# Duality Between Formal Description of Program Construction and Program Behaviour

Daniel Mihályi*
Department of Computers and Informatics
Faculty of Electrical Engineering and Informatics
Technical University of Košice
Letná 9, 042 00 Košice, Slovakia
Daniel.Mihalyi@tuke.sk

## Abstract
Symmetry between the construction of programs and generating their behaviour can be properly formulated in terms of category theory by using significant property of duality. On one hand, this consists of the algebraic construction of static data structures and relations between them (programs) and on the other hand it consists of the coalgebraic generating of dynamic behavioural structures of program systems.

## Categories and Subject Descriptors
D.2 [**Software**]: Software Engineering; G.2.2 [**Mathematics of Computing**]: Discrete Mathematics—*Graph Theory*; I.6.5 [**Computing Methodologies**]: Simulation and Modeling—*Model Development*

## Keywords
Category theory, Algebra, Coalgebra, Type theory, Linear Logic, Possible World Semantics.

## 1. Introduction
The beginning of the basic ideas of computer science and informatics in the 20th century can be considered from the date of the definition of Turing machine (1937). We were waiting following ten years for the first von Neumann's computer. In this time has started the modern history of computer science. The direction of the development of informatics is based on two equivalent models of computational functions: Turing machine and $\lambda$-calculus. These two discoveries are Laurasia and Gondwana of two most known programming paradigms: imperative programming and functional programming. The significant turning point in the other paradigm, logic programming, is the formulation of linear logic that gradually replaces predicate logic because of its possibility to deal with resources and time. Hence logic programming has achieved logical space-time basis in the environment of real world and its spiritual part is formulated in the teory Ludics concerning with the spirituality of logical formulae.

Owing to discovery of categorical duality between algebras and coalgebras can be modern informatics considered as a discipline concerning with the investigation of generated behaviour also in the application research. The first aim of formal specifications is to create a formal description of a system and its components from the accessible informal requirements obtained from a potentional user of this system.

Fundamental levels of such description are characterized above all on the base of analysis of relations that appear in such systems. As models of formal descriptions are often used sets or mathematical structures and their properties are expressed by using appropriate resource-oriented logical system.

- the first level is based on set theory: model dependencies are realized without respect to any logical interconnection or category morphisms;
- the second level form mathematical structures, e.g. algebras;
- the third level form categories; on this level the relations between categories of models can be formulated and investigated.

A logical system expressing relations between specifications or theories is a part of all levels mentioned above. It is important to keep the independence of specifications and results of logical formalisms in which they are formulated. An example can be behavioural formal specification of complex program system, where external behavioural observations appear as the results of system's logical formalisms and they are encapsulated in a model that is put in the environment of real world.

## 2. Contemporary trends
In traditional informatics often has been used Wirth's [17] idea formulated in 70-ties of the 20th century that programs consist of data structures and algorithms. These principles are applied in software engineering up to the

*Recommended by thesis supervisor:
Prof. Valerie Novitzká
Defended at Faculty of Electrical Engineering and Informatics, Technical University of Košice on September 30, 2009.



Mihályi, D. Duality Between Formal Description of Program Construction and Program Behaviour. Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 2, No. 1 (2010) 1-5

present. Many-typed algebra is used as a model of algebraic specification of an abstract data type [1]. In category theory programs can be expressed as fibred categories over classified categories capturing type theories [11].

Fundamental building blocks of present modern program systems are *data*, *behaviour* and *interaction*. The symmetry between constructing of programs and generating their behaviour can be properly formulated by using category theory and its important property – duality. Using duality principle we can algebraically construct statical data structures and relations between them and on the other side – dually – coalgebraically describe generating of dynamic behaviour structures of program systems. That means, algebras are appropriate for modelling program construction, its internal structure and coalgebras are appropriate for modelling behaviour of dynamic systems. The advantages of the investigations of such duality are based on idea [15] that proved properties and facts about algebras can be properly dualized in the framework of category theory and then applied to the behavioral theory of program systems based on coalgebras. Dualities between some notions of algebras and coalgebras are in Table 1.

By the work of D.Turi and J.Rutten [16], a program can be prescribed as the initial algebra over a many-typed signature in the form of denotation model whose structure can be inductively extended to compositional interpretation of programs using initiality.

On the other side, description of the behaviour of a program system can be understood as a final coalgebra over this many-typed signature in the form of operational model whose structure can express coinductive principles of definition and proof via finality.

It appears also no less important a selection of such logical system that is available to realize a description of state oriented dynamics and to construct proofs of bisimilarity of observed states in behaviour of infinite structures. Because external displays of this behaviour are mostly appreciated by observer in the descriptive manner, it is therefore need to respect expressivity power of logical connectives of a given logical system.

The most suitable candidates of multivalue logical systems with appropriate expressive power and including also some elements of uncertainity appears such ones whose language contains modal operators. These criteria satisfy classical propositional and predicate modal logics and linear logic.

In comparision with linear logic, the main disadvantage of classical propositional and predicate modal logics is *idealised nature* of their connectives that express infinite and pleonasmic handling with formulae and causalities.

The main benefit of modal logics rests upon semantics of possible worlds, which is in suitable (although idealised) manner applicable for description of complex system semantics, where system components represent possible however pleonasmic worlds. Linear logic could introduce into these semantics the resource-oriented representation and more rigorous manipulation with these worlds. Then also the semantic of complex systems would be represented in its applicable form at the circumstances of real possible worlds situated in time and space of Ludics theory [4, 5, 2].

## 3. Main goals

The main aim of my work was to demonstrate some possible ways to use the duality principle for incorporating categorical structures of algebras and coalgebras into modern computer science for solving their actual problems. The duality principle in our approach rests on the construction of finite statical data structures in terms of algebras and on generating behaviour of infinite dynamic data structures via coalgebras. This duality is of of the fundamental and invariant properties valid in category theory and can be usefuly used also in other areas of computer science.

The next goal of this work was to map conterporary trends applying resource-oriented logical systems with modal operators and investigate their possible interconnection with internal states of coalgebras.

No less important in this work is using of graphical notation that usually in discrete mathematics but in the present literature concerning with algebras is still used rarely.

Achieved results that are presented in this thesis and in our publications, e.g. [13, 9, 10]. They express the ideas of our long-termed research within two VEGA research projects.

In the first project *VEGA No. 1/2181/05: Mathematical Theory of Programming and its Application in the Methods of Stochastic Programming* we were interested in formal description of internal structure of programs in terms of categories and linear logic. In the second project *VEGA 1/0175/08: Behavioural categorical models for complex program systems* we concern with formal description of behaviour of program systems. The relation between these two approaches we can characterize also as a duality. At the one side of the mirror is the internal view based on investigated structures of algebraic programs (terms). On the another side of the mirror we look at a program from outside, i.e. we investigate behaviour of (program) system behaviour based on coalgebraic observations.

We illustrate our research areas and results together with the duality in Figure 1.

## 4. Methods and design

Presently, signatures are the basic syntactic languuage blocks in the formulation of algebraic and coalgebraic formal specifications. a signature can be considered as a natural abstraction of an abstract data type, where signature symbols are used to axiomatization of its basic structural properties by an algebra and of their observable behaviour by a corresponding coalgebra.

The main goal of so formulated (co)algebraic specification within the program development process appears to prescribe specification of program interface by using algebraic methods and to describe behavioural specification of program systems by coalgebras. In general, a signature is considered as a pair that consists of a finite set of type specifications and a familly of operational specifications on types used by description of particular problems in the area of computer science.

**Table 1: Mirroring of algebras and coalgebras**

| Algebra | Coalgebra |
|---|---|
| static data structures | dynamic infinite structures |
| structure construction of prog. components | behaviour generating of prg. system |
| inductive prescription of steps | coinductive description of behaviour |
| equalities of equational logic | modalities of linear logic |

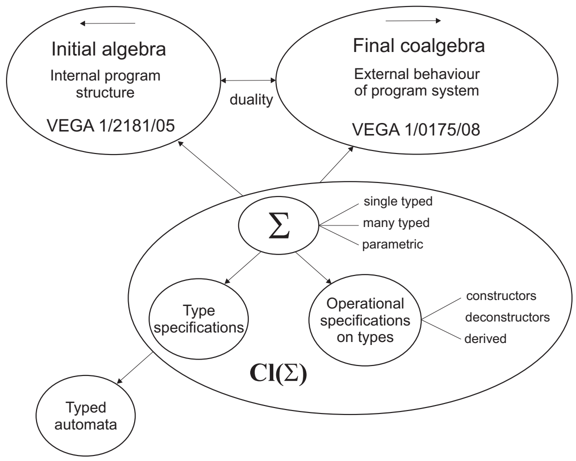


**Figure 1: duality of (co)algebras over Type theory**

Categorical declaratios types and operations are interpreted in form of tangible structures, those relations are depicted in terms of finite sets category or through more complicated generalized categorical interconnections. For example, signature for algebraic specification for stack finite structure is syntax and concrete data entity stack is semantics. Similar syntactic-semantic analogy: theories are syntactic formulation families of formulas (axioms) that are constructed from a given signature and which are interpreted by models of structures. These via fibred functors assigns to axioms their denotations.

Categorical declarations of signature types and operations are interpreted in the form of concrete structures. The relations between them an be expressed either in terms of the category $\mathbf{Set_{fin}}$ of finite sets or in more complex generalised categorical relations and interconnections. For instance, a signature for an algebraic specification of finite structure stack is syntactical entity and a concrete data structure stack is semantical entity.

Theories are similar syntactical-semantical categorical analogies, they are syntactical description of formulae, called axioms that are formulated over a signature and are interpreted by models of these structures, where morphism (functors) assign to axioms their meanings.

For purpose to investigate behavioural properties of (program) systems that are modelled by coalgebras in terms of category theory is needed to emphasize the importance of selecting appropriate logical system. Using appropriate logical system can be possible to realize description of state oriented dynamics and to construct proofs of bisimilarity of observable states via behaviour of infinite structures.

Because outside badges (symptoms) of behaviour are typically appreciates by observers in descriptive manner, it is needed to respect expressive power of logical connectives of a given logical system. Many-valued logical system semm be suitable candidates for these purposes.

**Table 2: Linear logic styles**

| positive | negative | modal |
|---|---|---|
| $\otimes$, $\oplus$, 1, 0 | &, $⅋$, $\top$, $\bot$ | !, ? |
| algebraic style | logic style | coalgebraic style |

Between many candidates wih appropriate expresive power that contain elements of uncertainty seem be suitable such that theeir language includes modal operators [3], i.e. multimodal language. On the present by this language operate many deductive and reasoning systems based on pleonasmic modal classical and predicate logical systems and their derivations. Many deduction and inference systems work over such language, mostly on the base of pleonasmic modal propositional and predicate logics and their derivates, e.g. temporal logics with linear and branching time, or modalities of deontic and epistemic logics.

For our purposes are especially suitable logical systems including resource-oriented features and modal operators, e.g. classical linear and intuitionistic linear logic. The main reason for choosing these logics is the fact, that they can explicitly manipulate with resources and this dealing is causal. In contrast to other candidates, e.g.modal propositional logic used in the imperative programming paradigm and predicate logic used in functional and logic programming paradigms, linear logic is able deal rigorously with resource restrictions (as Petri nets [6]) and with causalities in the area of application research. Different styles of linear logic are in Table 2.

These facts make from linear logic a perspective logic for resource oriented programming paradigm [8]. The large expressing algebraic, coalgebraic and logical potential can reach within the modern investigations to such areas as algebraic construction of program execution using Curry-Howard correspondence in notions of category theory over type theory [12]. Another perspective area is also investigation related to behavioural specifications and semantics of complex systems via coalgebras in terms of category theory [14], or resource oriented logical approach devoted to translation between linear logic and Petri nets [7]

## 5. Achieved results

In this section we gradually present main results achieved during our research. All these point are explained in detail in dissertation work.

- For our purposes we extended the definition of polynomial endofunctor to semipolynomial endofunctor, that enables us to consider also powersets and sequences in the frame of category **Set** of sets. This

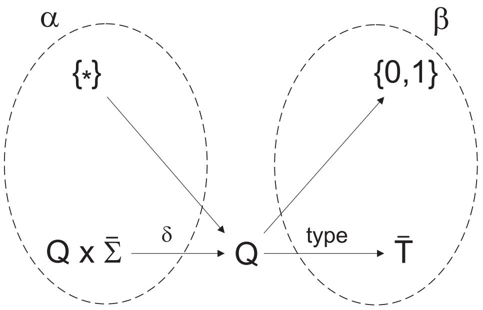


**Figure 2: Behaviour of typed sequential acceptor**

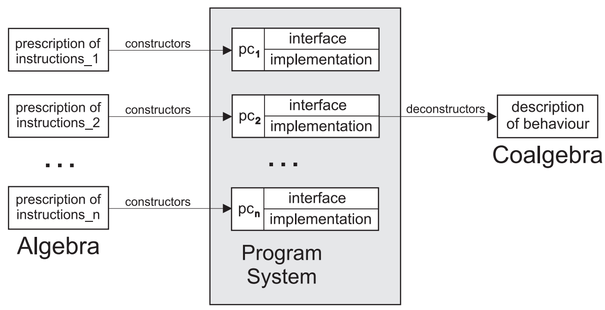


**Figure 3: (co)algebras duality at program system**

functor is induced by many-typed eventually generalised signature.

- We systematically analyzed the duality between algebraic models and coalgebraic structures based on sets, that we later generalised in the frame of category of sets.

- We constructed algebra of polynomial endofunctor as a pair consisting of carrier set and algebraic structural function. We illustrated it on well known data structure - stack.

- In the frame of category **Set** we introduced coalgebras as dual notions to algebras induced by polynomial endofunctors. A coalgebra is a pair consisting of a set called state spaces and coalgebraic structural function. This function represents set-oriented dynamics of a coalgebra. Particular coalgebraic observations we demonstrated on the structure stack mentioned above.

- We illustrated the duality between prescription internal structure represented by algebras and description of external behaviour represented by coalgebras by means of typed sequential automata, that realised one of real application of our approach. Figure 2 illustrate behaviour of typed sequential acceptor.

- We consider as the most important result of the thesis the generalization of duality between algebras and coalgebras to arbitrary bicartesian category $\mathcal{C}$. First, we extended the definition of the signature, then we formulated type theory in the frame of classifying category and finally we constructed model as a functor from classifying category to the category of types representations.

- As was mentioned above, we demonstrated the application of generalised notions in the example of Intrusion Detection System (IDS) as a real success by the solving real ptoblems of modern informatics. We can state that coalgebra for IDS treats with non trivial structures, records and packet lists.

- We demostrated that algebras resp. coalgebras in terms of categories can serve for description of structure resp. behaviour of complex program systems consisting of program components. We point out coalgebras describing generated behaviour of program systems observed outwardly in the interaction with adjacent environment as a significant observable result of execution of components in a dynamic system.

- We formulated fundamental principles of predicate lifting that enables us to interpret modal operators of chosen multimodal language and to realize its formulae over states captured by appropriate coalgebra of the semipolynomial endofunctor.

- We analyzed some possibilities related to application of modal and coalgebraic modalities and illustrated it in the example of IDS mentioned above. Esspecialy for modalities of multimodal language of linear logic, we attempted to formulate its semantics in terms of Kripke approach based on category of possible worlds.

## 6. Conclusions

During our research we have achieved several interesting results in the area of construction and behaviour of program system that open some directions of our research. We would like to investigate structures of structures using extended fusion. We would like to extend the construction of polynomial endofunctors by generalization of powersets to subobjects of toposes. Very interesting area of future research seems be the investigation of behaviour of large program systems on the base of linear logic and its semantics in terms of possible worlds. It is an invitation to introduce time and space into this semantics.

**Acknowledgements**. This work was supported by:

**VEGA No.1/2181/05:** Mathematical theory of programming and its application in the methods of stochastic programming, 2005-2007 and

**VEGA No. 1/0175/08:** Behavioural categorical models for complex programm systems, 2008-2010.

## References

[1] H. Ehrig and B. Mahr. *Fundamentals of Algebraic Specification 1: Equations and Initial Semantics*, volume 6 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, New York, Berlin, Heidelberg, New York, Tokio, 1985.

[2] C. Faggian. Travelling on designs: ludics dynamics, 2002.

[3] J.-Y. Girard. On the meaning of logical rules I: Syntax vs. semantics. In F. L. Bauer and R. Steinbrueggen, editors, *Computational Logic*, pages 215–272. Springer-Verlag, New York, New York, 1998.

[4] J.-Y. Girard. Locus solum: From the rules of logic to the logic of rules. *Mathematical Structures in Computer Science*, 11(3):301–506, 2001.

[5] J.-Y. Girard. From foundations to ludics. *Bulletin of Symbolic Logic*, 9(2):131–168, 2003.

[6] Š. Hudák. Rozšírenia petriho sietí, (habilitačná práca). EF VŠT Košice, 1980.

[7] D. Mihályi, V. Novitzká, and V. Slodičák. From petri nets to linear logic. In *CSE'2008 International Scientific Conference on Computer Science and Engineering*, pages 48–56, Stará Lesná - VysokéTatry, 2008. DCI FEI Technical University, Košice.

[8] V. Novitzká and D. Mihályi. Resource-oriented programming based on linear logic. *Acta Polytechnica Hungarica*, 4(2):157–166, 2007.

[9] V. Novitzká, D. Mihályi, R. Hužvár, V. Slodičák, and A. Verbová. Resources in linear logic proofs. In J. Kollár, editor, *Computer Science and Technology Research Survey*, page 1ï£¡8, Letná 9, 2007. Elfa.

[10] V. Novitzká, D. Mihályi, R. Hužvár, V. Slodičák, and A. Verbová. Categorical models for behavioural description of program systems. In J. Kollár, editor, *Computer Science and Technology Research Survey*, pages 7–12, Letná 9, 2008. Elfa.

[11] V. Novitzká, D. Mihályi, and V. Slodičák. Categorical logic over church's types. In *ECI'2006*, pages 122–127, Košice - Herl'any, 2006. DCI FEI Technical University, Košice.

[12] V. Novitzká, D. Mihályi, and V. Slodičák. How to combine Church's and linear types. In *ECI'2006*, pages 128–133, Košice - Herl'any, 2006. DCI FEI Technical University, Košice.

[13] V. Novitzká, D. Mihályi, and V. Slodičák. Foundations of correct programming of mathematical machines. In J. Kollár, editor, *Computer Science and Technology Research Survey*, pages 1–5, Letná 9, 2007. Elfa.

[14] V. Novitzká, D. Mihályi, and A. Verbová. Coalgebras as models of system's behaviour. In *AEI 2008, International Conference on Applied Electrical Engineering and Informatics '2008*, pages 31–36, Atény - Grécko, 2008. DCI FEI Technical University, Košice.

[15] J. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249(1):3–80, 2000.

[16] D. Turi and J. Rutten. On the foundations of final coalgebra semantics: nonwell-founded sets, partial orders, metric spaces. *Mathematical Structures in Computer Science*, 8(5):481–540, 1998.

[17] N. Wirth. *Algoritmy a ï£¡truktï£¡ry ï£¡dajov*. Alfa prekl. Prentice Hall Inc., Bratislava, 1989.

## Selected Papers by the Author

V. Novitzká and D. Mihályi. Resource-oriented programming based on linear logic. *In: Acta Polytechnika Hungarica.*, vol. 4, no. 2 (2007), p. 157-166. ISSN 1785-8860.

V. Novitzká and D. Mihályi. Polymorphic type theory as a base for categorical logic. *In: Acta Electrotechnica et Informatica.*, roč. 7, č. 3 (2007), s. 19-25. ISSN 1335-8243.

V. Novitzká and D. Mihályi and V. Slodičák. Categorical models of logical systems in the mathematical theory of programming. *In: Pure Mathematics and Applications.*, vol. 17, no. 3-4 (2006), p. 367-378. ISSN 1218-4586.

V. Novitzká and D. Mihályi and V. Slodičák. Linear logical reasoning on programming. *In: Acta Electrotechnica et Informatica.*, roč. 6, č. 3 (2006), s. 34-39. ISSN 1335-8243.

V. Novitzká and D. Mihályi and A. Verbová. Coalgebras as models of systems behaviour. In AEI '2008, *International Conference on Applied Electrical Engineering and Informatics*, pages 31–96, Greece, Athens 2008. Košice: FEI TU, 2008. ISBN 978-80-553-0066-5.

V. Novitzká and D. Mihályi and V. Slodičák. Finite automata in the mathematical theory of programming. In ICAI 2007, *Proceedings of the 7th International Conference on Applied Informatics*, pages 91–98, Eger, Hungary. Eger : BVB Nyomda és Kiadó, 2007.

V. Novitzká and D. Mihályi and V. Slodičák. From Petri nets to linear logic. In *CSE 2008: Proceedings of International Scientific Conference on Computer Science and Engineering*, pages 48–56. The High Tatras, Starï£¡ Lesnï£¡, 2008. ISBN 978-80-8086-092-9.

D. Mihályi. Behaviour of algebraic term sequences. In *7th PhD Student Conference and Scientific and Technical Competition of Students of Faculty of Electrical Engineering and Informatics Technical University of Košice*, pages 153–154. Proceedings from conference and competition. Košice, TU, 2007. ISBN 978-80-8073-803-7.

V. Slodičák and V. Novitzká and D. Mihályi. Tree automata in the mathematical theory. In *SAMI 2007: 5th Slovakian – Hungarian Joint Symposium on Applied Machine Intelligence and Informatics*, pages 447–456. Poprad, Slovakia, 2007. ISBN 978-963-7154-56-0.

D. Mihályi and V. Novitzká and V. Slodičák. Categorical type analysis for parsing algebras. In *SAMI 2007: 5th Slovakian – Hungarian Joint Symposium on Applied Machine Intelligence and Informatics*, pages 477–485. Poprad, Slovakia, 2008. ISBN 978-963-7154-56-0.

D. Mihályi. World of mathematical objects in category theory. In *6th PhD Student Conference and Scientific and Technical Competition of Students of Faculty of Electrical Engineering and Informatics Technical University of Košice*, pages 89–90. Proceedings from conference and competition. Košice, Elfa, 2006. ISBN 80-8086-035-1.

D. Mihályi and V. Novitzká and V. Slodičák. How to combine Church's and linear types. In *In: ECI 2006: Proceedings of the 7th international scientific conference 2006*, pages 477–485. Košice - Herl'any, Slovakia. 2006. ISBN 80-8073-598-0.

D. Mihályi and V. Novitzká and V. Slodičák. Categorical logic over Church's Types. In *In: ECI 2006: Proceedings of the 7th international scientific conference 2006*, pages 122–127. Košice - Herl'any, Slovakia. 2006. ISBN 80-8073-598-0.