The State of Arithmetic and Complex Dynamics in Sage

Benjamin Hutz

Department of Mathematical Sciences
Florida Institute of Technology

November 7, 2013
Sage-Days 55

Directory Structure

1. sage/schemes/generic
2. sage/schemes/projective
3. sage/schemes/affine

Creating Projective and Affine Spaces

1. Rings
    1. integer
    2. p-adic
    3. polynomial
2. Fields
3. Finite Fields

Affine and Projective Spaces

1. ==, ! =, _init_(), _copy_()
2. dimension
3. coordinate_ring()
4. change_ring()
5. normalize_coordinates()
6. base_ring
7. subscheme

Points

1. ==, ! =, _init_(), _copy_()
2. scale_by()
3. clear_denominators()
4. change_ring()
5. dehomogenize()
6. normalize_coordinates()
7. nth_iterate(), orbit()

inherited

1. codomain()
2. base_ring
3. __get__

Morphisms

1. normalize_coordinates()
2. ==, ! =, _init_(), _copy_()
3. scale_by()
4. dehomogenize()
5. degree()
6. nth_iterate_map()
7. dynatomic_polynomial([m,n])
8. resultant() - only dimension 1
9. is_morphism()
10. primes_of_bad_reduction()
11. conjugate()

inherited

1. domain, codomain
2. base_ring()
3. defining_polynomials(), __get__

Finite Fields

1. cyclegraph()
2. orbit_structure()
3. hash()

Heights: 14218 (5.13.beta2)

Points and Morphisms

1. greens_function()
2. height()
3. canonical_height()

Rational Preperiodic points: 14219 (needs review)

1. height_difference_bound()
2. multiplier()
3. possible_periods()
4. rational_preimages()
5. lift_to_rational_periodic()
6. rational_(pre)periodic_points() (or graph)

Reviews

1. 14219 - rational preperiodic points
2. products of projective space
3. Wehler K3

To Do: Minor Changes

1. add switch to dynatomic polynomial to remove all multiple roots at each step ($gcd(f, f')$).
2. global_height for ZZ.
3. tutorials
4. is_periodic(), is_preperiodic(), cyclestrucure() for rational points
5. primes of bad reduction, is_morphism - add defining equations of subscheme to ideal to make these work over subschemes
6. _validate() in projetive_space does not check that the polynomials are in the coordinate ring. (neither does affine_space)

```
1 R.<t,s,w>=PolynomialRing(GF(5),3)
2 P.<x,y>=ProjectiveSpace(QQ,1)
3 P._validate([t-s])
```

To Do: More Involved

1. What finite field functionality can also work in Zmod(n) for composite n.
2. Lazy imports wherever possible.

To Do: Algorithm implementation

1. FMV algorithm (automorphisms groups)
2. Krumm-Doyle Algorithm (points of small height for number fields)
3. Bruin-Molnar algorithm - minimal models
4. Macualay resultant `http://minimair.org/mr/`

To Do: Algorithm implementation

1. FMV algorithm (automorphisms groups)
2. Krumm-Doyle Algorithm (points of small height for number fields)
3. Bruin-Molnar algorithm - minimal models
4. Macualay resultant `http://minimair.org/mr/`

More amorphous tasks

1. What to do with critical points and PCF maps?
2. *p*-adic dynamics