## EQUIVARIANT COHOMOLOGY FOR TORUS ACTION

## Leyla YARDIMCI Middle East Technical University Ankara, Turkey

In this talk, I will describe the equivariant cohomology for some special space with  $T^n$  torus action that has finitely many fixed points and 1-dimensional orbits. Equivariant cohomology is a functor from category of topological spaces and equivariant continous maps to the category of graded rings and their homomorphisms and this functor reflects both the topology of the space and the action of the group. In favorable situations under the torus action equivariant cohomology in a combinatoric picture. That is, for certain spaces we can describe equivariant cohomology in a combinatorial way. Therefore we need not to use chain complexes that are used in ordinary cohomology anymore. Instead of chain complexes we use fixed points and 1-dimensional orbits to construct a graph, called the moment graph, and use this graph to compute the equivariant cohomology ring.

## References

- [1] Allen Hatcher 2006. Algebraic Topology, Cambridge University Press, http://www.math.cornell.edu/ hatcher/AT/
- [2] Allen Hatcher 2003. Vector Bundles and K-Theory (2.0 ed.)
- [3] John M.Lee 2003. Introduction to Smooth Manifolds, *New York: Springer, ISBN 0-387-95448-1*
- [4] Julianna S. Tymoczko 2005. An introduction to equivariant cohomology and homology, following Goresky, Kottwitz, and MacPherson. In Snowbird lectures in algebraic geometry, volume 388 of Contemp.Math., pages 169-188. Amer. Math. Soc., Providence, RI.
- [5] J. J. Duistermaat and J. A. Kolk Lie Groups, Universitext, Springer
- [6] Loring W.Tu 2011. What Is... Equivariant Cohomology? (PDF). AMS Notices 58 (03): 423?426.
- [7] M.Goresky, R.Kottwitz, R.MacPherson 1998. Equivariant Cohomology, Koszul Duality, and The Localization Theorem, *Invent. Math.* 131 (1998), 25?83.
- [8] Richard Gonzales Localization in Equivariant Cohomology and GKM Theory (PDF)
- [9] V.W.Guillemin and S.Sternberg 1999 Supersymmetry and equivariant de Rham theory. *Springer-Verlag, Berlin.*