COURSE SUMMARY

GABRIELA BROWN
MATH B49
05/23/2022

Contents

1. Introduction

TODO

2. Fundamentals of Shift Spaces

[1][Ch. 1, Ch. 2]

Key realizations:

- The topology (open sets) of a shift space has a basis in cylinder sets. It is thus natural to define a shift space by forbidden words, because we need the shift space to be closed, and the forbidden words give us a countable number open sets that are excluded, the union of which is open, and the complement of which is closed.
- Why do we bother with progressive overlap? (i.e. using Nth higher block shift vs Nth higher power shift) Both are shift spaces, but the former are what give us "sliding block codes" (through factoring?) which when they are invertible are what give us topological conjugacies between shift spaces. Higher block shifts come up later in the Finite Coding Theorem to get around a particular entropy requirement.
- Is 0 needed for the closure?

1.1.

1.2. defs: shift space forbidden blocks subshift shift invariance shift map

give "word definition" and forbidden blocks definition e.g. golden mean shift e.g. even shift e.g. run length limited shift e.g. S-gap shift e.g. a shift of finite type e.g. context free shift e.g. failure of closure

1.3. defs: language irreducible

prop: correspondence between language and shift space // TODO: this can be intuitive, not formal

1.4. defs: progressive overlap Nth higher block shift (and notation) Nth higher power shift (and notation)

prop: higher block shift also shift space prop: Nth higher power shift also a shift

e.g. golden mean shift

1.5. defs: block map sliding block code with memory m and anticipation n topologically conjugate

some e.gs. also golden mean

prop: sliding block code diagram commutes prop: dynamical properties preserved by sliding block code prop: existence of a factoring of a sliding block code through a higher block shift thm: image under a sliding block code is a shift space thm: invertible sliding block codes are conjugacies

2.1. defs: shift of finite type M-step (memory M)

e.g. full shift, golden mean, 3 letter alphabet example from before n.e.g. even shift

thm: conjugate to a shift of finite type =¿ shift of finite type

... rest of chapter graphs of shifts state splitting data storage

ch 3 - sofic shifts

questions - is the centering condition on a sliding block code equivalent to it being continuous coding is always continuous based on the metric on the shift space it's expanding, points are pushed away every iteration don't worry too much about sliding block codes - example of a full shift code that is not a sliding block code

- e.g. 1.2.6, do you need $10^k1orjust0^k, doyouneed0inthesettogetclosure?closed-wewanttododynamicsonacom$
$compactsubsetsergodictheoryworkspoorlyifyoudon'thaveacompactphasespaceifyoudon'thavecompactness, youm$

coded maps and sfts from dynamics are basically the same, because even if you don't have the conjugacy you still have the factor (finite to 1)

every sofic shift is a factor of an sft every sft is a sofic shift not every sofic shift is an sft finite of 1.2.6 is a sofic shift co-finite case of 1.2.6 is an sft

- if you have a lift of a blockmap through a higher alphabet that gives you a 1-block code, why doesn't this recoding not in general give you a 1-block coding for the inverse? Also, what does it mean for the inverse?

References

[1] Brian Marcus Douglas Lind. *An Introduction to Symbolic Dynamics and Coding*. Ed. by Cambridge University Press. Second. Cambridge Mathematical Library, 2021.