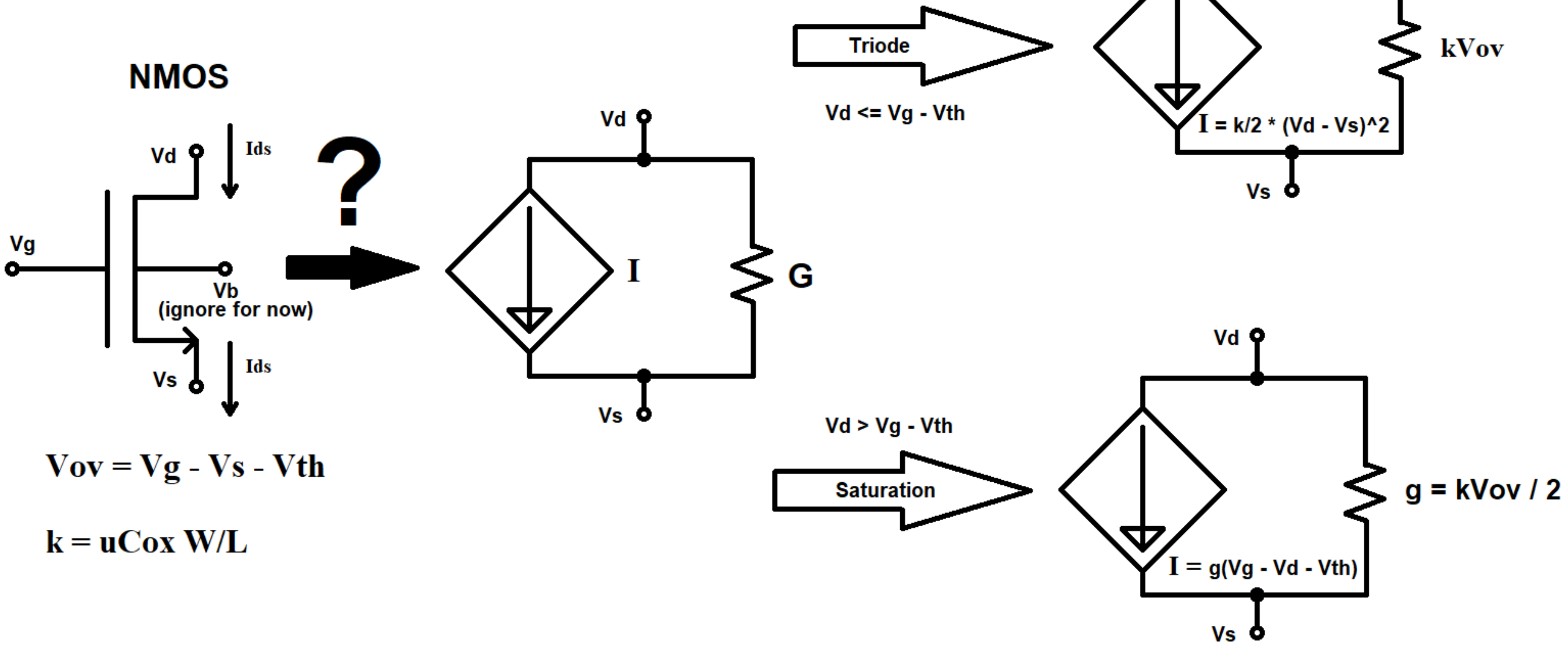
CUDA SPICE CIRCUIT SIMULATOR

MILESTONE 2

ANGELINA RISI, EE 2018

PROGRESS SINCE MILESTONE 1: TRANSISTORS

- Added transistor parsing, not including any extra geometry/layout parameters
- Currently use a default hard-coded NMOS model, need to add model parsing
- Transistor linearization to G and I elements
- Solution of circuits including transistors by:
    - Using previous solution as "Guess" voltages to linearize model
    - Iterating solving circuit until new solution and previous guess converge to within some absolute tolerance (currently 1µV)
    - This implicitly performs Newton-Raphson method
    - Currently very serial method, only matrix solving on GPU



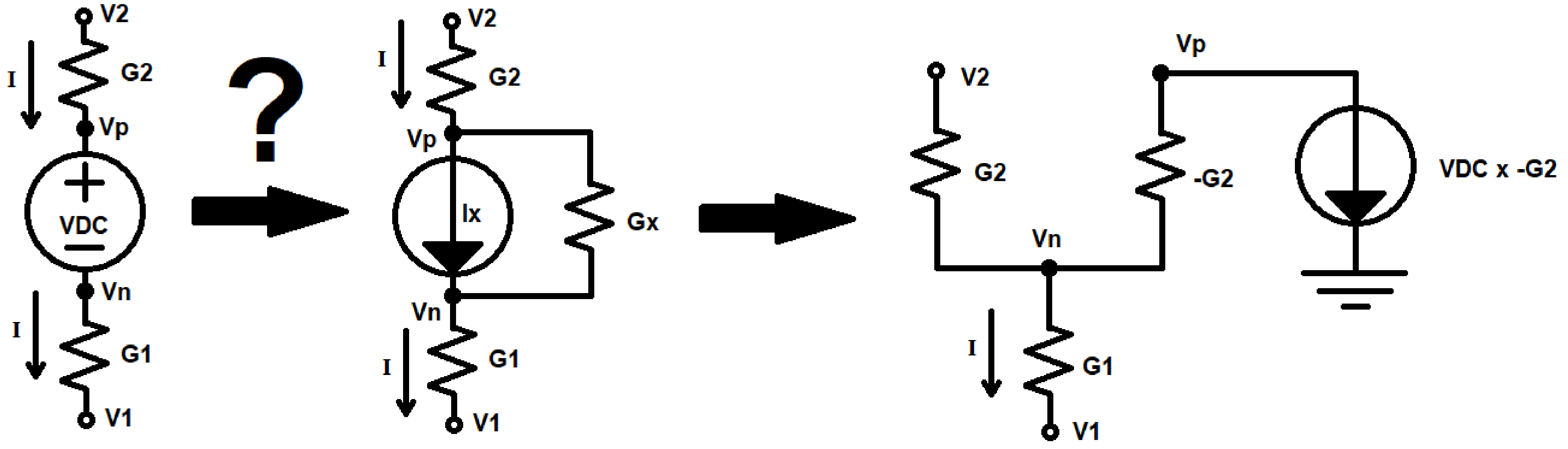
$$V_{ov} = V_g - V_s - V_{th}$$

$$k = uC_{ox} W/L$$

PROGRESS SINCE MILESTONE 1:

- Code cleanup: overhaul on how we handle circuit elements
  - Instead of separate structs and lists for each type of element, all passives are the same Element with node and parameter arrays (instead of hardcoding how many they have)
    - Other than voltage sources, all passives can be stored on the same list (because VDC needs to be applied last to matrices for reasons described below
- More edge case testing to locate and correct bugs
  - Located a problem with VDC modelling (problem with conversion to I and G when neither node grounded)
    - Before: Used the fact that the currents flowing out of both positive and negative node are equal to generate G and I matrix entries
    - Problem: Even though $Vp = Vn + Vdc$ used to replace on of the voltages in one of the equations to assert Vdc dependence, still dependent on G seen at each node
    - Solution: seems really obvious, but $Vp – Vn = Vdc$, => $Vp(Gx) – Vn(Gx) = Vdc(Gx)$ => set $Gx = 1$. We still use the equal currents to generate the Vn row of the matrix, but the Vp row is replaced with 1 in Vp column and -1 in Vn column and Vdc in I va

Voltage Source Transformation



CURRENTLY WORKING ON:

- Model parsing and multiline content parsing
    - Models generally specified in separate .incude files, but can be in netlist file, .model line
    - Parameters spilling into multiple lines start new line with ‘+’ character, need to look ahead to next line
- DC Sweep – already have idea of how to setup
    - Find swept element, save original parameter value, overwrite with start value
    - Solve, increment value by step size, repeat until stop value reached
- Porting matrix setup to GPU:
    - Parallelize by elements
    - Atomics for race conditions between elements on same node

NEXT MILESTONE:

- Transient (Time step) Simulation
  - Add in capacitors, inductors, AC sources (time-dependent elements)
- Second-order effects (improved transistor modelling)
- Graphical interface or data export and plotting
- Performance optimizations & comparison
- Stream compaction of matrices (expected to be sparse in large circuits)