Winter Research Result 2024-2025

Jonathan Huang (Shao-Kai Huang)
Physics Department, National Taiwan University
IMB, Academia Sinica

February 13, 2025

1 Improved Effective Bounds For SAPs

Definition 1.1. *Define the* $(n-1)$*-order polynomial* P_n *over the positive reals:*

$$P_n(s) \equiv \left(1+\frac{s}{a_1}\right)\left(1+\frac{s}{a_2}\right)\cdots\left(1+\frac{s}{a_{n-1}}\right) = \prod_{k=1}^{n-1}\left(1+\frac{s}{a_k}\right). \tag{1}$$

1.1 The $n = 3$ Case

The system is described by the figure ??. The long term growth rate λ satisfies the equation

$$\lambda = A(\lambda) = b\left(\frac{a_1}{a_1+\lambda}\right)\left(\frac{a_2}{a_2+\lambda}\right).$$

This is simply $b/\lambda = P_3(\lambda)$, and finding a solution is equivalent to solving for the largest (modulus-wise) solution of the cubic

$$\lambda^3 + (a_1+a_2)\lambda^2 + a_1a_2\lambda - a_1a_2b = 0.$$

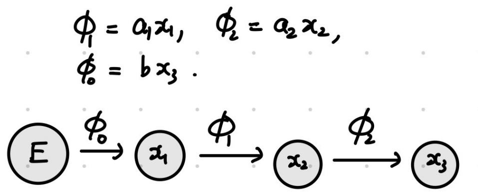

Figure 1: Example of a LRN with three nodes ($n = 3$).

The solution is

$$\lambda_0 = \frac{1}{3\sqrt[3]{2}}\left(\Delta+\sqrt{\Delta^2+4B^3}\right)^{1/3} - \frac{\sqrt[3]{2}}{3}\frac{B}{\left(\Delta+\sqrt{\Delta^2+4B^3}\right)^{1/3}} - \frac{1}{3}(a_1+a_2) \in \mathbb{R}, \tag{2a}$$

$$\begin{aligned}\lambda_\pm = &-\frac{1}{6\sqrt[3]{2}}\left(1\mp i\sqrt{3}\right)\left(\Delta+\sqrt{\Delta^2+4B^3}\right)^{1/3} \\ &+\frac{\sqrt[3]{2}}{6}\left(1\pm i\sqrt{3}\right)\frac{B}{\left(\Delta+\sqrt{\Delta^2+4B^3}\right)^{1/3}} - \frac{1}{3}(a_1+a_2) \in \mathbb{C},\end{aligned} \tag{2b}$$

where

$$\Delta \equiv (a_1+a_2)^3 - 3(a_1^3+a_2^3) + 27a_1a_2b, \quad B \equiv 3a_1a_2 - (a_1+a_2)^2.$$

Since the desired solution must be real, the growth rate is given by λ_0.

1.2 Effective Bounds For SAPs

The current upper and lower bounds for an effective one-step chain are (Lin)

$$a_L^{-1} = (A(b) - 1)/b, \tag{3a}$$

$$a_U^{-1} = a_1^{-1} + \cdots + a_{n-1}^{-1}. \tag{3b}$$

For the following paragraph, refer to figure ??. Notice that $P_n'(0) = \frac{1}{a_U}$, so the current bound yields the optimal upper bound for λ_U, if we require that $P_{\mathrm{U}}(x), P_{\mathrm{n}}(x)$ for all $x > 0$. However, this needs not be true, since we can introduce a cutoff $c \in (0, \lambda)$ to make the bound stricter, such that. $P_U(x) < P_n(x)$ only when $c < x < \lambda$.

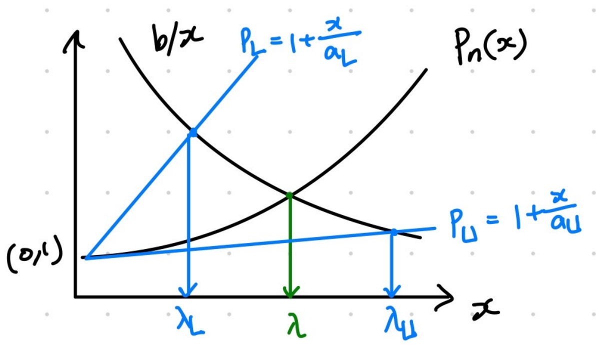

Figure 2: Geometric interpretation of λ in the case $n = 3$.

Here I propose a simple algorithm to derive stricter bounds by means of iteration. We continue to use the geometric interpretation of λ as the intersection of the polynomial $P_n(x)$ and the hyperbola b/x in our analysis.

For the $n = 2$ case, we can solve explicitly for λ_U using the equation for a one-step chain:

$$\frac{b}{\lambda_U} = a + \frac{\lambda_U}{a_U} \implies \lambda_U = \frac{-1 + \sqrt{1 + 4b/a_U}}{2/a_U}.$$

A similar procedure can be done for λ_L, but this is not important for the following analysis. Define $\lambda_U^{(0)} = \lambda_U$ and consider the following lines

$$\begin{cases} U^{(0)}: \quad 1 + m_U^{(0)} x, \quad m_U^{(0)} = \dfrac{1}{a_U} = \dfrac{1}{a_1} + \cdots + \dfrac{1}{a_{n-1}}, \\ L^{(1)}: \quad 1 + m_L^{(1)} x, \quad m_L^{(1)} = \dfrac{P_n\left(\lambda_U^{(0)}\right) - 1}{\lambda_U^{(0)}}, \\ U^{(1)}: \quad 1 + m_U^{(1)} x, \quad m_U^{(1)} = \dfrac{P_n\left(\lambda_L^{(1)}\right) - 1}{\lambda_L^{(1)}}. \end{cases}$$

Going from the first equation to the second, I solved for $\lambda_L^{(1)}$ again using the solution for the $n = 2$ case:

$$\lambda_L^{(1)} = \frac{-1 + \sqrt{1 + 4bm_L^{(1)}}}{2m_L^{(1)}}, \quad m_L^{(1)} = \frac{P_n\left(\lambda_U^{(0)}\right) - 1}{\lambda_U^{(0)}}.$$

This is demonstrated in figure ??. Geometrically it is interpreted as extending the line $x = \lambda_U^{(0)}$ upwards to intersect with $y = P_n(x)$, and using that point and $(0,1)$ to construct $L^{(1)}$. Similarly, going from the second equation to the third requires solving for

$$\lambda_U^{(1)} = \frac{-1+\sqrt{1+4bm_U^{(1)}}}{2m_U^{(1)}}, \quad m_U^{(1)} = \frac{P_n\left(\lambda_L^{(1)}\right)-1}{\lambda_L^{(1)}}.$$

This is interpreted as using $c = \lambda_L^{(1)}$ as the cutoff point for $U^{(1)}$, which is key to making the iteration scheme work.

Using the equation of the lines $L^{(1)}$ and $U^{(1)}$, we can derive the effective coefficient $a_U^{(n)}$ and $a_L^{(n)}$ by noticing that

$$a_L^{(n)} = \frac{1}{m_L^{(n)}}, \quad a_U^{(n)} = \frac{1}{m_U^{(n)}}. \tag{4}$$

This process can be continued indefinitely, with the upper and lower bounds closing in on the true value λ. The main result is summarized as follows:

$$m_L^{(j)} = \frac{P\left(\lambda_U^{(j-1)}\right)-1}{\lambda_U^{(j-1)}}, \quad a_L^{(j)} = \frac{1}{m_L^{(j)}}, \quad \lambda_L^{(j)} = \frac{-1+\sqrt{1+4bm_L^{(j)}}}{2m_L^{(j)}}, \tag{5a}$$

$$m_U^{(j)} = \frac{P\left(\lambda_L^{(j)}\right)-1}{\lambda_L^{(j)}}, \quad a_U^{(j)} = \frac{1}{m_U^{(j)}}, \quad \lambda_U^{(j)} = \frac{-1+\sqrt{1+4bm_U^{(j)}}}{2m_U^{(j)}}, \tag{5b}$$

$$\lambda_U^{(0)} = \frac{-1+\sqrt{1+4b/a_U}}{2/a_U}, \quad j = 1,2,3,\dots. \tag{5c}$$

This process is illustrated in figure ??, where the pink lines represent the original bounds.

To decouple the recurrence relations, notice that the map $\eta : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ defined by

$$x \mapsto \frac{1}{2}\left(-1+\sqrt{1+4b\left[\frac{P_n\left(\left(-1+\sqrt{1+4b\left[P_n(x)-1\right]/x}\right)/(\left[P_n(x)-1\right]/x)\right)-1}{\left(-1+\sqrt{1+4b\left[P_n(x)-1\right]/x}\right)/(\left[P_n(x)-1\right]/x)}\right]}\right) \times \left[\frac{P_n\left(\left(-1+\sqrt{1+4b\left[P_n(x)-1\right]/x}\right)/(\left[P_n(x)-1\right]/x)\right)-1}{\left(-1+\sqrt{1+4b\left[P_n(x)-1\right]/x}\right)/(\left[P_n(x)-1\right]/x)}\right]^{-1}, \tag{6}$$

where

$$P_n(x) = \prod_{p=1}^{n-1}\left(1+\frac{x}{a_p}\right), \tag{7}$$

maps $\lambda_L^{(j)}$ to $\lambda_L^{(j+1)}$ and $\lambda_U^{(j)}$ to $\lambda_U^{(j+1)}$. Therefore, the equation (5) can be recast in the following form:

$$\lambda_L^{(j+1)} = \eta\left(\lambda_L^{(j)}\right), \quad \lambda_U^{(j+1)} = \eta\left(\lambda_U^{(j)}\right), \quad j = 1,2,\dots. \tag{8}$$

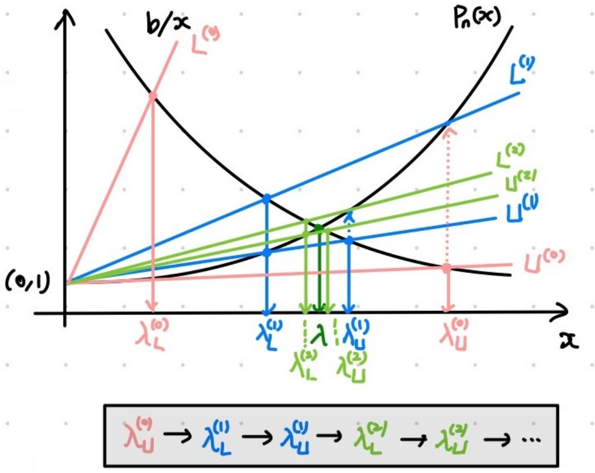

Figure 3: Illustration of the iteration shceme. Each color represents a different order of approximation, higher order approximations bound the true value λ more tightly.

1.3 Example Calculations

Example 1.1. We apply the above bounds to a system with $n = 5$, where $a_1 = 2, a_2 = 4, a_3 = 5, a_4 = 3, b = 6$, and compare them with the original bounds a_U and a_L. The exact solution according to WolframAlpha is $\lambda = 1.3966432600825643010$, and the following python code gives the order one to order three bounds according to my theory:

```
import numpy as np

# n = 5
a_1, a_2, a_3, a_4, b = 2, 4, 5, 3, 6
arr = [a_1, a_2, a_3, a_4, b]
sumarray = 1/a_1 + 1/a_2 + 1/a_3 + 1/a_4
exact = 1.3966432600825643010

def P(s): # defines the polynomial P(s)
    return (1 + s/a_1)*(1 + s/a_2)*(1 + s/a_3)*(1 + s/a_4)

# lambda 0
mu0 = sumarray
lu0 = ( -1 + np.sqrt(1 + 4*b*mu0)) / (2*mu0)

# In the following code, "m" is the reciprocal of the corresponding "a" coefficients
# lambda 1
ml1 = ( P(lu0) - 1 ) / lu0
ll1 = (-1 + np.sqrt(1 + 4*b*ml1)) / (2*ml1)
mu1 = ( P(ll1) - 1 ) / ll1
lu1 = (-1 + np.sqrt(1 + 4*b*mu1)) / (2*mu1)
```

```
# lambda 2
ml2 = ( P(lu1) - 1 ) / lu1
ll2 = (-1 + np.sqrt(1 + 4*b*ml2)) / (2*ml2)
mu2 = ( P(ll2) - 1 ) / ll2
lu2 = (-1 + np.sqrt(1 + 4*b*mu2))/ (2*mu2)

# lambda 3
ml3 = ( P(lu2) - 1 ) / lu2
ll3 = (-1 + np.sqrt(1 + 4*b*ml3)) / (2*ml3)
mu3 = ( P(ll3) - 1 ) / ll3
lu3 = (-1 + np.sqrt(1 + 4*b*mu3))/ (2*mu3)

# lambda 4
ml4 = ( P(lu3) - 1 ) / lu3
ll4 = (-1 + np.sqrt(1 + 4*b*ml4)) / (2*ml4)
mu4 = ( P(ll4) - 1 ) / ll4
lu4 = (-1 + np.sqrt(1 + 4*b*mu4))/ (2*mu4)

# print out the result
print("the 1st order lower bound is ", ll1)
print("the 2nd order lower bound is ", ll2)
print("the 3rd order lower bound is ", ll3)
print("the 4th order lower bound is ", ll4)
print("the exact solution is        ", exact)
print("the 4th order upper bound is ", lu4)
print("the 3rd order upper bound is ", lu3)
print("the 2nd order upper bound is ", lu2)
print("the 1st order upper bound is ", lu1)
print("the 0th order upper bound is ", lu0)

print("\nThe corresponding a_U's are: ")
print("a_U =    ", 1 / mu0)
print("a^(1)_U =", 1 / mu1)
print("a^(2)_U =", 1 / mu2)
print("a^(3)_U =", 1 / mu3)
print("a^(4)_U =", 1 / mu4)

print("\nThe corresponding a_L's are: ")
print("a^(1)_L =", 1 / ml1)
print("a^(2)_L =", 1 / ml2)
print("a^(3)_L =", 1 / ml3)
print("a^(4)_L =", 1 / ml4)
```

The result of computation is summarized:

```
the 1st order lower bound is  1.299583559601074
the 2nd order lower bound is  1.3906073366962926
the 3rd order lower bound is  1.3962709994738929
the exact solution is         1.3966432600825642
the 3rd order upper bound is  1.396735687054176
the 2nd order upper bound is  1.3981429249149382
the 1st order upper bound is  1.421026861073852
the 0th order upper bound is  1.8074605994605224
```

Figure 4: Calculation result for $n = 5$, bounds on λ.

```
The corresponding a_U's are:
a_U =     0.7792207792207793
a^(1)_U = 0.440997856643241
a^(2)_U = 0.4247858215921593
a^(3)_U = 0.42380155621397647
a^(4)_U = 0.4237409425602482

The corresponding a_L's are:
a^(1)_L = 0.3593122970700142
a^(2)_L = 0.4195322260715242
a^(3)_L = 0.42347686055131173
a^(4)_L = 0.4237209249695411
```

Figure 5: Calculation result for $n = 5$, effective coefficients.

1.4 Convergence

From construction we make the following observation: for all $n \geq 1$, we have

$$\lambda \leq \lambda_U^{(n+1)} \leq \lambda_U^{(n)}, \tag{9a}$$

$$\lambda_L^{(n)} \leq \lambda_L^{(n+1)} \leq \lambda, \tag{9b}$$

$$a_U \geq a_U^{(n)} \geq a_U^{(n+1)}, \tag{9c}$$

$$a_L^{(1)} \leq a_L^{(n)} \leq a_L^{(n+1)}. \tag{9d}$$

In particular, notice that $\left(\lambda_U^{(n)}\right)_{n\in\mathbb{N}}$ is a decreasing sequence, while $\left(\lambda_L^{(n)}\right)_{n\in\mathbb{N}}$ is an increasing sequence.

Theorem 1.1 (Monotone Convergence Theorem)**.** *If $(a_n)_{n\in\mathbb{N}}$ is a monotone sequence of real numbers, then this sequence has a finite limit if and only if the sequence is bounded. In particular, if the sequence is increasing, then it converges to its supremum; if the sequence is decreasing, then it converges to its infimum.*

By the above theorem, we know that each of the sequences $\left(\lambda_U^{(n)}\right)_{n\in\mathbb{N}}$ and $\left(\lambda_L^{(n)}\right)_{n\in\mathbb{N}}$ converges to λ.

1.5 Topics To Be Explored

Consider the simple case of $n = 3$. We can analyze λ as a function of the three variables a_1, a_2, b, so that

$$\lambda = \lambda(a_1, a_2, b) = b \cdot f\left(\frac{a_1}{b}, \frac{a_2}{b}\right), \tag{10}$$

where λ satisfies $\lambda = A(\lambda)$.

The function f of two variables (or, more generally, $n-1$ variables), should satisfy the following conditions:

1. $f(p_1, p_2) = f(p_2, p_1)$ (symmetry): The network exhibits no branching, so every process has to pass through a_1 and a_2 no matter the order. More generally, let

   $$\lambda = \lambda(a_1, \dots, a_{n-1}, b) \equiv f\left(\frac{a_1}{b}, \dots, \frac{a_{n-1}}{b}\right),$$

   then for all permutations $\pi \in S_{n-1}$ we have $f(\mathbf{p}) = f(\pi\mathbf{p})$.

2. λ is an increasing function of a_1, a_2, b, so f is an increasing function of a_1 and a_2. More generally,

   $$f\left(\frac{a_1}{b}, \dots, \frac{a_{n-1}}{b}\right)$$

   is an increasing function of $a_1, \dots, a_{n-1}$.