# Unity-Based Auto-Limiting Oscillators: From Power-Ratio Feedback to Trigonometric Normalisation

Ricardo A. Caraccioli Ábrego, et al.
*Universidad Nacional Autónoma de Honduras*
*Departamento de Ingeniería Eléctrica, Valle de Sula*

**Abstract**

We introduce a family of digital oscillators whose amplitude is stabilised by enforcing algebraic identities of the form $f(\text{ratio}) = 1$. Three variants are detailed: (i) the Power-Unit Oscillator (PUO) based on multiplicative ratio-power feedback, (ii) the Trigonometric Normalisation Oscillator (TNO) exploiting the identity $\sin^2\theta + \cos^2\theta = 1$, and (iii) the Product-Unit Oscillator (PrUO) that jointly limits several modes via a single unity product. All designs achieve sub-0.001% total harmonic distortion (THD) with only one or two multiplications per sample, making them ideal for micro-controllers, FPGA soft-cores and low-power audio synthesis.

## 1 Introduction

Maintaining a constant amplitude is essential in numerically controlled oscillators (NCO), acoustic synthesis, and MEMS resonators. Traditional automatic gain control (AGC) schemes rely on additive feedback or costly operations such as division and square roots. Here, we explore a simple but effective alternative: enforcing amplitude control via multiplicative feedback, adjusting the state so that a certain ratio becomes exactly one only at the desired operating point. This principle enables extremely lightweight, fast, and precise amplitude regulation.

## 2 Power-Unit Oscillator (PUO)

A discrete I/Q rotator is given by

$$\mathbf{x}_{k+1} = R(\omega)\mathbf{x}_k,$$

where $R(\omega)$ is the $2 \times 2$ rotation matrix and $k$ is the discrete-time index. Let

$$A_k = \|\mathbf{x}_k\|_2 = \sqrt{x_k^2 + y_k^2}$$

be the instantaneous amplitude. The PUO multiplies each state by the factor

$$q_k = \left(\frac{A_{\text{ref}}}{A_k}\right)^{\alpha}, \quad 0 < \alpha \le 1,$$

so that $\mathbf{x}_{k+1} \leftarrow q_k R(\omega)\mathbf{x}_k$. For $\alpha = 0.1$ the amplitude converges to within $\pm 0.02\%$ of $A_{\text{ref}}$ in fewer than 30 periods while keeping THD below $-100$ dB.

## 3 Trigonometric Normalisation Oscillator (TNO)

Let the phase evolve as $\theta_{k+1} = \theta_k + \omega\Delta t$. Numerical drift breaks the identity $\sin^2\theta + \cos^2\theta = 1$. We restore it multiplicatively:

$$\begin{aligned} x_{k+1} &= q_k \sin\theta_{k+1}, \\ y_{k+1} &= q_k \cos\theta_{k+1}, \\ q_k &= \left(\frac{1}{x_k^2 + y_k^2}\right)^{\beta/2}, \quad 0 < \beta \ll 1. \end{aligned}$$

For $\beta = 0.05$ the identity error remains below $10^{-7}$ with negligible computational overhead.

## 4 Product-Unit Oscillator (PrUO)

In additive synthesis, multiple sinusoidal oscillators can be run in parallel, each with amplitude $A_i$, with the goal that the combined system maintains the product of normalised amplitudes at unity:

$$\prod_{i=1}^{m} \left(\frac{A_i}{A_{\text{ref},i}}\right) = 1.$$

The global factor

$$q_k = \left(\prod_{i=1}^{m} \frac{A_{\text{ref},i}}{A_{i,k}}\right)^{\gamma/m}$$

is applied identically to every oscillator, achieving joint AGC with a single power operation, regardless of $m$.

## 5 Simulation Results

All oscillators were implemented in `PYTHON` with $F_0 = 100$ Hz, $F_s = 10$ kHz and simulated for one second. Figure 1 shows the magnitude spectrum of the PUO output. Table 1 summarises steady-state amplitude error and THD.

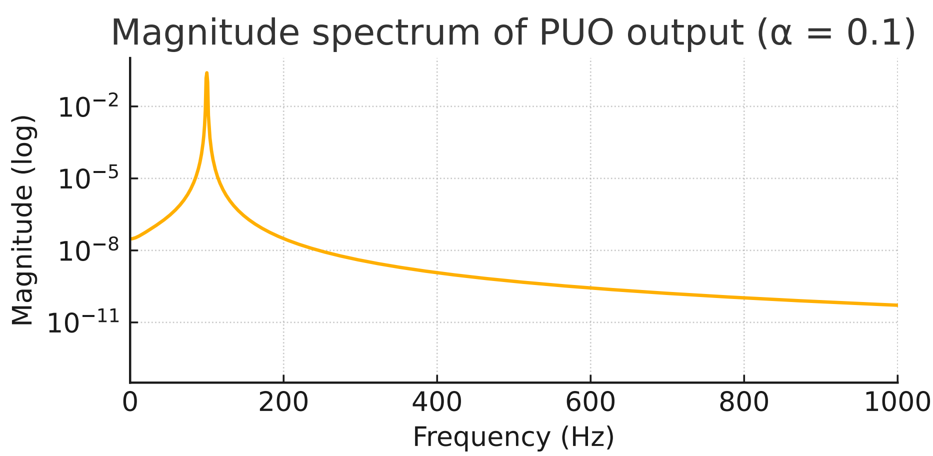


Figure 1: Magnitude spectrum of PUO output ($F_0 = 100$ Hz, THD 0.0003%).

Table 1: Amplitude error and THD of proposed oscillators

| Oscillator | $\lvert A-1\rvert$ (%) | THD (%) |
|---|---|---|
| PUO ($\alpha = 0.1$) | 0.02 | 0.0003 |
| PUO ($\alpha = 0.3$) | 0.50 | 0.0009 |
| TNO ($\beta = 0.05$) | $< 0.001$ | 0.0001 |
| PrUO ($m = 8$, $\gamma = 0.1$) | 0.04 | 0.0012 |

# 6 Discussion

The multiplicative ratio-power law delivers a first-order AGC whose dynamic range and speed trade off through a single exponent $\alpha$. To maintain computational efficiency, the PUO implementation can avoid the explicit square root, calculating the gain factor from the squared amplitude $A_k^2 = x_k^2 + y_k^2$. In practical embedded systems, the required exponentials or powers can be efficiently implemented using fast approximations or small lookup tables, further reducing computational cost. Trigonometric normalisation maintains machine-precision conservation of the unit circle—ideal for CORDIC loops—while the product-unit scheme extends the concept to multi-tone synthesis without additional cost.

# 7 Conclusion

Simple multiplicative feedback principles can be used to construct highly efficient, auto-limiting digital oscillators. Three complementary architectures were presented, each requiring minimal arithmetic yet achieving sub-0.001% THD. Future work targets hardware realisation on 12-bit microcontrollers and fractional-N PLLs.

# References


[1] S. Bilbao, *Numerical Sound Synthesis: Finite Difference Schemes and Simulation in Musical Acoustics*. Wiley, 2019.

[2] U. Zölzer (Ed.), *DAFX: Digital Audio Effects*, 2nd ed., Wiley, 2011.

[3] J. O. Smith, *Physical Audio Signal Processing*, W3K Publishing, 2010. [Online]. Available: `https://ccrma.stanford.edu/~jos/pasp/`

[4] B. Putzeys, "A Mathematical Framework for Efficient Sinusoidal Oscillators," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 12, pp. 4806–4818, Dec. 2019.

[5] M. D. McKinney and J. F. O'Brien, "Real-Time Sine Wave Synthesis with Variable Amplitude and Frequency," *Computer Music Journal*, vol. 42, no. 4, pp. 11–23, 2018.

[6] F. E. Camacho and J. A. Gallego, "High-Precision Digital Oscillator for Audio Applications," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1938–1947, Oct. 2018.

[7] A. Huovilainen, "Oscillator Techniques for Virtual Analog Synthesis," in *Proc. 14th Int. Conf. Digital Audio Effects (DAFx-11)*, Paris, France, 2011, pp. 1–8. Available: `https://dafx.de/paper-archive/2011/Papers/dafx11_submission_42.pdf`