Athens 24.05.1996-24.07.1996

Fiedler József

Correlation

The theoretical questions of correlation technology were clarified a long time ago as it is known in the relevant literature.
The application of this extremely interesting and valuable technology has got greater significance in the past few years, since modern digital technology was widely used in correlation technology.

The fact that noisy signs and systems can be handled well with the help of correlation technology, and during the measuring one can get accurate results gave the biggest stimulus in the development of it.
Thus correlation technology appears in modern identification methods, for example.
Correlation technology makes it possible to examine the system during its normal functioning. This is an important characteristic of the measuring method because the system parameters often change during natural functioning.

Certain parameters need to be changed then according to the new situation for the optimal functioning of the system.
The role of the correlation methods is also significant in regulation technology.
The internal parameters of systems can be kept within the given limits with the help of them, and functioning conditions can be made optimal. The possibility to search cause-result connections, the identification of malfunctioning units became general with the help of correlation methods.
The fast fault detection and identification during functioning save time and are very significant from operation economic aspects.

The rapid development of microprocessor technology gave a big stimulus to the development of correlation analyzers. New equipment could be built with the help of digital technology, and the development of computer software was connected to this. This way numerous new computing methods, widely applicable program packs were published.
FFT (Fast Fourier transformation), which is one of the main aids of modern correlation measuring, is probably the most widely known among them.

Briefly about the classification of signs

We can talk about correlation technology only if we know its place and role in sign theory.
We distinguish two main sign categories in practice:

- **deterministic and**
- **stochastic signs**

The deterministic signs can be divided further into two categories; into

- **periodical and**
- **non-periodical categories.**

Periodical signs have the following features:

The changes in the amplitudes of the signs are repeated after a certain time and this time is the period time. These signs were named periodical because of this feature.
The momentary values of the periodical signs are know at every moment, even if we choose that moment in the past, present or the future. That is why we can tell the size of the periodical sign without doubt at every moment. This feature is called deterministicism.
For non-periodical signs, only the last feature is true: signs that belong to this category are deterministic.

The signs that run only once, without repetition, and where the amplitude can be determined at every moment just like in case of periodical signs, belong to this category.
Signs that lack the deterministic characteristics belong to the other category. Telling the momentary values in advance is not possible here, the changes in the signs that far do not determine the value taken at the next moment. We call these signs stochastic signs.
Stochastic signs can be divided into two groups according to whether they are stationary or not. Stationary stochastic signs have two types:

ergodic and non-ergodic

We regard stochastic procedures as statistic phenomena, thus their description can be done by statistical parameters. These statistical parameters are not necessarily constant, but they can be parameter functions, too.
Every technical process that has chance characteristics can be described by stochastic signs. Stochastic signs describe a time function in practice that is governed by the principles of probability, that is why their accurate process cannot be given.
The analytic description of such signs is not possible directly, we can give only statistical characteristics or characteristic functions. The probability distribution function, the probability density function and the different averages are suitable for such description.
It may happen that these simple statistical characteristics are not enough for the description of the stochastic process. In this case the examination of higher level distribution and density parameters are necessary. There are stochastic processes for the probability description of which some characteristic values or functions are enough. One has to know only the dispersion of the process to describe the Gauss process, for example. This can be seen from the probability density function given by the Gaussian noise equation.Modelling according to the Gauss-process is widespread. In those cases where the statistic features of the noises are not known in advance, or not undeniable, the examined sign can be regarded as one that has Gauss distribution on the basis of practical consideration. The function is usually defined only for the behaviour of the noise in the amplitude interval.The concepts of white, pink or other colour noise should not be mistaken with this. These concepts are used for the description of the noise processes in the frequency interval. For example, the name white noise means the following: every frequency value appears in equal measure in the chance process. Pink noise derives from white noise in a way where the lower frequency components are stronger. Colour noise is an optional chance process in which the different frequency values appear in different measure.

Correlation functions

So it follows from the aforesaid that the change of stochastic signs cannot be given in advance, the change is accidental.

One of the most important point of the examination of the sign is when we try to point out whether there is a connection between the values of the sign in one particular and a later moment.
Obviously our result will be appreciable only if we carry out a lot of measuring and we average the data.

The basis of this technology is the so called correlation function. The correlation function can be given by the following formula:

$$k_{xx}(t_1) = \lim \frac{1}{2T} \int_{-T}^{T} x(t)x(t+t1)dt$$

This expression gives information about what the connection, or correlation in other words, is between the values that are successive in time of one sign. Since the $k_{xx}(t_1)$ function refers only to the x(t) sign, we call it auto-correlation function.The auto-correlation function has the following interesting features:
The value of the auto-correlation function $k_{xx}(t_1)$ at the $t_1=0$ place is the x(t) equal to the x^2 square mean of the stochastic sign, and this the maximum value of the function.
The $k_{xx}(t_1)$ auto-correlation function gives the square of the mean of the stochastic sign at the $t = \infty$ place.

The $k_{xx}(t_1)$ auto-correlation function is an even function

$$\mathbf{k_{xx}(t_1)= k_{xx}(-t_1)}$$

The $k_{xx}(t_1)$ bigger is the t_1 interval the smaller is the value of the auto-correlation function. The faster changes describe the sign, the faster the value of $k_{xx}(t_1)$ decreases by the increasement of t1. Or the faster the value of decreases, the higher the frequency components of the stochastic signs are.
The $k_{xx}(t_1)$ auto-correlation function makes it possible to show the hidden periodical sign components in the x(t) stochastic sign.This feature enables it to select periodical signs from stochastic signs in case of relatively high noise level too. The period sign of the auto-correlation function is equal to the period time of the periodical component of the sign.CorrelatersThe development of sign and system analysis gave the correlation theory stimulus. Its significance was known a few decades ago. However, the correlation technology could occupy its real place in the last few years. This could be explained by the widespread of the cheap digital technology. The realization of even a rather complicated correlater is easy by these tools.

We divide the correlaters into two main categories:

- **real time correlaters**
- **numeric correlaters**

Both analogue and digital machines belong to the real time correlaters, however the digital solutions are used in nearly all the cases nowadays. Real time correlaters process the sign to be measured continually and their procedure usually follows the defining algorithm.

The advantages of real time correlaters are quickness and the fact that it is relatively cheap to produce them, and it is more flexible to handle them.
The advantages of numeric correlaters are the available accuracy and the multi-channel function.

The realization that the mathematical form of the correlation function can be useful mainly in the field of the oppression of noises was one of the most important result of the correlation technology research. Let us consider two signs that are in proportion with a useful sign. The input and output sign of a linear system can be such. The effect of the noise can be oppressed with the help of correlation.We usually regard the effect of the noise as an additive member, -z(t), the oppression of disturbances that take effect in an other way is not so easy.The two measured signs can be written by the following equations:

$$\mathbf{y1(t)=x(t)+z1(t)}$$
$$\mathbf{y2(t)=x(t)+z2(t)}$$

Let us form the cross-correlation function of y1and y2 this means the averaging of the y1(t)y2(t+t1) product by time, it can be written down formally by the following equation:

$$\mathbf{KKF\{y_1y_2\} = y_1(t)y_2(t+t_1)}$$

By putting the signs given in the previous equation into this one, we may write the following:

$$\mathbf{y1(t)y_2(t+t_1) = x(t)x(t+t_1) + x(t)z_2(t+t_1) + z_1(t)x(t+t_1) + z_1(t)z_2(t+t_1).}$$

It can be seen from the aforesaid that the correlation function of the two individual signs that do not influence each other disappears. That is why it should be examined in these cases if there is some kind of connection between the noise and the useful sign.

Noise oppression in practice

Measuring exercises which correlation technology is the best method to solve are frequent in medical measure technology.

It is widely known that physiological processes are followed by small value (microvoltage) biovoltage.
The measuring of these signs runs into difficulties because of two reasons:
On one hand they can be registered with a high noise level, on the other hand both the sign and the living organism are constantly changing. That is why the examination of the signs in a time interval gives result only in a few special cases (for example, ECG examination), and it is never enough for a more intensive examination. Thus it is obvious to use the measuring theory and measuring technology developed for the stochastic signs. Thus we can eliminate the effect of the undesirable signs, noise, and we can establish a few important statistical characteristics about the observed organism, which give a chance to the better understanding of the procedure.

The examination of fetal cardiac sounds can be carried out by similar methods. In this case the cardiac sign of the mother appears as an interfering sign at ECG measuring, its elimination is possible by the correlation interference oppression.

Modelling and examining noise on Helios operative system

I modelled the noise that is the target of the examination in the following way:

1.About the Helios

The helios is an unix-like operating system, that is designed to run on a wide range of transputer based architectures. Helios supports environments where transputers simply act as accelerator for existings machines.

Helios network:

A collection of processors controlled by Helios is called a Helios network.Helios handles the control of all the resources within the network, and provides a consistent mechanism for accesssing these resourches.Examples of these machies include the IBM PC and the SUN workstations.Standard server for the host machine, provided with the system software, acts as an interface to the transputer back and. This server allows the user to access all the standard features of the host environment, and appears to the Helios network like anz other Helios processor.
Helios networkk can be made up of different combinations of processors. It could, for example, be made up of a Motorola 6800, or an Intel 80286, plus one or more Inmos transputers.

Basic of Programming

We can use the following conventions:

I/O channel
Task

Pipe

Processor

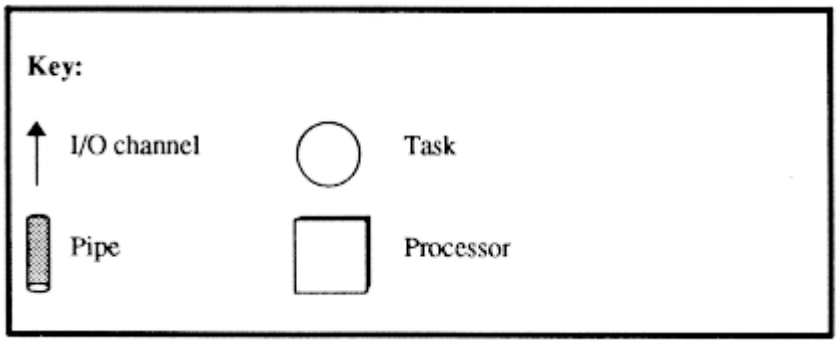

Figure 1.

The task

The smallst unit of parallelism in Helios is called: the task. A task is a self-contained unit, wich has been separately compiled and linked.

Helios provided an environment wich enables more than one task to be running at any time, either all running on the same processor or distributed amongst many processors.

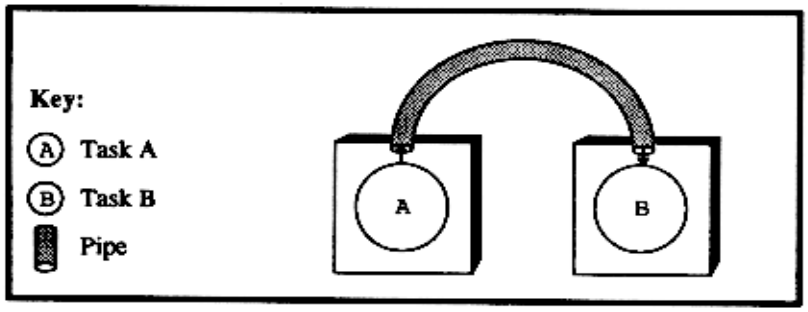

Figure 2

The Pipe

The tasks can communicate with each other, from within the program code, using standards reads and writes over channels of comunication called pipes. The pipe construct mentioned earlier, is so called because it connects pipes of communications between tasks.

The use of standards reads and writes for communication means that Helios tasks can written in standard languages such as C, Pascal or Fortran.

Them written in differnt languages can be mixed together.The routing of pipes between tasks, across the network of processors is handled automatically by Helios.

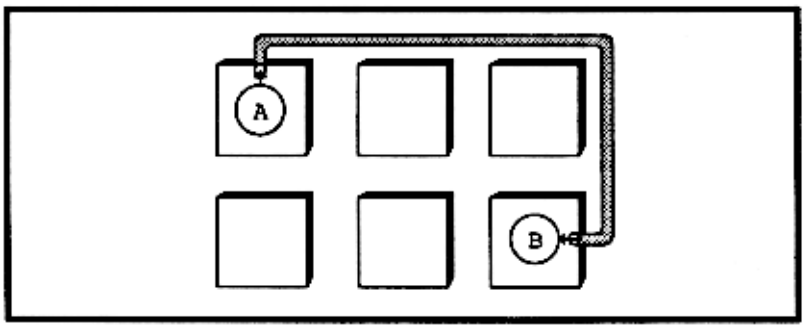

Figure 3

Pipeline

The simplest form of parallelism is one way pipeline of sequential tasks. The C compiler takes its input the file.c and sends assembly code the standard output channel(stdout). This channel is automatically routed by Helios to standard input channel(stdin) of the assembler. The assembler sends its output to the file file.o.

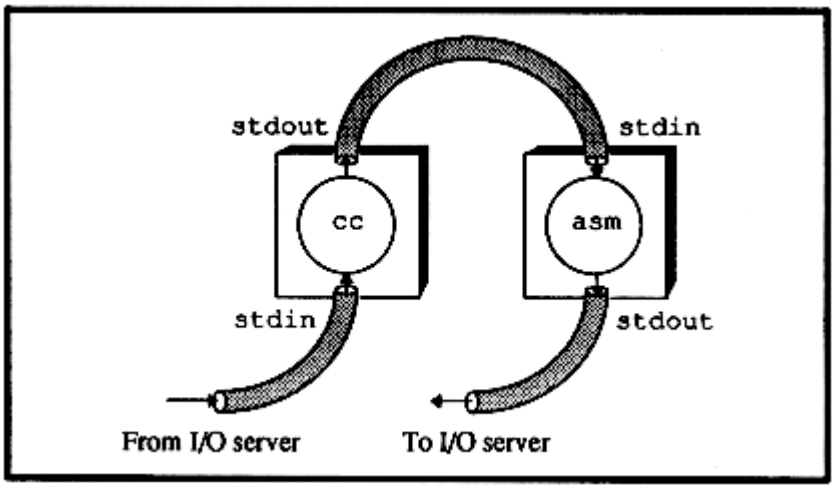

Figure 4.

The task force manager automatically distributes the tasks in the pipeline across the processor network.

Bidirecordional pipelines

Bidirecordional pipelines can be created just as easily as unidirectional pipelines using the subordinate construct. This construct works in the same way as the pipe construct, except it provides two-way link between the taks.

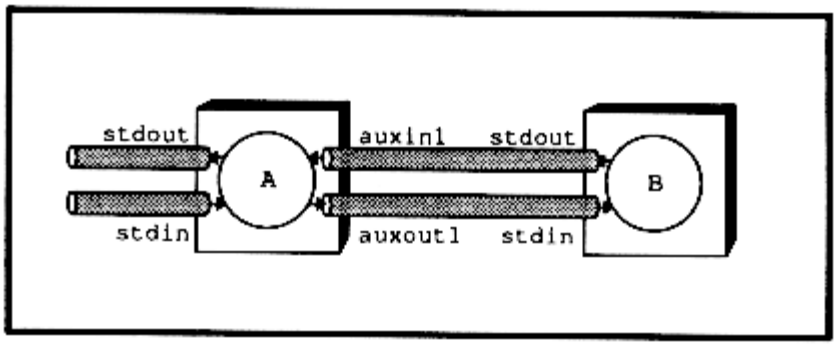

Figure 5.

Farm topology

Helios provides a powerful mechanism called farming for applications wich process large amounts of data in an identical way.

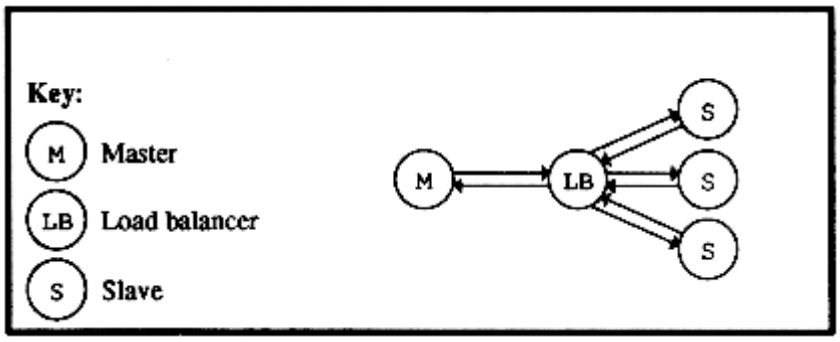

Figure 6

Load balancer, as its name suggest, balances the load across the slave tasks.

Transputers

The transputers consists of a high-performance processor, one - chip RAM. The one - chip RAM is very fast (40 ns access time on the 25 Mhz part) and allows fast data access.
The transputer family includes 16 -bit and 32 -bit architecture processors. We can use the T800 and T414 Inmos transputers, but we have used the T800 processors.
The T414 Transputer is a 32-bit processor, consists of a RISC CPU, 2 K of fast on-chip RAM, an external memory interface and four serial links wich may run at 5, 10 or 20 Mbits/second.The T800 is similar eexcept that it containstins floating ppoint processor and 4K of RAM.

Task forcees

A group of related tasks is called a task force. Bz related we mean tasks wich work together within a single application. Helios has a piece of code called the task force manager. It is resides on the root processor. The task force manager automatically dstributes task across the processors in the network, matching the resources avaiable with those required bz the task force.
Helios nucleos
The nucleus is the minimum system that must be present on every processor under helios control. The nucleus consists of the Kerel, System, Library, Loader and processor manager.

Client-server model

Helios is based on the Client-Server model in wich client wanting some service sends a message to a server wich performs the operation and returns a replay message.
All tasks are either clients or servers.

1., The program and the hardware environment

The resource map file describes in detail the physical characteristics of the network of processors available to operating system. It is used by the network manager, when loading operating system onto the network for the first time, and by the task force manager, when deciding how to allocate tasks to processors. The resource map file includes the following information:

- **the number of processors**
- **the processor types**
- **how the processors are interconnected**
- **how much external memory**

The network manager use this information to determine which processor need to have the operating system nucleus loaded onto them. The task force manager uses this information to work out how to load the network, as specified in the resource map file, with the resources requested by the task force, as specified in the CDL file.
The resource map file must be written when the network is first put together and amended if the network is reconfigured. It must be compiled by the utility program (rmgen) before it can be used

by the network manager. This program simply converts textual resource map file into a more efficient binary representation.

Resource map file consist of a set of network descriptions called subnets. Each subnet consist of a set of processor descriptions. A processor description is a terminal or a subnet description. It is the terminal descriptions which describe the resource available to a particular processor and how the processor and how the processors in the network are connected. The subnet construct enables complicated networks to be divided into a set of related subnetworks. It is entirely up to you how you decide to divide your networks. You may decide to have one large network or several small networks intended for different applications. for example, a network of processors which is shared between several users might be divided into a number of subnetworks, one per user. It is always a good idea to draw a picture of our network before attempting to write the research map file.
The component distribution language (CDL) is used to defined groups of related tasks called task forces. A CDL definition consist of two parts:

- a description of how the tasks are connected together
(e., how the communicate with each other).
-an optional description of the resource requirements
for each task.

The CDL task force descriptions are used by the task force manager when loading task forces onto the network. The task force manager attends to reconcile the research requirements us specified in the CDL file, with the network resources us specified in the resource map file. The task force manager also ensures the tasks are correctly connected together no matter how the tasks are physically distributed across the network.. CDL allows any task force to be described in a language and network independent way. The language independent aspect of CDL enables task forces to include task forces written in more than one language. The network independent aspect of CDL enables task forces to be run on a single processor network as easily as on a multiprocessor network without any change to the CDL description. It is the task force manager which decides where to place the tasks on the available processors and which ensures that all the communication pipes are correctly connected.

The noise analyse

In my work, I tried to find the details of periodical signals in the noise.
I have used the feature of autocorrelation that I wrote the last pages: The auto-correlation function

makes it possible to show the hidden periodical sign components in the stochastic sign.This feature enables it to select periodical signs from stochastic signs in case of relatively high noise level too.
We cav see the model of noise on the on the figure 8. The file name is bit3.
The ring model that I choosed, it works following:
The T00 modul read the noisefile from **stdio** and the samples, that we try to find in the noise. The lenght of samples: 1,2 and 3 bytes.
When the noise and samples have been in **T00**, it send 1 byte and the samples to the **T01**. T01 will compare the 1 byte lenght sample and details of noise and it send the second and third samples to the **T02** as vell as the details of noise continuously.
The **T02** will make the compare with 2 bytes and send result and the bytestream to the T03 also.
When finished the **T01, T02, T03**, they send results to the **T00**. T00 store the results and send them ti the stdio.

The used topology

The definitions file for the ring structure is the following:

```
subnet /noise_anal              {
CONTROL Rst_Anl [/ac/00];
terminal 00{ ~IO,,~01, ;        SYSTEM;
                Mnode Rst_Anl[pa_ra.d];
                ptype T800;
}
terminal 01 { ~00, ~02, ,~03 ; HELIOS;
                        memory #100000;
                        ptype T800;
        }
terminal 02 { ~01 , , , ~03  ;   HELIOS;
                        memory #100000;
                        ptype T800;
        }
terminal 03 { ~01 , ~02 , ,   ;   HELIOS;
                        memory #100000;
                        ptype T800;
        }
terminal IO { ; IO; }
}
```

In the Figure 7. we show the ring, the used transputers, numbered from **T00** to **T03**, the links of the processors from **0** to **3** in each transputer box, the data streams from **stream 0** to **stream 4** allocated by the **HELIOS** *parallel operating system*, and the I/O numbers in braces.

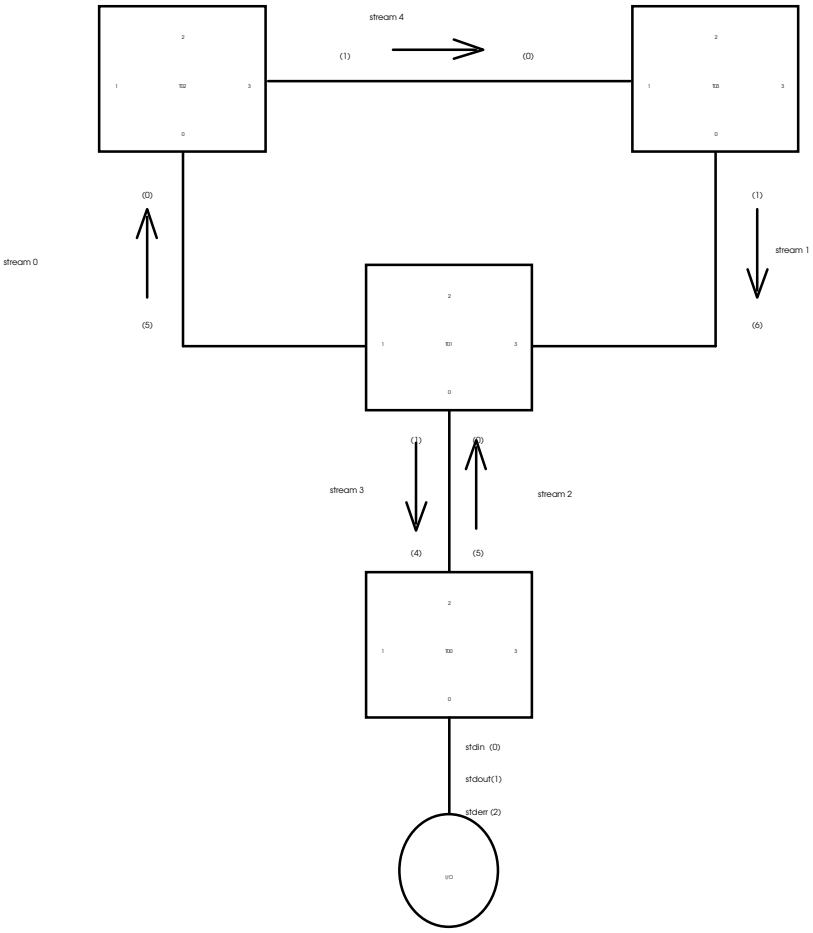

Figure 7. The ring structure, the I/O numbers and the data streams

Run of the program

/helios/usr/jfiedler/fj noise.txt

Ask the sample: ab0

result[0]: 717
result[1]: 46
result[2]: 2

Ask the sample: 65b

result[0]: 838
result[1]: 44
result[2]: 0

Ask the sample: 291

result[0]: 736
result[1]: 43
result[2]: 5

Ask the sample: fff

result[0]: 501
result[1]: 12
result[2]: 0

Ask the sample: ff0

result[0]: 501
result[1]: 12
result[2]: 0

Ask the sample: 73a

result[0]: 592
result[1]: 30
result[2]: 0

This work was sponsored by the TEMPUS. The 16 transputer containing PARSITEC and the HELIOS parallel programming tool was provided by the University of Athens. Special thanks to Prof. Sergios Theodoridis, dr. Yianis Cotronis and Costas Vassilakis.