Отношению КА к другим мат. моделям

- Клеточный автомат обладает алгоритмической универсальностью (https://wpmedia.wolfram.com/uploads/sites/13/2018/02/15-1-1.pdf)
- Любая Машина Тьюринга может быть симулирована элементарным клеточным автоматом (https://link.springer.com/content/pdf/10.1007%2F978-0-387-30440-3_54.pdf, https://www.researchgate.net/publication/220431575_Simple_Computation-Universal_Cellular_Spaces)

  ### Comparison with Turing Machines

  It is well-known that a TM with one tape and one head on that tape can be simulated by a one-dimensional CA. See for example the paper by Smith [27]. But even multi-tape TM can be simulated by a one-dimensional CA without any significant loss of time.

  Several methods are presented above for obtaining substantial computing power from quite simple cellular automata. In particular, one dimension is shown to be sufficient for universality on the set of partial recursive functions. A cellular autom-

- Клеточный Автомат имеет (*как минимум*) такую же вычислительную мощность, как и Машина Тьюринга (http://hpc-education.ru/files/lectures/2011/ershov/ershov_2011_slides02.pdf)

  Алгоритмическая универсальность

  - Большое внимание в исследованиях по клеточным автоматам уделяется вопросам алгоритмической универсальности той или иной модели.
  - Легко показать, что уже одномерные клеточные автоматы с локальной окрестностью ранга один являются алгоритмически универсальными.
  - Это достигается построением клеточного автомата по заданной машине Тьюринга.
  - Множество состояний такого автомата есть декартово произведение алфавита машины Тьюринга и множества ее состояний (плюс одно пустое состояние).
  - Далее каждое правило машины Тьюринга преобразуется в набор правил клеточного автомата. Идея заключается в том, что любая команда машины Тьюринга затрагивает максимум две ячейки ленты, т. е. является локальной.

- Клеточный автоматы распознают неопределенные полиномиальные языки (https://link.springer.com/chapter/10.1007/3-540-58338-6_95, http://www.umsl.edu/~siegelj/information_theory/classassignments/Lombardo/03_nondeterministicpolynomialtime.html)

space. While finite automata generate only regular languages, cellular automata generate non-deterministic polynomial languages which may be non-regular.

Nondeterministic polynomial time

A ***verifier*** for a language *A* is an algorithm *V*, where
A = {*w* | *V* accepts <*w*,*c*> for some string *c*}.

We measure the time of a verifier only in terms of the length of *w*, so a ***polynomial time verifier*** runs in polynomial time in the length of *w*.

A language *A* is ***polynomially verifiable*** if it has a polynomial time verifier.

NP is the class of languages that have polynomial time verifiers.

NTIME(*t*(n)) = {L| L is a language decided by a *O*(*t*(n)) time nondeterministic Turing machine}.

$NP = \bigcup_k NTIME(n^k)$

- Определение рекурсивных автоматов (https://link.springer.com/article/10.1007/s00224-011-9369-9?shared-article-renderer)

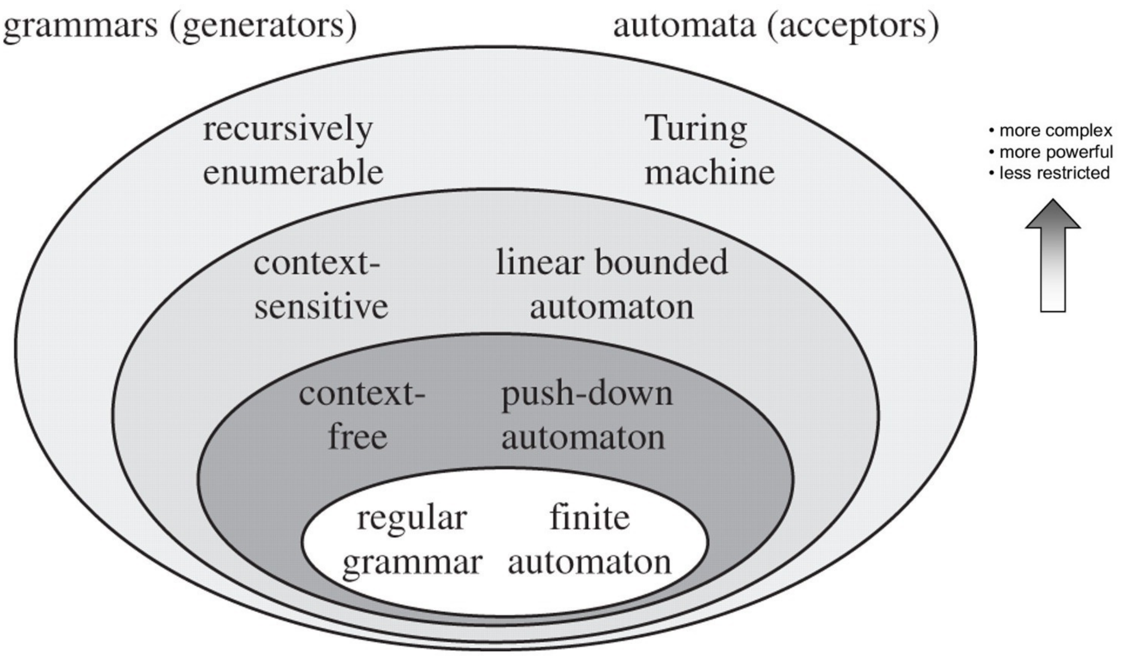
Definition 3

Classes of algorithms and automata that have the same computing/accepting power as Turing machines are called *recursive*.

- Связь КА с формальными языками (https://www.sciencedirect.com/science/article/abs/pii/016727899090197W)

A connection is made between the description of one-dimensional cellular automata as dynamical systems acting on a compact metric space, and as computational systems acting on strings. Operators are defined which determine a bijective correspondence between subshifts of infinite configurations and languages of finite strings. Under this correspondence, families of languages give rise to families of subshifts. Regular languages correspond to sofic subshifts. Families of subshifts corresponding to context-free, context-sensitive, and recursively enumerable languages are introduced and their closure properties under forward and backward cellular automaton images studied.

- Иерархия формальных языков Чомского (https://devopedia.org/chomsky-hierarchy)



- Клеточный Автомат может симулировать Машину Тьюринга (https://wpmedia.wolfram.com/uploads/sites/13/2018/02/04-3-4.pdf)

included), is shown. It is also demonstrated that a Turing machine with m tape symbols and n internal states can be simulated by a cellular automaton of range $r = 1$ with $m + n + 2$ states per cell.