Deduction System for TIL-2010

Marie Duží[1], Marek Menšík[1,2], Lukáš Vích[1]

[1] VŠB - Technical university Ostrava
17. listopadu 15, 708 33 Ostrava, Czech republic
[2] Silesian university in Opava,
Bezrucovo namesti 13, 746 01 Opava, Czech Republic

Abstract. The goal of this paper is to introduce a deductive system for Tichý's Transparent Intensional Logic (TIL). Tichý defined a sequent calculus for pre-1988 TIL, that is TIL based on the simple theory of types. Thus we first briefly recapitulate the rules of this simple-type calculus. Then we describe the adjustments of the calculus so that it be applicable to hyperintensions within the ramified hierarchy of types. TIL operates with a single procedural semantics for all kinds of logical-semantic context, be it extensional, intensional or hyperintensional. We show that operating in a hyperintensional context is far from being technically trivial. Yet it is feasible. To this end we introduce a substitution method that operates on hyperintensions. The syntax of TIL is the typed lambda calculus. Its semantics is based on a procedural redefinition of, inter alia, functional abstraction and application. The only two non-standard features are a hyperintension (called *Trivialization*) that presents objects, including hyperintensions, and a four-place substitution function (called *Sub*) defined over hyperintensions.

Key words: Transparent Intensional Logic, TIL, deductive system, inference

1 Foundations of TIL

From the formal point of view, TIL is a hyperintensional, partial typed λ-calculus. Thus the syntax of TIL is Church's (higher-order) typed λ-calculus, but with the all-important difference that the syntax has been assigned a procedural (as opposed to denotational) semantics, according to which a linguistic sense is an abstract *procedure* detailing how to arrive at an object of a particular logical type. TIL *constructions* are such procedures. Thus, abstraction transforms into the molecular procedure of forming a function, application into the molecular procedure of applying a function to an argument, and variables into atomic procedures for arriving at their assigned values.

There are two kinds of constructions, atomic and compound (molecular). Atomic constructions (*Variables* and *Trivializations*) do not contain any other constituent but themselves; they specify objects (of any type) on which compound constructions operate. The *variables* $x, y, p, q, \ldots,$ construct objects dependently on a valuation; they v-construct. The *Trivialisation* of an object X (of

Aleš Horák, Pavel Rychlý (Eds.): Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2012, pp. 33–41, 2012.

any type, even a construction), in symbols $^{\mathbf{0}}\boldsymbol{X}$, constructs simply X without the mediation of any other construction. *Compound* constructions, which consist of other constituents as well, are *Composition* and *Closure*. *Composition* $[\boldsymbol{F}\ \boldsymbol{A}_{\mathbf{1}}\ldots\boldsymbol{A}_{\mathbf{n}}]$ is the operation of functional application. It v-constructs the value of the function f (*valuation-*, or *v-*, –constructed by F) at a tuple – argument A (v-constructed by $A_1, \ldots, A_n$), if the function f is defined at A, otherwise the Composition is *v-improper*, i.e., it *fails* to v-construct anything.[3] *Closure* $[\lambda \boldsymbol{x}_{\mathbf{1}}\ldots\boldsymbol{x}_{\mathbf{n}}\ \boldsymbol{X}]$ spells out the instruction to v-construct a function by abstracting over the values of the variables $x_1,\ldots,x_n$ in the ordinary manner of the λ-calculi. Finally, higher-order constructions can be used twice over as constituents of composite constructions. This is achieved by a fifth construction called *Double Execution*, $^{\mathbf{2}}\mathbf{X}$, that behaves as follows: If X v-constructs a construction X', and X' v-constructs an entity Y, then ^2X v-constructs Y; otherwise ^2X is *v-improper*, failing as it does to v-construct anything.

TIL constructions, as well as the entities they construct, all receive a type. The formal ontology of TIL is bi-dimensional; one dimension is made up of constructions, the other dimension encompasses non-constructions. On the ground level of the type hierarchy, there are non-constructional entities unstructured from the algorithmic point of view belonging to a *type of order 1*. Given a so-called *epistemic (or objectual) base* of *atomic types* (o-truth values, ι-individuals, τ-time moments / real numbers, ω-possible worlds), the induction rule for forming functional types is applied: where α, $\beta_1,\ldots,\beta_n$ are types of order 1, the set of partial mappings from $\beta_1 \times\ldots\times \beta_n$ to α, denoted '$(\alpha\beta_1\ldots\beta_n)$', is a type of order 1 as well.[4] Constructions that construct entities of order 1 are *constructions of order 1*. They belong to a *type of order 2*, denoted '$*_{\mathbf{1}}$'. The type $*_1$ together with atomic types of order 1 serves as a base for the induction rule: any collection of partial mappings, type $(\alpha\beta_1\ldots\beta_n)$, *involving* $*_1$ in their domain or range is a *type of order 2*. Constructions belonging to a type $*_2$ that identify entities of order 1 or 2, and partial mappings involving such constructions, belong to a *type of order 3*. And so on *ad infinitum.*

The principle of hyperintensional individuation would slot in between Church's Alternatives (0) and (1) as Alternative (3/4), in that α-conversion and η-conversion together with a restricted principle of β-conversion determine the procedural individuation of hyperintensions we are operating with.

Laying out the required semantics requires a fair amount of footwork. Once this is in place, however, all that remains is filling in the nitty-gritty details of extensional rules such as quantifying-into hyperintensional contexts and substitution of identicals. The devil is in the detail, as ever, and defining

---

[3] We treat functions as partial mappings, i.e., set-theoretical objects, unlike the *constructions* of functions.

[4] TIL is an open-ended system. The above epistemic base $\{o, \iota, \tau, \omega\}$was chosen, because it is apt for natural-language analysis, but the choice of base depends on the area and language to be analysed. For instance, possible worlds and times are out of place in case of mathematics, and the base might consist of, e.g., o and ν, where ν is the type of natural numbers.

extensional rules of inference for hyperintensional contexts is far from being technically trivial. But it is feasible, which we are going to show in the rest of the paper. When defining extensional rules for operating in (hyper-)intensional contexts we encounter two main problems, namely the problem of *substitution* of identicals (Leibniz) and *existential generalization*. A common idea is that extensional (etc.) contexts are those that validate quantifying-in and substation of identicals. And conversely, if a context resists some of these rules, it is deemed to be in violation of one or more of the laws of extensional logic and as eluding full logical analysis. What we are saying is that also intensional and hyperintensional contexts may be quantified into, but that the feasibility of doing so presupposes that it be done within an extensional logic of hyperintensional contexts.

2 Tichý's sequent calculus

Tichý proposed a solution of the substitution and existential generalization problem in his (1982, 1986) and defined a sequent calculus for the pre-1988 TIL, that is for extensional and intensional contexts. The solution is restricted to the so-called linguistic constructions of the form $\lambda w \lambda t[C_1 C_2 \ldots C_m]$ or $\lambda w \lambda t[\lambda x_1 \ldots x_m C]$.

2.1 Substitution and existential generalization

a) *Substitution*. $a = b; C(a/x) \vdash C(b/x)$
This rule seems to be invalid in intensional contexts. For instance, the following argument is obviously invalid:

The President of ČR is the husband of Livie.
Miloš Zeman wants to be the President of ČR.
―――――――――――――――――――――――――――――――
Miloš Zeman wants to be the husband of Livie.

b) *Existential generalization*. $C(a/x) \vdash \exists x C(x)$
Again, in intensional contexts this rule seems to be invalid. For instance, the following argument is obviously invalid:

Miloš Zeman wants to be the President of ČR.
―――――――――――――――――――――――――――――――
The President of ČR exists.

Ad a) Tichý defines in (1986) *hospitality* for substitution. In principle, there are four cases. If a variable z is (1,1) hospitable, then the construction of the form $[X_{wt}]$ is substitutable for z. That is, z occurs in an extensional (*de re*) context. If a variable z is (1,0) hospitable, then the construction of the form $[X\ \boldsymbol{w}]$ is substitutable for z. That is, z occurs in an intensional (*de dicto*) context with respect to time t. If a variable z is (0,1) hospitable, then the construction of the form $[X\ \boldsymbol{t}]$ is substitutable for z. That is, z occurs in an intensional (*de dicto*) context with respect to a world w. Finally, if a variable z is (0,0) hospitable, then the construction of the form X is substitutable for z. That is, z occurs in an intensional (*de dicto*) context with respect to both t and w.

Ad b) Exposure and *existential generalization*. Let x be (1,1)-hospitable, $D(k,l)$ substitutable for x in C. Then the following rule is valid:

$$C(D(k,l)/x) \vdash \lambda w \lambda t \exists x\, C(x)$$

Example. $\lambda w \lambda t\ [Ekonom_{wt}\ PCR_{wt}] \vdash \lambda w \lambda t \exists x [Ekonom_{wt}\, x]$; ($Ekonom/(o\iota)_{\tau\omega}$; $PCR/\iota_{\tau\omega}$; $x \rightarrow_v \iota$.)

2.2 Sequent calculus

Basic notions we need are these.

Match is a pair $a : C$, where $a, C \rightarrow \alpha$ and a is an atomic construction. A match a:C is *satisfied* by a valuation v, if a and C v-construct the same object; match :C is satisfied by v, if C is v-improper; matches a:C # b:C are *incompatible,* if a, b construct different objects; matches a:C # :C are *incompatible.*

Sequent is a tuple of the form a_1:C_1, …, a_m:C_m $\rightarrow b$:D, for which we use a generic notation $\Phi \rightarrow \Psi$; A sequent $\Phi \rightarrow \Psi$ is *valid* if each valuation satisfying Φ satisfies also Ψ;

Next we define rules preserving validity of sequents.

Structural rules.

1. $\| \Phi \rightarrow \Psi$, if $\Psi \in \Phi$ (trivial sequent)
2. $\Phi \rightarrow \Psi \| \Phi_s \rightarrow \Psi$, if $\Phi \subseteq \Phi_s$ (redundant match)
3. $\Phi, \vartheta \rightarrow \Psi; \Phi \rightarrow \vartheta \| \Phi \rightarrow \Psi$ (simplification)
4. $\| \Phi \rightarrow$ y:y (trivial match)
5. $\Phi \rightarrow \vartheta 1; \Phi \rightarrow \vartheta 2 \| \Phi \rightarrow \Psi$, if $\vartheta 1$ and $\vartheta 2$ are incompatible
6. $\Phi, :\vartheta \rightarrow \Psi; \Phi, y:\vartheta \rightarrow \Psi \| \Phi \rightarrow \Psi$($y$ is not free in ...)

Application rules.

7. *a-instance* (modus ponens):

$\Phi \rightarrow y$:$[FX_1 \ldots X_m]$, Φ, f:F, x_1:X_1,…,x_m:X_m $\rightarrow \Psi \| \Phi \rightarrow \Psi$, ($f$, x_i, different variables, free in Φ, Ψ, F, X_i)

8. *a-substitution*:
   (i) $\Phi \rightarrow y$:$[FX_1 \ldots X_m]$, $\Phi \rightarrow x_1$:X_1,…,$\Phi \rightarrow x_m$:X_m $\| \Phi \rightarrow y$:$[Fx_1 \ldots x_m]$
   (ii) $\Phi \rightarrow y$:$[Fx_1 \ldots x_m]$; $\Phi \rightarrow x_1$:X_1,…, $\Phi \rightarrow x_m$:X_m $\| \Phi \rightarrow y$:$[FX_1 \ldots X_m]$
9. *extensionality*:

Φ, y:$[fx_1 \ldots x_m] \rightarrow y$:$[gx_1 \ldots x_m]$; Φ, y:$[gx_1 \ldots x_m] \rightarrow y$:$[fx_1 \ldots x_m] \| \Phi \rightarrow f$:$g$ (y, x_1,…,x_m are different variables that are not free in Φ, f, g.)

λ-rules.

10. Φ, f:$\lambda x_1 \ldots x_m Y \rightarrow \Psi \| \Phi \rightarrow \Psi$($f$ is not free in Φ, Y,Ψ)
11. β-reduction:
    $\Phi \rightarrow y$:$[[\lambda x_1 \ldots x_m Y]\ X_1 \ldots X_m] \|$
    $\Phi \rightarrow y$:$Y(X_1 \ldots X_m / x_1 \ldots x_m)$; ($X_i$ is substitutable for x_i)
12. β-expansion:
    $\Phi \rightarrow x_1$:X_1;…; $\Phi \rightarrow x_m$:X_m; $\Phi \rightarrow y$:$Y(X_1 \ldots X_m / x_1 \ldots x_m) \|$
    $\Phi \rightarrow y$:$[[\lambda x_1 \ldots x_m Y]\ X_1 \ldots X_m]$

3 Generalization for TIL 2010

Our goal is to generalize the calculus so that it involves *ramified theory of types*, all kinds of constructions, existential generalization to any contexts and substitution of identicals in any kind of context. To this end we first specify the free kinds of context.[5]

3.1 Three kinds of context

Constructions are full-fledged objects that can be not only used to construct an object (if any) but also serve themselves as input/output objects on which other constructions (of a higher-order) operate. Thus we have:

Hyperintensional context: the sort of context in which a construction is not used to v-construct an object. Instead, the construction itself is an argument of another function; the construction is just mentioned.

Example. "Charles is solving the equation $1 + x = 3$". When solving the equation, Charles wants to find out which set (here a singleton) is constructed by the Closure $\lambda x[^0{=}\ [^0{+}\ ^01\ x]\ ^03]$. Hence this Closure must occur hyperintensionally, becuase Charles is related to the Closure itself rather than its product, a particular set. Otherwise the seeker would be immediately a finder and Charle's solving would be a pointless activity. The analysis comes down to:

$$\lambda w \lambda t[^0 Solve_{wt}\ ^0 Charles\ ^0[\lambda x[^0{=}\ [^0{+}\ ^01\ x]\ ^03]]].$$

Intensional context: the sort of context in which a construction C is used to v-construct a function f but not a particular value of f; moreover, C does not occur within another hyperintensional context.

Example. "Charles wants to be The President of Finland". Charles is related to the office itself rather than to its occupier, if any. Thus the Closure $\lambda w \lambda t[^0 President\_of_{wt}\ ^0 Finland]$ must occur intensionally, because it is not used to v-construct the holder of the office (particular individual, if any). The sentence is assigned as its analysis the construction

$$\lambda w \lambda t[^0 Want\_to\_be_{wt}\ ^0 Charles\ \lambda w \lambda t[^0 President\_of_{wt}\ ^0 Finland]].$$

Extensional context: the sort of context in which a construction C of a function f is used to construct a particular value of f at a given argument, and C does not occur within another intensional or hyperintensional context.

Example. "The President of Finland is watching TV". The analysis of this sentence comes down to the Closure

$$\lambda w \lambda t[^0 Watch_{wt}\ \lambda w \lambda t[^0 President\_of_{wt}\ ^0 Finland]_{wt}\ ^0 TV].$$

The meaning of 'the President of Finland' occurs here with *de re* supposition, i.e. extensionally.

[5] For exact definitions see [5, §2.6] and also [2, Chapter 11].

3.2 Extensional calculus of hyperintensions

First we specify the rules of existential generalization and substitution rules for all kinds of context and for any constructions. In order to operate in hyperintensional context we need to introduce a four-place substitution function, $Sub/(*_n *_n *_n *_n)$, defined over hyperintensions. When applid to constructions C_1, C_2 and C_3 the function returns as its value the construction D that is the result of correctly substituting C_1 for C_2 into C_3.

Let F/($\alpha\beta$); a/α. First we specify the ***rules for existential generalisation***.[6]

a) extensional context.

Let an occurrence of the Composition $[\ldots[^0F\ ^0a]\ldots]$ be extensional and let it *v*-construct the truth-value **T**. Then the following rule is valid:

$$[\ldots[^0F\ ^0a]\ldots] \vdash \exists x[\ldots[^0Fx]\ldots]; x \rightarrow_v \alpha$$

Example. „Pope is wise." $\models$ „Somebody is wise".

$$\lambda w\lambda t[^0Wise_{wt}\ ^0Pope_{wt}] \models \lambda w\lambda t\exists x[^0Wise_{wt}\ x];$$

b) intensional context.

Let $[^0F\ ^0a]$ occur intensionally in $[\ldots\lambda y\ [\ \ldots[^0F\ ^0a]\ \ldots]]$ that *v*-constructs **T**. Then the following rule is valid:

$$[\ldots\lambda y[\ldots[^0F\ ^0a]\ldots]] \vdash \exists f[\ldots\lambda y[\ldots[f\ ^0a]\ldots]]; f \rightarrow_v (\alpha\beta)$$

Example. „*b* believes that Pope is wise". $\models$ „There is an office such that *b* believes that its holder is wise".

$$\lambda w\lambda t[^0Believe_{wt}\ ^0b\ \lambda w\lambda t[^0Wise_{wt}\ ^0Pope_{wt}]] \models$$
$$\lambda w\lambda t\exists f[^0Believe_{wt}\ ^0b\ \lambda w\lambda t[^0Wise_{wt}\ f_{wt}]];$$

c) *hyperintensional* context.

Let $[^0F\ ^0a]$ occur hyperintensionally in a construction $[\ldots{}^0[\ldots[^0F\ ^0a]\ldots]]$ that *v*-constructs **T**. Then the following rule is truth-preserving:

$$[\ldots{}^0[\ldots[^0F\ ^0a]\ldots]] \vdash \exists c\ ^2[^0Sub\ c\ ^{00}F\ ^0[\ldots{}^0[\ldots[^0F\ ^0a]\ldots]]];$$
$$c \rightarrow_v *_n;\ ^2c \rightarrow_v (\alpha\beta)$$

Example. „*b* believes* that Pope is wise." $\models$ „There is a concept of an office such that *b* believes* that the holder of the office is wise."

$$\lambda w\lambda t[^0Believe^*_{wt}\ ^0b\ ^0[\lambda w\lambda t[^0Wise_{wt}\ ^0Pope_{wt}]] \quad \models$$
$$\lambda w\lambda t\exists c[^0Believe^*_{wt}\ ^0b\ [^0Sub\ c\ ^{00}Pope\ ^0[\lambda w\lambda t[^0Wise_{wt}\ ^0Pope_{wt}]]]];$$
$(Believe^*/(o\iota *_n)_{\tau\omega}$: hyperpropositional attitude; $c \rightarrow_v *_n;\ ^2c \rightarrow_v \iota_{\tau\omega}.)$

[6] For details see [1].

Second, here are the ***rules for substitution (Leibniz).***

a) In an *extensional context* substitution of ***v-congruent*** constructions is valid.
b) In an *intensional context* (modalities, notional attitudes, ...) substitution of ***equivalent*** (but not only v-congruent) constructions is valid.
c) In a *hyperintensional context* (propositional attitudes, mathematical sentences, ...) substitution of ***procedurally isomorphic*** (but not only equivalent) constructions is valid.

Third, we must specify how to manage partial functions, that is ***compositionality*** and ***non-existence.*** If a function F has no-value at an argument a (value gap) then the Composition $[^0F\ ^0a]$ is *v-improper*, and so is any construction C occurring extensionally and containing $[^0F\ ^0a]$ as a constituent; *partiality is strictly propagated up*:

$[\ldots[\ \ldots[^0F\ ^0a]\ \ldots]\ \ldots]$ is *v-improper* until the context is raised up to *hyper/intensional* level:

intensional context : $\lambda x\ldots[\ldots[\ \ldots[^0F\ ^0a]\ \ldots]\ \ldots]$ is *v-proper*
hyperintensional context: $^0[\ldots[\ \ldots[^0F\ ^0a]\ \ldots]\ \ldots]$ is *v-proper*

The ***rules of sequent calculus*** remain as specified by Tichý with one important exception. Tichý's λ-rules involve β-reduction 'by name'. This rule is validity preserving, but we need a stronger rule that would guarantee equivalency between redex and reduct in the sense that both either v-construct the same object or both are v-improper. Moreover, β-reduction 'by name' can yield a loss of analytic information.[7]

β-reduction 'by name' in the sequent calculus:

$\Phi \rightarrow y{:}[[\lambda x_1\ldots x_m Y]\ X_1\ldots X_m]\ \|\ \Phi \rightarrow y{:}Y(X_1\ldots X_m/x_1\ldots x_m)$; ($X_i$ is substitutable for x_i)

In logic of *partial functions* the rule of transformation $[[\lambda x_1\ldots x_m Y]\ X_1\ldots X_m] \vdash Y(X_1\ldots X_m/x_1\ldots x_m)$ is not equivalent, because the left-hand side can be v-improper while the right-hand side v-proper by constructing a degenerated function that is undefined for all its arguments. To illustrate the loss of analytic information, consider two redexes $[\lambda x[^0{+}\ x\ ^01]\ ^03]$ and $[\lambda y[^0{+}\ ^03\ y]\ ^01]$. They both β-reduce to $[^0{+}\ ^03\ ^01]$. In the resulting Composition we lost the track of which function has been applied to which argument. As a solution we propose the rule of ***β-reduction by value*** that is valid and applicable in any context. Let $x_i \rightarrow_v \alpha_i$ be mutually distinct variables and let $D_i \rightarrow_v \alpha_i (1 \leq i \leq m)$ be constructions. Then the following rule is valid:

$$[[\lambda x_1\ldots x_m Y]D_1\ldots D_m] \vdash {}^2[^0Sub\ [^0Tr_{\alpha 1}D_1]^0x_1 \ldots [^0Sub\ [^0Tr_{\alpha_m}D_m]^0x_m{}^0Y]]$$

Example. "John loves his own wife. So does the Mayor of Ostrava."

$\lambda w\lambda t\ [\lambda x\ [^0Love_{wt}\ x\ [^0Wife\_of_{wt}\ x]]\ ^0\text{John}] =_{\beta v}$
$\lambda w\lambda t\ ^2[^0Sub\ ^{00}John\ ^0x\ ^0[^0Love_{wt}\ x\ [^0Wife\_of_{wt}\ x]]]$

[7] For details see [3].

$\lambda w \lambda t\ [so\_does_{wt}\ {}^0MO_{wt}] \rightarrow$
$\lambda w \lambda t\ {}^2[{}^0Sub\ {}^0[\lambda w \lambda t \lambda x [{}^0Love_{wt}\ x [{}^0Wife\_of_{wt}\ x]]]\ {}^0so\_does\ {}^0[so\_does_{wt}\ {}^0MO_{wt}]] =_{\beta v}$
$\lambda w \lambda t\ [\lambda x\ [{}^0Love_{wt}\ x\ [{}^0Wife\_of_{wt}\ x]]\ {}^0MO_{wt}] =_{\beta v}$
$\lambda w \lambda t\ {}^2[{}^0Sub\ [{}^0Tr\ {}^{00}MO_{wt}]\ {}^0x\ {}^0[{}^0Love_{wt}\ x\ [{}^0Wife\_of_{wt}\ x]]]$.

One can easily check that in all these construction whether reduced or non-reduced the track of the property of loving *one's own* wife is being kept. This property is constructed by the Closure $\lambda w \lambda t \lambda x\ [{}^0Love_{wt}\ x\ [{}^0Wife\_of_{wt}\ x]]$. When applied to John it does not turn into the property of loving John's wife. And the same property is substituted for the variable *so_does* into the second sentence. Thus we can easily infer that John and the Mayor of Ostrava share the property of loving their own wives.

4 Conclusion

We described generalization of Tichý's sequent calculus to the calculus for TIL 2010. The generalization concerns these issues. First, the extensional rules of existential generalization and substitution of identicals were generalized so that to be valid in any context, including intensional and hyperintensional ones. Second, we showed that the sequent calculus remains to be the calculus for TIL based on the ramified hierarchy of types with one important exception, which is the rule of β-reduction. We specified a generally valid rule of β-reduction 'by value' that does not yield a loss of analytic information about which function has been applied to which argument. No doubt that these are valuable results.

Yet some open problems and questions remain. Among them there are in particular the question on completeness of the calculus and the problem of its implementation.

Acknowledgements

The research reported herein was funded by Grant Agency of the Czech Republic Projects No. 401/10/0792, "Temporal Aspects of Knowledge and Information", 401/09/H007 'Logical Foundations of Semantics' and also by the internal grant agency of VSB-Technical University Ostrava, Project SP2012/26, "An utilization of artificial intelligence in knowledge mining from software processes".

References

1. Duží, M. (2012): Towards an extensional calculus of hyperintensions. *Organon F*, 19, supplementary issue 1, pp. 20–45.
2. Duží, M., Materna P. (2012): TIL jako procedurální logika. Průvodce zvídavého čtenáře Transparentní intensionální logikou.
3. Duží, M., Jespersen, B. (to apper), Procedural isomorphism, analytic information, and beta-conversion by value, forthcoming in *Logic Journal of the IGPL*, Oxford.

4. Duží, M., Číhalová, M., Ciprich, N., Frydrych, T., Menšík, M. (2009): Deductive reasoning using TIL. In RASLAN'09, Recent Advances in Slavonic Natural Langure Processing. Ed. Sojka, P., Horák, A.. Brno: Masarykova universita, pp. 25–38.
5. Duží, M., Jespersen B., Materna P. (2010): Procedural Semantics for Hyperintensional Logic; Foundations and Applications of Transparent Intensional Logic. Series Logic, Epistemology and the Unity of Science. Berlin, Heidelberg: Springer.
6. Tichý, P. (1982): Foundations of partial type theory. *Reports on Mathematical Logic*, 14: pp. 52–72. Reprinted in (Tichý 2004: pp. 467–480).
7. Tichý, P. (1986): Indiscernibility of identicals. *Studia Logica*, 45: pp. 251–273. Reprinted in (Tichý 2004: pp. 649–671).