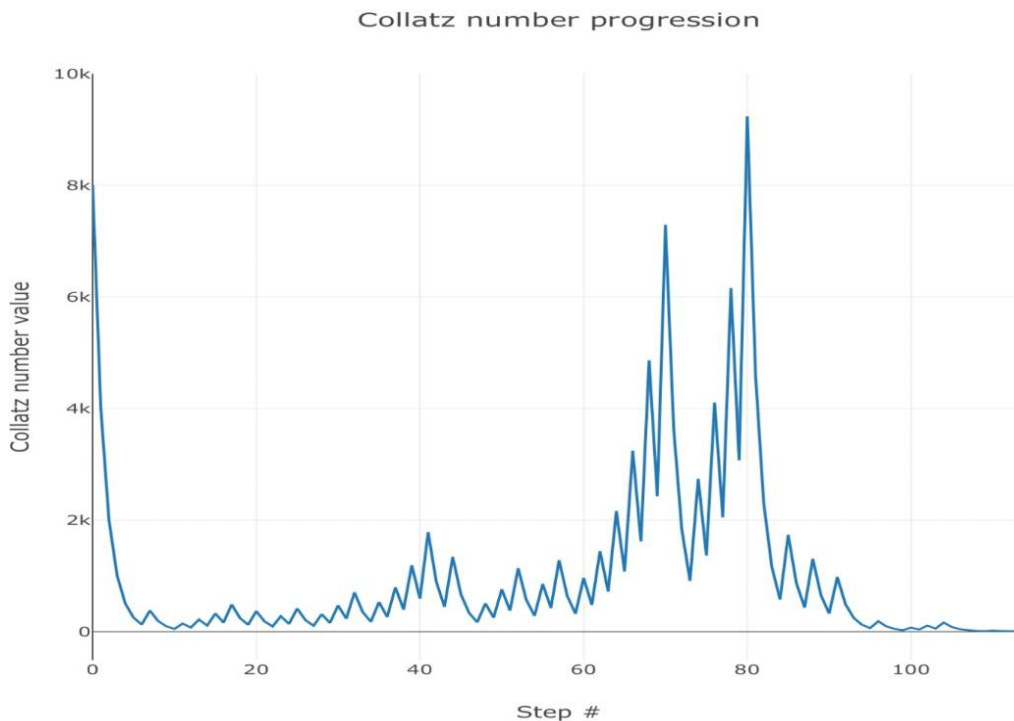


Which is the infinitively iterative graph . And the collatz conjecture graph



Graph of collatz conjecture

Can you see some equality between the graph of collatz conjecture and zaman giant equation graph . Yest it seems that it is a bad weather graph and if you zoom the zaman giant equation you will see that both have a property of increasing and decreasing functionality . If we want to find the similarity between bessel graph,zaman graph and collatz graph we will see

that collatz conzepture is the subset of zaman graph , zaman graph is the subset of Bessel graph in previous paper which I called the “The ultimate proof of collatz conjecture using simple number theory . Which was so easy to prove that if we have a set of only odd numbers collatz conjecture proof exist for all natural number. And remember odd number is the subset of

Natural number . Then if you deny that collatz conjecture then you have to deny the zaman function which is based on Bessel and euler identity . If you deny this then math is over . So I think Collatz Conjecture is proved for all types of function . Here is the source code if you want to check the graphs are correct

#### Source Code:

```
export LANG=C export
SERVER_NAME=pcbarina.fit.vutbr.cz echo
"hostname=$(hostname)"echo "pwd=$(pwd)"echo
"HOME=$HOME"echo "cpu model name=$(cat
/proc/cpuinfo | grep "model name" | head -n1)"echo
"cpus=$(cat /proc/cpuinfo | grep processor | wc -
l)"echo "TMPDIR=$TMPDIR" set -uset -e # check
the connectionif ! ping -c1 -q
"${SERVER_NAME}"; then echo "No
connection!" exitfi umask 077 CC=gccif type clang
> /dev/null 2> /dev/null && clang --version | grep -
qE "version (8|9|10|11|12|13)"; then echo "INFO:
clang available" CC=clangfi # don't forget git clone
git@github.com:xbarin02/collatz.git into
$HOMESRCDIR=$HOME/collatz/MAPDIR=$HO
ME/collatz-sieve/TMP=$(mktemp
collatz.XXXXXXXXXX --tmpdir) echo
"SRCDIR=$SRCDIR"echo "TMP=$TMP" if ! test
```

sourcecodeforpython

```
def collatz(n):
    while n > 1:
        print(n, end=' ')
        if (n % 2):
            # n is odd
            n = 3*n + 1
        else:
            # n is even
            n = n//2
        print(1, end='')

n = int(input('Enter n: '))
print('Sequence: ', end='')
collatz(n)
```

#### LastComment

If you can run all these programs you will see the relation between the collatz conjecture function and thefundament law of nature in physics “ Everything wants to come inequilibrium”

```
-d "$SRCDIR"; then pushd "$HOME" git clone
git@github.com:xbarin02/collatz.git popdelse
pushd "$SRCDIR" git pull || echo "cannot sync
repo" popdfi if ! test -d "$MAPDIR"; then pushd
"$HOME" git clone
git@github.com:xbarin02/collatz-sieve.git
popdelse pushd "$MAPDIR" git pull || echo
"cannot sync repo" popdfi mkdir -p --
"$TMP"pushd -- "$TMP" cp -r "${SRCDIR}" . cd
collatz/src HOSTNAME=$(hostname -s | tr
[:upper:] [:lower:]) TASK_UNITS=16if [[
"$HOSTNAME" =~ ^pco204-..$ ]]; then
TASK_UNITS=20fi # build mclient &
workermake -C worker clean all USE_LIBGMP=1
CC=$CC USE_SIEVE=1 USE_PRECALC=1
SIEVE_LOGSIZE=34 USE_SIEVE3=0
USE_SIEVE9=1
USE_LUT50=1make -C gpuworker clean all
CC=$CC TASK_UNITS=${TASK_UNITS}
SIEVE_LOGSIZE=24 USE_SIEVE3=1 || echo
"unable to build gpuworker"make -C mclient clean
all pushd "$MAPDIR"./unpack.sh esieve-34.lut50
"$TMP"/collatz/src/worker./unpack.sh esieve-24
"$TMP"/collatz/src/gpuworkerpopd cd mclient
CLEANUP_DIR=$TMP screen -d -m ./spawn.sh
$* popd
```