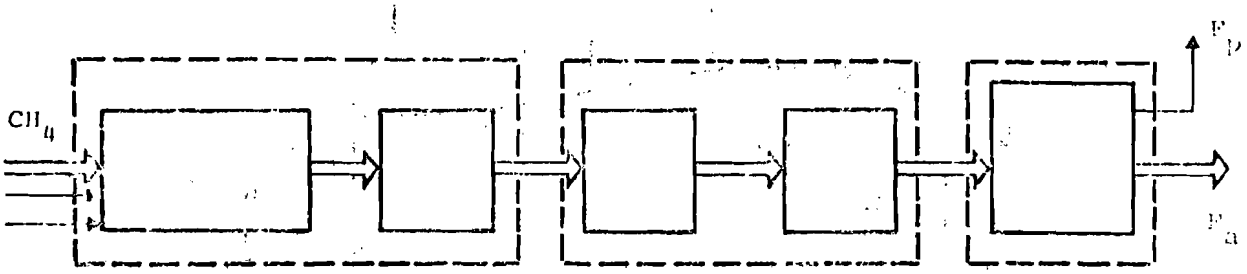
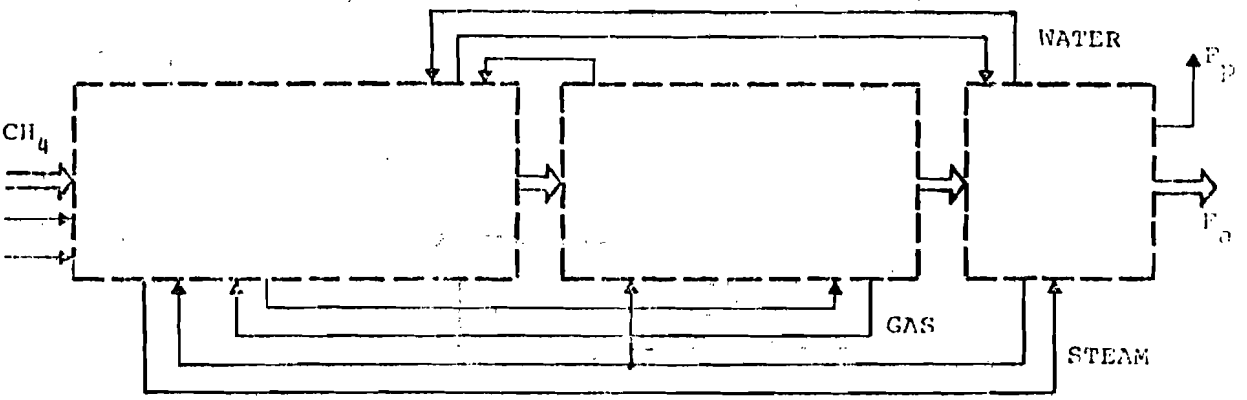
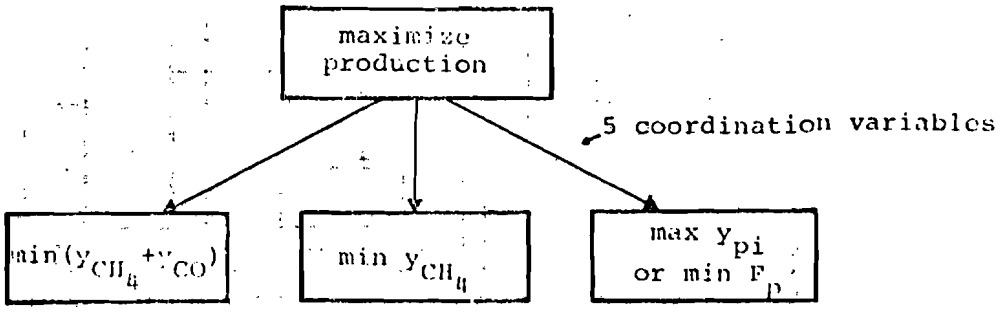


Fig.15 Ammonia plant divided into three subsystems.



a) Subsystem linkages, resulting in 40 interaction variables.



b) Control structure proposed.

Fig.16 Subsystem interactions and the control structure.

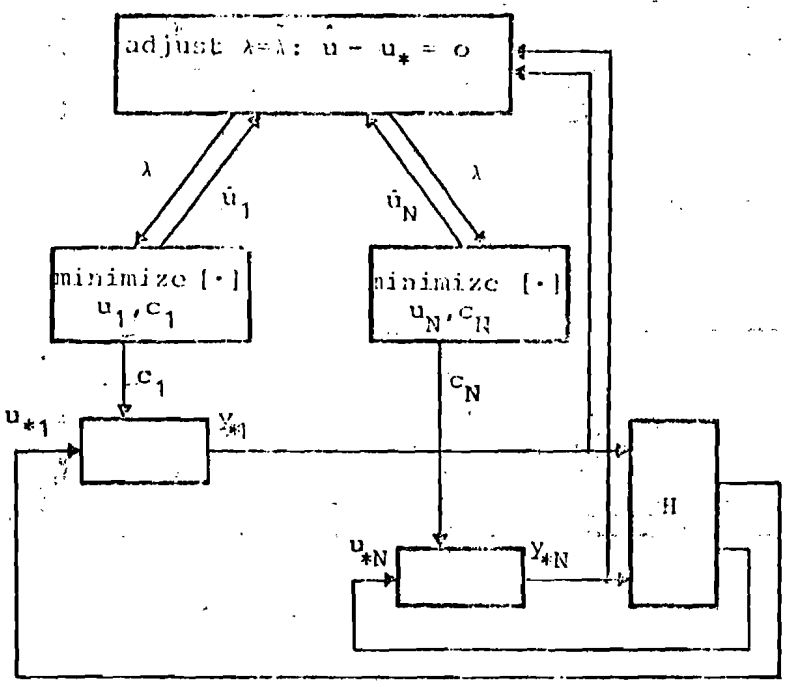


Fig.17 Iterative price coordination with feedback to the coordinator.

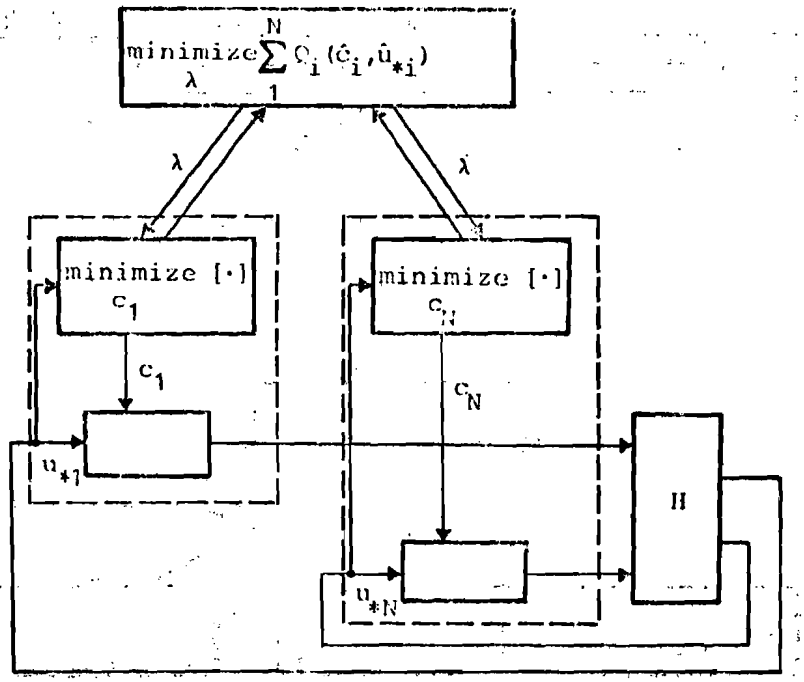


Fig.18 Decentralized control with on-line price coordination.

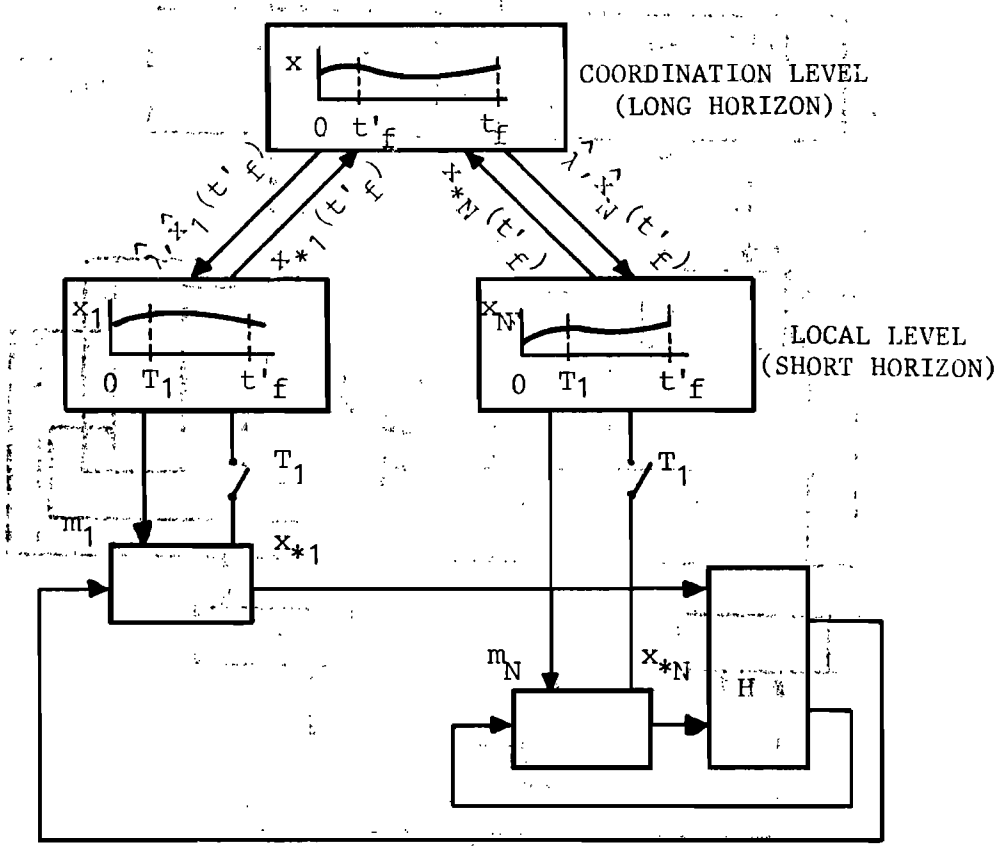


Fig. 19 Structure of on-line dynamic price coordination.

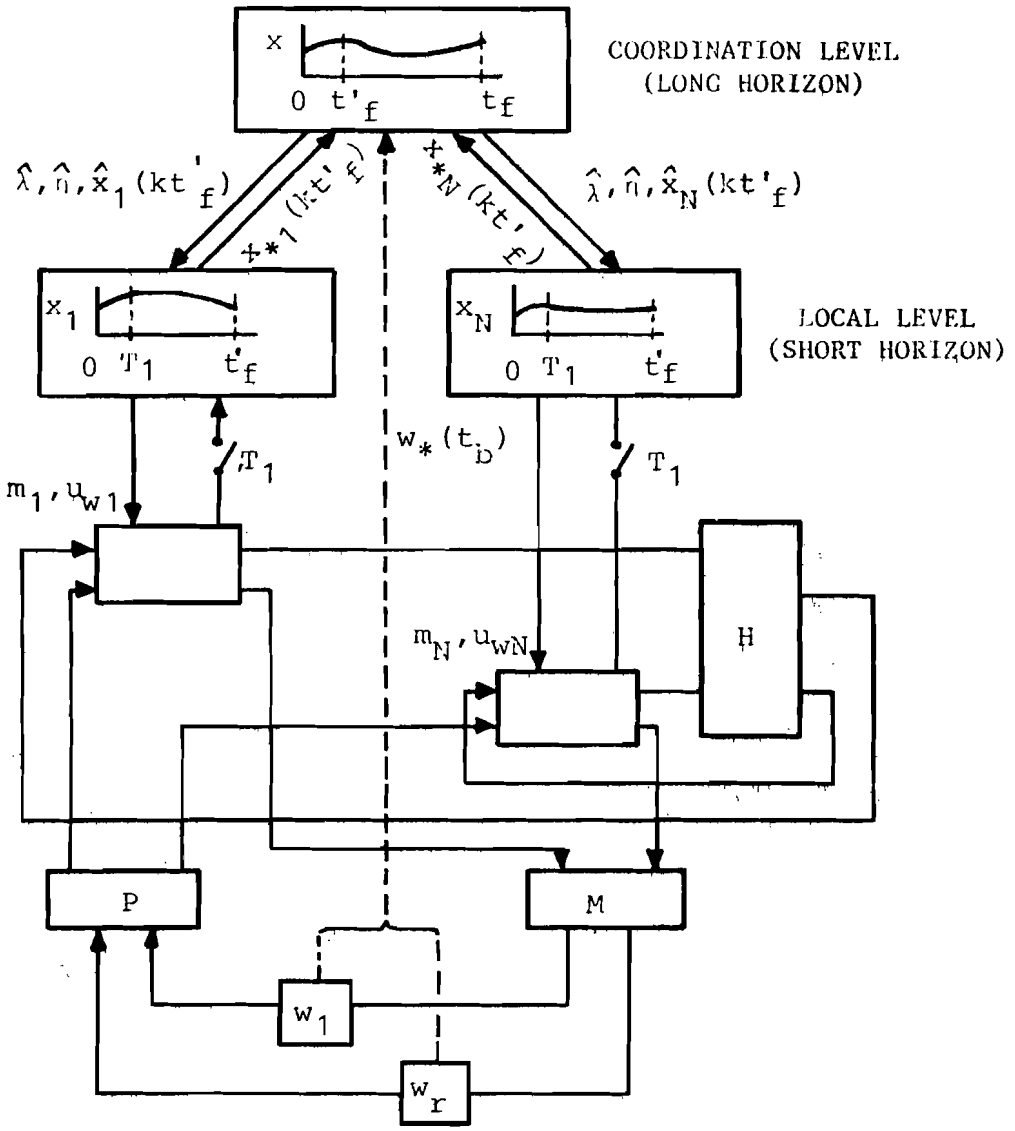


Fig. 20 On-line dynamic price coordination in a system containing stores in the interconnections.

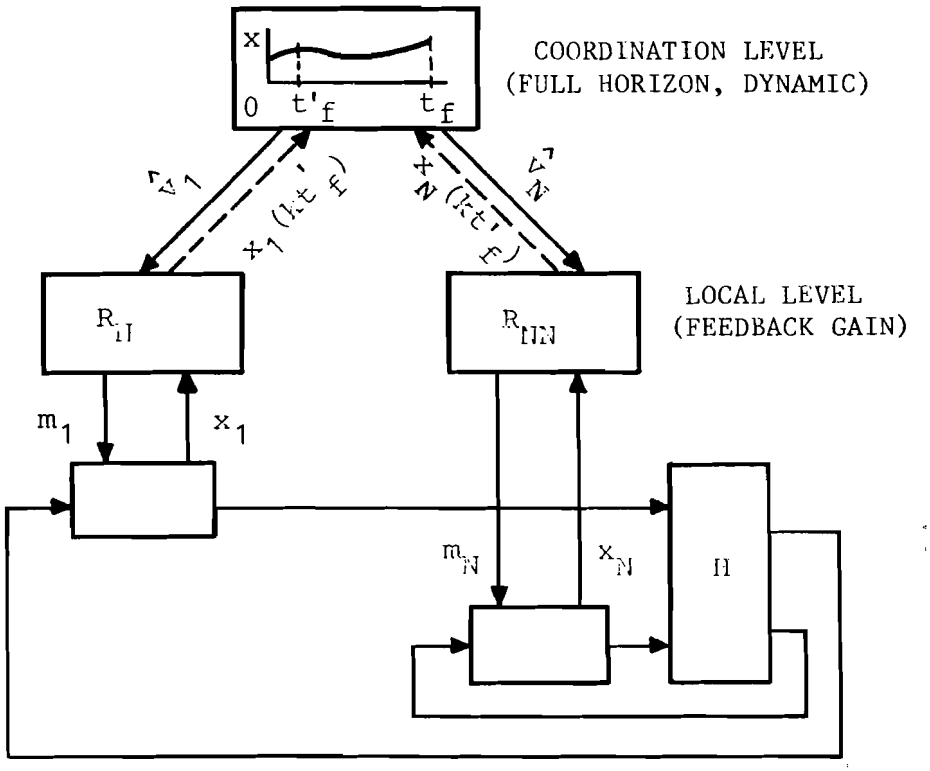


Fig. 21 Dynamic multilevel control based on feedback gain concept.

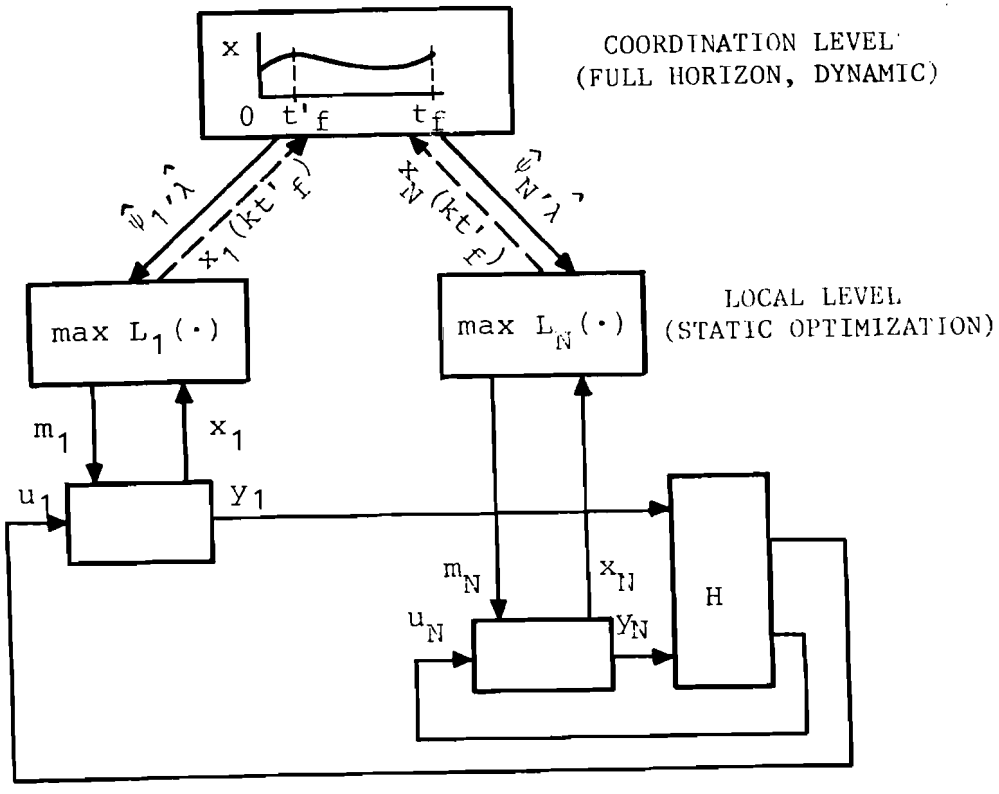


Fig. 22 Dynamic multilevel control using conjugate variables