TECHNICAL UNIVERSITY OF CLUJ-NAPOCA

ACTA TECHNICA NAPOCENSIS

Series: Series: Applied Mathematics and Mechanics
Vol. 54, Issue I, 2011

ALGORITHMIC APPROACH TO BASIC THERMODYNAMIC NOTIONS

Gabriela Raluca MOCANU, Dan Anastasiu POP, George ARGHIR

Abstract: *We propose a framework that might lead to new insight in physics namely that, conceptually, the underlying mechanisms of physics phenomena are basically algorithmic in nature. Our interest is to show that physical phenomena can be viewed as a manifestation of different programming concepts. In this context, some basic thermodynamics notions are studied.*
Key words: *vacuum, thermostat, temperature, OOP (object oriented programming).*

1. INTRODUCTION

Over the years physical subtleties have been better described using mathematical notions. We propose a model that offers a different approach to physics through algorithmic and finally through OOP (object oriented programming). OOP has modeled various physical phenomena, but the physical phenomena were never treated as a particular case of some more general programming notions. As mathematics has done (for example group theory) we do not doubt that informatics will also contribute to a new perspective: various algorithms and properties already studied in the field of informatics will find their expression in the real world with a different name but showing the same behavior. The aim of this paper is to link programming conceptual entities and fundamental notions in physics and not to develop another computational algorithm to model physical phenomena.

We wish to present a proposal for how the mechanisms of interaction work at the level of the computing space, and not of how the space itself may or may not be digitized.

Physics studies events over an extended range of spatial and temporal scales. In doing so, it employs different definitions for the same concept and uses more or less the same mathematical toolkit. We will begin by defining some fundamentals of thermodynamics and then we will link them with OOP notions.

2. CONCEPTS IN THERMODYNAMICS AND THEIR OOP COUNTERPARTS

As a starting point we will use a fundamental concept of thermodynamics: the **vacuum**. Vacuum can be defined in a number of ways, each area of physics having its favorite definition. In the present paper the simple and accepted definition given by molecular physics will be used: vacuum is a volume of space that is essentially empty of matter. The “causal” notion in programming will be the function that has no parameters and does not return any values

In pseudo code language:

```
function vacuum
begin

end
```

In C:

```
void vacuum(void)
{
}
```

Two thermodynamic systems in **thermal contact** behave as two objects (both containing methods) that call for data from each other (Figure 1). We can say that the two objects are in thermal contact as long as there is a data exchange flux between them. The flux will stop when a condition is met.

The pseudocode program sequence will use two objects, OBJn (method n1.. method ni..), with i=1..n and OBJm (method m1.. method mj..) with j=1..m, n not necessarily equal to m.

```
while(condition(OBJn,OBJm)) do
    begin

    instructions

    end
```

where condition(OBJn, OBJm) is a mathematical proposition; the instructions inside the loop will tend to deny this condition in order to achieve thermal equilibrium. The instructions consist of a set of calls from OBJn to OBJm and vice-versa. These calls may or may not imply modifying parameters in the methods and functions defining the two objects. Also, it is not necessary that one object calls on all the methods of the other object. The necessary condition for thermal contact is the existence of at least one iteration.

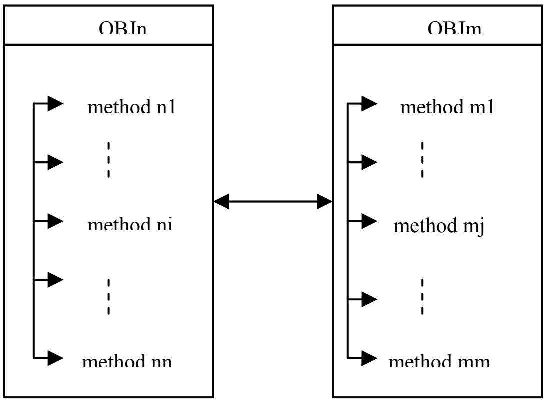

Figure 1. Thermal contact between two objects. (where the line connecting the two object stands for the data fluxand the arrowheads show direction of changing of state)

Further more, let us consider the situation when one of the objects receives/sends data without changing its state. In this case, it is a **thermostat** (Figure 2) for the object which sends/receives the data.

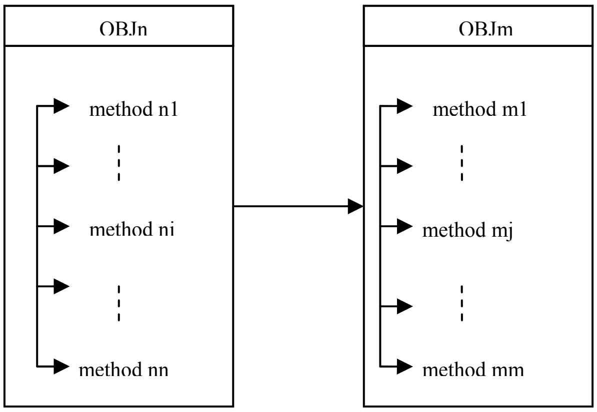

Figure 2. Thermostat (where the line connecting the two objects stands for the data flux and the arrowheads show direction of changing of state).

Generalization for more than two objects is possible. Consider OBJk (method k1.. method ki..), with i=1..k. If the programming entities from OBJk do not interact with each other but their parameters can be modified internally, it can be said that the system is at **absolute 0** (Figure 3).

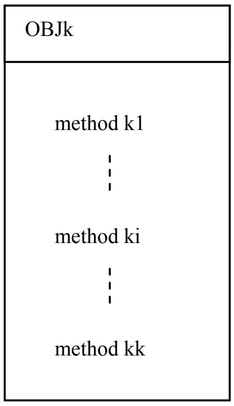

Figure 3. An object at absolute zero.

The data flow is a measure of the difference in temperature between two objects. If one of the objects is at absolute 0 (OBJk) then the data flow will be a measure of the **temperature** of the second object. As a consequence of the above discussion, each Object not at absolute 0 needs to have an exchange of data between its own methods.

In the following section we will give two examples of materializations in physics of this type of interactions taking place at the level of the “computing space”.

3. EXAMPLES

As a materialization of this interaction in physics we propose the reaction of a molecule which dissociates and forms two ions [6].

Consider the reaction of a molecule MX which, by dissociation leads to the formation of two ions, M^+ and X^-. The equilibrium constant for this reaction, K_{eq} would be, in terms of the concentrations (denoted by [])

$$K_{ech} = \frac{[M^+][X^-]}{[MX]}.$$

If we view this reaction from a kinetic point of view and define the rates of forward reaction f^+ and backward reaction f^-, it obtained that

$$K_{ech} = \frac{f^+}{f_-}.$$

We propose that the algorithm followed at the level of the "calculating space" in order to produce this observed reaction is the one given below

Reactant MX = new Reactant (…,C);
Product M+ = new Product (…,0); *// initial concentration of M+ product of reaction is 0*
Product X- = new Product (…,0); *// initial concentration of X- product of reaction is 0*

Const Kech = K; *// known*

while (M+.concentration * X-.concentration / MX.concentration ≠ K)

begin

modify concentrations such that a consistency rule is satisfied

end

This algorithm ensures that there is an information exchange (***thermal contact***) between the three objects up until equilibrium is reached.

The second example refers to how the computing spaces behaves so that the observable physical result is the process of Comptonization (the redistribution of photons in interstellar space that results from the scattering of photons on electrons). If comptonization is achieved through Inverse Compton Scattering, the seed photon spectrum (Obj1) gains energy at the expense of the kinetic energy of a distribution of relativistic electrons (Obj2) such that greater quantities of seed photons lead to lower temperatures of the plasma [8]. The interesting characteristic of the two programming object are described in the following

PhotonFlux Seed = new PhotonFlux (…, F_s)

ElectronSpectrum ES1 = new ElectronSpectrum (…, E_{kin})

where F_s is the seed photon flux impinging on the plasma and E_{kin} is the kinetic energy of the electrons in the plasma. There will be information exchange (***heat flux***) between the two objects until physical observable thermal equilibrium is reached.

4. CONCLUSIONS

In physics, digital physics is a collection of theoretical perspectives based on the premise that the universe is, at heart, describable by information, and is therefore computable. Therefore, the universe can be conceived as either the output of a computer program or as a vast, digital computation device (or, at least, mathematically isomorphic to such a device). The hypothesis that the universe is a digital computer was pioneered by Konrad Zuse in his book *Rechnender Raum* (translated into English as *Calculating Space*). The term *digital physics* was first employed by Edward Fredkin, who later came to prefer the term *digital philosophy*. Others who have modeled the universe as a giant computer include Stephen Wolfram, Juergen Schmidhuber, and Nobel laureate Gerard 't Hooft. These authors hold that the apparently probabilistic nature of quantum physics is not necessarily incompatible with the notion of computability [7].

In our view, future work will have to answer subtle and important questions such as what algorithms does the calculating space implement in the real world such that we can witness entropy, the principles of thermodynamics, mean free path, phase transitions and so on. There is another ramification to this train of thought, if one looks at it from the programmer's point of view: what are the correspondents in the physical world of algorithms already familiar to programmers and will the effort in trying to identify them lead to new advances in physics. As a possible answer to this question we will focus our future work on "generating" the world of elementary particles using OOP.

5. REFERENCES

[1] Atwood, W.B., Blankenbecler, R., Kunz, P., Burnett, T., Storr, K.M., *Monte Carlo and Detector Simulation in OOP*, SLAC-PUB-5359, 1990

[2] Budnik, P., *What is and what will b: Integrating spirituality and science*, published online, http://www.mtnmath.com/willbe.html, 2005.

[3] Lewis, J., Loftus, W., *Java software solutions*, Addison-Wesley Publishing, 4th Edition, 2004.

[4] Zuse, K., *Calculating space*, PhD Thesis, MIT, Cambridge, 1970.

[5] Yavorski, B., Detlaf, A., *Aide-memoire de physique*, MIR Publishing, 1975.

[6] Arghir, G., *Termodinamica solidului*, U.T. Press, Cluj Napoca, 2000.

[7] http://en.wikipedia.org/wiki/Digital_physics

[8] Krolik, J.H., *Active Galactic Nuclei. From the Central Black Hole to the Galactic Environment*, Princeton Series in Astrophysics, 1999.

ABORDARE ALGORITMICĂ A CONCEPTELOR TERMODINAMICE DE BAZĂ

Rezumat: *Propunem un model care ar putea duce la obținerea unor idei profunde în fizică, anume că mecanismul fundamental/primar care dictează comportamentul observabil al naturii este algoritmic. Scopul lucrării este de a demonstra că fenomenele fizice pot fi interpretate ca manifestări ale unor concepte de programare, mai exact:* ***nu dorim să modelăm*** *fenomene fizice ci să demonstrăm că fizica rezultă în urma comportamentului algoritmic al spațiului informațional. In acest context vom studia câteva noțiuni termodinamice de bază.*

Mocanu Gabriela Raluca, Drd. Fiz, Universitatea Babes-Bolyai, Cluj Napoca, Facultatea de Fizica.

Pop Dan Anastasiu, Dr. Fiz, Universitatea Tehnica din Cluj – Napoca, Facultatea de Stiinta si Ingineria Materialelor.

Arghir George, Prof. Dr. Ing. Fiz. Universitatea Tehnica din Cluj – Napoca, Facultatea de Stiinta si Ingineria Materialelor.