

Opportunities for GP researchers in blockchain

Trent McConaghy

Opportunity: Massive GP Compute

- Golem and iExec are decentralized marketplaces for massive compute
- Promise to be far cheaper than centralized cloud compute

Opportunity: *Way* more datasets, via Ocean😊

AI Practitioners /
Problem Solvers



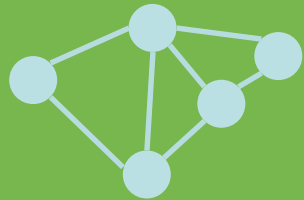
Data & compute
suppliers



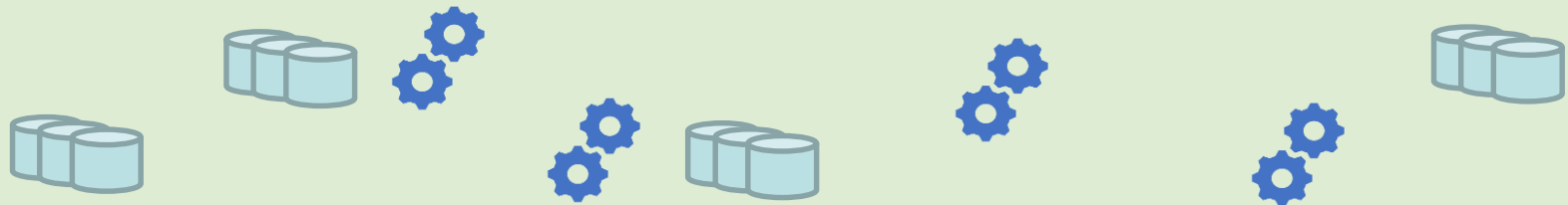
Data science frontends
(sklearn, Anaconda, etc)

Data Commons
Frontends

Marketplaces



Ocean Public Utility Network
Decentralized w/ data governance



Opportunity: Evolve Solidity

- Evolve Solidity smart contracts for whatever use case you want.
- Fitness function: Eth testnet running many agents
- Design space: a grammar with many Solidity code blocks

```
/// @title Voting with delegation.
contract Ballot
{
    // This declares a new complex type which will
    // be used for variables later.
    // It will represent a single voter.
    struct Voter
    {
        uint weight; // weight is accumulated by delegation
        bool voted; // if true, that person already voted
        address delegate; // person delegated to
        uint vote; // index of the voted proposal
    }
    // This is a type for a single proposal.
    struct Proposal
    {
        bytes32 name; // short name (up to 32 bytes)
        uint voteCount; // number of accumulated votes
    }

    address public chairperson;
    // This declares a state variable that
    // stores a `Voter` struct for each possible address.
    mapping(address => Voter) public voters;
    // A dynamically-sized array of `Proposal` structs.
    Proposal[] public proposals;

    /// Create a new ballot to choose one of `proposalNames`.
    function Ballot(bytes32[] proposalNames)
    {
        chairperson = msg.sender;
        voters[chairperson].weight = 1;
        // For each of the provided proposal names,
        // create a new proposal object and add it
        // to the end of the array.
        for (uint i = 0; i < proposalNames.length; i++)
            // `Proposal({...})` creates a temporary
            // Proposal object and `proposal.push(...)`
```

Opportunity: Evolve EVM Bytecode

- Ethereum Virtual Machine has about 100 operators
- GP could evolve these directly

The screenshot displays three sections of EVM data:

- Runtime Bytecode:** A hexadecimal string: `60606040525b600080fd00a165627a7a7230582012c9bd00152fa1c4`
- Opcodes:** A list of opcodes: `PUSH1 0x60 PUSH1 0x40 MSTORE PUSH1 0x18 PUSH1 0x0 SSTORE`
- Assembly:** A list of assembly instructions and their corresponding Solidity code:

```
.code
PUSH 60          contract MyContract {\
PUSH 40          contract MyContract {\
MSTORE          contract MyContract {\
PUSH 18         (10 + 2) * 2
PUSH 0          uint i = (10 + 2) * 2
SSTORE         uint i = (10 + 2) * 2
CALLVALUE      contract MyContract {\
ISZERO         contract MyContract {\
PUSH [tag] 1   contract MyContract {\
JUMPI          contract MyContract {\
PUSH 0         contract MyContract {\
DUP1           contract MyContract {\n    uin
REVERT        contract MyContract {\
```

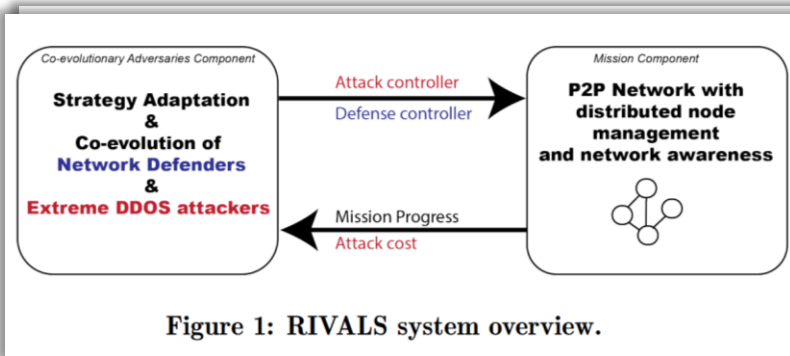
Opportunity: Evolve WASM Bytecode

- WebAssembly (WASM) is the future of smart contract VMs. Smart contracts in C, C++, Rust, ..
- WASM is already supported by major browsers

C input source	Linear assembly bytecode (intermediate representation)	Wasm binary encoding (hexadecimal bytes)
<pre>int factorial(int n) { if (n == 0) return 1; else return n * factorial(n-1); }</pre>	<pre>get_local 0 i64.eqz if (result i64) i64.const 1 else get_local 0 get_local 0 i64.const 1 i64.sub call 0 i64.mul end</pre>	<pre>20 00 50 04 7E 42 01 05 20 00 20 00 42 01 7D 10 00 7E 0B</pre>

Opportunity: Evolve Networks of Attackers & Defenders

- To improve security
- Like work from Una-May's lab



Developing proactive defenses for computer networks with coevolutionary genetic algorithms

Conference Paper · July 2017 with 12 Reads

DOI: 10.1145/3067695.3089234

Conference: Conference: the Genetic and Evolutionary Computation Conference Companion

[Cite this publication](#)



Anthony Erb Lugo



Dennis Garcia



Erik Hemberg

15.66 · Massachusetts Institute of Technology

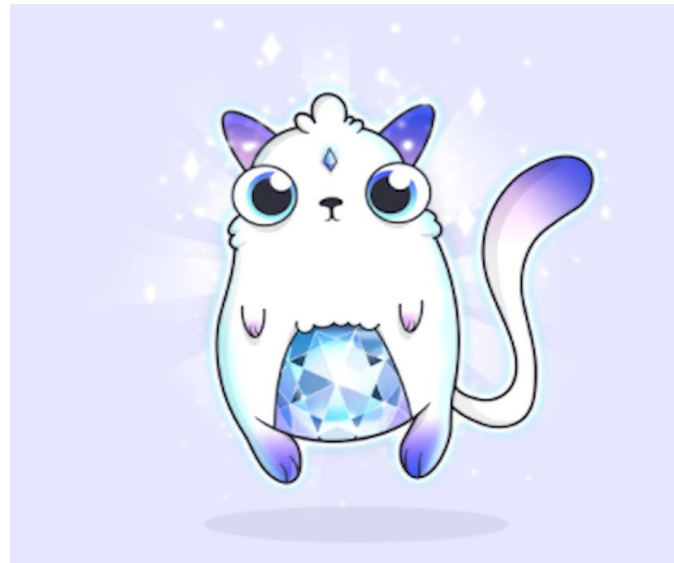


Una-May O'Reilly

19.56 · Massachusetts Institute of Technology

Opportunity: CryptoKitties ^ 10

- CryptoKitties starts with human-designed cats
- And is a GA with massive constraints
- Free it!

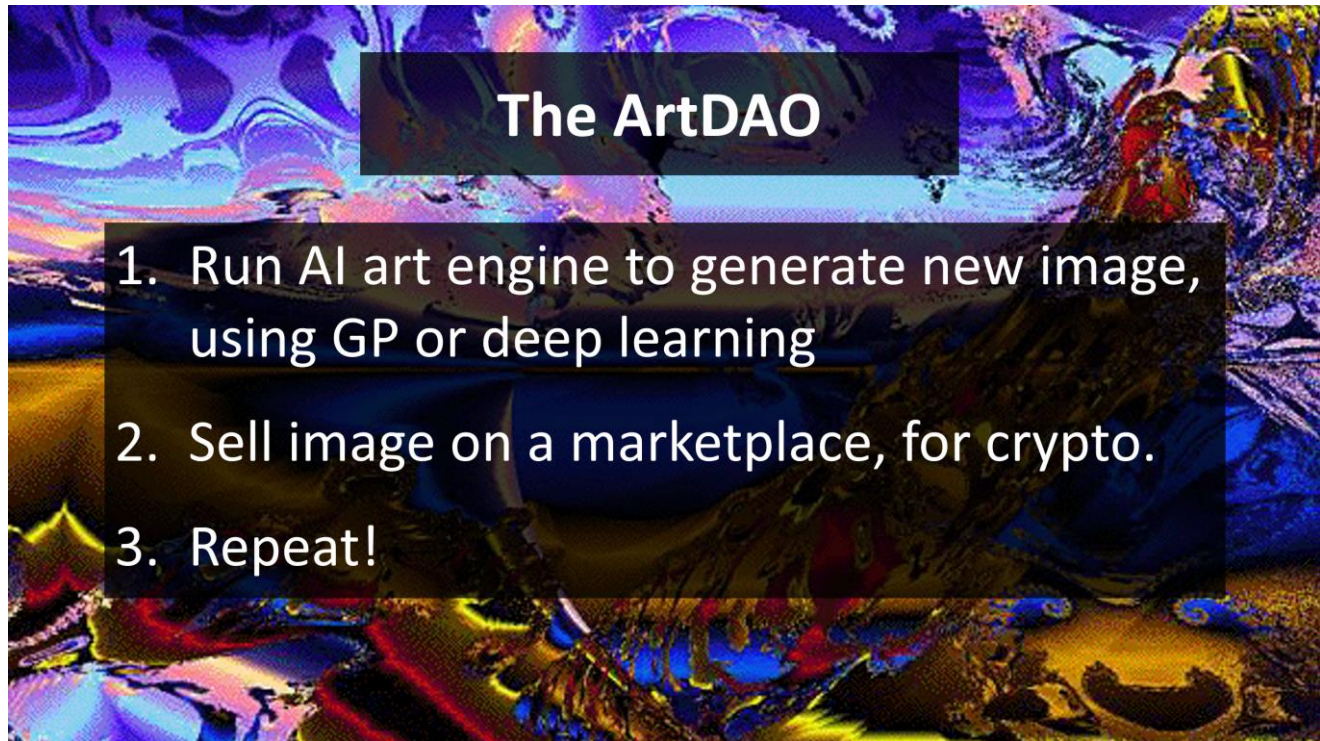


Opportunity: Unstoppable GP

- Imagine a GP run that never stops.
- If it runs out of resources, it stops. If it gets new resources, it continues.
- Riff: allow the algorithm to modify its own code
- “If you had a GP that could run forever, what substrate would you give it?” (James Foster)
 - E.g. Novelty search on WASM bytecode
 - E.g. MABE, Julian’s stuff, POE, ...

Opportunity: Build ArtDAO

- The first AI millionaire?



Opportunity: Evolvable ArtDAO

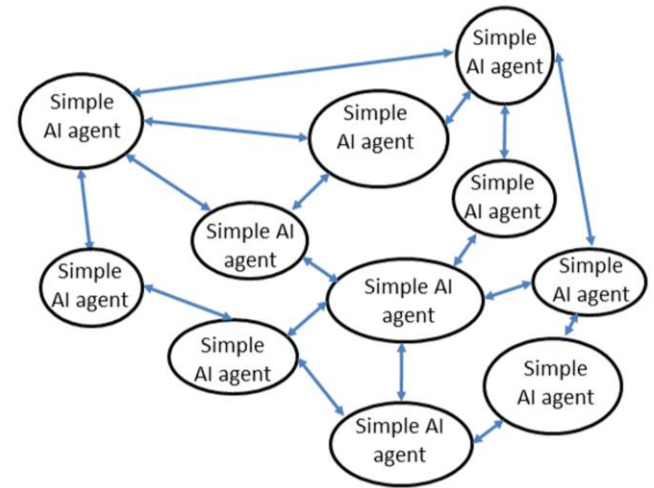
Auto adapt at the code level.

Here, the ArtDAO modifies its own code, in hopes of improving.

- It creates a copy of itself, changes that copy's code just a little bit, and gives a tiny bit of resources to that new copy.
- If that new copy is bad, it will simply run out of resources and be ignored.
- But **if that new copy is truly an improvement, the market will reward it**, and it will be able to amass resources and split more on its own.
- Over time, ArtDAO will spawn more children, and grandchildren, and the ones that do well will continue to spread. We end up with a mini-army of AI DAOs for art.
- If buyers are DAOs too, it's a network of DAOs, leading to swarm intelligence

Opportunity: Agent Life Forms

- One agent = a smart contract
- Launch a pop. of 100 agents
- Each agent “does its thing”, eg generate a random number, do escrow, etc.
- Each agent can reproduce, to make a new agent, using Solidity “factory”.
- Mutation: reproduce w variation.
- Crossover: reproduce from 2 parent smart contracts.
- Agents will “die” if they run out of resources (gas). But people can feed them. If people find a service by an agent interesting, it will keep getting fed
- Agents can interact with other agents, humans, ..
- Ultimately, the most useful services will survive and thrive
- Not simulated evolution. Simply **evolution**.



Summary

- Blockchains are a huge opportunity for GP.
- A path to automatic programming.
- Moonshots!
 - Unstoppable GP
 - Agent life forms on a natural substrate
- Actual life. Actual evolution.