CPD-12 Related research

I would like an overview of people who have tried to do PDCA as genetic algorithms.

So I wonder if GA may be the core idea in my thesis.
1. Introduction

2. Related Work
   - Multimodal Simulation
   - PDCA

3. Formulation

4. Implementation

5. Results

6. Discussion

This is my research. I should explain how my research gaps current knowledge.
The genius of this approach is that I'm documenting the very thing I'm writing anyway.

1) I always need an adequate number of problems. If I don't do this, I may get trapped, just like the CORBA case.

This approach is similar to the AR network idea in KTOP.

2)
3) When I design my own system, I also use GA.

How?

1. Initial phase = to proceed
2. Fitness = mutation level (≥ 0)
3. Repeat
   3.1 Select individuals to replace P(t)
   → 3.2 Breed new genome P(t+1)
   3.3 Terminate phase of new genome P(t+1)
   3.4 Replace worst of P(t) with best of P(t+1)
Does it make sense?

1) The 70 processes in 1808804
provide a fitness function,
not processes themselves.

- OR, each of the 70 processes could be TASKS.

It is the population of TASKS that is important.

Taylor's to be merged.

P1

T1 T2 T3 T4

Procedure

Tasks
2) How do I decide tasks to reproduce in a piece system?

- If something looks generic, I try to apply it all over.

- Andcet is a typical example. TheAndcet task becomes a control test for all 70 processes.

- Planning (sleeping RC) is another general task, identify RC considering under varying demands for all 70 processes.

- Doing condens of the actual process of moving papers. This pedestrian process is the core and anything related to like Taylor illustration, the pedestrian process can be break down into tasks.
1. Initial population
   Task 1 = randomly create
   Task 2 = produce (within paper) collect data
   Task 3 = incorporate (selective pressure to reproduce)

2. Fitness = ISO 9004 annual (30 points)

3. Repeat
   3.1 Select tasks to reproduce (survive)
      = select survivors until to reproduce
   3.2 Breed new population (new tasks) \( C(t) \)
   3.3 Define new population and evaluate (ISO 9004 annual)
   3.4 Replace most of \( P(t) \) with off-spring \( R(t+1) = P(t) + C(t) \)
The software process consists of ISO 9004 assessment, but also LEAN.

3. The idea is to focus on the most, trying to improve.

3. The idea is similar to breeding ideas (my phobia) in science.

breed ideas

Growing the tree (eliminate stupid ideas)

Everything should be as simple as possible. We don't want waste.
How should I select incoming projects?

- Projects should have impact,
- Projects should be simple to explain,
- Solutions should be simple to maintain.

Solutions should contextual to the small affinity and scalability of the ground production problem.

What are the processes in making a PAH? (a few)
What are the tasks? (many)
"Smallest task"
KUBAS quality imprint

1. Populace = complex program

2. Fitness = heuristically Assess and according to general system (map of processes)

3. Repeat

3.1

Heuristics: Pappy is not a program, they contain as

offspring, they are useful.

P(H) Q(H) 

most problematic

"best" ranking (by fitness) 

R(H) 

\[ P(H+1) = Q(H) + R(H) \]
3.1 Best ranking programs to reproduce.

3.2 Breed new programs (new programs + improved production rules)

3.3 Rank off-spring (find out which programs are ready for production)
\[ \text{Rank} = \{0, 1\} \]
1 = ready for produc
0 = not ready

3.4 \( P(t) = R(t) + Q(t) \)

Fitness = internal errors (Pareto) + external fitness with overall system.