On the computational power of photosynthesis

Niall Murphy, Damien Woods and Thomas J. Naughton

Technical Report

NUIM-CS-TR-2005-03

NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

Department of Computer Science
National University of Ireland, Maynooth
County Kildare, Ireland

Abstract

We describe a simplified explanation of photosynthesis. We characterise the complexity of photosynthesis by interpreting its chemical equation as a language acceptance problem. A model of computation is generalised from our description of photosynthesis. It is then proved that this model is Turing universal. An instance of the model is constructed that acts like photosynthesis.

1 Introduction

Photosynthesis is the process by which certain plant cells use water, carbon dioxide and light energy to produce sugar. Here we examine photosynthesis from the point of view of computational complexity. We show that photosynthesis can be described as a context-sensitive language acceptance problem. We begin with a biological description of photosynthesis.

2 A brief introduction to photosynthesis

We introduce a simplified overview of the process of photosynthesis as it applies to many green plants [1, 2]. In the cells of all green parts of a plant there are organelles called chloroplasts. In particular, high numbers of chloroplasts are found in the leaf cells. The chloroplast is a body containing thylakoid membranes in a dense fluid called the stroma. The thylakoid membranes contain chlorophyll molecules.

2.1 Light dependant phase

The light dependant phase takes place in the thylakoid membranes of the chloroplast.

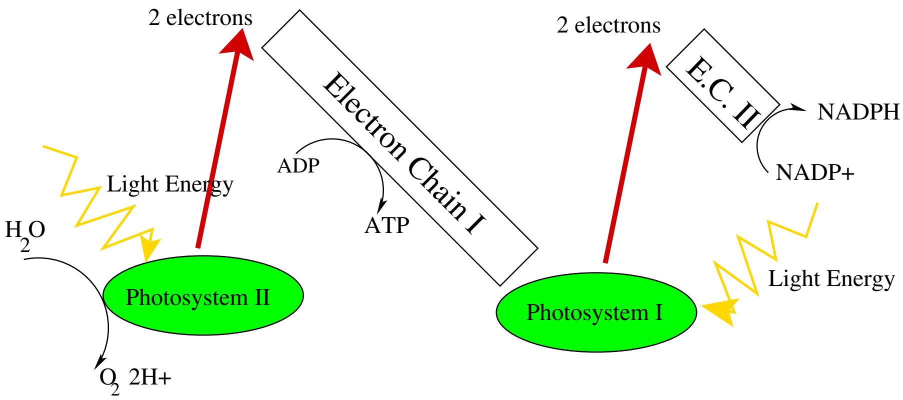

Figure 1: The light dependant phase of photosynthesis showing non-cyclic photophosporylation

2.1.1 Photosystem II

Light energy is absorbed by chlorophyll which becomes oxidised by losing two electrons. H_2O is split by an enzyme to provide electrons to reduce the oxidised chlorophyll. Four H^+ ions and an O_2 molecule are released for every two H_2O molecules. The electrons are passed to photosystem I via an electron transport chain. As the electrons move down the chain they are used to pull a H^+ ion across the thylakoid membrane. This creates a "proton gradient" which will cause a H^+ ion to exit the thylakoid membrane via ATP synthase. This is called a "proton pump". ATP synthase combines a phosphate and adenosine diphosphate (ADP) to produce adenosine triphosphate (ATP). The ATP molecules produced then proceed to the stroma where the Calvin cycle takes place.

2.1.2 Photosystem I

When the electrons of photosystem II reach photosystem I they reduce the oxidised chlorophyll there. Light energy again excites the chlorophyll which looses two electrons to a second electron transport chain. Here the electrons energy is used to combine H^+ ions and $NADP^+$ to produce NADPH. The NADPH molecules proceed to the Calvin cycle in stroma.

2.2 The Calvin cycle

The Calvin cycle occurs in the stroma of the chloroplast. Here the energy stored in the ATP and NADPH molecules that were produced in the light dependant phase are used to catalyse reactions that manipulate carbohydrates to produce a sugar. This sugar is used or stored elsewhere in the plant.

The cycle begins when each of three CO_2 molecules are attached to three ribulose bi-phosphate (RuBP) in the presence of the enzyme rubisco. The product is so unstable it splits immediately into six molecules of phosphoglycerate (PhGly). Six ATP are used to convert the phosphoglycerate into six molecules of 1,3-Bisphosphoglycerate (1,3BiPhGly). Six NADPH are then used to convert these into six molecules of glyceraldehyde-3-phosphate (G3P). One of these molecules is the output of the system, it is converted into other sugars before being used or stored in the plant. The five remaining molecules of G3P are converted back into three RuBP using three ATP and the cycle begins again.

3 Photosynthesis as a language acceptance problem

Based on its chemical equation we express photosynthesis as a language acceptance problem. We take the standard [2] photosynthesis chemical equation.[1]

$$6\mathrm{H_2O} + 6\mathrm{CO_2} + {}^{\text{light}}_{\text{energy}} \rightarrow \mathrm{C_6H_{12}O_6} + 6\mathrm{O_2} \tag{1}$$

For brevity we replace the four molecules with symbols b, c, d, and e, respectively. We quantify the light energy into units and we represent such a unit

[1] The use of $C_6H_{12}O_6$ instead of G3P is to produce a simple and balanced chemical equation.

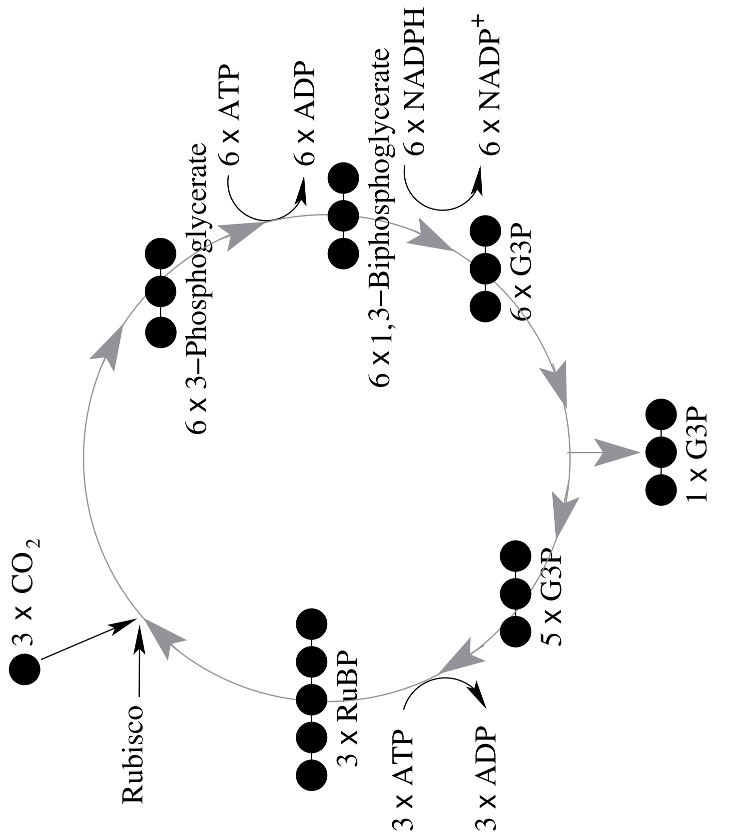

Figure 2: The Calvin cycle. This cycle occurs in the stroma of the chloroplast.

using the symbol a. Essentially a represents the amount of light energy needed to generate one molecule of $C_6H_{12}O_6$.

As usual the notation x^y denotes y juxtaposed copies of the symbol x. We claim that the process of photosynthesis has exactly the computational power required to compute the transformation on words

$$a^{k_0}b^{k_1}c^{k_2}d^0e^0 \rightarrow a^{k_0-n}b^{k_1-6n}c^{k_2-6n}d^ne^{6n} \tag{2}$$

where $n \in \mathbb{N}$ is the number of $C_6H_{12}O_6$ molecules produced in the process. Also $k_0, k_1, k_2 \in \mathbb{N}$ are the numbers of units of light energy, H_2O molecules, and CO_2 molecules, respectively available to the system during the transformation. So for example when k_0, k_1, k_2 are large, photosynthesis creates many $C_6H_{12}O_6$ molecules. Equation (2) recasts Equation (1) into a transformation on words. Additionally Equation (2) generalises Equation (1) somewhat, so that the transformation works in the presence of surplus H_2O, CO_2, and light energy.

In a slightly more simplified language form, we claim that the process of photosynthesis computes (or accepts) the language

$$\left\{a^{n+k_0}b^{6n+k_1}c^{6n+k_2}d^ne^{6n} : n, k_0, k_1, k_2 \in \mathbb{N}\right\} \tag{3}$$

Again the values k_0, k_1, k_2 refer to the fact that we may have surplus H_2O, CO_2, and light energy. This language is at least as hard as the textbook [5] context-sensitive language $\{a^nb^nc^n : n \in \mathbb{N}\}$ and is itself context-sensitive. Thus this problem requires a model of computation that has at least the power of a linear bounded automaton [5], that is, a Turing machine [6] that uses at most linear space.

4 Generalised model of photosynthesis

From the description of the process of photosynthesis in Section 2 we generalise the following model of computation. The model contains three individual automata that each have access to a subset of a set of global counters.

Definition 1 (Generalised photosynthesis automaton) *Each instance of a generalised photosynthesis automaton is a tuple* $(S_A, S_B, S_C, \sigma_A, \sigma_B, \sigma_C, T, \delta_A, \delta_B, \delta_C)$ *where*

- S_A *is a finite set of states for automaton* A*,*
- S_B *is a finite set of states for automaton* B*,*
- S_C *is a finite set of states for automaton* C*,*
- $\sigma_A \in S_A$ *is the start state of* A*,*
- $\sigma_B \in S_B$ *is the start state of* B*,*
- $\sigma_C \in S_C$ *is the start state of* C*,*
- T *is a set of counters, each counter has a value from* $\{0,\ 1,\ 2,\ 3,\ \ldots\}$*,*
- $\delta_A : S_A \times 2^{(\{-\}\times T)} \to S_A \times 2^{(\{+\}\times T)}$ *is the transition function for* A*,*
- $\delta_B : S_B \times 2^{(\{-\}\times \mathbb{N} \times T)} \to S_B \times 2^{(\{+\}\times \mathbb{N} \times T)}$ *is the transition function for* B*,*
- $\delta_C : S_C \times \left\{2^{(\{-\}\times T)} \cup (\{>, \leqslant\} \times T \times T)\right\} \to S_C \times 2^{(\{+\}\times T)}$ *is the transition function for* C*.*

The tuple $(+, t_i) \in \{+\} \times T$ means increment counter t_i. The tuple $(-, t_i) \in \{-\} \times T$ means decrement counter t_i. As a parameter to a transition it also becomes a comparison with zero. For example, $(s_i, \{(-, t_j)\}) \to (s_k, \{(+, t_l)\}) \in \delta_A$ means if in state s_i and counter $t_j > 0$ then decrement t_j, increment t_l, and make the transition to s_k. The tuple $(-, i, t_j) \in \{-\} \times \mathbb{N} \times T$ means decrement t_j by i units. The tuple $(s_i, (>, t_j, t_k)) \to (s_l, \{(+, t_m)\}) \in \delta_C$ means that this transition can be made if in state s_i and if the value in t_j is greater than that in t_k. Similarly, if the $>$ is replaced with a $\leqslant$ then the transition can only be made if the value in t_j is not greater than that in t_k.

A configuration in an instance of this model is a tuple $(\alpha \in S_A, \beta \in S_B, \gamma \in S_C, T)$, where each mapping in T is defined. The starting configuration of an instance of the model is a tuple $(\sigma_A, \sigma_B, \sigma_C, T)$ where the mappings of each element of T are provided by the user and constitute the inputs to the computation. We do not define what a halting configuration of an instance of the model is, and so do not mathematically define what a photosynthesis computation is at this time.

4.1 Model is Turing universal

The model of computation of generalised photosynthesis is universal. It has been proved by Minsky that even a single automaton with no more than two counters is universal [4]. This means that photosynthesis in its general form is capable of computing any function that is computable by a digital electronic computer or Turing machine. We argue that this generalised model of photosynthesis is a reasonable generalisation of a machine to accept the language defined in Section 3. In the next section we show this by supplying a specific instance of the generalised model that accepts the language defined by Equation (3) and illustrating that it uses all of the features provided by the generalised model.

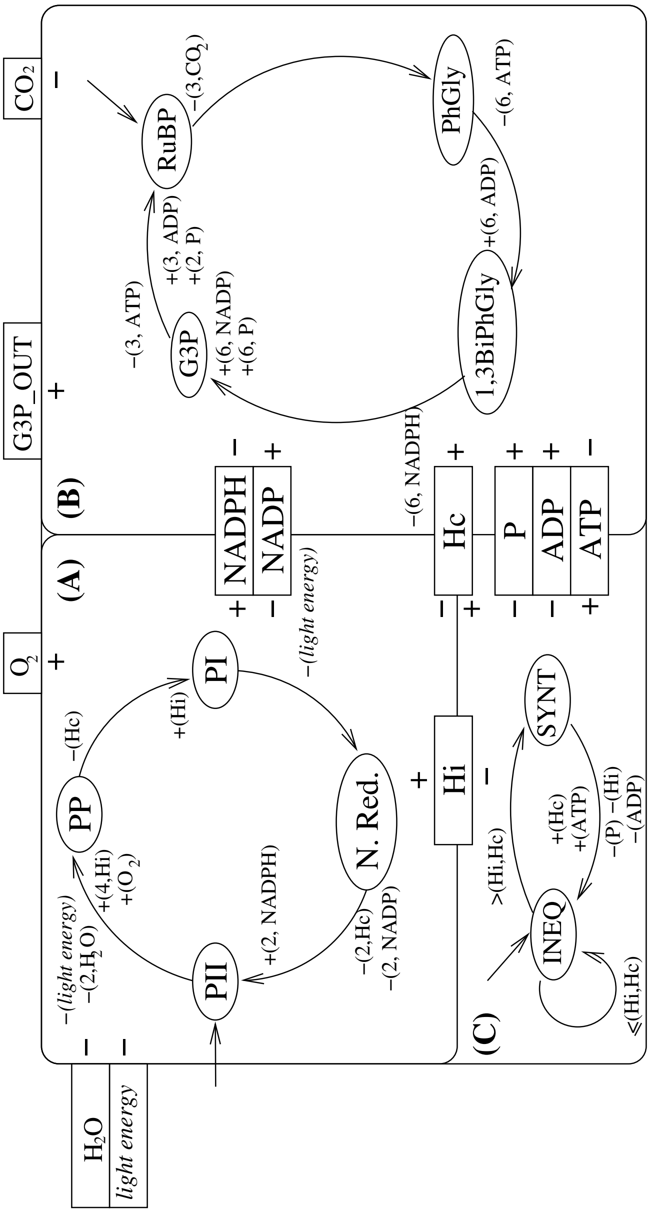

Figure 3: An instance of our generalised photosynthesis automaton. This machine represents the process of photosynthesis. Automaton A represents the light phase of photosynthesis (see Section 2.1). Automaton B represents ATP synthesis, part of the light dependant phase. Automaton C represents the Calvin cycle (see Section 2.2). The items in square boxes are counters of a named substance. A counter between two automata means it is available to each one. The plus or minus symbol indicates what effect each system can have on this counter. The abbreviation PP stands for Proton Pump and N. Red. stands for NADP Reductase.

Table 1: The transition function $\delta_{\mathcal{A}}$ for the automaton in Definition 2.

State	Counter	Next state	Counter
PII	$\{-(2, \text{H2O}), -(\text{light energy})\}$	PP	$\{+(4, \text{Hi}), +(\text{O2})\}$
PP	$\{-(\text{Hc})\}$	PI	$\{+(\text{Hi})\}$
PI	$\{-(\text{light energy})\}$	N.Red	$\{\}$
N.Red	$\{-(2,\text{Hc}), -(2,\text{NADP})\}$	PII	$\{+(2,\text{NADPH})\}$

Table 2: The transition function $\delta_{\mathcal{B}}$ for the automaton in Definition 2.

State	Counter	Next state	Counter
RuBP	$\{-(3,\text{CO2})\}$	PhGly	$\{\}$
PhGly	$\{-(6,\text{ATP})\}$	1-3BiPhGly	$\{+(6,\text{ADP})\}$
1-3BiPhGly	$\{-(6,\text{NADPH})\}$	G3P	$\{+(6,\text{NADP}), +(6,\text{P})\}$
G3P	$\{-(3,\text{ATP})\}$	RuBP	$\{+(3,\text{ADP}), +(2,\text{P}), +(\text{G3P\_OUT})\}$

5 Instance of the model for photosynthesis

In Definition 2 we give a specific instance of our generalised photosynthesis automaton, this instance is illustrated in Figure 3 and accepts the photosynthesis language defined in Equation (3).

Definition 2 *Let $M = (S_A, S_B, S_C, \sigma_A, \sigma_B, \sigma_C, T, \delta_A, \delta_B, \delta_C)$ be an instance of the generalised photosynthesis automaton where*

- $S_A = \{$*PI, PP, PII, N.Red*$\}$,
- $S_B = \{$*RuBP, PhGly, 1-3BiPhGly, G3P*$\}$,
- $S_C = \{$*INEQ, SYNT*$\}$,
- $\sigma_A =$ *PII,*
- $\sigma_B =$ *RuBP,*
- $\sigma_C =$ *INEQ,*
- $T = \{$*H2O, Hc, Hi, O2, light energy, ATP, ADP, NADPH, NADP, P, CO2, G3P_OUT*$\}$,
- δ_A *is defined in Table 1,*
- δ_B *is defined in Table 2,*
- δ_C *is defined in Table 3.*

A typical starting configuration of the machine (encoding the inputs) would be (PII, RuBP, INEQ, {H2O $\rightarrow 10^3$, Hc $\rightarrow 10^2$, Hi $\rightarrow 10^2$, O2 $\rightarrow 0$, light energy $\rightarrow 10^5$, ATP $\rightarrow 9$, ADP $\rightarrow 9$, NADPH $\rightarrow 9$, NADP $\rightarrow 9$, P $\rightarrow 9$, CO2 $\rightarrow 10^3$}). This instance accepts the photosynthesis language from Section 3 and uses all of the features from our generalised model of photosynthesis.

Table 3: The transition function $\delta_{\mathcal{C}}$ for the automaton in Definition 2.

State	Counter	Next state	Counter
INEQ	$\{> (\text{Hi, Hc})\}$	SYNT	$\{\}$
INEQ	$\{\leqslant (\text{Hi, Hc})\}$	INEQ	$\{\}$
SYNT	$\{-(\text{ADP}), -(\text{P}), -(\text{Hi})\}$	INEQ	$\{+(\text{ATP}), +(\text{Hc})\}$

6 Conclusions

Using our formalism it can be seen that the context-sensitive process of photosynthesis is composed of several simpler processes that accept regular or context-free languages. The combination and communication of these processes results in the greater computational power of the system.

Our generalised photosynthesis model permits instances of automata that do not faithfully model biological photosynthesis. Could it be possible to actually create such an instance that is implementable by a chloroplast and yet does not destroy the plant? From the biological point of view, programming (the DNA of) plants in this way would be notoriously difficult. However in the next few years, thanks to efforts such as the Registry of Standard Biological Parts (`http://parts.mit.edu/`) this may become more achievable.

Another possibility is to manipulate the default behaviour of the processes of photosynthesis to accept and reject words in its language at different times. This could be achieved by controlling and changing the levels of various substances involved in the process.

In our model the light dependant phase compares two natural numbers of arbitrary size in constant time. Also, in a given chloroplast there would be many 'instances' of photosynthesis taking place. Thus it might be interesting to study the computational power of photosynthesis from the parallel computing point of view.

References

[1] Alberts, Johnson, Lewis, Raff, Roberts, and Walter. *Molecular Biology of the Cell, 4th Edition.* Garlend Science, 2002.

[2] Neil A. Campbell and Jane B. Reece. *Biology 6th Edition.* Pearson Education Inc., 2002.

[3] Dexter C. Kozen. *Automata and Computability.* Springer, 1997.

[4] Marvin L. Minsky. *Computation, Finite and Infinite Machines.* Prentice Hall, 1967.

[5] Michael Sipser. *Introduction to the Theory of Computation.* PWS Publishing Company, 1996.

[6] Alan M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 1936.