# How can FPGAs contribute to sustainable animal breeding?

**Jeremie Vandenplas**, Mario Calus

6 December, 2023

# Acknowledgements



Luke
NATURAL RESOURCES INSTITUTE FINLAND

TU Delft

VORTECH

UNIVERSITY OF MANNHEIM

Cobb
ONE FAMILY.
ONE PURPOSE.

CRV
BETTER COWS | BETTER LIFE

icbf

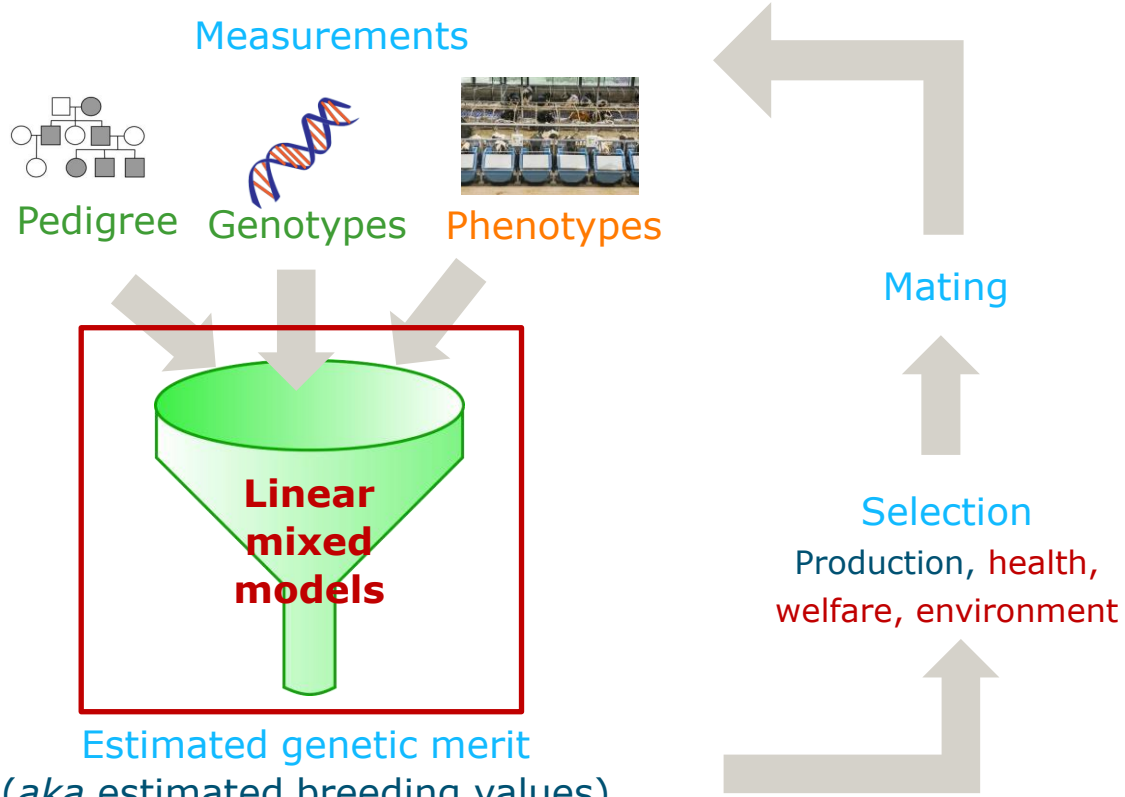
Topigs Norsvin
PROGRESS IN PIGS

HENDRIX GENETICS

WAGENINGEN
UNIVERSITY & RESEARCH

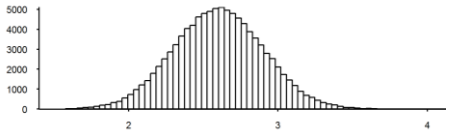
# Animal breeding

To contribute to making animal production more sustainable



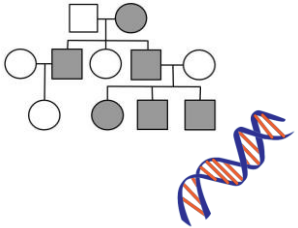

WAGENINGEN
UNIVERSITY & RESEARCH

# Phenotype = Environment + Genetic




=

+



Thousands/millions of

- Phenotypic observations (so called 'traits')
  - Production, health, welfare, environment
- Animals with known genealogy (several generations)
- DNA-profiles
  - Summarized with ~50,000 values (SNP; 0, 1, 2 or missing) per individual

# Linear mixed model

$$\text{Phenotype} = \text{Environment} + \text{Genetic}$$

$$\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{W}\mathbf{a} + \mathbf{e}$$

$\mathbf{y}$: vector of phenotypic records

$\mathbf{b}$: vector of fixed effects (e.g., herd, sex)

$\mathbf{a} \sim MVN(\mathbf{0}, \mathbf{\Sigma})$: vector of random additive genetic (animal) effects

- $\mathbf{\Sigma}$: pedigree and genomic-based relationship matrix

$\mathbf{e} \sim MVN(\mathbf{0}, \mathbf{R})$: vector of random residuals

# Genotype matrix $\mathbf{Z}$ – main challenges

- Typically
  - Rows: Up to millions of individuals
  - Columns: ~50-100K Single Nucleotide Polymorphisms

```
id1 112121110221002020201111211211012002021110111122012112110100212020100101011011020210
id2 111011101111112111201101112111002001010111201111102112111111111121001212221121011011
id3 111011101102122202202202202002002000000022210011201121011012000100211002122120121001211
id4 021120211112112111202212121102012001011011120111110211211111111112100121222112101101
id5 112001011112111201111101101221111112111101120111110211211111111112100121222112101101
```

# Genotype matrix $\mathbf{Z}$ – main challenges

- Challenges
  - Storage
  - Multiplication with
    1. itself,
    2. a real dense matrix, or
    3. a real sparse matrix

```
id1 112121110221002020201111211211012002021110111122012112110100212020100101011011020210
id2 111011101111112111201101112111002001010111120111111021121111111111210012122211210110
id3 111011101102122202202202202002002000000022210011201121011012000100211002122120121001
id4 021120211112112111202212121102012001011011120111111021121111111111210012122211210110
id5 112001011112111201111101101221111112111101120111111021121111111111210012122211210110
```

# Genotype matrix $\mathbf{Z}$ – storage

- Storage

  ➔Compressed genotypes

  ➔Bit-level

E.g. 2 million genotypes with 50,000 SNPs

- Double precision: 800 GB ➔ Compressed: 25 GB

➔Limited memory requirements

# Genotype matrix Z – storage

| SNP genotype | Homozygous first allele | Heterozygous | Homozygous second allele | Missing |
|---|---|---|---|---|
| **Decimal** | 0 | 1 | 2 | 3 |
| **2-bit** | 00 | 10 | 11 | 01 |

3210 ⇔ 00101101 ⇔ 45

4 SNP genotypes (4-32 bytes)

1-byte integer

# 1) Genotype matrix $\mathbf{Z}$ – multiplication with itself

- Two ways
  - $\mathbf{ZZ'}$ ➔ genomic relationship matrix
  - $\mathbf{Z'Z}$ ➔ linkage disequilibrium
- Multiple applications
  - Genomic evaluations
  - Population genomics
  - Genome-wide association studies
- Also needed in plant and human genomics

# 1) Genotype matrix **Z** – multiplication with itself

- Potential of FPGAs already demonstrated

2016 IEEE International Parallel and Distributed Processing Symposium Workshop

**Efficient Computation of Linkage Disequilibria as Dense Linear Algebra Operations**

Nikolaos Alachiotis, Thom Popovici, Tze Meng Low
*Carnegie Mellon University*
{*nalachio, lowt, dpopovic*}*@andrew.cmu.edu*

**High Performance Linkage Disequilibrium: FPGAs Hold the Key**

Nikolaos Alachiotis
Carnegie Mellon University
nalachio@cmu.edu

Gabriel Weisz
Carnegie Mellon University
gweisz@cmu.edu

**Table 3: Performance comparison based on simulated datasets D.1 and D.2 with sample sizes of 10,000 and 100,000 sequences, respectively.**

| Threads | PLINK 1.9 Exec. time (sec) | | PLINK 1.9 mLD/sec | | FPGA LD Proc. Speedup (X) | |
|---|---|---|---|---|---|---|
| | D.1 | D.2 | D.1 | D.2 | D.1 | D.2 |
| 1 | 41.1 | 389.1 | 1.2 | 0.128 | 171.4 | 159.3 |
| 2 | 31.4 | 297.6 | 1.6 | 0.168 | 128.5 | 121.4 |
| 4 | 19.2 | 180.2 | 2.6 | 0.277 | 79.1 | 73.6 |
| 8 | 11.3 | 109.4 | 4.4 | 0.456 | 46.8 | 44.7 |
| 12 | 9.9 | 88.3 | 5.0 | 0.566 | 41.1 | 36.0 |

# 2) Genotype matrix $\mathbf{Z}$ – multiplication with a real dense matrix

- Various applications
  - Genome-wide associations studies
  - Genomic evaluations
  - …

# 2) Genotype matrix $\mathbf{Z}$ – multiplication with a real dense matrix

- Genomic evaluations
  - Solved by PCG
  - Required two multiplications
    - $\mathbf{Z}\boldsymbol{\Lambda}_1$ and $\mathbf{Z}'\boldsymbol{\Lambda}_2$
    - Up to 90% of computational load of one PCG iteration

➔Tailored algorithms needed for compressed genotypes

- CPU and GPU

# 2) Genotype matrix $\mathbf{Z}$ – multiplication with a real dense matrix

- Multiplication with compressed genotypes

➔Two strategies investigated for CPUs and GPUs

1. Decompression of (blocks of) $\mathbf{Z}$ followed by the multiplication with the corresponding $\boldsymbol{\Lambda_i}$

Or

2. Storage of all possible results in a hash table followed by looking up the results corresponding to $\mathbf{Z}$

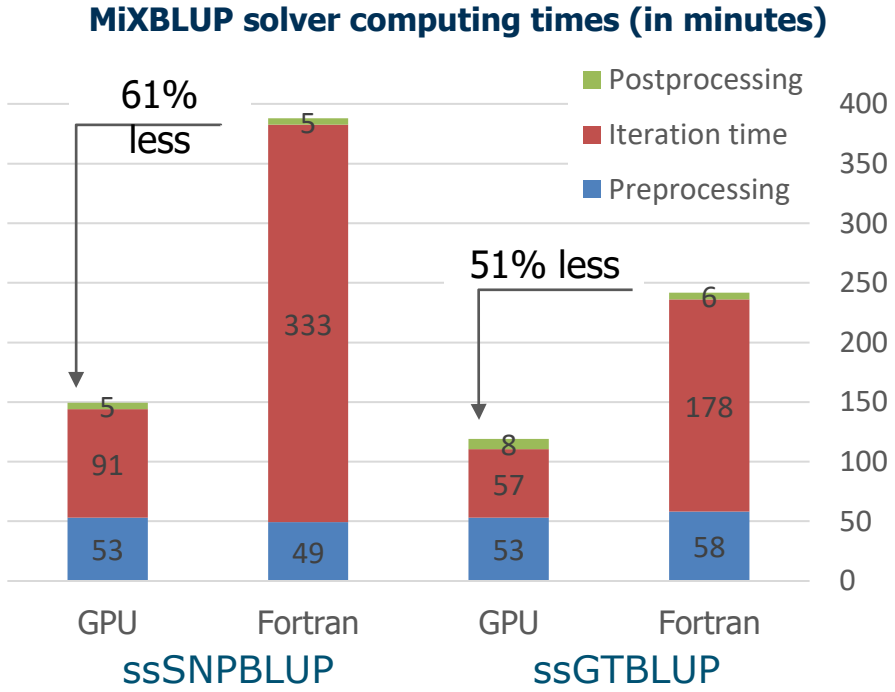
**➔Can we perform these multiplications with FPGAs?**

# Example – Irish dataset

Accelerated matrix-vector multiplications for matrices involving genotype covariates with applications in genomic prediction

Alexander Freudenberg[1*], Jeremie Vandenplas[2], Martin Schlather[1], Torsten Pook[2], Ross Evans[3] and Jan Ten Napel[2]

- 2.61 million genotypes
  - 47,006 SNPs
- CPU: AMD EPYC 7513
- GPU: Nvidia H100
  - 80 GB
- 20 threads




WAGENINGEN UNIVERSITY & RESEARCH

Freudenberg et al. Interbull 2023

# 3) Genotype matrix $\mathbf{Z}$ – multiplication with a real sparse matrix

- $\mathbf{S}$: sparse matrix
- Needed for
  - Preconditioning: $\mathbf{Z}'\mathbf{S}\mathbf{Z}$
  - PCG iteration: $(\mathbf{Z}'\mathbf{S}\mathbf{Z})^{-1}\boldsymbol{\Lambda}_1$
- Result: dense matrix
  - Rows: ~#SNPs

# Other problems…

- DNA sequencing analysis and storage
- DNA sequence alignment
- Genotype imputation
- …

# Conclusions

- Genotype matrix - several challenges
  - Storage
  - Multiplication with 1) itself, 2) a real dense matrix, or 3) a real sparse matrix
- Possible with current datasets, tailored algorithms, software, hardware

BUT datasets get bigger every day!

➔ Current large datasets will be "small" tomorrow!

**How can FPGAS contribute to sustainable animal breeding?**

# Thank you!

# Mixed model equations

$$\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{W}\mathbf{a} + \mathbf{e}$$

$$\begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{W} \\ \mathbf{W}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{W}'\mathbf{R}^{-1}\mathbf{W} + \boldsymbol{\Sigma}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{b} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{y} \\ \mathbf{W}'\mathbf{R}^{-1}\mathbf{y} \end{bmatrix}$$

- Large (>$10^9$ equations) and ill-conditioned
- Mainly sparse
- Dense block (up to a few million rows/columns)
    - Due to genomic information

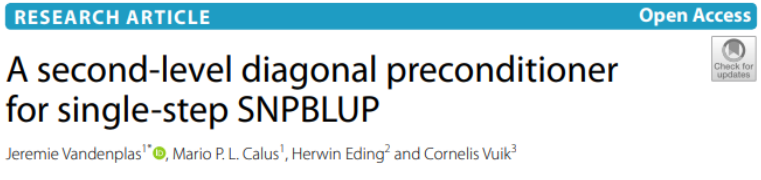
# Solving large mixed model equations

- Two-level PCG


RESEARCH ARTICLE Open Access

Deflated preconditioned conjugate gradient method for solving single-step BLUP models efficiently

Jérémie Vandenplas[1*], Herwin Eding[2], Mario P. L. Calus[1] and Cornelis Vuik[3]


RESEARCH ARTICLE Open Access

A second-level diagonal preconditioner for single-step SNPBLUP

Jeremie Vandenplas[1*], Mario P. L. Calus[1], Herwin Eding[2] and Cornelis Vuik[3]

- Matrix-free approaches
  - Main cost: (compressed) genotype matrix


RESEARCH ARTICLE Open Access

Computational strategies for the preconditioned conjugate gradient method applied to ssSNPBLUP, with an application to a multivariate maternal model

Jeremie Vandenplas[1*], Herwin Eding[2], Maarten Bosmans[3] and Mario P. L. Calus[1]

WAGENINGEN UNIVERSITY & RESEARCH

TUDelft VORTECH CRV BETTER COWS | BETTER LIFE

# Large example – Irish dataset

- 17 million records (6 traits)
- 26.5 million animals
- 2.61 million genotypes with 47,006 SNPs
  - ➔Double precision:~982 GB
  - ➔Compressed: ~31 GB
- Equations: 372 million
- RAM required for PCG: 86 GB

Accelerated matrix-vector multiplications for matrices involving genotype covariates with applications in genomic prediction

Alexander Freudenberg[1*], Jeremie Vandenplas[2], Martin Schlather[1], Torsten Pook[2], Ross Evans[3] and Jan Ten Napel[2]

RESEARCH ARTICLE Open Access

Efficient large-scale single-step evaluations and indirect genomic prediction of genotyped selection candidates

Jeremie Vandenplas[1*†], Jan ten Napel[1], Saeid Naderi Darbaghshahi[2], Ross Evans[2], Mario P. L. Calus[1], Roel Veerkamp[1], Andrew Cromie[2], Esa A. Mäntysaari[3] and Ismo Strandén[3†]

WAGENINGEN UNIVERSITY & RESEARCH

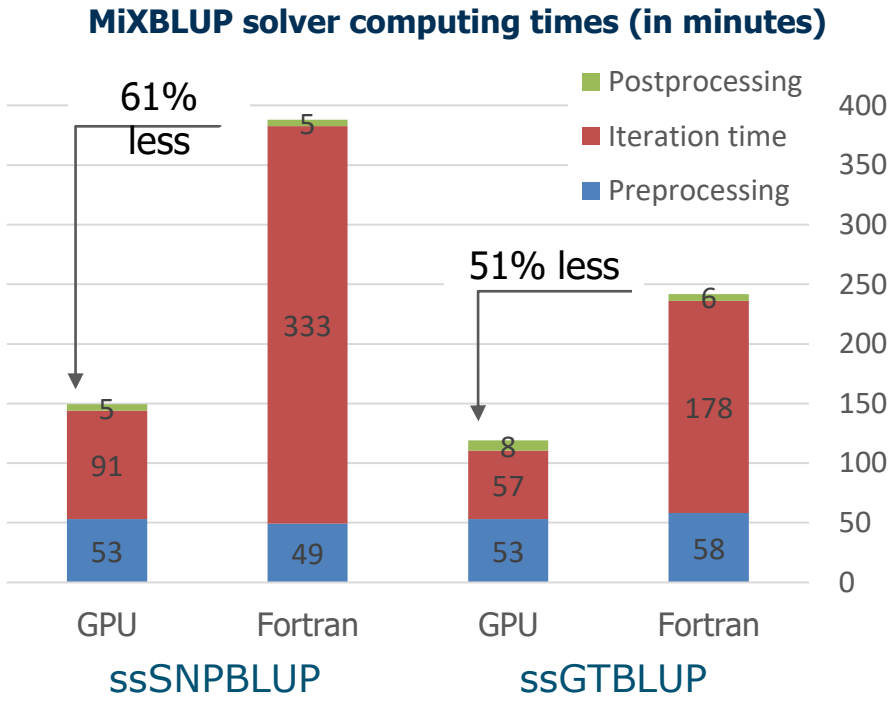
icbf

Luke NATURAL RESOURCES INSTITUTE FINLAND

UNIVERSITY OF MANNHEIM

# Large example – Irish dataset

- CPU: AMD EPYC 7513
- GPU: Nvidia H100
  - 80 GB
- 20 threads




WAGENINGEN UNIVERSITY & RESEARCH

Freudenberg et al. Interbull 2023

# 2) Genotype matrix $\mathbf{Z}$ – multiplication with a real dense matrix

- At each PCG iteration: $\mathbf{Z\Lambda_1}$ and $\mathbf{Z'\Lambda_2}$
  - Up to 90% of computational load of one PCG iteration

➔Tailored algorithms for compressed genotypes

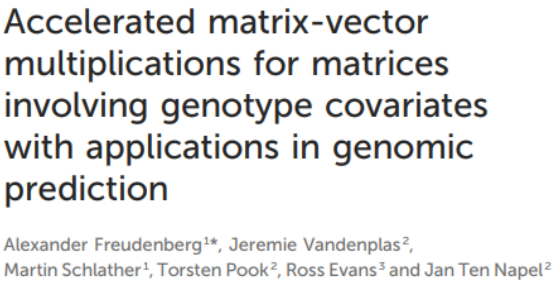
- Optimal use of the hardware (e.g. vectorization, loop unrolling)
- CPUs and GPUs

**➔Can we perform these multiplications with FPGAs?**


RESEARCH ARTICLE Open Access

Computational strategies for the preconditioned conjugate gradient method applied to ssSNPBLUP, with an application to a multivariate maternal model

Jeremie Vandenplas[1*], Herwin Eding[2], Maarten Bosmans[3] and Mario P. L. Calus[1]


Accelerated matrix-vector multiplications for matrices involving genotype covariates with applications in genomic prediction

Alexander Freudenberg[1*], Jeremie Vandenplas[2], Martin Schlather[1], Torsten Pook[2], Ross Evans[3] and Jan Ten Napel[2]

WAGENINGEN UNIVERSITY & RESEARCH