change cofinalities or cardinalities $\leq \kappa$. The possible cofinalities for μ in the $\mathbb{P}$-generic extension depend on the Mitchell order of μ. Inner model theory tells us that these assumptions are optimal in terms of consistency strength assuming $\kappa \geq \aleph_2$. (Note: this is joint work with Peter Koepke.)

We end with a short discussion on the connection between forcings that change cofinalities of several cardinals simultaneously and mutually stationary sequences of sets.

Abstract prepared by Dominik Thomas Adolf
URL: `http://www.uni-muenster.de/imperia/md/content/logik/dissadolf.pdf`

CAROLIN ANTOS, ***Foundations of Higher-Order Forcing***, University of Vienna, 2015. Supervised by Sy-David Friedman. MSC: 03Exx, 03E40, 03E70. Keywords: forcing, class forcing, class theory.

Abstract

Forcing notions can be classified via their size in a general way. Until now two different types were developed: set forcing and definable class forcing, where the forcing notion is a set or definable class, respectively. Here, we want to introduce and study the next two steps in this classification by size, namely class forcing and definable hyperclass forcing (where the conditions of the forcing notion are themselves classes) in the context of (an extension of) Morse–Kelley class theory. For class forcing, we adapt the existing account of class forcing over a ZFC model to a model $\langle M, \mathcal{C}\rangle$ of Morse–Kelley class theory. We give a rigorous definition of class forcing in such a model and show that the Definability Lemma (and the Truth Lemma) can be proven without restricting the notion of forcing. Furthermore we show under which conditions the axioms are preserved. We conclude by proving that Laver's Theorem does not hold for class forcings. For definable hyperclass forcing, we use a symmetry between MK^{**} models and models of ZFC^{-} plus there exists a strongly inaccessible cardinal (called SetMK^{**}). This allows us to define hyperclass forcing in MK^{**} by going to the related SetMK^{**} model and use a definable class forcing there. We arrive at a definable class forcing extension from which we can go back to a model of MK^{**}. To use this construction we define a coding between MK^{**} and SetMK^{**} models and show how definable class forcing can be applied in the context of an ZFC^{-} model. We conclude by giving an application of this forcing in showing that every β-model of MK^{**} can be extended to a minimal β-model of MK^{**} with the same ordinals.

Abstract prepared by Carolin Antos
E-mail: `carolin.antos-kuby@uni-konstanz.de`

ANUSH TSERUNYAN, ***Finite Generators for Countable Group Actions*; *Finite Index Pairs of Equivalence Relations*; *Complexity Measures for Recursive Programs***, University of California at Los Angeles, 2013. Supervised by Alexander S. Kechris and Itay Neeman. MSC: Primary 03E15, 37B10, 03D15, Secondary 37A35, 37A20. Keywords: Borel group actions, generating partitions, entropy, countable Borel equivalence relations, treeable-by-finite, finite index, recursive programs, complexity measures.

Abstract

Part 1: For a continuous action $\Gamma \curvearrowright X$ of a countable group Γ on a Polish space X, a finite Borel partition (coloring) $\mathcal{P} = \{C_i\}_{i<n}$ of X induces the so-called *coding map* from X to the shift n^Γ by sending each $x \in X$ to the sequence of colors that x encounters when moved by the group elements, i.e., $x \mapsto (i_\gamma)_{\gamma\in\Gamma}$, where $\gamma \cdot x \in C_{i_\gamma}$. $\mathcal{P}$ is called a *generator* if its coding map is injective. A finite generator exists if and only if the action is embeddable into a finite shift action.

For $\Gamma := \mathbb{Z}$ or any other amenable group, the existence of a finite generator is precluded by the existence of an invariant Borel probability measure of infinite entropy. It was asked by B. Weiss in the late 80s if the actions that do not possess any invariant Borel probability measure must admit a finite generator. For σ-compact (e.g., locally compact) actions, I give