

CO 330 LECTURE 9 SUMMARY

FALL 2026

SUMMARY

Definition 1. *The dependency digraph of the specification*

$$\begin{aligned}\mathcal{A}^{(1)} &= \Theta_1(\mathcal{A}^{(1)}, \mathcal{A}^{(2)}, \dots, \mathcal{A}^{(r)}) \\ \mathcal{A}^{(2)} &= \Theta_2(\mathcal{A}^{(1)}, \mathcal{A}^{(2)}, \dots, \mathcal{A}^{(r)}) \\ &\vdots \\ \mathcal{A}^{(r)} &= \Theta_r(\mathcal{A}^{(1)}, \mathcal{A}^{(2)}, \dots, \mathcal{A}^{(r)})\end{aligned}$$

is the directed graph with vertices $1, 2, \dots, r$ and directed edge from i to j if $\mathcal{A}^{(i)}$ appears on the right hand side of the j th equation.

Note in particular that self-loops are allowed.

A specification is *iterative* if the dependency digraph has no directed cycles and a specification is *recursive* if the dependency digraph has at least one directed cycle (including a self-loop as a directed cycle). A specification is *rational* if it is an iterative specification using only $\cup, \times, \text{Seq}, \mathcal{E}, \mathcal{Z}$. As class described by a rational specification will always have a rational generating series. Also all regular languages have a rational specification but there's a subtlety in that a regular expression may not be unambiguous, but it is always possible to rewrite it to be unambiguous, see the references for details.

Next we stated the Lagrange implicit function theorem (LIFT)

Theorem 2 (Lagrange Implicit Function Theorem). *Let G be a formal power series whose constant term has a multiplicative inverse. Then there is a unique formal power series R satisfying*

$$R(x) = xG(R(x))$$

and for any formal power series F we have

$$[x^n]F(R(x)) = \frac{1}{n}[u^{n-1}]F'(u)G(u)^n \text{ for } n \geq 1$$

and

$$[x^0]F(R(x)) = F(0)$$

Most of the time we don't really need F , so we'll just take $F(u) = u$. With this theorem we can quickly calculate things that were doable but hard before and we can calculate things we couldn't do before. We did one example and set ourselves up for a multivariate example.

NEXT TIME

We'll do the multivariate LIFT example and then move on to the proof of LIFT.

REFERENCES

- For the relation between regular languages and regular specifications see Flajolet and Sedgewick “Analytic Combinatorics” appendix A.7.
- <https://enumeration.ca/toolbox/lift/>
- Chapter 8 of <https://melczer.ca/330/WagnerNotes.pdf>.