Generalizing Rules via Algebraic Constraints
(Extended Abstract)

Alexander V. Gheorghiu David J. Pym

University College London, London WC1E 6BT, United Kingdom

Logics are often characterized by proof systems that are composed of rules. These rules give meaning to the logic — see, for example, proof-theoretic semantics [7]. We [4] propose a framework called *generalizing rules via algebraic constraints* (GRvAC), within which a rule may be decomposed into another rule together with some constraints over an algebra. The effect on the logic as a whole is more easily understood in the other direction: one enriches a logic $\mathfrak{L}$ with an algebra $\mathcal{A}$ to form a presentation of another logic $\mathfrak{L}'$. In short, we make precise the meaning of equations of the following form:

$$\text{Proof in } \mathfrak{L}' = \text{Proof in } \mathfrak{L} + \text{ Algebra of Constraints } \mathcal{A}$$

By doing reasoning in $\mathfrak{L}$ enriched by $\mathcal{A}$, one recovers reasoning in $\mathfrak{L}'$ through a transformation that is parametrized by solutions to the algebraic constraints. Consequently, $\mathfrak{L}$ is thought of as more *general* than $\mathfrak{L}'$. More precisely, one begins by labling the syntax of $\mathfrak{L}$ by (a syntax for) $\mathcal{A}$. Assignments I of the variables of $\mathcal{A}$ then determine valuations ν_I mapping the syntax of $\mathfrak{L}$ enriched by $\mathcal{A}$ to the syntax of $\mathfrak{L}'$. A rule of $\mathfrak{L}'$. is *generalized* when a rule of $\mathfrak{L}$ (taken over the enriched language together) with constraints (i.e., equations) over $\mathcal{A}$ is used to expresses it.

The GRvAC phenomenon is essentially a generalization of the *resource-distribution via boolean constraints* (RDvBC) mechanism introduced by Harland and Pym [5] for the study of proof-search in the presence of multiplicative (or intensional) connectives. Indeed, RDvBC is an important example of GRvAC. We may use it in the case of linear logic (LL) to illustrate the ideas given abstractly so far: one labels the formulas of LL with a syntax for boolean algebra $\mathcal{B}$ (e.g., one has formulas such as $\phi \cdot x \otimes \psi \cdot \bar{x}$) such that assignments I determine valuations ν_I that keep formulas labelled by variables that I map to 1 and delete formulas labelled by variables that I map to 0 (e.g., if $I(x) = 0$, then $\nu_I(\phi \cdot x \otimes \psi \cdot \bar{x}) = \phi$). This setup allows multiplicative conjunction rules to be *generalized* to additive conjunction rules; for example,

$$\frac{\Gamma \vdash \phi \quad \Delta \vdash \psi}{\Gamma, \Delta \vdash \phi \otimes \psi} \quad \text{generalizes to} \quad \frac{\Gamma \cdot x, \Delta \cdot \bar{x} \vdash \phi \quad \Gamma \cdot x, \Delta \cdot \bar{x} \vdash \psi}{\Gamma, \Delta \vdash \phi \otimes \psi}$$

In this way, RDvBV allows one to witness the following equation in which LL is a proof system for LL and LK is a proof system for classical logic: $\mathsf{LL} = \mathsf{LK} + \mathcal{B}$

Other examples of GRvAC are present in the literature too. For example, the authors [3] have studied the application of algebraic constraints to unification during logic programming, which can be regarded as witnessing that propositional logic is more general than predicate logic. Moreover, though generalization allows one to relate two logics, the idea of algebraic constraints is useful in itself and present elsewhere in the literature — see, for example, work by Negri [6] on relational calculi. Indeed, the concept of enrichment here is strongly related to the framework of *Labelled Deductive Systems* introduced by Gabbay [2].

The GRvAC phenomenon is useful for the theory of logic. It is a technology that allows one to express formally relationships *between* logics; for example, it supports the folkore that classical logic (CL) is a combinatorial core of logics (i.e., that CL generalizes most logics). It also allows one to study the metatheory of particular logics. Moreover, by understand how a logic arises from CL by means of an algebra, GRvAC allows one to derive model-theoretic semantics for the logic. Inversely, it allows one to generate sound and complete proof systems for a logic from its model theory. These semantics uses of GRvAC are prefigured by Docherty [1].

An example of the effectiveness of the GRvAC principle for metatheory is captured by a case study on intuitionistic logic (IL). Here, GRvAC allow one to construct from a single-conclusioned calculus a multiple-conclusioned sequent calculus, which witnesses that CL is the combinatorial core of IL. By studying the new calculus' relationship to CL using GRvAC, one can *derive* a model-theoretic semantics of IL. The derivations provides a new technique for proving soundness and completeness that proceeds by showing the equivalence of proof-search of the two logics relative to the constraints captured by the algebra.

There are also practical uses for GRvAC, particularly for proof-search. For example, RD-vBC was introduced as a solution to the context-management problem during proof-search in substructural logics. In general, GRvAC allows one to separate the combinatorial aspects of a logic from the internal choices made during search (e.g., the use of a quantifier rule vs the choice of a substitution); that is, the combinatorial aspects of proof-search in $\mathfrak{L}'$ can be understood by proof-search in $\mathfrak{L}$ with the specifics of original the logic recovered by constraints over $\mathcal{A}$. Among other things, this allows one to capture certain amount of global reasoning for proof-search, which can be interpreted as capturing a certain amount of backtracking within a proof system.

The GRvAC framework allows one to express complex rules as simple rules together with algebraic constraints that recover the former from the latter by means of transformations parameterized by solutions to equations over the algebra. It is useful for both intra-logic metatheory (i.e., proof theory and semantics), inter-logic metatheory (i.e., connexions between logics) and in applied logic tasks involving proof-search. Though we have outlined it conceptionally, substantial work remains in developing the space of examples and using it to develop uniform approaches to metatheory. Moreover, on the question of proof-search, GRvAC may be used to give a general mathematical theory of control, which is currently lacking, and relate the control problems of proof-search to other aspects of the logic (e.g., the clauses of its semantics).

References

[1] Docherty, S., and Pym, D. J. Modular Tableaux Calculi for Separation Theories. In *Foundations of Software Science and Computation Structures — FOSSACS* (2018), C. Baier and U. Dal Lago, Eds., pp. 441–458.

[2] Gabbay, D. M. *Labelled Deductive Systems.* Oxford University Press, 1996.

[3] Gheorghiu, A. V., Docherty, S., and Pym, D. J. Reductive Logic, Coalgebra, and Proof-search: A Perspective from Resource Semantics. In *Samson Abramsky on Logic and Structure in Computer Science and Beyond*, A. Palmigiano and M. Sadrzadeh, Eds., Springer Outstanding Contributions to Logic Series. Springer, 2021. to appear

[4] Gheorghiu, A. V., and Pym, D. J. Generalizing Rules via Algebraic Constraints. `http://www0.cs.ucl.ac.uk/staff/D.Pym/grvac.pdf`, 2022. submitted.

[5] Harland, J., and Pym, D. J. Resource-distribution via Boolean Constraints. *ACM Transactions on Computational Logic 4* (2003), 56–90.

[6] Negri, S. Proof Analysis in Modal Logic. *Journal of Philosophical Logic 34*, 5 (2005), 507–544.

[7] Schroeder-Heister, P. Proof-Theoretic Semantics. In *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2018.