Virtual Labs for Data Structures: An Algodynamics Approach

Venkatesh Choppella

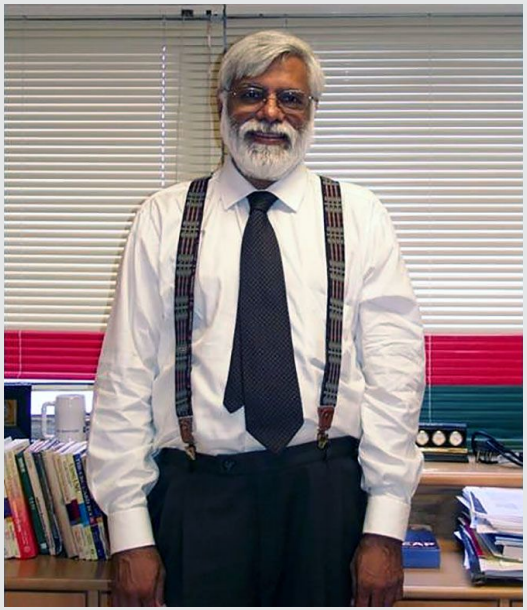
9th Jun, 2021

IIIT Hyderabad

Contents

Dedication: Kesav V Nori

13th December 1945 – 29th May 2021



- IIIT Hyderabad (Distinguished Professor), IIT Hyderabad, IIT Kanpur, Pune Univ.
- Executive Vice President TCS
- Founding Head, Tata Research Design and Development Centre (TRDDC) Pune

Contents

What are Virtual Labs?

1. Online simulation experiments
2. Over 1100 experiments in a dozen disciplines
3. Developed over the last 10 years. Sponsored by Ministry of Education.
4. http://vlab.co.in

Virtual Labs Usage since 1st Jan 2020

1. Online Analytics
2. 42 Million+ Views.
3. 6 Million Users
4. Data Structures-I and Data Structures-II have a combined cumulative views of 1Million+.

Virtual Labs: Practice, Theory, Practice

- How do Virtual Labs Enhance Learning?
- Can the act of designing and doing experiments affect the way we think about a subject?

Animation vs. Interaction

1. Bubblesort on Wikipedia
2. Bubblesort on Youtube

Bridging the gap between Understanding and Implementing

1. Theory
2. **Virtual Labs**
3. Code

1. Modeling
2. **Interacting**
3. Implementing

Contents

Engineering vs. Engineering Science

The fundamental question in any branch of engineering is "how does it work?" The fundamental question in any branch of any engineering science is "why does it work?" These are exactly the questions of software engineering and computer science. We must therefore, inspire our students with curiosity about these questions, and then satisfy it. ...

... Perhaps they don't even need to know how these systems work, but in a university they should be taught all the same, because one day it might help.

— C. A. R. Hoare

Quoted in Sec 2.2.3

Grand Challenges in Computing — Education

British Computer Society, 2004

Engines and Engineering Sciences

Engine	**Flow**	**Engineering Science**
Heat Engines (Locomotives)	Heat flow	*Thermodynamics*
Jet Engines (Turbines)	Fluid flow	*Fluid Dynamics*
Electromagnetic Engines (Generators/Motors)	Current flow	*Electrodynamics*
Computational Engines (Algorithms/Apps)	Information flow	*Algodynamics*

Computer Science and Systems

1. **Computer** Organization
2. **Computer** Networks
3. **Computer** Graphics

But also

1. Operating **Systems**
2. Database **Systems**
3. Information **Systems**

Not to mention other disciplines

1. Eco**Systems**
2. Transportation **Systems**
3. Education **Systems**
4. etc.

What is a system?

1. **Observations:** Quantities that may be sensed/seen
2. **Behaviour:** Traces of observatoins
3. **State:** Internal machinery
4. **Display:** Dashboard
5. **Actions:** Controls that affect state
6. **Dynamics:** Laws that decide how an action affects a state

Continuous vs. Discrete Systems

	Continuous	**Discrete**
Time t	$\mathbb{R}$ or $\mathbb{R}^+$	$\mathbb{Z}$ or $\mathbb{N}$
State x	$x(t) : \mathbb{R}$	$x_n : X$
Change of state	Rate $\dot{x}$ (dx/dt)	Next state x' (x_{n+1})
Dynamics	$\dot{x} = F(x)$	$x' = F(x)$
	Differential Equations	Transition Equations
Discipline	Science	Computing

Closed vs. Open Systems

	Closed	**Open**
Type	Autonomous	Interactive
Action set	Singleton	No restriction
Dynamics	$x' = F(x)$	$x \xrightarrow{u} x'$
Example	Algorithms	Apps
Discipline	(Computer) Science	(Software) Engineering

What is an Algorithm?

1. Iterative System: $\langle X, F : X \to X \rangle$
2. Trajectory: $x_0, F(x_0), F^2(x_0), \ldots$
3. Fixed point: $x = F(x)$

What is an Algorithm?

```
def loop(x,F):
    while True:
        next_x = F(x)
        if next_x = x: # fixed point!
            return x
        else:
            x = next_x
```

How to define an algorithm

1. Specify X: State Space
2. Specify F: Transition function
3. Specify x_0: initial state

Contents

Bubblesort as an iterative system

Bubblesort as an initial value problem.

1. **State space**: $x : X = (i, b, a) : (\mathbb{N}, \mathbb{N}, (Array(\mathbb{N}))$
2. **Initial state**: $(a^0, 0, |a^0|)$
3. **Transition Equation**: $(a', i', b') = F(a, i, b)$

Bubblesort as an iterative system

Bubblesort as an initial value problem.

1. **Transition Function**:

$$
\begin{aligned}
F(i, 1, a) &= (i, b, a) \\
F(i, b, a) &= (i, b, a) && i \geq b \\
F(b-1, b, a) &= (0, b-1, a) \\
F(i, b, a) &= (i+1, b, a) && a_i \leq a_{i+1} \\
F(i, b, a) &= (i+1, b, swap(a, i, i+1) && a_i > a_{i+1}
\end{aligned}
$$

Bubblesort Example Run

$[\underline{6}\ 8\ 7\ 4 \mid] \rightarrow$

$[6\ \underline{8}\ 7\ 4 \mid] \rightarrow$

$[6\ 7\ \underline{8}\ 4 \mid] \rightarrow$

$[6\ 7\ 4\ \underline{8} \mid] \rightarrow$

$[\underline{6}\ 7\ 4 \mid 8\ ] \rightarrow$

$[6\ \underline{7}\ 4 \mid 8\ ] \rightarrow$

$[6\ 4\ \underline{7} \mid 8\ ] \rightarrow$

$[4\ \underline{6} \mid 7\ 8\ ] \rightarrow$

$[\underline{4} \mid 6\ 7\ 8\ ]$ fixed point

Bubblesort as an interactive system

Bubblesort as an initial value problem.

1. **State space**: $X = Array(\mathbb{N})$
2. **Initial state**: a^0
3. **Actions**: swap(i, j),
4. **Transition Relation**:

$$a \xrightarrow{\text{swap}(i,j)} a' \qquad \text{iff } 0 \leq i, j < |a| \text{ and } a' = swap(a, i, j)$$

Interactive run of Bubblesort

$$[\underline{8}\ 6\ \underline{4}\ 7] \xrightarrow{\text{swap}(0,2)}$$

$$[4\ 6\ \underline{8}\ \underline{7}] \xrightarrow{\text{swap}(2,3)}$$

$$[4\ 6\ 7\ 8] \not\rightarrow$$

Bubblesort Demo

Bubblesort Demo

Contents

Mergesort Demo

Search Demo

Contents

Mergesort Demo

Mergesort Demo

Contents

Conclusion

1. Learning by Doing is a well-known principle in Learning Science.
2. Virtual Labs encourage learning by doing
3. Virtual Labs correspond to Open Systems
4. The systems view of Computer Science is Algodynamics
5. Focus on Modelling, Interaction along with coding.