INTELLIGENT SIMULATION ENVIRONMENTS

Edited by

Paul A. Luker, PhD

Heimo H. Adelsberger, PhD

Simulation Series
Volume 17
Number 1

SCS THE SOCIETY FOR COMPUTER SIMULATION

INTELLIGENT SIMULATION ENVIRONMENTS

Proceedings of the Conference on
Intelligent Simulation Environments

23-25 January 1986
San Diego, California

Edited by
Paul A. Luker, PhD
California State University, Chico
and
Heimo H. Adelsberger, PhD
Texas A&M University

Simulation Series
Volume 17
Number 1
January 1986

A Society for Computer Simulation (Simulation Councils, Inc.) publication.
San Diego, California

Philippe Geril, Managing Editor

(Society for Computer Simulation)
P.O. Box 17900
San Diego, California 92117

ISSN 0735-9276

PRINTED IN THE UNITED STATES OF AMERICA

INTELLIGENT SIMULATION ENVIRONMENTS

Titles in the *SIMULATION SERIES*

Vol. 1 No. 1 *Mathematical Models of Public Systems*
George A. Bekey, PhD, Editor
January 1971

Vol. 1 No. 2 *Systems and Simulation in the Service of Society*
David D. Sworder, PhD, Editor
July 1971

Vol. 2 No. 1 *The Mathematics of Large-Scale Simulation*
Paul Brock, PhD, Editor
June 1972

Vol. 2 No. 2 *Recent Developments in Urban Gaming*
Philip D. Patterson, PhD, Editor
December 1972

Vol. 3 No. 1 *Computer Simulation in Design Applications*
Said Ashour, PhD, and
Marvin M. Johnson, PhD, Editors
June 1973

Vol. 3 No. 2 *Simulation Systems for Manufacturing Industries*
Marvin M. Johnson, PhD, and
Said Ashour, PhD, Editors
December 1973

Vol. 4 No. 1 *Annotated Bibliographies of Simulation*
Tuncer I. Ören, PhD, Editor
June 1974

Vol. 4 No. 2 *Spanning the Applications of Simulation*
Paul Brock, PhD, Editor
December 1974

Vol. 5 No. 1 *New Directions in the Analysis of Ecological Systems: Part 1*
George S. Innis, PhD, Editor
June 1975

Vol. 5 No. 2 *New Directions in the Analysis of Ecological Systems: Part 2*
George S. Innis, PhD, Editor
December 1975

Vol. 6 No. 1 *Toward Real-Time Simulation (Languages, Models, and Systems), Part 1*
Roy E. Crosbie, PhD, and
John L. Hay, PhD, Editors
June 1976

Vol. 6 No. 2 *Toward Real-Time Simulation (Languages, Models, and Systems), Part 2*
Roy E. Crosbie, PhD, and
John L. Hay, PhD, Editors
December 1976

Vol. 7 No. 1 *An Overview of Simulation in Highway Transportation: Part 1*
James E. Bernard, PhD, Editor
June 1977

Vol. 7 No. 2 *An Overview of Simulation in Highway Transportation: Part 2*
James E. Bernard, PhD, Editor
December 1977

Vol. 8 No. 1 *Simulation of Energy Systems: Part 1*
Kenneth E.F. Watt, PhD, Editor
June 1978

Vol. 8 No. 2 *Simulation of Energy Systems: Part 2*
Kenneth E.F. Watt, PhD, Editor
December 1978

Vol. 9 No. 1 *Simulation in Business Planning and Decision Making*
Thomas H. Naylor, PhD, Editor
July 1981

Vol. 9 No. 2 *Simulating the Environmental Impact of a Large Hydroelectric Project*
Normand Thérien, PhD, Editor
July 1981

Vol. 10 No. 1 *Survey of the Application of Simulation to Health Care*
Stephen D. Roberts, PhD, and
William L. England, MSEE, Editors
December 1981

Vol. 10 No. 2 *Computer Modeling and Simulation: Principles of Good Practice*
John McLeod, PE
June 1982

Vol. 11 No. 1 *Peripheral Array Processors*
Walter J. Karplus, PhD, Editor
October 1982

Vol. 11 No. 2 *Computer Simulation in Emergency Planning*
John M. Carroll, PhD, Editor
January 1983

Vol. 12 No. 1 *Lumped-Parameter Models of Hydrocarbon Reservoirs*
Ellis A. Monash, PhD, Editor
March 1983

Vol. 12 No. 2 *Computer Models for Production and Inventory Control*
Haluk Bekiroglu, PhD, Editor
January 1984

Vol. 13 No. 1 *Aerospace Simulation*
Monte Ung, PhD, Editor
February 1984

Vol. 13 No. 2 *Simulation in Strongly Typed Languages: ADA, PASCAL, SIMULA . . .*
Ray Bryant, PhD, and
Brian W. Unger, PhD, Editors
February 1984

Vol. 14 No. 1 *All About Simulators, 1984*
Vince Amico and
A. Ben Clymer, Editors
April 1984

Vol. 14 No. 2 *Peripheral Array Processors*
Walter J. Karplus, PhD, Editor
October 1984

Vol. 15 No. 1 *Emergency Planning*
John M. Carroll, PhD, Editor
January 1985

Vol. 15 No. 2 *Distributed Simulation 1985*
Paul F. Reynolds, PhD, Editor
January 1985

Vol. 16 No. 1 *Simulators*
John S. Gardenier, D.B.A., Editor
March 1985

Vol. 16 No. 2 *Aerospace Simulation II*
Monte Ung, PhD, Editor
January 1986

Vol. 17 No. 1 *Intelligent Simulation Environments*
Paul A. Luker, PhD and
Heimo H. Adelsberger, PhD, Editors
January 1986

Preface

Five years ago, the theme of this conference would not have been a candidate for a session in a larger conference such as the Summer Computer Simulation Conference. What, then, has changed to warrant this topic as a full conference?

The short answer is that users have at last become more demanding of their software. Established practitioners of simulation have sought more realism from their simulations, which requires considerably more support throughout the modeling process and the implementation of their results. However, simulationists have encountered barriers when trying to translate their simulation requirements into a working, representative tool and that is the software. From the experienced to the novice, all have found simulation software to be lacking.

A simulation language is clearly not enough for either extremes of the user spectrum. The expert needs more support than a language can provide; there must be an integrated software environment to support all stages of the simulation process from requirement specification to the analysis of results. The novice, on the other hand, does not have the knowledge or experience required to utilize a simulation language effectively. However simple a language is claimed to be, the inescapable fact remains that programs must be written. To get the best out of any language, some knowledge of simulation techniques and algorithms is essential. The software environment must provide a simple interface while offering expertise within the system to construct and conduct the simulation for the user. Given a software support environment with expertise, the user is then free to concentrate on the problem rather than having to be concerned with simulation techniques and practices.

This conference owes its existence to the fact that computer science has advanced significantly in the last five years and software environments in general, have received a lot of attention. Artificial intelligence techniques and tools are now being applied in simulation after having been perceived to be the province of theoreticians for so long. Our own perspective on the resulting marriage is obviously through the focus of simulation which is, after all, only one application of computers. However, the relevance of the union encompasses broad areas of computer science.

Papers published in proceedings never fall into neat categories; this proceeding is no exception. It would be sad if we were all neatly compartmentalized. If the allocations of papers incur the wrath of authors and/or presenters, then I apologize. The papers included address a wide range of topics – some discuss the design of simulation environments, others describe their application. Many of the contributions are concerned with the detailed building blocks from which intelligent simulation environments are constructed.

In addition to the papers published for this conference, we have also included papers presented at the joint session with Modeling and Simulation on Microcomputers. The papers for the two joint sessions with the conference on Continuous System Simulation Languages are published in that proceedings. It was not possible to duplicate them for both proceedings.

Heimo H. Adelsberger has organized the State-Of-The-Art Tutorials held as a parallel track in the ISE conference. These provide an interesting and informative background to the technical papers and all are published in this proceedings.

Simulation is at an exciting stage of development. Let us look forward to the next five years! Finally, I would like to take this opportunity to thank all contributors to the conference, all the delegates, and all the staff at SCS. Poor Rosemary Whiteside bore the brunt of having to co-ordinate the activities and inactivities of a discrete set of chairmen.

Paul A. Luker, PhD
Chico, California

SCS Multiconference Board Representative

Roy Crosbie

California State University at Chico
Computer Science Department
Chico, CA 95929

SCS Multiconference General Chairman

Ray Swartz

Berkeley Decision/Systems, Inc.
150 Belvedere Terrace
Santa Cruz,.CA 95062

INTELLIGENT SIMULATION ENVIRONMENTS CONFERENCE

Co-General Chairmen

Robert Shannon

Texas A&M University
c/o 3008 Brothers Road
College Station, TX 77840

Graham Birtwistle

University of Calgary
Dept. of Computer Science
Calgary, Alberta T2N 1N4, Canada

Co-Program Chairmen

Paul A. Luker

California State University, Chico
Dept. of Computer Science
Chico, CA 95929-0410

Heimo H. Adelsbserger

Texas A&M University
Dept. of Computer Science
College Station, TX 77843-3211

Conference Committee

Robert Howe

University of Michigan
Department of Aerospace Engineering
Ann Arbor, MI 48109

Floyd E. Nixon

Martin Marietta Corporation
P.O. Box 5738; M/P 170
Orlando, FL 32085

Wade P. Webster

Lockheed-EMSCO/C7
1830 NASA Road #1
Houston, TX 77258

CONTENTS

Volume 17, Number 1

	Page	Authors
Preface	vii	*Paul A. Luker, PhD*
Knowledge Bases		
Using cellular automata in graph theory modelling: A high performance solution of the HAMILTON problem	3	*J.C. Perez* *R. Castanet*
A rule based expert simulation environment	9	*Paul Robertson*
Knowledge bases for an advanced simulation environment	16	*Tuncer I. Oren*
Expert Systems		
An application of declarative modeling to aircraft fault isolation and diagnosis	25	*David Wahl*
The use of TC Prolog for medical simulation	29	*Tibor Deutsch* *Ivan Futo* *Imre Papp*
Plant experience with an expert system for alarm diagnosis	35	*K.L. Gimmy*
A framework for knowledge-based systems unifying expert systems and simulation	38	*Mildred L.G. Shaw* *Brain R. Gaines*
Simulation Environments		
SCCA: Simulation environment for concurrency control algorithms in distributed databases	47	*Karl Kleissner* *Markus Stumptner*
A simulator environment for an autonomous land vehicle	53	*Jay Glicksman*
An environment for discrete event simulation	58	*Frank J. Wales* *Paul A. Luker*
A distributed software prototyping and simulation environment: JADE	63	*Brian Unger* *Alan Dewer* *John Cleary* *Graham Birtwistle*
Applications		
Predicting X-tree network performance using the Jade environment	75	*Brian Unger* *Li Xining*
An initial framework for simulations of discontinuous recall and inference in multilayered nerve sets	80	*Rolf Martin*
PDP mechanisms for intelligent display control	87	*Timothy P. McCandless*
Application of artificial intelligence to improve plant availability	92	Michael V. Frank Steven A. Epstein
Complex Models		
System detected modeling tensions: System vs user intelligence	101	*R.T. Newkirk* *R.L. Walker*
Rule based object oriented simulation systems	107	*Heimo H. Adelsberger* *Udo W. Pooch* *Robert E. Shannon* *Glen N. Williams*

CONTENTS (Continued)

Volume 17, Number 1

	Page	Authors
Fifth generation simulation	118	Edward L. Davis
Joint session with Modeling and Simulation on Microcomputers and Intelligent Simulation Environments		
Development of an expert system for simulation model selecton	121	*Robert A. Campbell*
The microcomputer version of TC-Prolog	123	*Ivan Futo* *Imre Papp* *Janos Szeredi*
An expert simulation model builder	129	*Behrokh Khoshnevis* *An-Pin Chen*
Dialog-oriented and knowledge-based modeling in a typical PC environment	133	*A. Lehman* *B. Knodler* *E. Kwee* *H. Szczerbicka*
State-of-the-art Tutorials		
Introduction to artificial intelligence	141	*Heimo H. Adelsberger*
Expert systems and simulation in industrial applications	144	*Brian R. Gaines*
Intelligent simulation environments	150	*Robert E. Shannon*
A view of LISP	157	*Graham Birtwistle* *John Kendall*
A Prolog tutorial	163	*Gustaf Neumann*
Natural language processing	165	*Alexa T. McCray*
Author Index	167	

Knowledge Bases

Using cellular automata in graph theory modelling: A high performance solution of the HAMILTON problem

J.C. PEREZ*, R. CASTANET**

*IBM FRANCE Bordeaux ROBOTICS CENTER
Route de Canejan 33610 Cestas France

**BORDEAUX I University COMPUTER SCIENCE DEPARTMENT
351, Cours de la Libération 33405 Talence France

ABSTRACT

After general considerations on a parallel approach of graph theory problems, we present a specific problem : "To find hamiltonian circuits in a 3-vertex connected cubic graph".

This problem is similar to the "TRAVELLING SALESMAN PROBLEM". We present a parallel KNOWLEDGE REPRESENTATION of the graph. The parallel algorithm uses parallel PROPAGATION of CELLULAR AUTOMATA.

This EXHAUSTIVE approach is combined with a powerful HEURISTIC method. The result is a powerful polynomial algorithm which finds Hamiltonian circuits in complex graphs.

Meanwhile, an open-problem is discussion and research of cases where this algorithm fails.

I INTRODUCTION

Our major research directions in cellular automata area are : graph theory modelling, biology modelling, parallel computing, artificial intelligence.

This paper deals with the 1-st area : graph theory modelling. Graph theory is present in many A.I problems and knowledge modelling areas. Generally, graph problems are resolved with current knowledge representation methods (stacks, index methods etc...).

We propose new and original knowledge representation methods in graph modelling (based on parallel cellular automata).

II THE POWER OF CELLULAR AUTOMATA IN GRAPH THEORY

2.1 THE POWER OF CELLS :

The initial background of this idea is an exploration of the CONWAY cellular automata, also known as "game of life" (1),(2).

In this research, we conclude that elementary rules processing at cellular level can control complex relations and mechanisms.

For example, the process which deduces the following pattern "CLOWN" (figure b) from the initial pattern "U" (figure a) could be understood and applied by a "3-year old child".

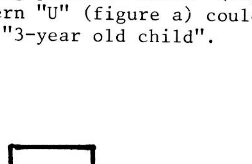


Figure a

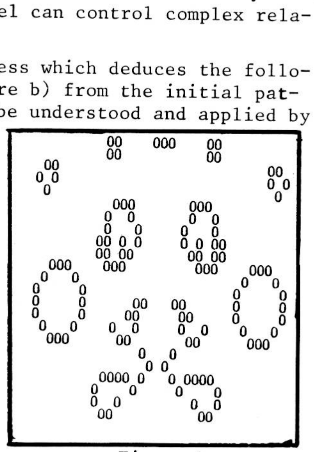


Figure b

Then, we choose to apply cellular automata in different areas such as A.I, biology, parallel computing and graph problems.

Definition :

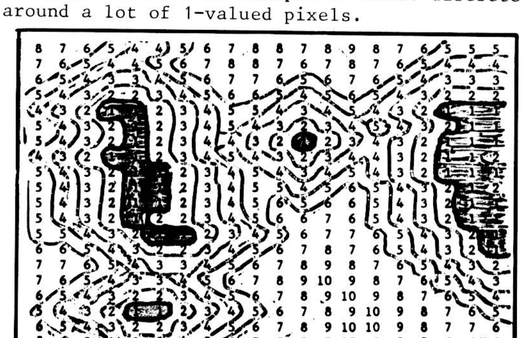
A cellular automata results from the dynamic network of cells (like a chessboard). All cells run, synchronously, the same simple algorithm. The next state of each cell depends on :
- its present state
- the state of its neighboring cells.

2.2 FIRST APPLICATION OF CELLULAR AUTOMATA IN GRAPH THEORY : "THE UNDULATORY CELLULAR AUTOMATA"

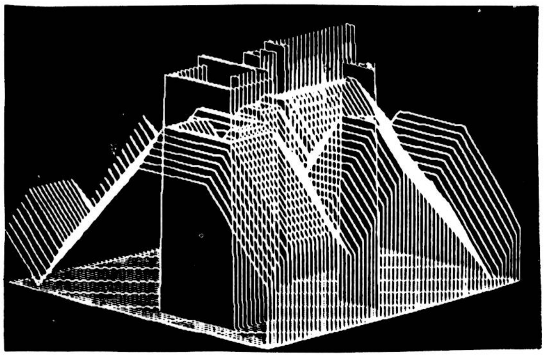
We define the following topics :

- A CELLULAR SPACE : there is a 2-dimension discrete network of cells. Each cell has a binary value (0/1).
- NEIGHBOURHOOD : Around each cell, we define a VON-NEUMANN neighbourhood : each cell can communicate with its 4 neighbours.
- PROPAGATION CELLULAR AUTOMATON : If a cell fires (at the time n), then it propagates a firing value (1) (at the time n+1) in the 4 neighbouring cells (and so on...). The operation is synchronous and run simultaneously for the whole cellular space.
- "UNDULATORY ANALOGY" : This propagation cellular automata is identical with discretisation of undulatory waves in a plan, like in a water surface (3).
- "FREE CELLULAR PROPAGATION" : The cellular space is free then waves can run and visit the whole cellular space. We use this approach in the parallel global vision expert-system "MONADES" (4).

Following you can see an example of these discrete waves around a lot of 1-valued pixels.



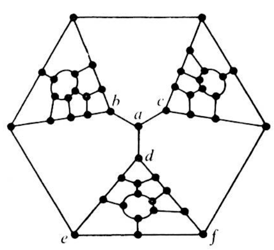
We remark that the full cellular space is self-organized around the "emission-pattern". Then, we obtain a potential distribution as shown by the following figure :



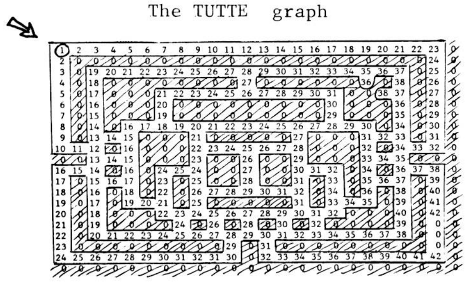
- "CONSTRAINT CELLULAR PROPAGATION" : The same undulatory cellular automata is used but the cellular space is restricted, like a labyrinth (with 2 classes of areas : propagation domains and forbidden domains).

If we continue using the "waterwaves analogy", it is like a "basin" with a set of paving-stones that canalize circulation of water.

Following is a labyrinth representing a discretization of the TUTTE graph :



The TUTTE graph



Discrete propagation in a labyrinth of the TUTTE graph

Immediatly, we can deduce from this figure the cellular parallel backgrounds for :
- algorithms in connexity search
- algorithms finding the shortest length path between 2 nodes in a labyrinth.

If we continue with "water analogies", this is equivalent to water which runs from the initial source point, with divisions when it arrives at each node, and so on ... This is, really, a "massive parallel approach".

2.3 AN OVERVIEW OF OUR WORK WITH CELLULAR GRAPHS :

At this point, we can imagine a large panel of experimentations in graph theory. We process graph problems using 3 successive approaches :

- ASYNCHRONOUS APPROACH : We work with asynchronous propagation in labyrinths (then, waves don't stop at nodes, and propagation is independant of nodes but depends on the length of edges).
- SYNCHRONOUS APPROACH : We use a synchronous propagation mechanism which is independant of the length of edges and which processes waves between nodes. There are 2 hierarchical levels of waves :
  . micro-waves inside edges (between 2 nodes)
  . macro-waves : synchronous waves between nodes (propagation nodes by nodes).
- RULES-BASED APPROACH : We implement such algorithms in the PROLOG5 inference system (5).

In this case, the graph is not discretized but described with PROLOG rules. The cellular propagation runs at the PROLOG5 parallel inferences level.

Then, we can summarize the major graph theory areas covered by our experimentations using such cellular methods :

=> IN GENERAL GRAPH PROBLEMS :
- . Find the faces, nodes, edges of a graph.
- . Minimum length path in a graph or labyrinth, using parallel cellular automata propagation.
- . Center, ray, and diameter localization in a graph.
- . Studying connectivity in graphs (like 2 vertex-connected etc...).

=> IN N-P COMPLETE COMPLEXITY GRAPHS PROBLEMS :
- . Find simple cycles in a hyper-graph (6).
- . Find Hamilton cycles in a graph (cubic and 3-vertex connected).
- . Find Steiner path tree problem. (Minimal spanning tree).

In these above problems, we demonstrate and run parallel polynomial algorithms, but we do not prove their validity.

All algorithms are based on cellular parallel propagation in labyrinths, graphs and hyper-graphs. Above algorithms provides discussion material about relationships between complexity, parallelism and cellular knowledge representation.

Another major remark is about "LOW-COST" computer technology : Present algorithms are very simple (cellular operations), consequently, there are realizable with VLSI technologies.

2.4 INTRODUCING NEW APPROACHES IN GRAPH THEORY

In graph theory, there are 3 major approaches :
- The ALGORITHMIC approach (polynomial computing time).
- The EXHAUSTIVE approach (exponential computing time).
- The HEURISTIC approach (polynomial but result not guaranteed).

... Global level and cellular level ...

All problems in graph theory are resolved working at the global level : Global level manipulates objects and relations, edges, nodes, faces etc...

Our algorithms work at the cellular level.

Then, the relation between EXHAUSTIVE and HEURISTIC must be reconsidered with this new approach.

For example, a problem like the HAMILTON problem is resolved :
- using EXHAUSTIVE search at CELLULAR level.
- using HEURISTIC search at GLOBAL level.

There are new situations ...

We say that :

> The change in knowledge representation could modify notions like complexity and algorithm efficiency (!)

Particularly, it is necessary to be very prudent in NP-complete complexity considerations.

Meanwhile, the following arguments go in the same direction :

- In BELL laboratories, J.J. HOPFIELD is studying cellular algorithms based on neuron-like circuits which solve the "travelling salesman problem" with highly efficient performances.
- In (7), A.K. DEWDNEY describes analog methods (based on cords, fluids, etc...) that can solve NP-complete problems in polynomial time (like the STEINER minimum spanning tree problem) !

We explain this phenomena, in the case of our systems, by the existence of a natural link between these properties of analog methods and our undulatory approach :

In fact, our undulatory algorithms are discretization in computers of analog properties : the propagation of waves in an undulatory area (waves etc...).

Naturally, this computerized approach benefits from the properties of analogic and continuous representations
(likely technique of numerical analysis conserve properties of the continuous phenomena modelled by numerical analysis discrete methods).

Finally, the major conclusion is a new view on mutual relations between EXHAUSTIVE and HEURISTIC searchs : a simple HEURISTIC method combined within EXHAUSTIVE process of cellular automata provides HIGH PERFORMANCE new algorithms and a new view of N-P complete and exponential problems.

III RESOLUTION OF THE HAMILTON'S PROBLEM

3.1 PROBLEM PRESENTATION

The Hamilton's problem is one of the major unresolved problems in the graph theory area (8). In this paper, we limit our research to the restricted case of a cubic 3-connected unoriented graph.

- Finding Hamilton circuits consists of drawing a circuit passing, only once, by each note.
- A cubic 3-connected graph has 3 edges by node and it is necessary to cut 3 edges to disconnect the graph in 2 distinct sub-graphs (8).
- The Hamilton's problem, in the case of a cubic 3-connected graph, is an NP-complete problem (9).

3.2 BACKGROUND RULES AND GUIDELINES

The following cellular automata solution of Hamilton's problem is based on the use of basic cellular tools :

- Automatic search of nodes, edges, faces in a cellular labyrinth.
- Research of the minimum length path between 2 nodes.
- Measurement of 1-connected, 2-connected, 3-connected.
- Potential topologic ordering of the graph around an emission node.

All these modules are fully parallel and use cellular automata propagation.

Using these background tools, we have redefined a new approach to the HAMILTON problem ; the major ideas are the following :

IDEA 1 : Relations between the Hamiltonian circuit and the maximum length path : The Hamilton problem is equivalent to the following problem : the Hamiltonian circuit is the longer path in the graph ; then if we consider one edge in the graph, we prove :

(a) This edge is included in the Hamiltonian circuit (if the graph is Hamiltonian)
(b) The union between this edge and the longer path joining the 2 nodes of the edge constitutes the Hamiltonian circuit.

Demonstrations :

(a) is deduced from a graph theory theorem (8) "if a graph has 1 Hamiltonian circuit, then this graph has $\geqslant 3$ Hamiltonian circuits." We can understand this theorem using coloration of edges : we can color the Hamiltonian circuit using 2 alternate colours and using the 3rd colour for the remaining edges.

Then, the 3 combinations of 2 out of 3 colours give the 3 Hamiltonian circuits.

(b) The Hamiltonian circuit is the union between the chosen edge and the maximum length path around this edge. Suppose a Hamiltonian graph with N nodes. Then, the Hamiltonian circuit has N edges. Then, the maximum length path around the edge must have N-1 edges.

IDEA 2 : Relations between maximum length path and minimum length path : "the school-boys path" :

Our heuristic strategy is the following :

- If at each time of the heuristic process, we can choose between the 2 disjointed minimum length paths, we keep the longer way path of the two and we eliminate the shorter path of the two.
- Then a flow of automatic deductions occurs and we must assure that the remaining sub-graph will be 3-connected ;
- then, it will be possible to choose, a new path, between 2 disjointed possibilities.
- and so on ...

In summarized terms, we assure that consecutive eliminations of shorter paths conduce to the larger path (!). For these reasons, we choose an analogy with a French proverb : "the school-boys path", school-boys when they leave school for home, at each choice, eliminate the shorter path ...

3.3 ALGORITHM SUMMARY

The algorithm is programmed in APL language and includes about 250 instructions distributed in about 30 modules ...

The major guideline is to build a major chain in which the 2 extremal nodes can be joined by 2 disjointed paths (3-connected graph). This property must be guaranteed at each step in the heuristic process.

Then, this chain grows and converges to the final step : the hamiltonian circuit (if the graph is hamiltonian) ; if the graph does not have a hamiltonian circuit, the algorithm generally converges towards to the hamiltonian path (!).

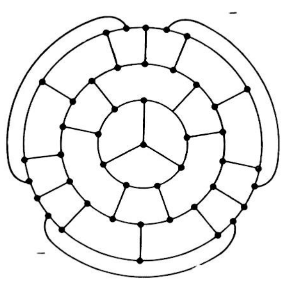
The algorithm uses in flip-flop the 3 following major functions :

- finding the "school boy path" at a local decision step. This choice must guarantee that the remaining sub-graph will be 3-connected.
- Absorption of secondary sub-chains by the major chain if the 2 chains are near and if the resulting chain guarantees a 3-connected property.
- Automatic deduction of edges after a decision. If an edge is included in the hamiltonian chain, then, other edges in relation becomes non-hamiltonian or hamiltonian (see figures).

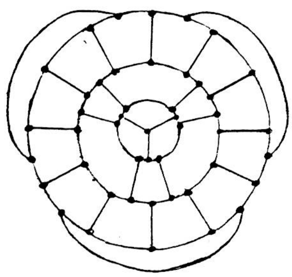
Example :

We can see, in the annex, at the end of this paper, an example running on a graph deduced from the KOZIREV and GRINBERG graph (8).

KOZIREV and GRINBERG is not hamiltonian.



The deduced following graph is Hamiltonian.



REFERENCES :

(1) M. GARDNER "Mathematical Games" (1970) Scientific American, oct 1970

(2) JH. CONWAY "Winning ways" New York Academic Press 1982

(3) J.C. PEREZ, Theoretical background rules for designing 5th generation intelligent computers, June 1985, Cognitiva 85 CESTA Paris France.

(4) J.C. PEREZ, "MONADES" A global vision expert-system based on a parallel undulatory knowledge propagation theory". ROVISEC 5 (ROBOT VISION), AMSTERDAM, the NETHERLANDS, 29-31 October 1985.

(5) J.C. PEREZ "Thinking by holograms in A.I", AI EUROPA Conference, WIESBADEN, West Germany, 24-26 September 1985.

(6) D.S. FUSSEL et al "A theory of correct locking protocols for database systems". Very large Data Bases, Cannes, France. September 9-11, 1981.

(7) AK DEWDNEY "Mathematical Games" Scientific American January 1985.

(8) C. BERGE "Graphes et hypergraphes". DUNOD 1970 Paris, France.

(9) M.R. GAREY et al. "the planar Hamiltonian circuit problem is N-P complete". SIAM. J. Comput. Vol 5 n°4 - December 1976

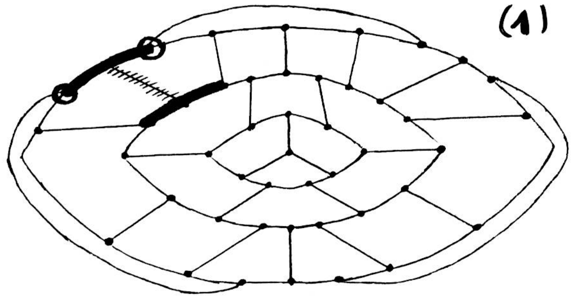
See ANNEX on the next page.

ANNEX : Using the algorithm for finding Hamiltonian circuit in the following graph (derived from KOZIREV and GRINBERG graph).

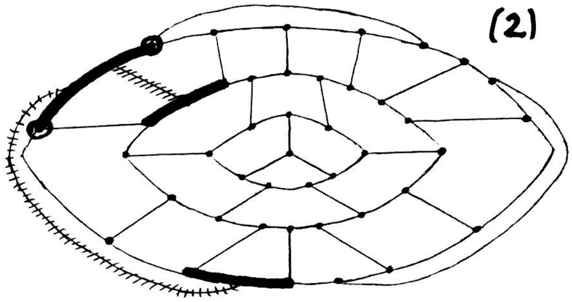
——— : unresolved edge   ///////// : forbidden edge

▬▬ : hamiltonian edge   o—o : major chain

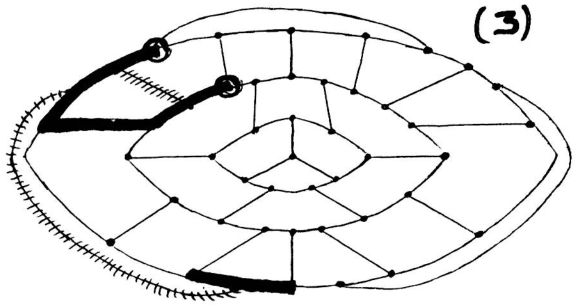
(1)



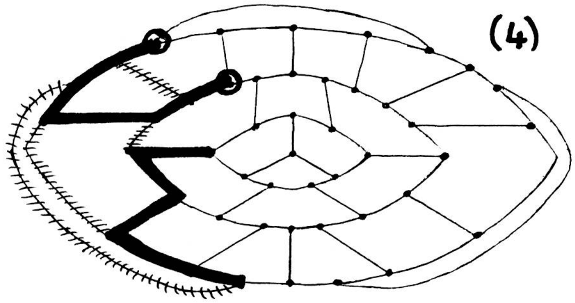
(2)



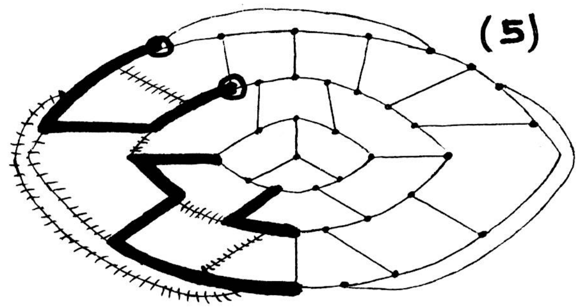
(3)



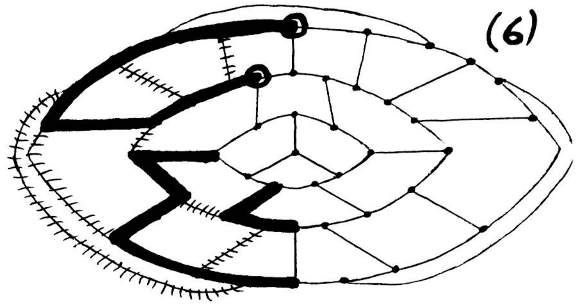
(4)



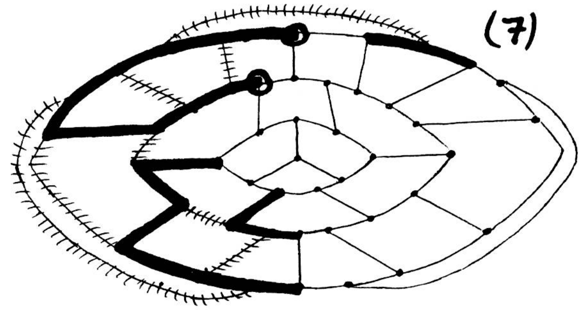
(5)



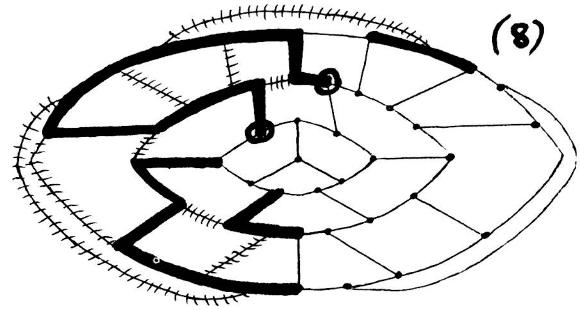
(6)



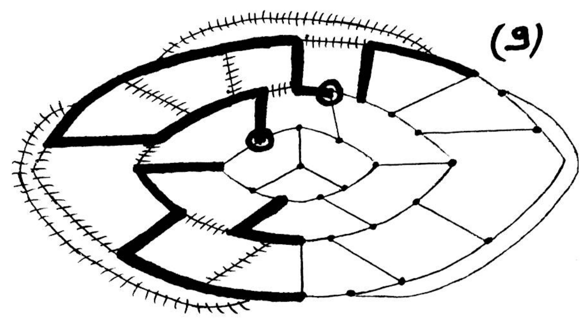
(7)



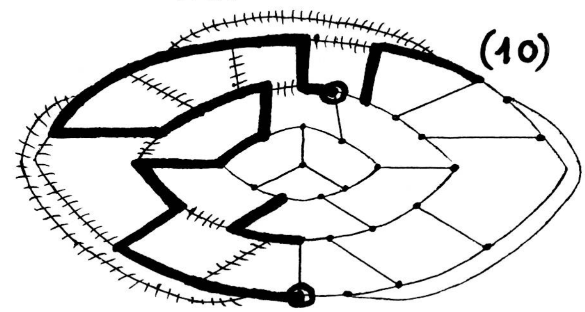
(8)



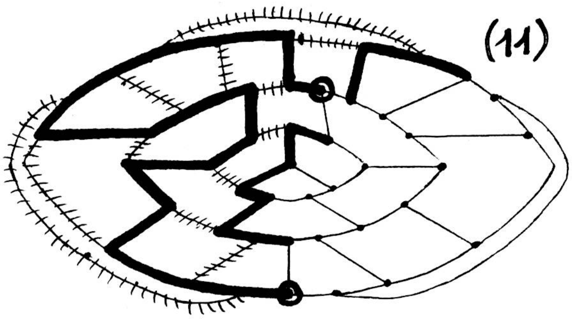
(9)



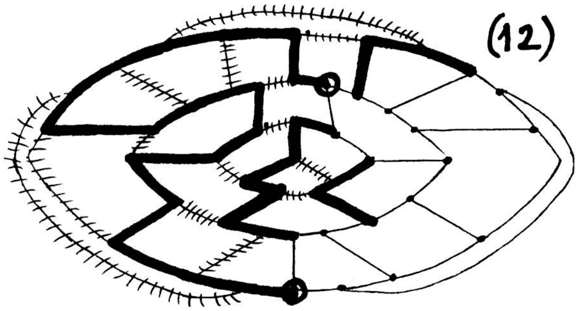
(10)



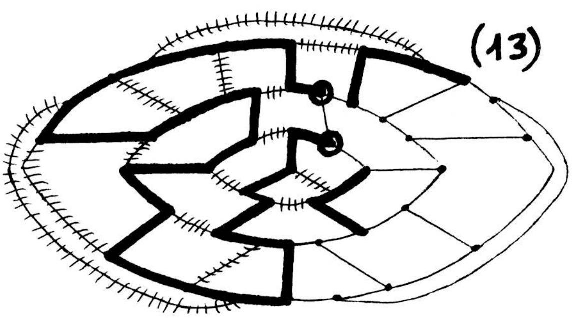
(11)



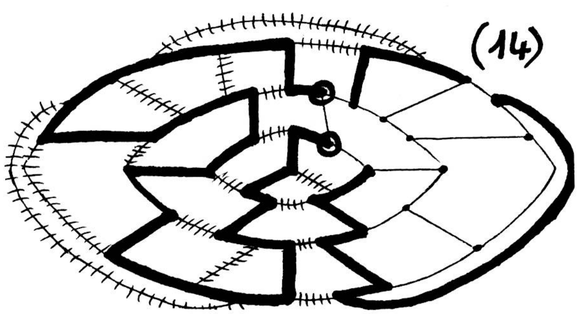
(12)



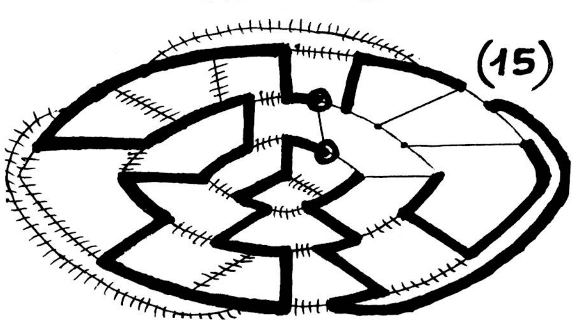
(13)



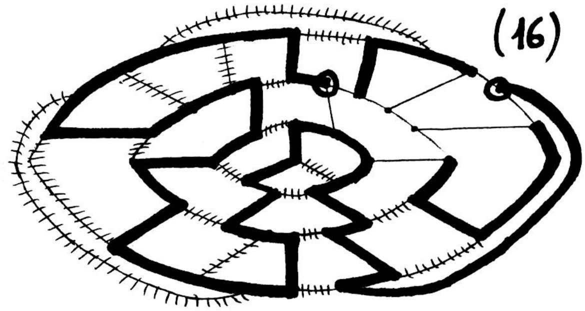
(14)



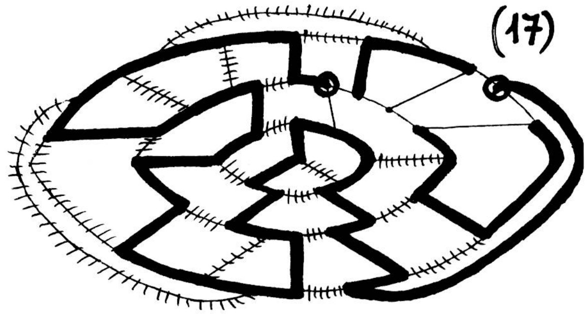
(15)



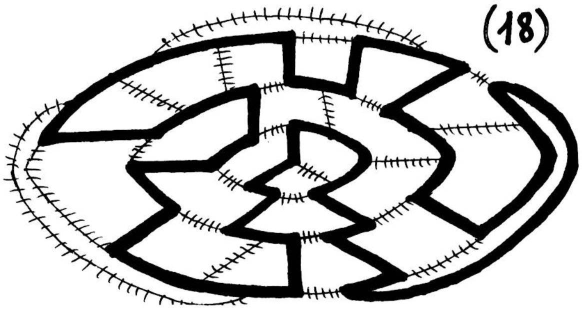
(16)



(17)



(18)



an overview
of the cellular
representation
of the graph