Performance of Photovoltaic Maximum Power Point Tracking Algorithms in the Presence of Noise

Alexander M. Latham, Charles R. Sullivan, and Kofi M. Odame
Thayer School of Engineering
Dartmouth College Hanover, NH 03755
Email: alexander.m.latham@dartmouth.edu, charles.r.sullivan@dartmouth.edu

Abstract—This paper introduces a probabilistic analysis of the effects of noise on various maximum power point tracking (MPPT) algorithms for photovoltaic systems, including how noise affects both tracking speed and overall efficiency. The results of this analysis are verified by simulations. This analysis provides a better understanding of how noise affects performance and it can be used to optimize an MPPT system.

I. INTRODUCTION

Maximum power point tracking (MPPT) has become a standard technique for high-performance photovoltaic systems. An intelligent controller adjusts the voltage, current, or impedance seen by a solar array until the operating point that provides maximum power for the connected array in the present temperature and insolation conditions is found. There is a large body of literature describing MPPT control techniques, including the surveys [1] and [2]. Although the established techniques are routinely implemented in industry, and generally give satisfactory performance, publication on the topic continues to accelerate, with dozens of publications per year in the last decade [1], in part because of the importance of getting the best possible output from an expensive solar array.

Key metrics for an MPPT algorithm include tracking speed and accuracy, as is discussed extensively in the literature (e.g., [3]). However, the fundamental constraint on tracking accuracy is often the effect of noise in the measurement on the behavior of the tracking algorithm. Noise can also affect tracking speed in some cases. Standard tracking algorithms involve directly or indirectly introducing a periodic perturbation in the operating point in order to measure the slope of some characteristic. This perturbation reduces the power obtained from the solar panel because the panel is no longer operated consistently at the maximum power point, even if the algorithm has successfully found that point. This provides an incentive to reduce the size of the perturbation. However, as the size of the perturbation is reduced, the signal-to-noise ratio in the measurement of the slope is degraded. Thus, noise fundamentally limits the performance. This is particularly important in methods that require a current measurement, as some current measurement methods (e.g. Hall-effect transducers) are inherently noisy [4], and the use of a sense resistor entails a tradeoff between signal-to-noise ratio in the measurement and power loss in the resistor [5].

The importance of noise is acknowledged in a subset of the literature on MPPT and is sometimes used to motivate particular algorithms or hardware configurations (e.g., [3], [6]–[10]) but with very few exceptions (e.g., [11]), the impact of the noise is not analyzed quantitatively. In [12], we quantitatively analyze the effect of noise on a continuous-time MPPT algorithm. In this paper, we develop quantitative analysis of the impact of noise on two discrete-time maximum power point tracking algorithms. The analysis is verified through dynamic simulations which include noise, and the performance of these algorithms are compared to the system analyzed in [12].

II. NOISE EFFECT ON SLEW RATE OF PERTURB AND OBSERVE

Consider an MPPT system with a simple perturb and observe (P&O) tracking algorithm, where one changes a variable $X$, which could be a voltage, current or duty cycle, that influences the operating point of the array, by a fixed $\Delta X$ each period, $\Delta T$, and measures the power output of the array to determine how to change $X$ next [1], [2]. The slew rate, how fast the algorithm will move toward the MPP, will be influenced by the amount of noise in the measurement of power. The maximum slew rate for the algorithm is $\frac{\Delta X}{\Delta T}$. However, with the addition of noise to the system, wrong decisions may sometimes be made about whether to increase or decrease $X$, leading to a slower average slew rate.

For this analysis, the noise considered is Gaussian white noise that shows up on the power measurement of the array. We assume the signal representing the output power is integrated during the period between decisions, and so the standard deviation of the noise being added to each measurement of power is $\sigma_p = k/\sqrt{\Delta T}$, where $k$ is a constant with units volts$/\sqrt{Hz}$. When the system makes a decision about whether to increase or decrease $X$, it looks at the change in power from the previous step to the current step ($\Delta P$). At each point on the power vs. $X$ curve, the signal that will be seen is $m\Delta X$, where $m$ is the slope of the curve. In order for the algorithm to make the wrong decision about whether to increase or decrease $X$, the noise must have a magnitude greater than $m\Delta X$ and a sign opposite to that of the slope. Also, as the signal used, $m\Delta X$, comes from two measurements, the standard
deviation of the noise added to the signal is $\sqrt{2}\sigma_n$. The noise will have a Gaussian distribution, and the probability of an error is, based on the $Q$-function or tail-probability function,

$$P_e = \frac{1}{2} - \frac{1}{2} \text{erf}\left(\frac{\text{abs}(m)\Delta X}{2\sigma_n}\right)$$  \hspace{1cm} (1)

With this error rate, the average slew rate will be

$$S_e = (1 - 2P_e) \frac{\Delta X}{\Delta T}$$  \hspace{1cm} (2)

Fig. 1: Slew Rate vs. $\Delta X$ (adaptation step size) and frequency with noise constant, $k = 6.25 \times 10^{-5}$ and slope, $m = 0.54$.

As can be seen from Fig. 1, as the frequency of tracking decisions increases, decreasing $\Delta T$, the slew rate increases; however, the increase is not linear, as it would be without noise. It decreases in slope with increasing frequency, representing a diminishing return in slew rate from increasing the frequency of tracking. While one might think that there could be an optimal frequency for a given noise constant, $k$, there is not; the slew rate will always increase with increasing frequency, at least without taking any other constraints of the system into consideration. This is because the increased error rate due to increasing the frequency does not overcome the increase due to the $\frac{\Delta D}{\Delta T}$ term in the slew-rate equation. Also evident from Fig. 1 is that the slew rate increases with positive curvature with increasing $\Delta X$. This is due to the fact that increasing $\Delta X$ both increases the $\frac{\Delta D}{\Delta T}$ term in the equation for slew rate, as well as decreases the error rate; however, the resulting slew rate will still be less than if there were no noise.

III. SLEW RATE SIMULATIONS

To simulate the effect of noise on the slew rate, a P&O controlled PV system was modeled with ordinary differential equations and solved numerically in MATLAB; the system model is shown in Fig. 2. While the switching effects of the dc-to-dc converter are not modeled, as they are considered to be at a much higher frequency than the algorithm, the passive components of the dc-to-dc converter were included, as they affect the response of the system to a change in $D$. The body diode of the MOSFET is included in the model because it clamps the output voltage in the case that the solar panel tries to sink current.

In the model, the output current, $I_{out}$, is kept constant, representing a constant-current load. The duty cycle, $D$, of the converter thus controls the current of the PV array and is $X$ in the analysis above. With a constant-current load, $V_{out}$ is proportional to the power from the array, and so it is used as the power signal, which is why it is being integrated in the model to give an average power over each cycle. Applying the analysis to other types of loads is easily done and will just change the shape of the power vs. $D$ curve.

For fixed temperature, the model has power vs. $D$ curves shown in Fig. 3. At a fixed irradiance and string current, one can take the slope at each point along the power vs. $D$ curve and, using the calculation above, get the slew rate at each $D$ value (Fig. 4). The slew rate is zero at the MPP, and because the slope of the power curve to the right of the MPP is greater than that to the left, the slew rate is greater to the right of the MPP.

With this curve, one can estimate how long it will take for the P&O controller to get from one $D$ value to another by computing the integral:

$$\int_{D_{start}}^{D_{end}} \frac{dD}{SlewRate(D)}$$  \hspace{1cm} (3)

Noise was added to the values of the voltage used for making decisions in the model, and the estimate from the analysis for how long it will take to move from one $D$ value to another is very close to that seen in the model. For example, Fig. 5 shows the $D$ values of multiple runs of the simulation. The analysis predicts that for the amount of noise added to this system, it should take 7.02 ms to go from $D = 0$ to $D = 0.53$, which is almost exactly what is seen in the model.

IV. NOISE EFFECT ON STEADY-STATE EFFICIENCY OF PERTURB AND OBSERVE

While it is important to know how the slew rate is affected by noise, what is of greater importance is how
noise affects the efficiency of the tracking at steady state. Efficiency, here, means the difference in output power compared to if one were to operate perfectly at the MPP.

Around the MPP, the slope of the power vs. $D$ curve is close to zero, and so, with just a little noise, the signal there is almost completely lost, resulting in a near-random walk, which will lead to some loss in efficiency. In fact, the entire P&O algorithm can be defined as a semi-random walk, where at each $D$ value, there is a probability of taking a step to the right or to the left. Using a similar analysis as above, the probability of going to the left is

$$
P_{\text{left}} = \frac{1}{2} - \frac{1}{2} \erf\left(\frac{m\Delta X}{2\sigma_n}\right),$$

and the probability of going to the right is $P_{\text{right}} = 1 - P_{\text{left}}$, except for $D = 0$, where the probability of going right is one; and $D = 1$, where the probability of going left is one.

Using the transition probabilities given in (4) the P&O method can be put in the form of a Markov chain [13] with an ergodic transition matrix

$$
P = \begin{bmatrix}
0 & 1 & 0 & 0 & \cdots & \cdots & 0 \\
p_{11} & p_{12} & p_{13} & \cdots & \cdots & \cdots & 0 \\
p_{21} & 0 & p_{23} & \cdots & \cdots & \cdots & 0 \\
p_{31} & 0 & 0 & \cdots & \cdots & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\
p_{n1} & 0 & 0 & \cdots & 0 & p_{nn} & 0 \\
0 & \cdots & \cdots & \cdots & \cdots & 0 & 1 \\
0 & \cdots & \cdots & \cdots & \cdots & 0 & 0
\end{bmatrix},$$

where the probabilities of going left or right at each $D$ value fill the diagonals around the central zero diagonal, with each row summing to 1.

Using this transition matrix along with a starting row vector $u_k$, representing the percent chance of being at each of the $D$ values at time step $k$, one can determine the probability, $u_{k+1}$, of being at each of the possible $D$ values at the next time step by $u_{k+1} = u_k P$. Then, the probability distribution at time step $k + n$ is $u_{k+n} = u_k P^n$. To get the steady state solution, one can use a large value of $n$, but because one does not know how large a value is sufficient, it is better to solve the equation $u\omega = u_0 P$ for $u_\omega$, which will be the steady-state distribution and is the left eigenvector of $P$ corresponding to the eigenvalue of 1. So, for each set of parameter values (such as noise, $\Delta D$, and frequency), one can determine the probability distribution function (pdf) at steady state and then calculate the efficiency,

$$
\eta = \frac{\sum_D \text{pdf}(D) \cdot \text{Power}(D)}{\text{Power}(D_{\text{opt}})}.
$$

Example pdfs are shown in Fig. 6.

The effects of frequency and $\Delta D$ on efficiency can be seen in Fig. 7(a). The efficiency goes up as frequency decreases, as lowering the frequency effectively decreases the noise. Decreasing $\Delta D$ also increases the efficiency, and
this is due to the oscillations around the MPP increasing as \( \Delta D \) increases. However, the efficiency improvements due to decreasing \( \Delta D \) are asymptotic, approaching a maximum possible efficiency for the given frequency. Making \( \Delta D \) smaller and smaller will yield smaller and smaller improvements in the efficiency while greatly decreasing the reaction rate of the controller. This leads to the usefulness of Fig. 7(b). Given a desired nominal slew rate, one can choose the optimal frequency and step size for greatest efficiency at steady state.

V. PERTURB AND OBSERVE WITH ADDED FORCED OSCILLATION

One enhancement that can be made to the perturb and observe algorithm is to add a constant oscillation on top of the normal change in \( D \), as this will improve the signal to noise ratio, and, as will be shown here, is able to achieve higher steady state efficiencies for the same slew rate.

The update equation for normal perturb and observe is

\[
X[k + 1] = X[k] + \Delta X \text{sgn} \left( \frac{P[k] - P[k - 1]}{X[k] - X[k - 1]} \right),
\]

(6)

where \( P \) is the output power of the system. Adding the forced oscillation results in the update equation becoming

\[
X[k + 1] = X[k] + \Delta X_p (-1)^k + \Delta X \text{sgn} \left( \frac{P[k] - P[k - 1]}{X[k] - X[k - 1]} \right),
\]

(7)

where \( \Delta X_p \) is the size of the added oscillation or perturbation. This method simply adds a square wave perturbation on top of the normal perturb and observe algorithm. With this added perturbation, the probability of going left becomes

\[
P_{\text{left}} = \frac{1}{2} - \frac{1}{2} e^{\frac{m\Delta X_p}{2\sigma_n}},
\]

(8)

with \( P_{\text{right}} = 1 - P_{\text{left}} \). This is a good approximation as long as \( \Delta X_p \) is much larger than \( \Delta X \), so that the perturbation in \( X \) is dominated by \( \Delta X_p \). When \( \Delta X_p \) and \( \Delta X \) are close, a better approximation is to average the transition probabilities for the step sizes \( \Delta X_p \pm \Delta X \), as those are the actual step sizes that will be taken. Fig. 8 was produced using these probabilities in the same way Fig. 7(a) was produced through the use of Markov transition matrices. It shows how the steady state efficiency depends on both \( \Delta X \) and \( \Delta X_p \) for a fixed slew rate. Again, for the system considered here, \( X = D \), the duty cycle of the converter. For small values of \( \Delta D_p \), the algorithm essentially returns to being the simple perturb and observe, where there is an optimal \( \Delta D \) value at which the trade-off between losses due to large oscillations around the MPP from using a large \( \Delta D \) and losses due to wandering around the MPP from using a small \( \Delta D \) are balanced to give the best steady state efficiency possible for the given slew rate and noise. It is clear from Fig. 8 that the optimal value of \( \Delta D \) for the perturb and observe with forced

![Diagram](image_url)
oscillation algorithm is the smallest feasible value; the limit on this is the highest frequency at which one can run the control algorithm, as decreasing $\Delta D$ means increasing the frequency in order to keep the slew rate constant.

The reason that a small value of $\Delta D$ is optimal can be explained as follows. Consider the equation for the probability of making a wrong decision, (1), which, with the substitution of $\sigma_n = k/\sqrt{\Delta T}$, becomes

$$P_e = \frac{1}{2} - \frac{1}{2} \text{erf} \left( \frac{\text{abs}(m)\Delta X \sqrt{\Delta T}}{2k} \right).$$  \hfill (9)

As the argument of the error function in (9) increases, the probability of making a wrong decision goes to zero. So, as one increases $\Delta X = \Delta D$, the chance of making a wrong decision goes down, leading to more accurate tracking. However, as can be seen from Fig. 7(a), this increase in accuracy is outweighed by the decrease in efficiency due to larger oscillations around the MPP, leading to increased efficiency as $\Delta X$ is decreased. So, with the argument of the error function in (9) proportional to $\Delta X$, the efficiency asymptotically approaches the maximum efficiency as $\Delta X$ is decreased, which is evident in Fig. 7(a).

Now consider holding the nominal slew rate fixed, as a certain slew rate is usually desired, and adjusting $\Delta X$. To maintain a fixed slew rate, we need to also adjust $\Delta T$, based on $\Delta T = \Delta X/S_r$. Thus, in (9), $\Delta T$ changes as well as $\Delta X$ when we change slew rate. To see the effect we can write (9) in terms of $\Delta X$ with fixed slew rate as

$$P_e = \frac{1}{2} - \frac{1}{2} \text{erf} \left( \frac{\text{abs}(m)\Delta X^{3/2}}{2k \sqrt{S_r}} \right).$$  \hfill (10)

In (10), the argument of the error function is proportional to $\Delta X^{3/2}$, and so with a stronger dependence on $\Delta X$, the noise more rapidly becomes worse as $\Delta X$ is decreased. This strong effect outweighs the benefits of decreasing $\Delta X$ and leads to there being an optimum value of $\Delta X$ (or $\Delta D$) for a fixed slew rate. This can be seen in Fig. 8 by looking at the front, right edge where $\Delta D_p$ is equal to zero, as this is the curve representing the simple perturb and observe algorithm.

For the perturb and observe with forced oscillation, equation (9) becomes

$$P_e = \frac{1}{2} - \frac{1}{2} \text{erf} \left( \frac{\text{abs}(m)\Delta X_{p} \Delta X^{1/2}}{2k \sqrt{S_r}} \right).$$  \hfill (11)

In (11), the argument of the error function is proportional to $\Delta X^{1/2}$. This is a weaker dependence on $\Delta X$ than in (9), so the increase in efficiency due to smaller oscillations around the MPP as one decreases $\Delta X$ will outweigh the decrease in efficiency from less accurate tracking. That is why the maximum efficiency continues to increase as one decreases $\Delta D$ in Fig. 8.

Because of the trends seen in Fig. 8, for the perturb and observe with forced oscillation algorithm, one should use the smallest $\Delta D$ that the system can handle, given the desired slew rate, $S_r$, and maximum frequency, $f_{\text{max}}$. The smallest $\Delta D$ one can use is then

$$\Delta D = \frac{S_r}{f_{\text{max}}}. \hfill (12)$$

Then, one must choose the optimal value of $\Delta D_p$ which balances the losses due to large oscillations and wandering.
This is done using the analysis presented above. For each value of $\Delta D_p$ of interest, one uses equation (8) to create the transition matrix and invert it to get the steady state pdf, from which the steady state efficiency can be found using (5). This will result in a plot of one of the lines of constant $\Delta D$ in Fig. 8 from which one can determine the optimum $\Delta D_p$.

Alternatively, it is possible to incorporate the limits on frequency into the analysis to find the optimum $\Delta D$ and $\Delta D_p$. For example, depending on the passive components of the converter in the system, see Fig. 2, the output voltage will ring before settling after each change in $D$. So, one should wait until the system settles before starting to integrate the output voltage. Adding in this blanking time means that as one increases the frequency of tracking, the actual integration time, $\Delta T_f$, will approach zero faster than the simplified case where $\Delta T = 1/f$. This results in a global maximum, which can be seen in Fig. 10.

Besides providing a method for determining optimum system parameters, there are many other conclusions that can be drawn from Fig. 8. For one, there is a minimum value of $\Delta D$ for which adding a forced oscillation can improve the efficiency; above that value, the added perturbations only cause more oscillations around the MPP without improving the accuracy of the tracking enough to make it worthwhile. Also, the optimum size of the added perturbation, $\Delta D_p$, is always larger than $\Delta D$, if improvement is possible. In fact, any added perturbation less than or equal to $\Delta D$ will degrade the efficiency of a normal perturb and observe method, as expected.

Finally, the improvement in efficiency expected from the perturb and observe method with forced oscillation over the simple perturb and observe can be seen by looking at the difference between the maximum along the left edge of Fig. 8, corresponding to the minimum value of $\Delta D$, and the maximum along $\Delta D_p = 0$ in Fig. 8. This difference will depend upon the noise and slew rate for the system. Fig. 11 shows the maximum efficiency of the simple perturb and observe and perturb and observe with forced oscillation methods versus slew rate. As is evident in Fig. 11, as the slew rate or noise increase (because both enter into equation (11) in the same place), the benefit from using the forced oscillation version of the perturb and observe method increases, as expected.

VI. SIMULATION RESULTS FOR STEADY STATE EFFICIENCY

Even though the analysis presented in this paper is fairly straightforward, simulations were done in order to validate the analytical results. While the MATLAB model presented earlier provides a fairly realistic simulation, it takes a long time to run, making it non-ideal for doing steady state analysis. However, as the dynamics of the modeled PV system are fast enough that they have little to no effect on the algorithms’ performance, simulating the algorithms with update equations is much faster. For these simulations, the same PV cell model was used as in the full model; however, for each new value of $D$, the steady state output voltage of the system was calculated. Then, before using the given algorithm to determine the next $D$ value, random noise with variance $\sigma_n^2 = k^2/\Delta T$ was added to the calculated voltage. After running the simulation sufficiently long, the average output power was compared to the maximum output power of the cell in order to determine the efficiency of the algorithm.

In order to make sure that this simplified model matched the full model closely enough, the same simulation was run for each model. The parameters for the simulation were a $\Delta D$ of 0.005, $\Delta T$ of 33.3$\mu$s, and a noise constant of $1.25 \times 10^{-5} \sqrt{\text{Hz}}$. The simplified, state update model resulted in an efficiency of 99.6159 percent, and the full model, which took considerably longer to run, resulted in an efficiency of 99.6171 percent. These values were close enough to decide that the simplified model was a good approximation to the full model.

The simplified model was then run for the same range of $\Delta D$ and $\Delta D_p$ in Fig. 8 in order to produce Fig. 12, which matches Fig. 8 well. Most importantly, the shape of the two figures is very similar, meaning that the trends in efficiency vs. $\Delta D$ and $\Delta D_p$ found in the analysis also show up in the simulations. Also, the optimum $\Delta D_p$ values for the analytical model and simulation are similar at 0.0073 and 0.0083 respectively, and the optimum $\Delta D$ values were reasonably close at 0.009 and 0.005 respectively. Finally, the efficiencies of the analytical model and simulation were quite close. As can be seen in Fig. 8 and Fig. 12, the contour line closest to the optimum for the simple perturb and observe is 99.711% for the analytical model and 99.722% for the simulation. Similarly, the contour line closest to the
optimization for the perturb and observe with forced oscillation algorithm is 99.817% for the analytical model and 99.798% for the simulation.

Fig. 12: Simulated Efficiency vs. $\Delta D$ and $\Delta D_p$ for $k = 1.25 \times 10^{-4} \frac{V}{\sqrt{Hz}}$ and a Slew Rate of 1 Hz. This matches well with Fig. 8, though it is more jagged due to a less dense evaluation grid.

VII. COMPARISON TO LINEAR MPPT SYSTEMS

In [12] the linear, continuous-time MPPT system shown in Fig. 13 along with its discrete-time counterpart were optimized in the presence of noise in the power measurement. They are considered linear because the steps in $D$ are proportional to the slope of the power vs. $D$ curve at the current operating point. In an approach similar to that which was taken here, the optimum perturbation magnitude, $d$ in Fig. 13, was found for this system for which the sum of the losses due to moving around the MPP due to the perturbation and due to noise were minimized. In [12], a linearized model for the system is presented, which allows one to use linear analysis techniques, both continuous-time and discrete-time, to calculate the effect of both the noise and perturbation on the system. Ultimately, it was found that the optimum perturbation size made the losses due to the noise and perturbation equal.

The following is a summary of the comparison of the systems analyzed here and those analyzed in [12]; for the full comparison and explanation see [12].

![MPPT System](image)

Fig. 13: MPPT System

In order to compare the linear systems to the two perturb and observe systems analyzed here, the reaction rates of the systems need to be normalized in some way. This is difficult, as the perturb and observe systems have a slew rate; whereas, the linear systems have a time constant. A simplistic way of doing this, which is explained in more detail in [12] is to set the slew rate equal to $S_r = \Delta D/\Delta T_{P&O} = cG/\Delta T$, where $c$ is an experimentally found constant. Using this normalization, the minimum power loss for the different systems for a slew rate of $1/6$ and noise constant of $k = P_n = 6.25 \times 10^{-4} \frac{1}{\sqrt{Hz}}$ was found and is shown in Fig. 14. While in Fig. 14 the continuous-time system has half the power loss of the perturb and observe with forced oscillation system, the approximations in the power loss for the continuous-time system are further from reality, as was found in simulation, and lead to a higher estimated efficiency than is actually achieved, meaning the perturb and observe with forced oscillation may be able to achieve a higher efficiency. Overall, all four systems, if optimized, can perform maximum power point tracking at a very high efficiency, meaning one can use other factors, such as cost, complexity, hardware needed, etc. to determine what algorithm best fits their system.

![Percent Power Loss](image)

Fig. 14: Percent power loss for the different algorithms with $k = P_n = 6.25 \times 10^{-4} \frac{1}{\sqrt{Hz}}$.

VIII. CONCLUSION

While it is well known that noise hurts the overall performance of MPPT systems, its impact is rarely quantified. The analysis in this paper quantifies the effect of noise on two MPPT algorithms in slowing down tracking, as well as in degrading steady state efficiency and compares their performance to algorithms analyzed in [12]. Dynamic simulation results have verified the analytical predictions. The analysis is useful for making informed decisions about algorithm and parameter choices and can aid development of improved algorithms. The end result will be more efficient designs and better performance.
References


