

A DEGENERATION OF THE MODULI SPACE OF STABLE BUNDLES

D. GIESEKER

1. Let k be an algebraically closed field and let d be an odd integer. Let $g \geq 2$ be an integer and suppose Y is a smooth projective curve of genus g . Let U_Y be the set of isomorphism classes of stable bundles E of rank two and degree d . (Note: we do not fix $\wedge^2 E$.) Following Mumford and Seshadri, we know that U_Y is in a natural way the set of (closed) points of a smooth projective variety again denoted by U_Y . Our aim in this paper is to develop a method of studying the topology of U_Y by degeneration methods. Our main application is the proof of the following theorem conjectured by Newstead and Ramanan.

Theorem 1.1 ($k = \mathbb{C}$). *The k th Chern class of the tangent bundle of U_Y is zero in the deRham cohomology of U_Y if $k > 2g - 2$.*

We hope that degeneration methods may be useful in other contexts. For instance, one can hope that the theory can be generalized to bundles of arbitrary degree and rank. One should also be able to compute the lower Chern classes of $\Omega_{U_Y}^1$.

The following is a brief outline of this paper: Let X_0 be an irreducible curve of genus g which is smooth except of one ordinary node N . We let X be the normalization of X_0 and let P_1 and P_2 be the inverse image of N . Our object is to find a (singular) projective variety U_{X_0} which will play the role to U_Y when Y is smooth. In particular, if $\{Y_i\}$ is a family of smooth curves degenerating to X_0 , then we desire that U_{Y_i} generates to U_{X_0} .

The first difficulty in constructing U_{X_0} is that one cannot hope that all the points of U_{X_0} will correspond to actual bundles on X_0 . There are two methods to resolve this difficulty. One is to consider certain torsion-free sheaves on X_0 to obtain a candidate for U_{X_0} [3]. However, such a U_{X_0} does not appear to have (analytic) normal crossings. The second method, which we will follow, is to consider certain bundles on certain semistable models of X_0 as is suggested by