Mathematics and AI NEU Math Circle, April 1, 2007, Oliver Knill

Artificial intelligence

Can a machine solve mathematical problems or prove theorems automatically? Are there computer programs which find solution strategies for puzzles? Can one reconstruct a city in three dimensional form, from a movie taken from an airplane? Can a machine read a mathematical formula written by hand on a piece of paper. What score can a computer reach in a standardized test like SAT? Can a machine pass the **Turing test**, can it behave in an online chat so that it can not be distinguished from an actual human?

Here are some problems types in artificial intelligence. Most of them are purely mathematical:

- **problem solving:** playing games, performing algorithms, solving puzzles
- **pattern recognition:** speech, music, image, face, ocr, movie, plagiarism, spam
- **geometric reconstruction:** tomography, 3D reconstruction and scanning
- **research problems:** computer assisted theorem proving, finding theorems, verifying proofs
- **data mining:** knowledge acquisition, knowledge organization, learning
- **translation:** language translation, porting applications to new programming language
- **creativity:** writing poems, jokes, novels, music pieces, painting, sculpture
- **simulation:** physics engines, evolution of bots, game development, design of aircrafts
- **inverse problems:** earth quake location, oil depository, tomography
- **prediction:** weather prediction, global climate global warming, epidemics, supplies

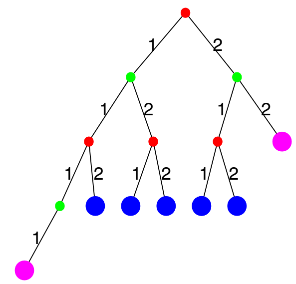
Moral: The field of artificial intelligence is extremely wide and interdisciplinary.

Problem List some human abilities, which computers are not yet able do.

Games

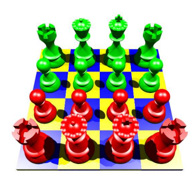
The ability to play games an important aspect of intelligence. Consider a game for two people A,B, which only has finitely many positions. The game is described by a graph with the initial position at the top and all the possible "end positions" at the bottom. Zermelo proved that there are three possibilities: either A has a winning strategy, or B has a winning strategy or both A and B have a tie strategy. For tic-tac-toe for example, there is a tie strategy for both players.

The theorem of Zermelo belongs to the origins of game theory: in a two player game with finitely many positions, where every player has full knowledge, there is either a winning strategy for one of the players or a draw strategy for both players.



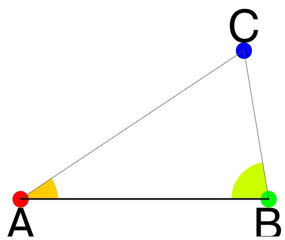
The game "4 wins": player A starts and chooses the number 1 or 2. Then player B chooses 1 or 2. The numbers are added. The player, who reaches 4 first wins. Clearly player A has a winning strategy. A plays 1, and then, after B has play plays 1 or 2 to get 4. If the sum is 3, then B has a winning strategy. Chess is an example, where one does not know which of the three possibilities happen. The game tree of chess has about 10^{120} nodes.

Moral: Many popular games have a winning strategy, but for most games, one does not know which of the three cases happen.



Problem a) (difficult) Is there a winning strategy for 4×4 chess, where all squares are occupied either by paws, rocks, queen or king?
b) (easy) See what happens if you remove all paws in the previous problem.
c) (easy) Finally look at a 4x8 board with no pawns. Who wins?

Space reconstruction



For us humans, the visualization of our surroundings appears not difficult. We can estimate the speed with which we drive for example, we can catch a ball, or recognize faces. The challenge for an artificial intelligent robot to do so is formidable. Here is an example. Cameras are located at points A, B. They photograph a point C. The angles α and β are known. It allows to compute the location of the point C.

- The DARPA Grand Challenge asks robots to race through a desert course autonomously. In order to find their paths, the bots have to spot and avoid obstacles without human interaction.
- The mars rovers "Spirit" and "Opportunity" can navigate autonomously for short distances on the planet mars. While remote controlled, there are times, when they are on their own. Stereo cameras allows the bots to read the terrain and to find paths which are safe to drive.
- When building a 3D model of a human body, the person has to be photographed from different directions. A computer program reconstructs the position of the surface.
- A bot playing ping pong has to estimate the velocity and position of the ball fast in order to calculate, where to hit it next.

Let us look at a simple problem. A stereo camera looks at an object and determines the location of the object.

Moral: Trigonometry is useful already for simple stereo reconstruction problems.

Problem a) Given the points A, B and the angles α, β, find formulas for the point C.
b) Assume we see several points $C_1, ..., C_n$ with the two cameras. How many points n do we have to see with the stereo camera so that we can reconstruct the point as well as the camera position up to a scale and translation?

Pattern matching

Pattern matching is surprisingly difficult for a machine. The human brain is still vastly superior. A machine has problems for example to drive a car, mainly because it is unable to identify and analyze the objects in the camera picture. A computer has also difficulties to recognize faces or to reliably read handwritten notes.

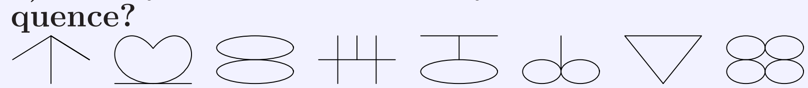
Optical character recognition: Given a scanned text, we want to regain the actual text. There are several problems which appear: we have to

- find the orientation of the text.
- separate individual lines and possibly pictures
- separate individual letters and possibly pictures
- identify the letters.

These things turn out to be harder than expected.
Face recognition: Airports would like to screen and identify passengers for security reasons, casinos like to identify gamblers from survey cameras and match them with databases.
Obstacle identification: An autonomous car needs to identify objects on the street. A plastic bag on the street does not produce any danger but a piece of metal could turn out to be deadly.
Finding corresponding points in a stereo picture: When taking a stereo picture of a town taken from an areoplane, we want to identify points which correspond to each other. This allows to gain three dimensional model of the scene.
3D canning: Also when reconstructing an object with a 3D scanner, we need to identify corresponding points when the picture is seen from different angles.
Captchas: A Captcha is a **C**ompletely **A**utomated **P**ublic Turing Test to tell Computers and Humans Appart.
Image recognition: Labeling images is an important task for search engines. Google tried to solve the image recognition by building a game, in which players are paired. They have to name pictures individually. If they match, they win a point. It is an example of a cooperative game.

Moral: Moral of this section: Pattern matching is surprisingly difficult.

Problem 1) You have scanned a ruled paper but the orientation is not known. Find a strategy to orient the paper correctly.
2) Can you find the next symbol of the following sequence?



Computer algebra

Computer algebra systems (CAS) can solve many mathematical problems with ease. Here are some simple examples what CAS can do. First of all they can solve equations

```
Solve[{x^3+2y==x,y^2-x y==x+1},{x,y}]
NSolve[x^6=x^5-2,x]
```

Manipulate algebra:

```
Expand[(2+x)^10]
1024 + 5120*x + 11520*x^2 + 15360*x^3 + 13440*x^4 + 8064*x^5 + 3360*x^6 +
 960*x^7 + 180*x^8 + 20*x^9 + x^10
```

Computing π to 1000 digits:

```
N[Pi,100]
 3.1415926535897932384626433832795028841971693993751
   058209749445923078164062862089986280348253421170 68
```

Doing numerical computations with arbitrary large numbers:

```
2^1000 =
 10715086071862673209484250490600018105614048117055336074437503883703
 51051124936122493198378815695858127594672917553146825187145285692314043 5
 98457757469857480393456777482423098542107460506237114187795418215304647 4
 98358194126739876755916554394607706291457119647768654216766042983165262 4
 386837205668069376
```

Do number theoretical operations:

```
FactorInteger[2^(2^8)+1]
1238926361552897 *
93461639715357977769163558199606896584051237541638188580280321
```

Differentiate and integrate functions.

```
In:     Integrate[Sin[Log[x]],x]
Out:    -(x*Cos[Log[x]])/2 + (x*Sin[Log[x]])/2
In:     D[Sin[Sin[Sin[x]]],x]
Out:    Cos[x] Cos[Sin[x]] Cos[Sin[Sin[x]]]
```

There are still integrals, which a computer can not do so easily but humans can. Here is an example:

```
Integrate[Sin[Sin[Sin[Sin[x]]]], {x, 0, 2 Pi}]
```

Mathematica refuses to compute it, but we know that the result is 0 because of symmetry reasons.

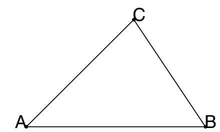
Pitfalls for computer algebra systems: many problems come from functions which are multi-valued. The square root, the logarithm (in the complex) or the inverse of trigonometric functions are examples.

```
Sqrt[(x+y)^2]==x+y
```

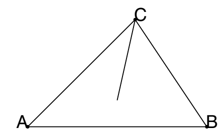
Proof verification

Here is a funny joke dealing with an error, which often can occur:
THEOREM: 1=2: Let $a = b$. Then $a^2 = ab$ and $a^2 + a^2 = a^2 + ab$ and $2a^2 = a^2 + ab$ $2a^2 - 2ab = a^2 + ab - 2ab$ and $2a^2 - 2ab = a^2 - ab$. Writing this as $2(a^2 - ab) = 1(a^2 - ab)$. Simplification gives 1=2.

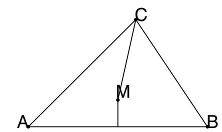
THEOREM: All triangles are isocline



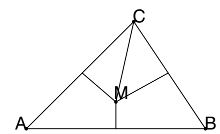
Start with an arbitrary triangle



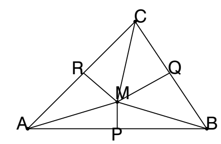
Dissect the angle at C with a line L



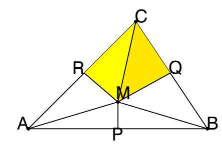
The mid normal to AB intersects L in M.



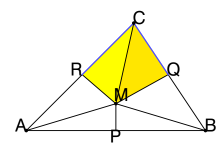
Form the normals from M to AC and BC



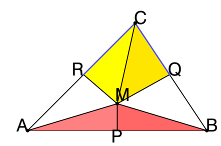
Leading to points R and Q.



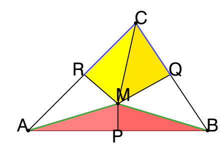
RMC and QCM are congruent (same angles and common side)



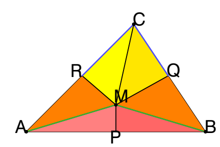
Therefore, RC = QC.



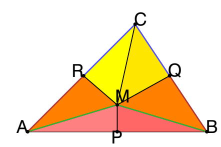
APM and BMP are congruent (right angle and two common sides)



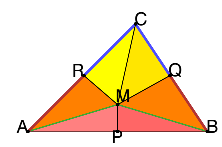
Therefore, AP=BP and AM=BM.



AMR and BQM are congruent (right angle and two common sides)



Therefore AR=BQ. This and RC=QC implies AC=BC.



Same argument for other sides shows AB=AC too.

Diagnosis

Medical doctors have to solve the problem to find a cure from symptoms. The simplest medical diagnosis is the international ABCD check for first aid, which everybody knows: after an incident with an injured or sick person, one has to proceed according to the ABCD method for **emergencies** or the PICH procedure for **sport injuries**:

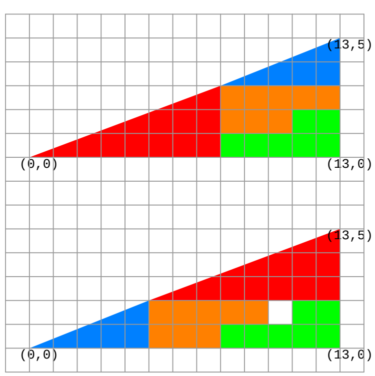
A) Airway (check the airways)	**P) Pause (don't move)**
B) Breathing (reanimate the breathing)	**I) Ice (cool with ice)**
C) Circulation (stop the bleading)	**C) Compression (with bandage)**
D) Defibrillation (check the pulse)	**H) High rest (body parts high)**

Decision trees have to be simple so that people can remember them. ABCD makes sense because one can not do B before A. C is important but B has priority. Breathing and bleading have priority over circulation problems. Things can become more complicated: when dealing with a badly injured fire fighter for example with a punctured lung, broken leg and toxicated by smoke, the doctor has to set priorities and make decisions.

Here is the core of the famous problem solving strategy by Polya of 1957: Much more detailed decision trees can be stored in databases and a computer can make a decision based on that. This is done in computer algebra systems already when solving integrals.

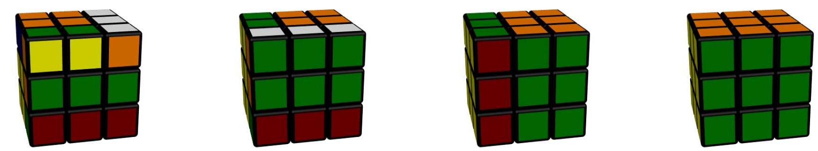
U) Understand the problem
D) Design a plan
P) Pursue the plan
L) Look back

Problem Here is a sick triangle and you are the doctor. What is wrong? Can you design a strategy which would simplify pinpointing the origin of the paradox? Does Polyas method help?



Puzzles

Many puzzles have a simple common form: an object is given in one state. The goal is to bring it to an other state. Examples of puzzles with a finite number of possible states are:



- the 15 puzzle
- the switch on-off game
- the Rubik cube

Mathematicians place all these problems in a single pot: group theory. The puzzles are actual mathematical objects which are groups and the puzzle is to find a path through that group connecting two points.

Moral: Many apparently different problems have a common underlying mathematical structure.

Problem Solve the two 2x2 on-off problems below. Which combinations of the four buttons switch all lights off?

	initial state	push 11	push 12	push 21	push 22
Problem A	1 0 / 0 0	1 1 / 1 0	1 0 / 1 1	1 0 / 1 1	0 1 / 1 1

	initial state	upper row	lower row	left column	right column
Problem B	1 0 / 0 0	1 1 / 0 0	0 0 / 1 1	1 0 / 1 0	0 1 / 0 1

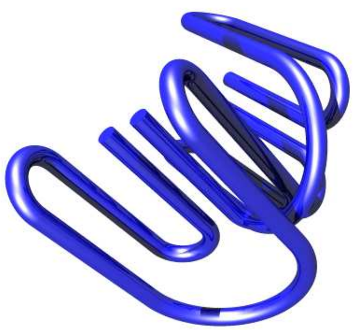
Space puzzles

Other puzzles have infinitely many states. Wire puzzles class of problems asks to separate two metal pieces. Mathematically, the pieces are given by two metallic wires. In the initial setup, the two cures are linked. The aim is to separate them. We can fix the position of one curve. The second curve can be translated and rotated.

- Linking puzzles with metallic hooks
- Knot and rope puzzles
- Wooden pieces, polyominos

These problems can also be seen as labyrinth problems. But the group is the set of possible affine transformations of all the objects. Moving the objects is a path in that group. Again, one has to find a connection between two points in the group.

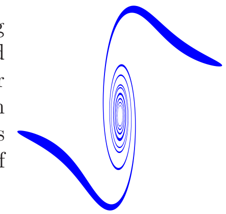
Here is a famous three dimensional "metal puzzle":



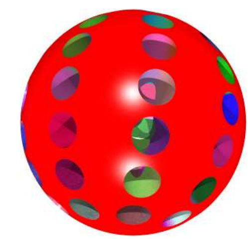
Here is a famous two dimensional "wood puzzle":

Mazes

A **2D labyrinth** is a simple closed curve in the plane dividing it into an interior and an exterior. The problem is to find connections between two arbitrary points A, B in the exterior of the curve without crossing the curve. 2D labyrinths can be solved by moving along the boundary of the region always yielding left. The solution path can not exceed the length of the curve because no point will be traversed twice.



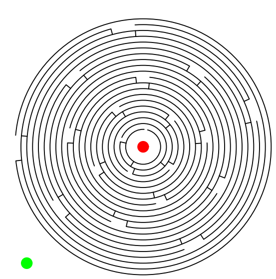
A **3D labyrinth** is given by a simple closed surface in space, dividing space into an interior and an exterior. The problem is to find a connection between two arbitrary points without crossing the surface. An autonomous bot moves inside the labyrinth and wants to get "out". Unlike in the two dimensional case, things are much harder now: there exists no universal strategy with finite computing resources which allows a bot to find an exit to any 3-dimensional labyrinth.



Remarks For any given Labyrinth, one can of course design a bot, which gets out but there is no universal strategy which works for all labyrinths in three dimensions. If a bot has access to a random number generator, there is an exit strategy: just move as a random walk bouncing off the boundary. It eventually will get out. But a random number generator can not be realized with finite computing resources. There is an other efficient strategy if the bot can have unlimited memory. The bot can measure the accelerations at all times and so always find its relative position to the initial point. The bot can explore larger and larger regions and produce a map. Eventually, the bot has mapped the labyrinth and see the exit.

Moral: Changing the dimension can make a simple problem impossible to solve.

Problem The following maze was mathematically defined by a simple rule. Try out the universal strategy to solve it.



Automatic theorem proving

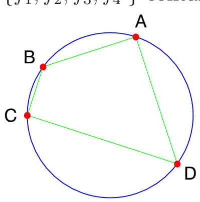
Can a computer find proofs by himself? Lets look at the Pythagorean theorem. Can we tell a computer the statement of the theorem and have the computer proof it? Some of the proofs need quite a bit of ingenuity. Let $A = (x_1, y_1), B = (x_2, y_2)$ and $C = (x_3, y_3)$ be three points in the plane so that the angle γ between AC and BC is a right angle. Show that $|AB|^2 + |BC|^2 = |AC|^2$. We can reformulate all the assumptions in terms of equations:

$$f_1 = (x_1 - x_3)(x_2 - x_3) + (y_1 - y_3)(y_2 - y_3) = 0$$
$$f_2 = a^2 - (x_2 - x_3)^2 - (y_2 - y_3)^2 = 0$$
$$f_3 = b^2 - (x_3 - x_1)^2 - (y_3 - y_1)^2 = 0$$
$$f_4 = c^2 - (x_1 - x_2)^2 - (y_1 - y_2)^2 = 0$$

The first equation tells that the angle near the point C is a right angle. The other three equations just define the lengths a, b, c (these are the definitions of the length and do not use Pythagoras!). Question: is $a^2 + b^2 - c^2 = 0$? Lets see:

```
as=(x2-x3)^2 + (y2-y3)^2; bs=(x3-x1)^2 + (y3-y1)^2;
cs=(x1-x2)^2 + (y1-y2)^2;
x1 = x3-(y1-y3) (y2-y3)/(x2-x3)
Expand[Simplify[as+bs-cs]]
```

How did the computer know, how to simplify this? There is a a mathematical branch of algebra called ideal theory which helps here. We have to know whether the ideal I generated by $F = \{f_1, f_2, f_3, f_4\}$ contains $g = a^2 + b^2 - c^2$. Here is a more advanced example from geometry:



THEOREM (Sadov): If ABCD is a quadrilateral on a circle, then $|AB|\ |BC|\ |CA| + |AC|\ |CD|\ |DA| = |BC|\ |CD|\ |DB| + |AB|\ |BD|\ |DA|$.

Proof. Here is a Mathematica version of a Maple proof by Shalosh B. Ekhad, the computer collaborator of Doron Zeilberg at Rutgers university:

```
H[A_,B_]:=(A[[1]]-B[[1]])^2+(A[[2]]-B[[2]])^2;
P[t_]:={t+1/t,(t-1/t)/I}/2;
T[A_,B_,C_,D_]:=((A+B-C-D)^2-4*(A*B+C*D))^2-64*A*B*C*D;
U[A_,B_,C_]:=H[A,B]*H[B,C]*H[C,A];
K={P[t[1]],P[t[2]],P[t[3]],P[t[4]]};
S=T[U[P[t[2]],P[t[3]],P[t[4]]],U[P[t[1]],P[t[3]],P[t[4]]],
U[P[t[1]],P[t[2]],P[t[4]]],U[P[t[1]],P[t[2]],P[t[3]]]]==0;
Simplify[S]
```

Moral: Computer algebra can tackle geometry.

Problem Write down equations which prove that for a equilateral quadrangle (rhombus), the diagonals are perpendicular.

Language

How can a computer understand natural language? There are many efforts to do so, from basic "bots" to sophisticated systems like "cyc", a large general knowledge base and common sense reasoning engine. There are many potential applications, to understand language:

- Improving search engines.
- Translating text to other languages.
- Identifying spam.
- Detecting plagiarism
- Assisting teaching
- Archiving news.

Here is a simple method which we implemented for a chat bot "Sofia", a tool, which aimed to solve basic calculus problems with the help of computer algebra systems. The idea is to build enough rules with parameters to cover as much problems as possible. Douglas Lenat, the father of cyc, once said: "Intelligence is 10 million rules".

- Simplify the sentence
- Map the sentence to a directory path
- Execute commands with parameters at the end of the path
- Collect the result

Here are examples of sentences sofia can tackle:

sofia "compute the derivative of sin(10*x)"
sofia "compute the eigenvalues of 3,2,4,5"
sofia "compute the norm of 3,2,5,2"
sofia "compute the eigenvectors of 3,2,4,5"
sofia "calculate the integral of log(tan(x))"
sofia "determine a factorization of 23425"
sofia "can you find the determinant of 1,3,4,23,2,4,12,123,1"
sofia "find gcd 26,39"
sofia "add 234 234 2 2 43 243 234"
sofia "add three and four and seven"
sofia "add million thousand hundred fifteen"
sofia "add a million and 234"
sofia "find the largest eigenvalue of 2,3,3,4"
sofia "find the row reduction of 2,3,3,4"
sofia "find the row reduced echelon form of 3,2,1,4"
sofia "find the continued fraction of Pi"
sofia "find the base form of 1234134,2"
sofia "compute 234123 in the base 2"
sofia "find the inverse of 2,3,4,2,2,1,2,3,1,2,3,4,1,2,3,12"
sofia "compute the surface area of u,v,u+v,u,0,1,v,0,1"
sofia "find the graphspectrum of 1,2,1,3,2,3"
sofia "compute the number of partitions of 10"
sofia "find a list of anagrams of the word test"
sofia "compute the curl of -y*z,x*z,0"

Final remarks

Why am I excited about AI? First of all, there is a wide overlap between mathematics and AI (very wide in algorithmic, probabilistic, and game theoretical aspects, but also in algebraic geometry harmonic analysis). **Neural nets** have connections with dynamical systems, reconstruction problems with linear algebra. There is considerable overlap between computer science and AI (logic programming, computability, search, sort and match algorithm). Practically every mathematical problem, one can also consider under the aspect of AI: how would a computer solve it? Are there patterns on how we come up with solutions or develop ideas.

Games are a typical place, where AI plays a role. Bots populate the game which need special strategies. The physics of the game itself needs mathematics, game designers have to identify players which use bots.

There are many mathematical problems still need to be explored in the area of AI. And most of the mathematics in AI is accessible for undergraduate mathematics and therefore for nonspecialists.

One major difficulty of the field is its vastness: essentially all of mathematics and all of computer science is involved additionally to the concrete situation in which the problem is posed.

To summarize, AI is a highly interdisciplinary subject. I myself am not an expert in the field but here are some places, where I had encounters with it:

- It was puzzles like the rubik cube, which contributed to get me excited about mathematics in high school. In college, I learned how a computer can find a strategy for such problems and also worked as a course assistant in an didactical implementation for the algebra language Cayley. Many puzzles are group theoretical and a computer knows, when given the group, the computer can find a strategy which navigates the group. The computer still can not find beautiful algorithms.
- In the summer of 2001, I worked on a OCR project. Starting from stratch, I wanted to see, how good moments work for distinguishing patterns. It turned out already quite a bit of work to align text, find lines in a text, separate individual letters. The pattern matching using "moments" (my motivation to start it) proved to be less efficient.
- In 2004-2005, we started Sofia, a project to have an intelligent agent answer mathematical questions in colloquial language. The project is asleep, but I still use the "robot" for looking up things or computing things.
- Having always liked photography, I got interested in geometric reconstruction methods from photographies. This happend for teaching purposes in multi-variable calculus or linear algebra.

Daily life shows that many tasks which were once considered doable by humans only, can now be achieved by machines. Do mathematicians have to fear artificial intelligence? Will it kill their job? Let me make an analogy with an activity of mountain climbing by ski, which I enjoyed in high school and college. Do ski lifts have stopped people climbing onto mountains by ski? No, it just has shifted their goals. It has become unattractive to climb up near the ski lift, but we can use the lift to go somewhere and then climb higher. Similarly, computers have made it unattractive to multiply two large numbers, to compute power series expansions for the motions of planets or asteriods or to integrate a complicated function. This is something, a computer can do much better. Finding efficient, robust and accurate algorithms has become the new and more exciting task, mathematicians can do. In the future, this will be taken over by computers too, but then Mathematicians may have moved on to enjoy thinking about strategies to make computers find strategies. Or to something completely new.

To the literature: a good introduction into AI is [11, 6, 1], about problem solving, I like [9, 13, 12, 2], about creativity [7, 8], about vision [3, 5, 4], pattern matching [10].

References

[1] Edward A. Bender. *Mathematical Methods of Artificial Intelligence*. IEEE Computer Society, 1996.

[2] Bela Bollobas. *The art of Mathematics, coffee time in memphis*. Harvard University Press, Cambridge, 1994. 9'th printing.

[3] Oliver Faugeras and Quang-Tuan Luong. *The Geometry of Multiple Images*. The MIT Press, Cambridge, Massachusetts, London, England, 2001.

[4] David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach*. Pearson, 2003.

[5] Richard Hartley and Andrew Zissermann. *Multiple View Geometry in computer Vision*. Cambridge University Press, 2003. Second edition.

[6] Philip C. Jackson. *Introduction to artificial intelligence*. Dover publications, 1985. Second, Enlarged Edition.

[7] D. Perkins. *The Eureka Effect*. WW. Nortaon, New York, London, 2000.

[8] D.N. Perkins. *The mind's Best Work*. Harvard University Press, 1981.

[9] G. Polya. *How to Solve It, a new aspect of mathematical method*. Princeton University Press, Princeton and Oxford, 1945.

[10] P.E. Hart R.O. Duda and D.G. Stork. *Pattern Classification*. John Wiley and Sons, Inc, New York, second edition edition.

[11] S. Russell and P. Norvig. *Artificial Intelligence: A modern approach*. Prentice Hall, 1995, 2003. second edition.

[12] T. Tao. *Solving mathematical Problems, A personal Perspective*. Oxford University Press, 2005.

[13] W. Wickelgren. *How to Solve Mathematical Problems*. Dover Publications, Inc, New York, 1974.