# Quantum Generators: a Self-Driving Experimentation for New Reproducibility of Complex Micro-Scale Protein Synthesis.

Poondru Prithvinath Reddy



December 28, 2021

# Quantum Generators: A self-driving experimentation for new reproducibility of complex micro-scale protein synthesis.

. Poondru Prithvinath Reddy

**ABSTRACT**

Quantum Generators is a means of achieving mass food production with short production cycles and when and where required by means of machines rather than land based farming which has serious limitations. The process for agricultural practices for plant growth in different stages is simulated in a machine with a capacity to produce multiple seeds from one seed input using computational models of multiplication (generating multiple copies of kernel in repetition). In this paper, we present a method to replicate curiosity-driven learning that can accurately identify and analyse an unknown and complex set of conditions of an experimental or computational procedure in such a way that the method autonomously chooses the experiments that maximize the number of new and reproducible observations/targets for effective protein synthesis. This model is part of robotic synthesis (equipped with CA) that can effectively explore a complex phenomenon exhibited by protein folding in cell synthesizer and is designed in an open-ended way with no explicit discovery or optimization. By applying CA-based robotic model to multicomponent cell synthesis, we may discover specific response to environmental parameters like temperature, etc. in a CellSynputer where abstraction representing cell synthesis is embodied. In this way, it is possible to script and run desired synthesis with reconfigurable system for automated experimentation of diverse protein folding outcomes depending on the crop tissues. Since curiosity learning(CA) is nothing but Reinforcement learning, we show an implementation of it with small model in obscene of real-world model of CellSynputer. Although the platform model given us a method of automating and optimizing cellular assemblies however, this need to be tested using natural crop cells for quantum generation.

## INTRODUCTION

A **Quantum** (plural quanta) is the minimum amount of any physical entity (physical property) involved in an interaction. On the other hand, **Generators** don't actually create anything instead, they generate quantity prescribed by physical property through multiplication to produce high quality products on a mass scale. The aim of Quantum Generators is to produce multiple seeds from one seed at high seed rate to produce a particular class of food grains from specific class of **seed** on mass scale by means of machine rather than land farming.

The process for agricultural practices include preparation of soil, seed sowing, watering, adding manure and fertilizers, irrigation and harvesting. However, if we create same conditions as soil germination, special watering, fertilizers addition and plant growth in different stages in a machine with a capacity to produce multiple seeds from one seed input using computational models of multiplication( generating multiple copies of kernel in repetition ) then we will be closure to achieving mass food production by means of quantum generators( machine generated ) rather than traditional land based farming which has very serious limitations such as large space requirements, uncontrolled contaminants, etc. The development of Quantum Generators requires specialized knowledge in many fields including Cell Biology, Nanotechnology, 3D Cellprinting, Computing, Soil germination and initially they may be big occupying significantly large space and subsequently small enough to be placed on roof-tops.

The Quantum Generators help world meet the food needs of a growing population while simultaneously providing opportunities and revenue streams for farmers. This is crucial in order to grow enough food for growing populations without needing to expand farmland into wetlands, forests, or other important natural ecosystems. The Quantum Generators use significantly less space compared to farmland and also results in increased yield per square foot with short production cycles, reduced cost of cultivation besides easing storage and transportation requirements.

In addition, Quantum Generators Could Eliminate Agricultural Losses arising out of Cyclones, Floods, Insects, Pests, Droughts, Poor Harvest, Soil Contamination, Land Degradation, Wild Animals, Hailstorms, etc.

## METHODOLOGY

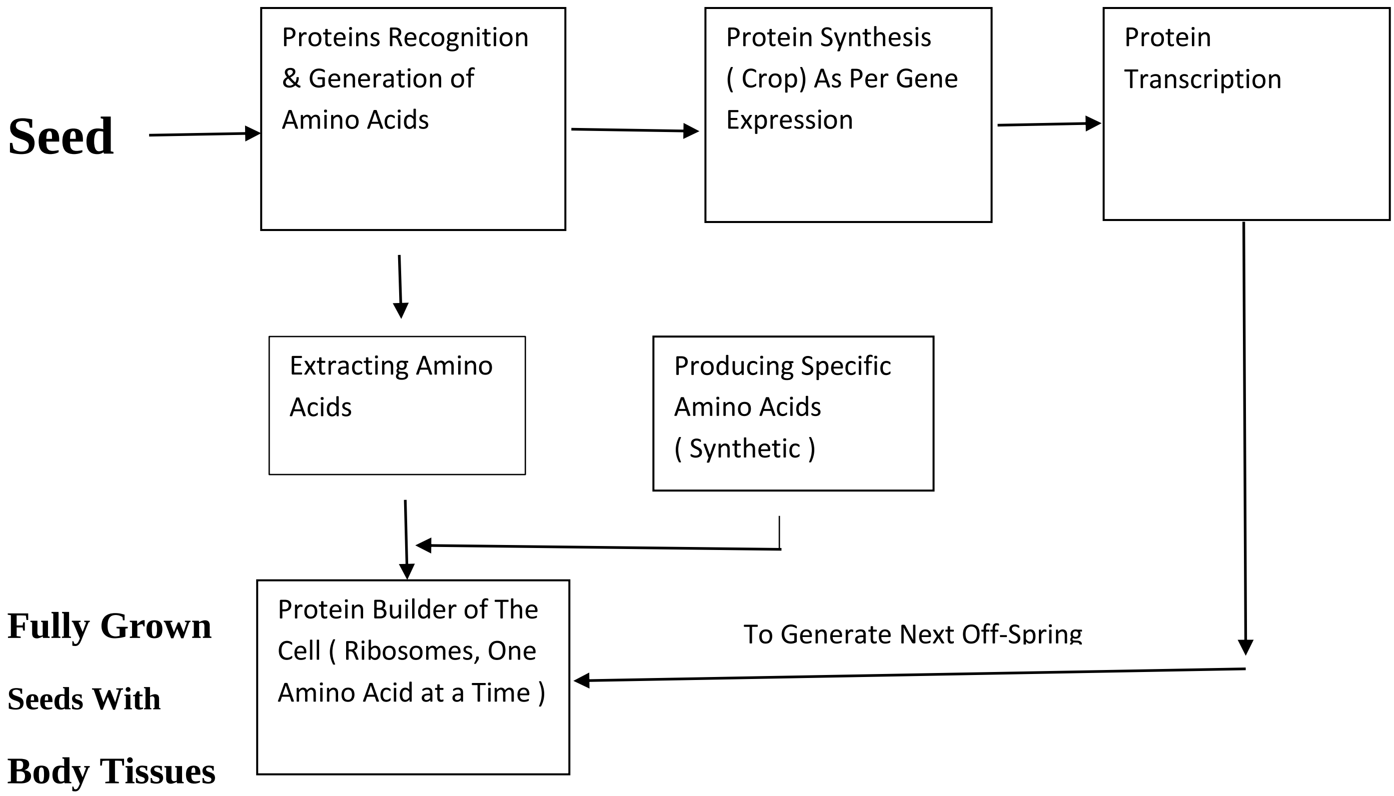



**Fig 1. Process Flow Diagram of Seed Builder**

**Protein** from input seeds is broken down into individual amino acids which are reassembled by Quantum Generating ribosomes into proteins that Crop cells need to be generated. The information to produce a protein is encoded in the **cell's** DNA. When a protein is produced, a copy of the DNA is made (called mRNA) and this copy is transported to a ribosome.

Protein **synthesis** is the process used by the QG( Quantum Generator ) to make proteins. The first step of protein **synthesis** is called Transcription. It occurs in the nucleus. During transcription, mRNA transcribes (copies) DNA.

Body tissues **grow** by increasing the number of cells that make them up. Every **cell** in the crop body contains protein. The basic structure of protein is a chain of amino acids. We need protein in our diet to help human body repair cells and make new ones. Protein is also important for growth and development in children, teens, and pregnant women.

The major steps in protein synthesis are:

- DNA unzips in the nucleus.
- mRNA nucleotides transcribe the complementary DNA message.
- mRNA leaves nucleus and goes to ribosome.
- mRNA attaches to ribosome and first codon is read.
- tRNA brings in proper amino acid from cytoplasm.
- a second tRNA brings in new amino acid.

**Protein** synthesis is the process in which **cells make proteins**. It occurs in two stages: transcription and translation. Transcription is the transfer of genetic instructions in DNA to mRNA in the nucleus. Translation occurs at the ribosome, which consists of rRNA and proteins**.**

**Ribosomes** are the protein builders or the protein synthesizers of the cell. Ribosomes, large complexes of **protein** and ribonucleic acid (RNA), are the cellular organelles responsible for protein synthesis. They are like construction guys who connect one amino acid at a time and build long chains. Ribosomes are special because they are found in both prokaryotes and eukaryotes.

During the **process** of transcription, the information stored in a gene's DNA is passed to a similar molecule called RNA (ribonucleic acid) in the cell nucleus. A type of RNA called transfer RNA (tRNA) assembles the protein, one amino acid at a time.

**Amino acids** can be produced by breaking down proteins, known as the extraction method. However, the amount of amino acids in the source protein limits the amount of amino acids made. Extraction is not good for making mass quantities of specific amino acids**.** So Synthetic Methods of making amino acids is necessary in protein synthesis.

The Quantum Generator contains pre-programmed Protein Synthesizer relevant to specific Crop/Tissue which essentially reassembles ribosomes ( Sites in a Cell ) into proteins that your crop cells need. The sequence and information to produce a protein is encoded in the synthesizer of Quantum Generator.

## Robotics for Automation and Optimization in Cell Synthesis

We believe that the potential of rapidly developing technologies (e.g., machine learning and robotics) are more fully realized by operating seamlessly with the way that synthetic biologists currently work. To reproduce this fundamental mode of operation, a new approach to the automated exploration of biological space is needed that combines an abstraction of biological synthesis with robotic hardware and closed-loop programming.

Here, we outline an approach to this problem beginning with an abstract representation of the practice of cell synthesis that then informs the programming and automation required for its practical realization. Using this foundation to construct closed-loop robotic synthesis engine, we can generate new synthesises that may be optimized, and repeated entirely automatically. These robots can perform synthesis reactions and this leads to a road map whereby molecules can be synthesized, optimized, and made on demand from a digital code.

The ability to make small molecules autonomously and automatically will be fundamental to many applications, including quantum generators. Additionally, automated synthesis requires (in many cases) optimization of reaction yields; following optimization, the best conditions can be fed to the synthesis robot to increase the overall yield. There are different approaches to automated yield optimization, and as optimization of reaction conditions requires live feedback from the robotic system, many different detectors are required to monitor progress of the reactions, including benchtop nuclear magnetic resonance spectroscopy, Raman spectroscopy, UV-Vis spectroscopy, etc. Harvested data are then fed to optimization algorithms to explore often the multidimensional parameter space.

## Machine Learning towards Biological Space Exploration

Machine learning approaches are fundamental to scientific investigation in many disciplines. In biological studies, many of these methods are widely applicable, we explore how robotics/automation are helping to progress cell synthesis through exploring biological space and beyond. Scientists have begun to embrace the power of machine learning coupled with statistically driven design in their research to predict the performance of synthetic reactions. For our study, the yield of a synthetic reaction can be predicted using **random forest ( machine learning algorithm )** in the multidimensional space obtained from robotic automation to map the yield landscape of intricate synthesis following synthesis code allowing improved prediction of high-yielding conditions and replication mechanisms. Synthesis through automation offers far better efficiency and accuracy. In addition, the machine learning algorithm explored a wider range of biological space that would need to be performed  purely automated random search and it is observed that self-driven laboratories/robots lead the way forward to fast-track synthesis.

In general, this approach allows for faster and more efficient retrosynthetic analysis than any other well-known method. Figure 2 shows a graphical representation of workflow for joining automated retrosynthesis with a synthesis robot and reaction optimization. The retrosynthetic module will generate a valid synthesis of the target that can then be transferred into synthesis code that can be executed in a robotic platform. The optimization module can optimize the whole sequence, getting the feedback from the robot.

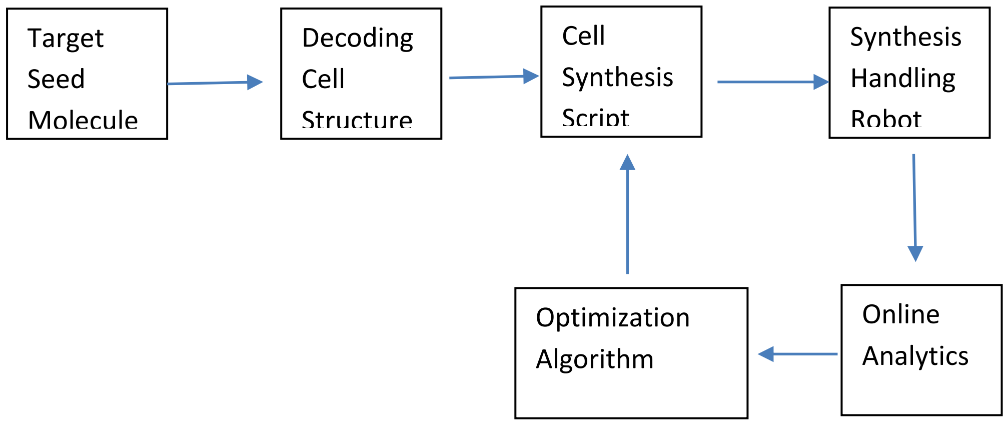



**Fig. 2 Architecture of Robotic Synthesis of Crop Cells in a Quantum Generator**

## Platform Design in Cell Synthesis

Methodologies for the automation of cell synthesis, optimization, and crop yields have not generally been designed for the realities of  crop-based yields,  instead focussed on engineering solutions to practical problems. To reproduce fundamental mode of operation in synthesis, a new universal approach to the automated exploration of cell synthesis space is needed that combines an abstraction of cell synthesis with robotic hardware and closed-loop programming. However, this leads biochemists to constantly test the reactions with different synthetic parameters and conditions.

**Automation Approach**

There are different automation approaches for cell synthesis these include block based, iterative, multistep however, we considered CellSynputer which is integration of abstraction, programming and hardware interface, which is given below depicted as in Fig 3.

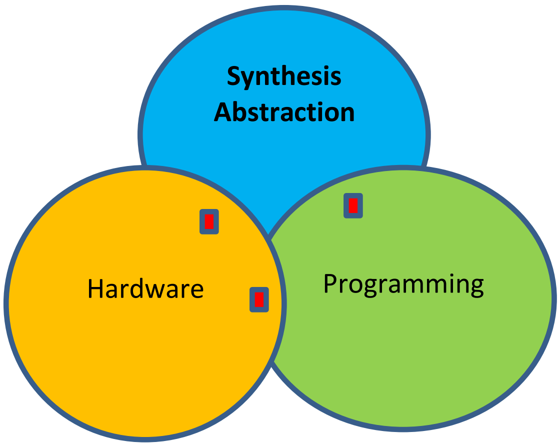



**Fig. 3 Approach – Cell Synthesis Automation**

Synthetic biologists already benefit from algorithms in the field of cell synthesis and, therefore, automation is one step forward that might help biologists and chemists to plan and develop biological space more quickly, efficiently, and importantly, CellSynputer is a platform that employs a broad range of algorithms interfacing hardware and abstraction to solve synthesis-related problems and surely can very well be established for quantum generation.

**Synthesis via Programmable Modular System: 'The CellSynputer'**

We presented a modular platform for automating cell synthesis, which embodies our synthesis abstraction in 'the CellSynputer'. Our abstraction of cell synthesis contains the key four stages of synthetic protocols: recognition, gene expression, transcription, and protein builder that can be linked to the physical operations of an automated robotic platform. Software control over hardware allowed combination of individual unit operations into multistep cell synthesis. A CellSynputer was created to program the platform; the system creates low-level instructions for the hardware taking graph representation of the platform and abstraction representing cell synthesis. In this way, it is possible to script and run published syntheses without reconfiguration of the platform, providing that necessary modules are present in the system. The synthesis of different small crop molecules on the system can be successfully scripted and performed automatically with yields comparable to traditional methods.

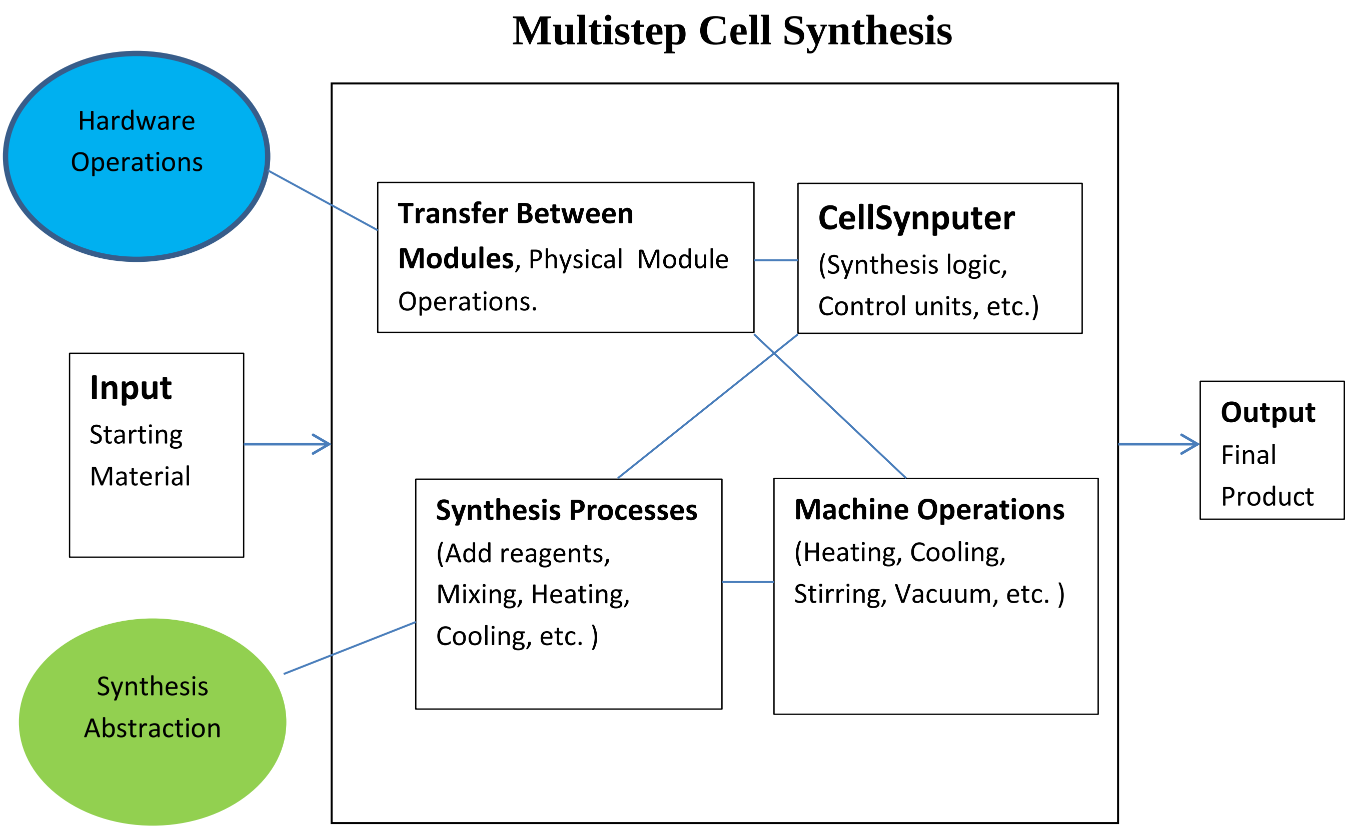



**Figure 4. CellSynputer Operational Architecture**

Finally, by combining CellSynputer platform and robotic systems with AI, it is possible to build autonomous systems working in closed loop, making decisions based on prior experiments and reactive conditions. We already presented a flow system for navigating a network of synthesis reactions utilizing an infrared spectrometer for on-line analysis and as the sensor for data feedback. The system will be able to select the most reactive or suitable starting materials autonomously on the basis of change in the infrared spectra between starting materials and end products

## ARCHITECTURE

### Robot Driven Curiosity Algorithm for Cell Synthesis

**Curiosity algorithm -** developed to replicate curiosity-driven learning in humans that can accurately analyse an unknown and complex micro-scale system. The knowledge for the synthesis is designed in such a way that the algorithm autonomously chooses the experiments that maximize the number of new and reproducible observations.

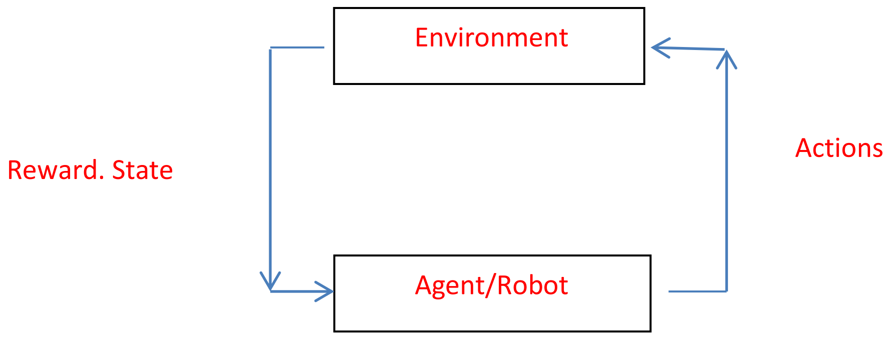
The idea of curiosity-driven learning is **to build a reward function that is intrinsic to the agent (generated by the agent itself)**. In this sense, the agent will act as a self-learner since it will be the student, with its own feedback. Intrinsic curiosity refers to **the idea of self-motivating the agent through an additional reward signal**, in order to guide the agent towards the desired outcome.

Reinforcement learning is a machine learning method based on rewarding desired behaviours and/or punishing undesired ones. In general, a reinforcement learning agent **is able to perceive and interpret its environment, take actions and learn through trial and error** and in reinforcement learning, Curiosity is **a type of intrinsic reward function which uses prediction error as reward signal**..

### Reinforcement Learning

- Reinforcement Learning is a feedback-based Machine learning technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or penalty.
- In Reinforcement Learning, the agent learns automatically using feedbacks without any labelled data, unlike supervised learning.

- Since there is no labeled data, so the agent is bound to learn by its experience only.
- RL solves a specific type of problem where decision making is sequential, and the goal is long-term, such as **game-playing, robotics**, etc.




**Approaches to implement Reinforcement Learning**

There are mainly three ways to implement reinforcement-learning in ML, which are:

1. **Value-based:**
   The value-based approach is about to find the optimal value function, which is the maximum value at a state under any policy. Therefore, the agent expects the long-term return at any state(s) under policy π.
2. **Policy-based:**
   Policy-based approach is to find the optimal policy for the maximum future rewards without using the value function. In this approach, the agent tries to apply such a policy that the action performed in each step helps to maximize the future reward.
   The policy-based approach has mainly two types of policy:
   - **Deterministic:** The same action is produced by the policy (π) at any state.
   - **Stochastic:** In this policy, probability determines the produced action.
3. **Model-based:** In the model-based approach, a virtual model is created for the environment, and the agent explores that environment to learn it. Model-based reinforcement learning can be very advantageous in applications like robotics and self-driving cars.

**Reinforcement Learning** is used in **Robotics, and is used for**, **adaptive control** such as Factory processes, telecommunication, Helicopter pilot navigation, etc. We used Reinforcement Learning in the cell synthesis for evaluating and planning protein folding strategies.

**Reinforcement Learning: Q-learning Algorithm Implementation**

We are going to look at implemention of a basic Reinforcement Learning algorithm which is called the Q-Learning technique and where we attempt to teach an agent/ robot to plan its strategies using the **Q-Learning technique**. **And this self-driving automation assisted agent/robot that is part of CellSynputer where abstraction representing cell synthesis is embodied.**

Now, Let's bring this agent/robot to a more realistic setting. Let us imagine that the agent is a process control supervisor and is trying to find out the best combination of amino acids, enzymes, environmental conditions, etc. He naturally concludes that there are no fixed set of environmental parameters the protein synthesis of desired crops can take and the proportionate mix of amino acids, enzymes, etc. Also, the result of synthesis leave a trace of their components when product is produced and this can help the agent/robot in finding out the required combination including environmental parameters. We want to train our agent/robot to find the best plan for synthesis using these Environmental Clues.

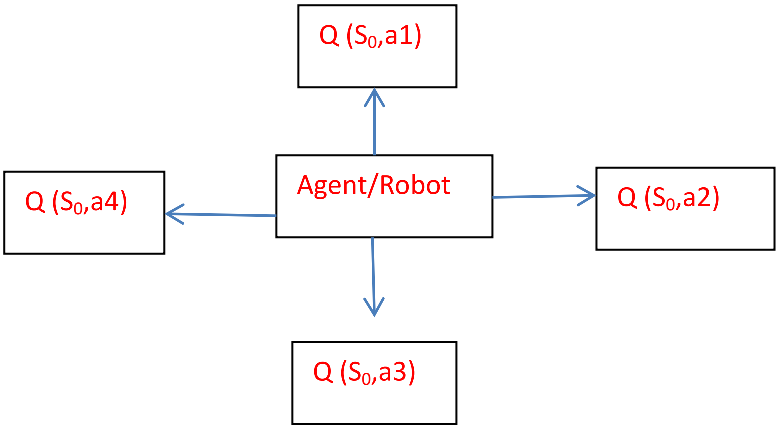
**Q-Learning:**

- Q-learning is a popular model-free reinforcement learning algorithm based on the Bellman equation.
- The main objective of Q-learning is to learn the policy which can inform the agent that what actions should be taken for maximizing the reward under what circumstances.
- It is an **off-policy RL** that attempts to find the best action to take at a current state.
  - 
- It learns the value function Q (S, a), which means how good to take action "**a**" at a particular state "**s**."
- The goal of the agent in Q-learning is to maximize the value of Q.
- The value of Q-learning can be derived from the Bellman equation. Consider the Bellman equation given below:

$$V(s) = \max [R(s,a) + \gamma \textstyle\sum_{s'} P(s,a,s')V(s')]$$

In the equation, we have various components, including reward, discount factor (γ), probability, and end states s'.

In this setup, our agent has been given different value options, V(s1), V(s2), V(s3), etc. and the agent can choose any option, so he needs to decide which one to go for the best or optimal plan/strategy. Here agent will take a move as per probability bases and changes the state.




Q- represents the quality of the actions at each state. So instead of using a value at each state, we will use a pair of state and action, i.e., Q(s, a). Q-value specifies that which action is more worthy than others, and according to the best Q-value, the agent takes his next move. The Bellman equation can be used for deriving the Q-value.

To perform any action, the agent will get a reward R(s, a), and also he will end up on a certain state, so the Q -value equation will be:

$$Q(S, a) = R(s, a) + \gamma \textstyle\sum_{s'} P(s,a,s')V(s')$$

Hence, we can say that, ***V(s) = max [Q(s, a)]***

**The above formulas are used to estimate the Q-values in Q-Learning.**

**Q-table:**

A Q-table or matrix is created while performing the Q-learning. The table follows the state and action pair, i.e., [s, a], and initializes the values to zero. After each action, the table is updated, and the q-values are stored within the table.

The RL( Reinforcement Learning ) agent uses this Q-table as a reference table to select the best action based on the q-values.

The setup of reinforcement learning implemented above was a very basic one and however, in practical case like protein synthesis for crop yields involve environmental complexity mainly requiring the concept of game theory. Hence, the illustration presented above was a very elementary one representing the agent behaviour and many practical examples like **Self Driving Cars, UAVs, etc.** involve more evolved the concepts of Game Theory with multiple parameters.

## CONCLUSION

Quantum Generators (QG) creates new seeds iteratively using the single input seed and the process leads to a phenomenon of generating multiple copies of kernels in repetition. We presented a robotic synthesis equipped with curiosity-driven learning that can effectively explore unknown and complex phenomenon of protein folding in cell synthesizer and is designed in an open-ended way with no explicit discovery or optimization. In this way, an automation assisted synthesizer with reconfigurable system that is part of CellSynputer is feasible for automated experimentation of diverse protein folding outcomes depending on the crop tissues and in that respect an implementation of Reinforcement Learning based on small model is presented. Although the platform model given us a method of automating and optimizing cellular assemblies however, this need to be tested using natural crop cells for quantum generation.

## REFERENCE


1. Poondru Prithvinath Reddy: "**Quantum Generators: A Platform for Automated Synthesis in a Modular Robotic System Driven by Cell Programming**", Google Scholar.
2. Jonathan Grizou, Laurie J. Points, Abhishek Sharma , Leroy Cronin: "**A Closed Loop Discovery Robot Driven by a Curiosity Algorithm Discovers Proto-Cells That Show Complex and Emergent Behaviours",** University of Glasgow