

Recurrent Cartesian Genetic Programming

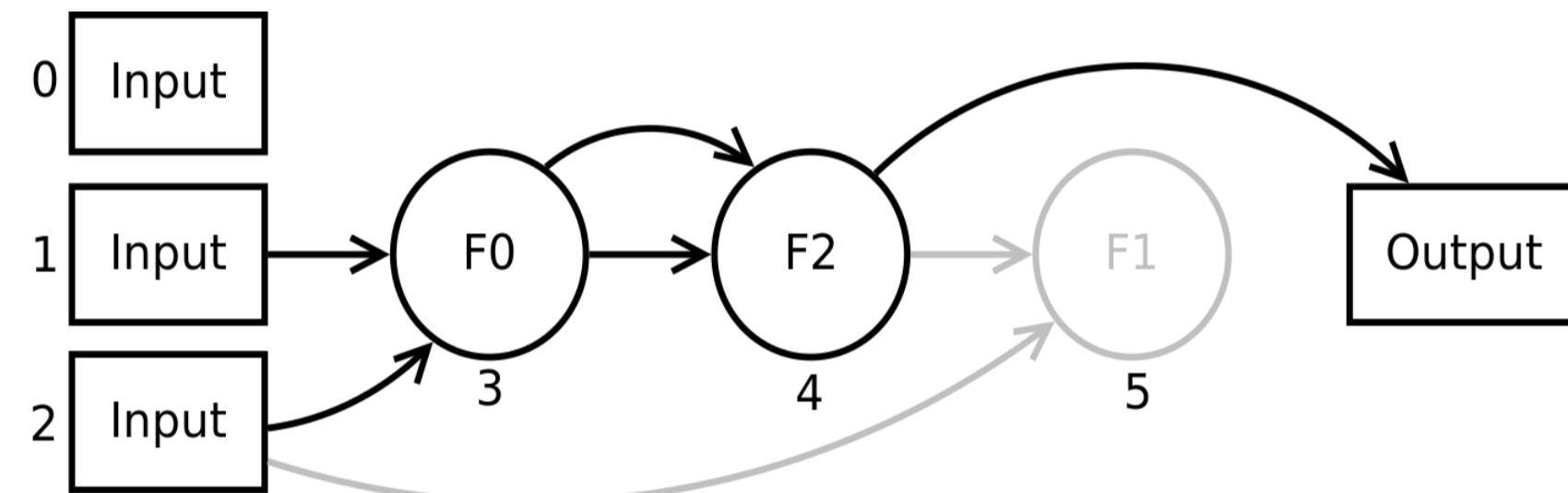
Andrew James Turner & Julian Francis Miller

andrew.turner, julian.miller@york.ac.uk



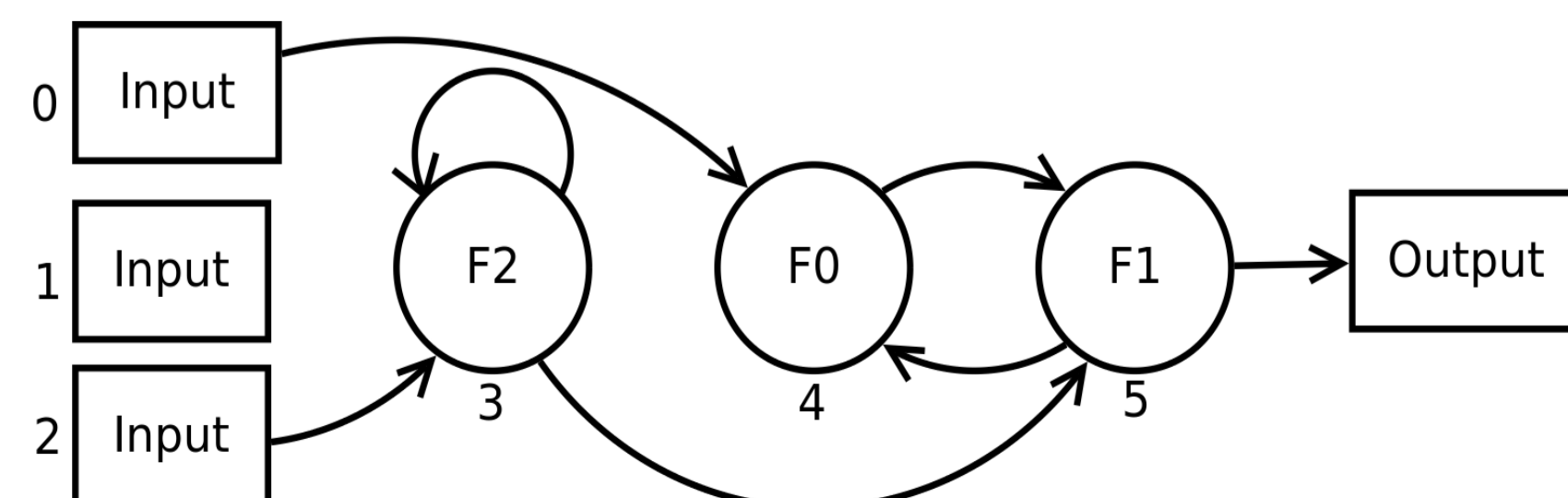
Cartesian Genetic Programming (CGP) [1]

- Very general form of Genetic Programming
- Evolves acyclic computational graphs (not just trees)
- Does not suffer from bloat [2]
- Utilises genetic redundancy via neutral genetic drift [3]
- Allows node output values to be reused (unlike GP)



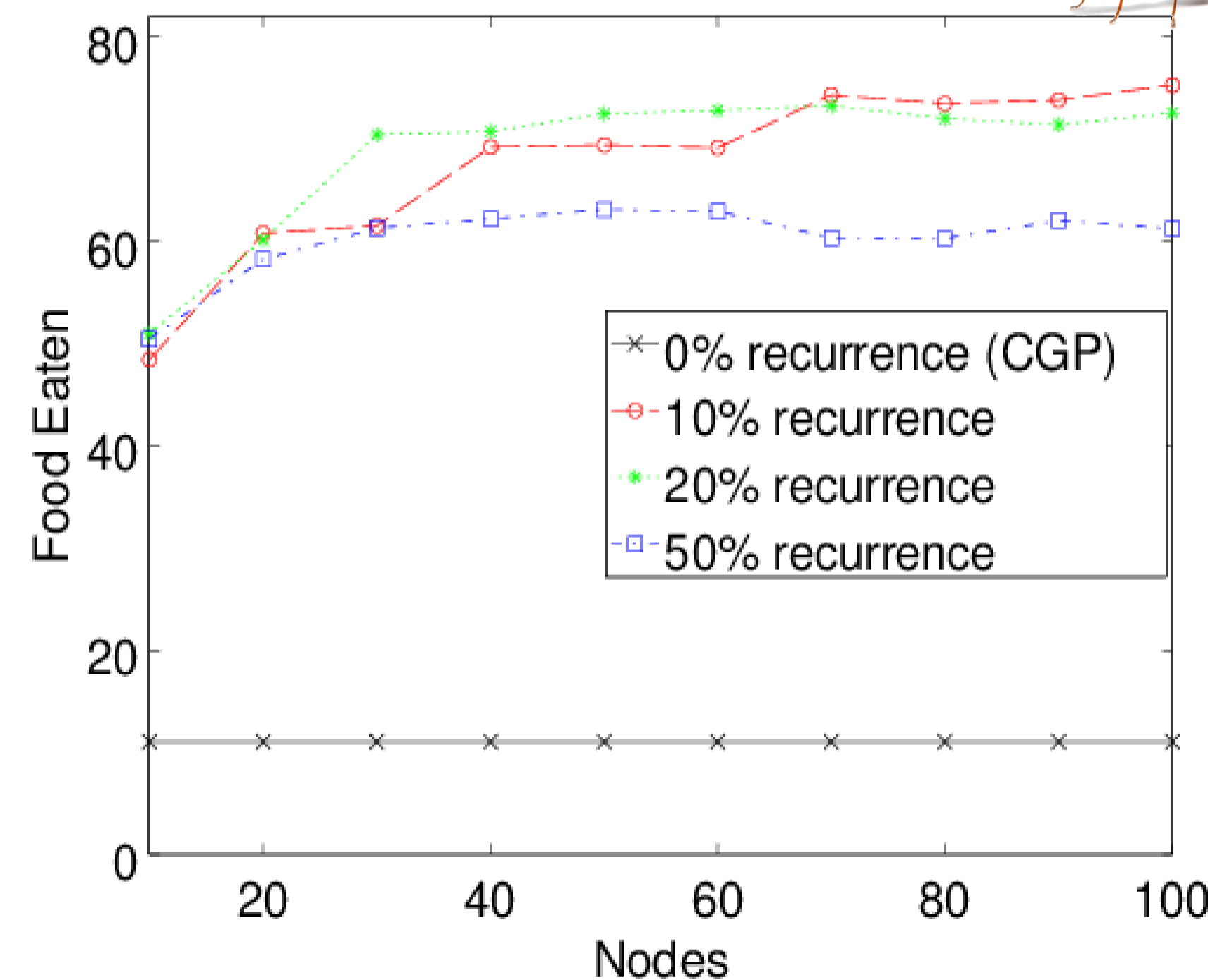
Recurrent Cartesian Genetic Programming (RCGP)

- Superset of CGP, can create acyclic and cyclic graphs
- Allows application to partially observable tasks
- Can hold internal state information (memory)
- Can produce recurrent as well as explicit equations
- Biased via a recurrent connection probability
- All the benefits of standard CGP



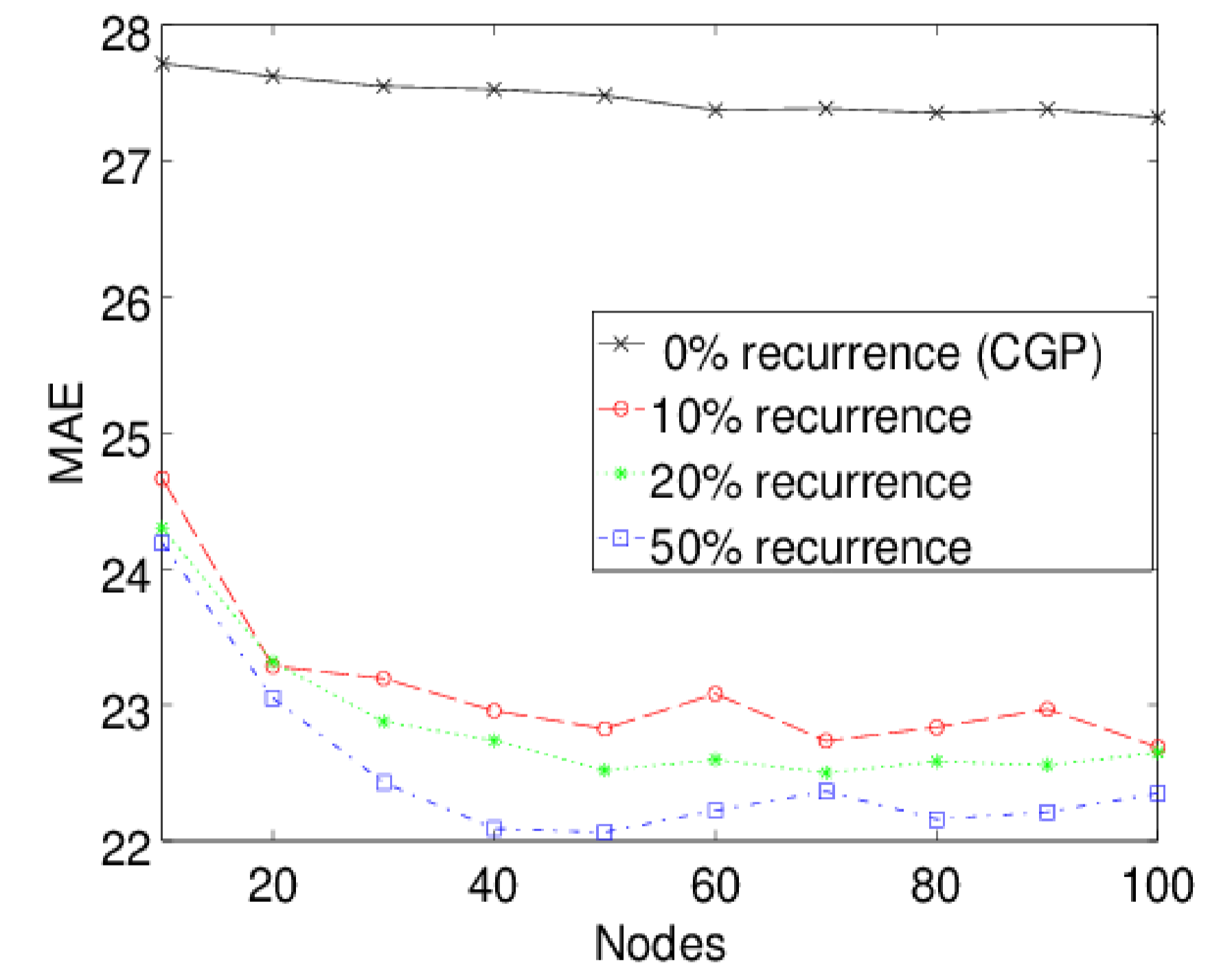
Application: Artificial Ant Variant

- Santa Fe Ant Trail
- Figure shows amount of food eaten
- RCGP is much more effective than CGP



Application: Predicting Yearly Sun Spots

- Predict the number of sun spots 35 years ahead
- Figure shows mean abs error on training set
- Again RCGP is much more effective than CGP



References

- [1] Miller, J. F. & Thomson, P., Cartesian Genetic Programming, EuroGP, 121-132, 2000
- [2] Turner, A. J. & Miller, J. F., Cartesian Genetic Programming: Why No Bloat?, EuroGP, 193-204, 2014
- [3] Yu, T. & Miller, J. F., Through the interaction of neutral and adaptive mutations, evolutionary search finds a way, Artificial Life, 12, 525-551, 2006
- [4] Turner, A. J. & Miller, J. F., Introducing A Cross Platform Open Source Cartesian Genetic Programming Library, Genetic Programming and Evolvable Machines, to Appear

Open Source CGP Library [4]

- www.cgplib.co.uk/
- Cross Platform
- CGP, RCGP & CGPANN
- Documented & tutorials
- Highly Customisable
- Multiple visualisation tools

