COMPUTATION FOR DIFFEOMORPHISM GROUPS OF CIRCLE BUNDLES OVER INTEGRAL KÄHLER AND SYMPLECTIC MANIFOLDS

SATOSHI EGI, YOSHIAKI MAEDA, AND STEVEN ROSENBERG

1. Introduction

This article presents a program for computing Wodzicki-Chern-Simons form of certain Kähler manifold with the Egison programming language.

2. Computation

In [1]:
```
(define $x~ [| θ₁ θ₂ θ₃ θ₄ |])
```

In [2]:
```
(define $g__
  [|[| 1 0 0 0 |]
    [| 0 1 0 0 |]
    [| 0 0 (/ ϰ (sqrt β)) (/ (* -1 θ₂ ϰ) (sqrt β))  |]
    [| 0 0 (/ (* -1 θ₂ ϰ) (sqrt β)) (/ (* '(+ 1 θ₂) ϰ) (sqrt β)) |]|])
```

In [3]:
```
(define $g~~
  [|[| 1 0 0 0 |]
    [| 0 1 0 0 |]
    [| 0 0 (/ '(+ 1 θ₂) (* ϰ (sqrt β))) (/ θ₂ (* (sqrt β) ϰ)) |]
    [| 0 0 (/ θ₂ (* (sqrt β) ϰ)) (/ 1 (* (sqrt β) ϰ)) |]|])
```

In [4]:
```
(define $β '(+ 1 θ₂ (* -1 (** θ₂ 2))))
```

In [5]:
```
(define $Γ~c_a_b
  (. (/ 1 2)
     g~c~e
     (+ (∂/∂ g_b_e x~a)
        (∂/∂ g_a_e x~b)
        (* -1 (∂/∂ g_a_b x~e)))))
```

In [6]:
```
(define $R_i_j_k~l
  (with-symbols {a}
    (+ (- (∂/∂ Γ~l_j_k x~i) (∂/∂ Γ~l_i_k x~j))
       (- (. Γ~l_i_a Γ~a_j_k) (. Γ~l_j_a Γ~a_i_k)))))
```

In [7]:
```
(define $R_i_j_k_l
  (with-symbols {a}
    (. R_i_j_k~a g_a_l)))
```

```
In [8]: (define $J__
          [|[| 0 1 0 0 |]
            [| -1 0 0 0 |]
            [| 0 0 0 ϰ |]
            [| 0 0 (* -1 ϰ) 0 |]|])
```

```
In [9]: (define $J_a~c (. J_a_b g~b~c))
```

```
In [10]: (define $∇J_m_a_b
           (with-symbols {n}
             (- (∂/∂ J_a_b x~m)
                (. Γ~n_m_a J_n_b)
                (. Γ~n_m_b J_a_n))))
```

```
In [11]: (define $∇J~m_a_b
           (with-symbols {t}
             (. ∇J_t_a_b g~t~m)))
```

```
In [12]: (define $∇J_m~a_b
           (with-symbols {t}
             (. ∇J_m_t_b g~t~a)))
```

```
In [13]: (define $∇J_m_a~b
           (with-symbols {t}
             (. ∇J_m_a_t g~t~b)))
```

```
In [14]: (define $δ
           (generate-tensor
             (match-lambda [integer integer]
               {[[$n ,n] 1]
                [[_ _] 0]})
             {5 5}))
```

```
In [15]: (define $R'___~
           (generate-tensor
             (match-lambda [integer integer integer integer]
               {
                [[,1 ,1 _ _] 0]
                [[_ _ ,1 ,1] 0]
                [[,1 $b ,1 $d] (* -1 p^2 δ~(- b 1)_(- d 1))]
                [[$a ,1 ,1 $d] (*    p^2 δ~(- a 1)_(- d 1))]
                [[,1 $b $c ,1] (*    p^2 g_(- b 1)_(- c 1))]
                [[$a ,1 $c ,1] (* -1 p^2 g_(- a 1)_(- c 1))]
                [[,1 $b $c $d] (* -1 p ∇J_(- b 1)_(- c 1)~(- d 1))]
                [[$a ,1 $c $d] (*    p ∇J_(- a 1)_(- c 1)~(- d 1))]
                [[$a $b ,1 $d] (* -1 p ∇J~(- d 1)_(- a 1)_(- b 1))]
                [[$a $b $c ,1] (* p ∇J_(- c 1)_(- a 1)_(- b 1))]
                [[$a $b $c $d] (+ R_(- a 1)_(- b 1)_(- c 1)~(- d 1)
                                  (* -1 p^2 J_(- b 1)_(- c 1) J_(- a 1)~(- d 1))
                                  (*    p^2 J_(- a 1)_(- c 1) J_(- b 1)~(- d 1))
                                  (* 2  p^2 J_(- a 1)_(- b 1) J_(- c 1)~(- d 1)))]})
             {5 5 5 5}))
```

```
In [16]: (define $S
           (with-symbols {i j k}
             (let {[[$es $os] (even-and-odd-permutations 5)]}
               (- (sum (map (lambda [$σ] (. R'_(σ 1)_j_1~i R'_(σ 2)_(σ 3)_k~j R'_(σ 4)_(σ 5)_i~k)) es))
                  (sum (map (lambda [$σ] (. R'_(σ 1)_j_1~i R'_(σ 2)_(σ 3)_k~j R'_(σ 4)_(σ 5)_i~k)) os)))
```

Rakuten Institute of Technology
Email address: egi@egison.org

Tohoku Center for Creativity, Tohoku University
Email address: yoshimaeda@m.tohoku.ac.jp

Department of Mathematics and Statistics, Boston University
Email address: sr@math.bu.edu