3G GW Computing

Peter Couvares (LIGO Laboratory - Caltech)

February 16, 2019

LIGO-G1900241-v1

Issues

- 3G Computing Resources Needed
- 3G Data Analysis Software Infrastructure
- 3G Computing - Human Effort
- 3G GW Community Services
- Data Analysis Optimization Support
- Collaborations with non-GW computing efforts.

LIGO-Virgo 2G Computing

- Compute
  - For the third observing run (O3), the LVC expects to need ~500 million CPU core-hours of data analysis for ~80 astrophysical searches, followup activities, and detector characterization.
  - The 10 most demanding of these 80 analyses comprise ~90% of the demand; long tail of remaining 70.
  - Most of this computing is "pleasingly parallel" High Throughput Computing (HTC) for "deep" offline searches; ~10% is low-latency data analysis needed to generate rapid alerts for multi-messenger (EM, neutrino) followup. Very little HPC needed (NR simulation is the exception — not included here).
  - Currently ~90% provided by dedicated LIGO-Virgo clusters vs. ~10% from external shared computing resources — but growth of the dedicated resources has flattened while shared component is growing.
  - Growing shared, external computing resources are presenting new distributed computing and data access challenges. This is something we share in common with LHC computing.
  - Careful consideration of parallel GPU and MIC architectures for our most compute-intensive searches; CUDA GPU has been the most successful and cost-effective: deploying at scale for the first time in O3.
  - Currently no cloud usage; no major technical obstacles (cloud looks similar to other shared resources), but logistics are unclear — we need to learn how to fund and manage collaboration usage of unlimited, metered computing when it's cost-effective.
- Data:
  - LIGO h(t) strain data is O(10TB) per IFO per observing year.
  - LIGO raw data (all channels, full sample rate) is O(1PB) per IFO per observing year.
  - No longer "big data" by 2018 standards — but non-trivial in a distributed HTC environment nonetheless.

3G Computing Resources

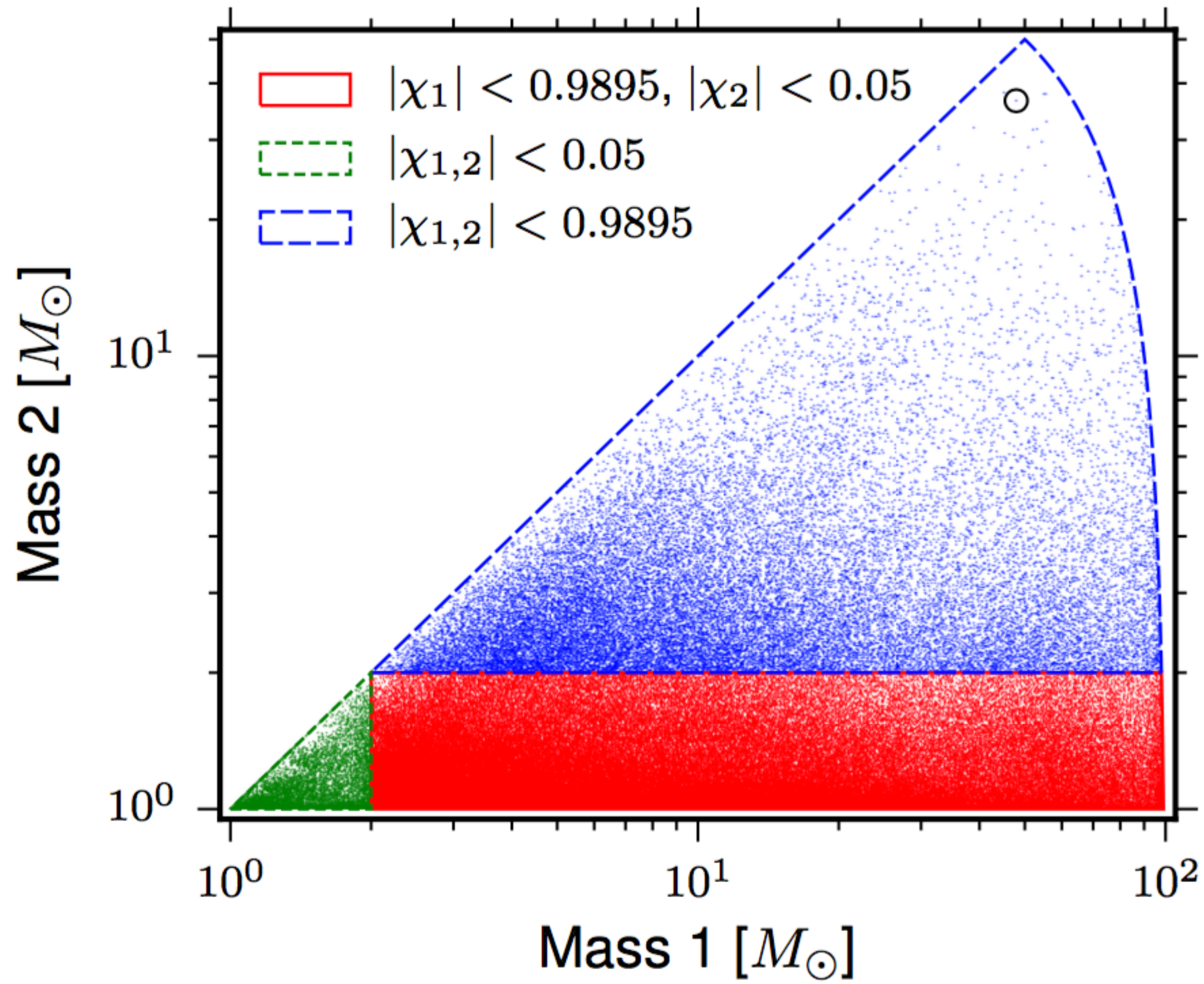
- Hardware needed for development, simulation, testing, and production data analysis:
  - **CBC Detection** - big problem!
    - Better frequency sensitivity -> more computing.
    - Dominant costs independent of predicted rates, thankfully.
    - Many "pleasingly parallel" algorithms — independent tasks, horizontally scalable.
  - **CBC Parameter estimation (Bayesian inference)** - bigger problem?
    - Better detector sensitivity -> more computing.
    - Higher detection rate -> more computing.
    - Some inherently sequential algorithms (e.g., MCMC).

3G Computing Resources

- Hardware needed for development, simulation, testing, and production data analysis:
    - **Burst Detection** - not a problem.
        - Burst searches should scale ~linearly with livetime, and # of IFOs. Unlike CBC, there is no scaling with detector sensitivity or low-frequency cut-off.
    - **Continuous-Wave (pulsar) Detection** - ?
        - It is less clear how CW searches will scale for 3G, but they have always had to live with many orders of magnitude less computing than optimal…
    - **Detector Characterization** - not a problem.
        - Unless 3G glitches are fundamentally more problematic than 2G.
- GW computing problems not very data-intensive.
    - O(10) TB/year of strain data to distribute to CPUs.
    - O(100) TB/year of data analysis results to store.
    - O(1) PB/year raw data computing is modest and raw data (probably) need not be distributed.

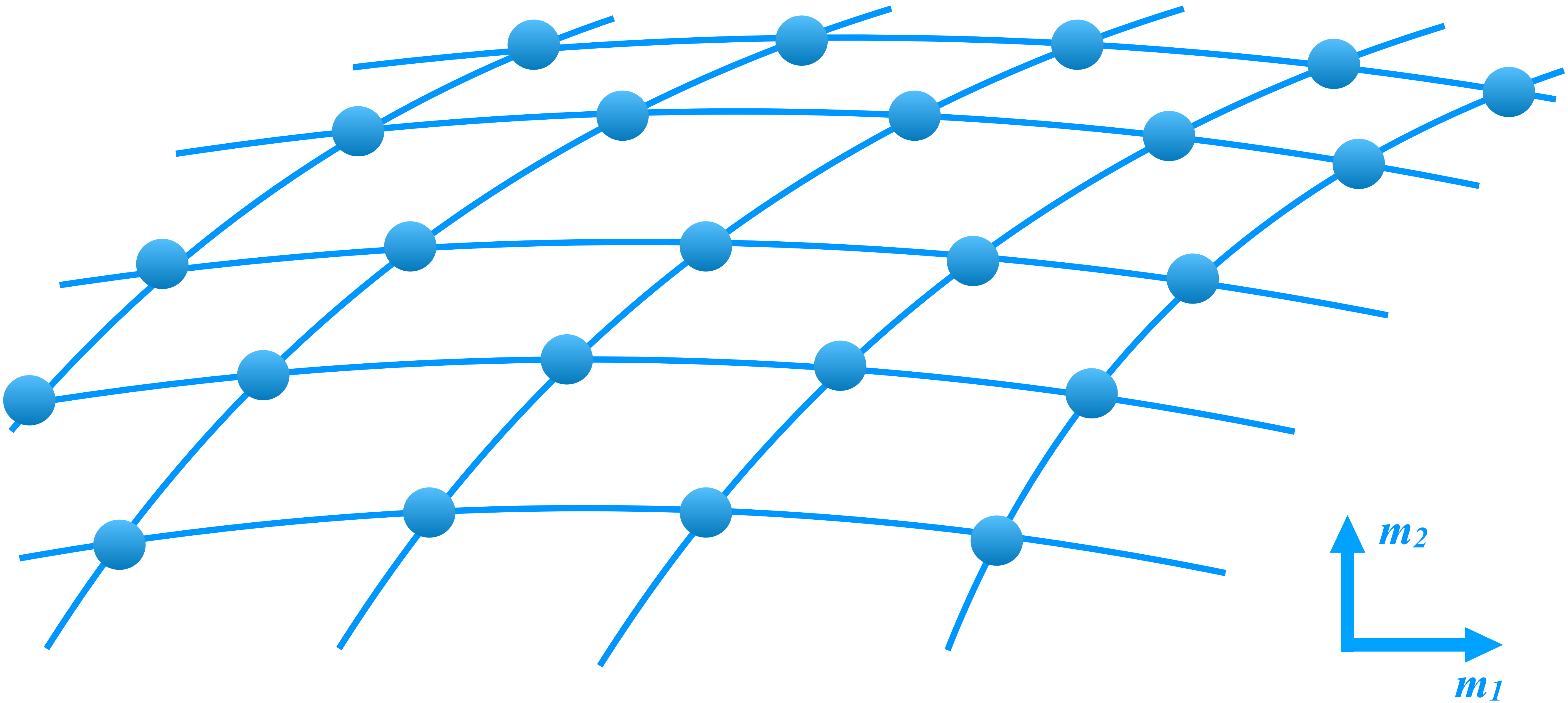
Matched Filter Search

- GW signals from compact binaries are well-modeled.
- Matched filtering of data in the frequency domain used to search for these signals — optimal for modeled signals in Gaussian noise.
- Since we do not know a priori the parameters of individual binaries we may detect, a bank of template waveforms is generated that spans the astrophysical signal space (mass, spin)
- We need enough templates to match the full range of signals we expect. 2G Template banks are currently made “dense” enough so that <1% of signals have a matched-filter SNR loss greater than 3% — this requires ~1M templates in a bank.
- Every chunk of time-series data from the detector must be compared to every template. This is the dominant computational cost of our search.



Detection

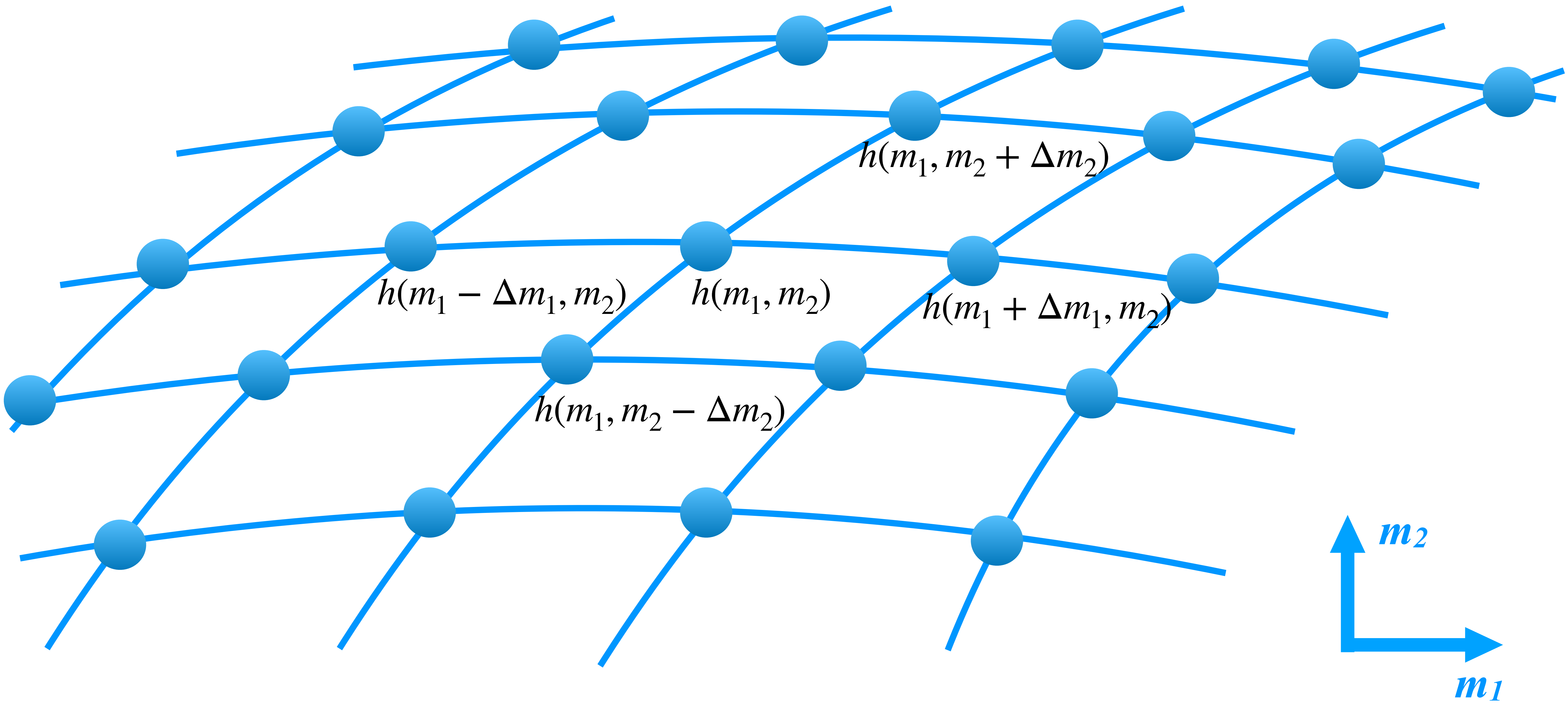
- Template Grid : assume 2D



credit: Ed Porter

Detection

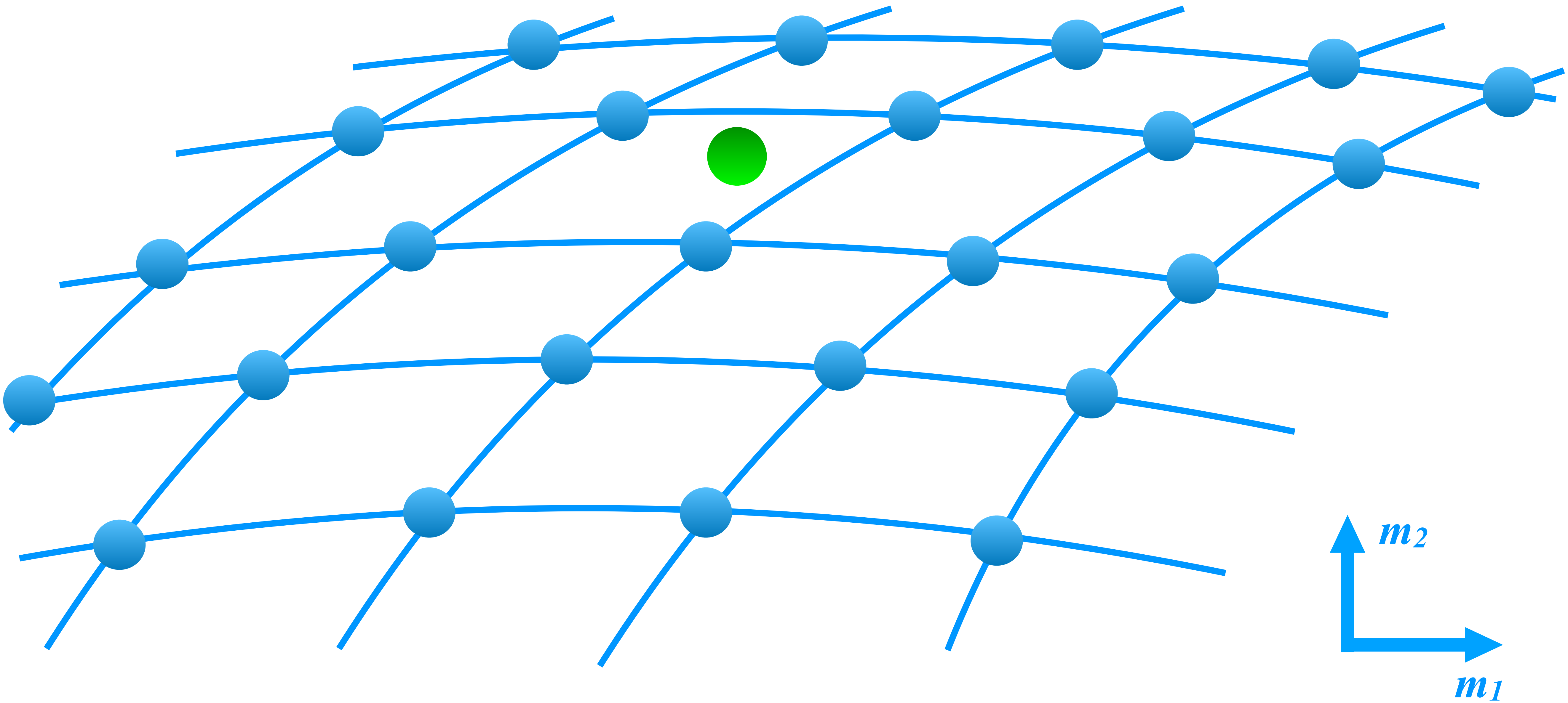
- Template Grid



credit: Ed Porter

Detection

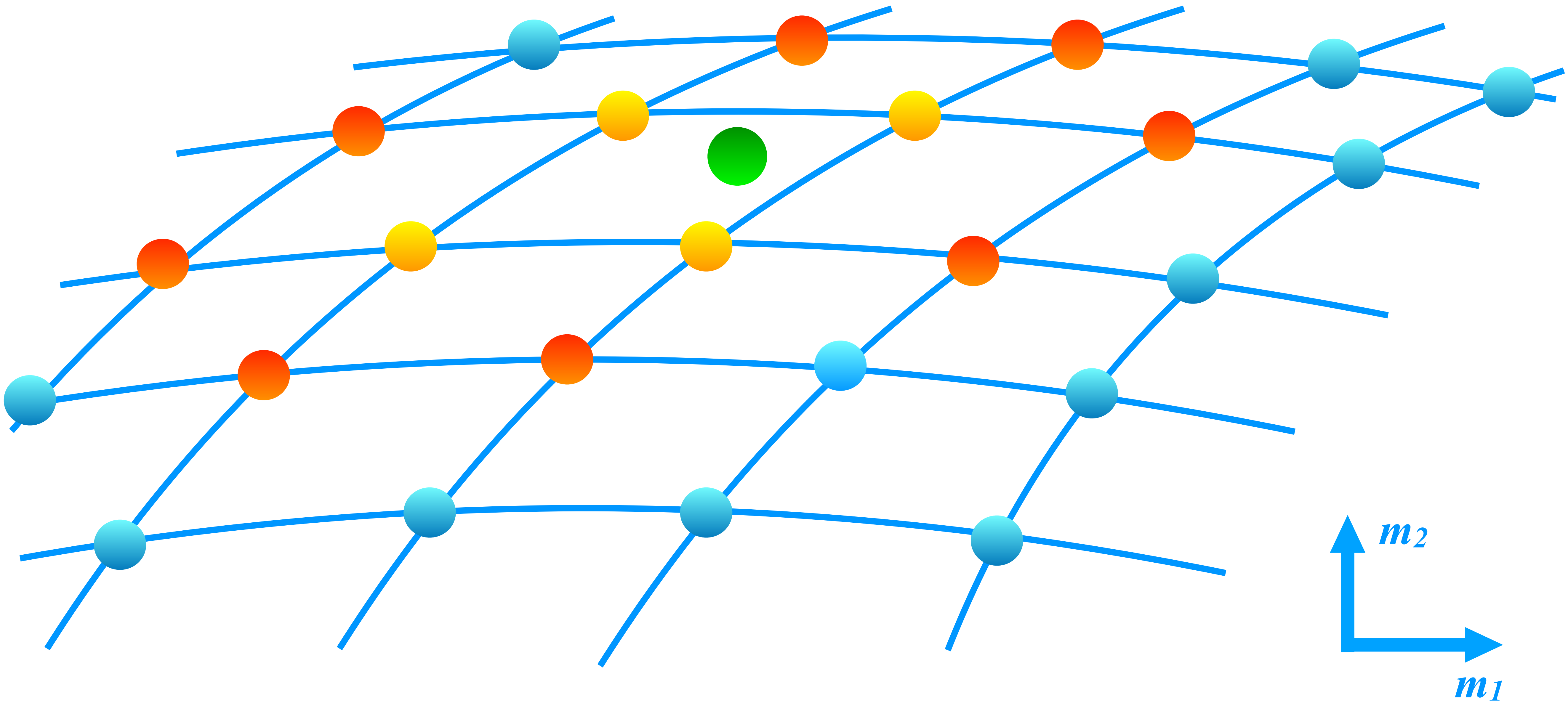
- Template Grid



credit: Ed Porter

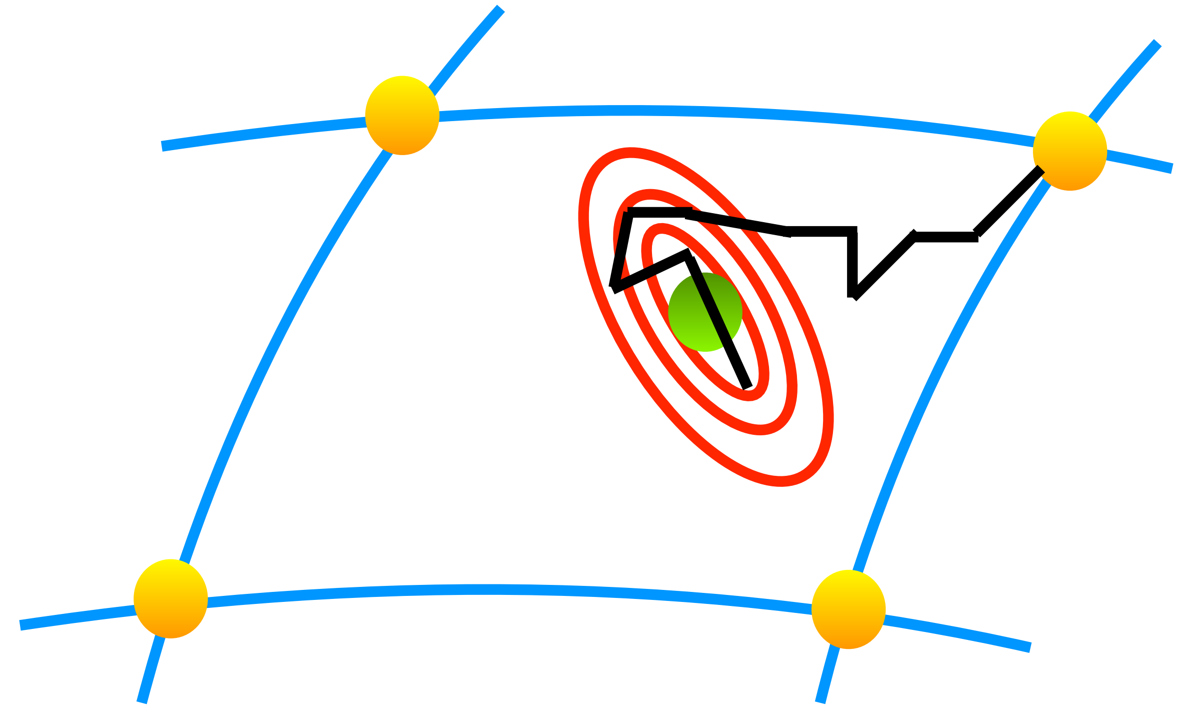
Detection

- Template Grid

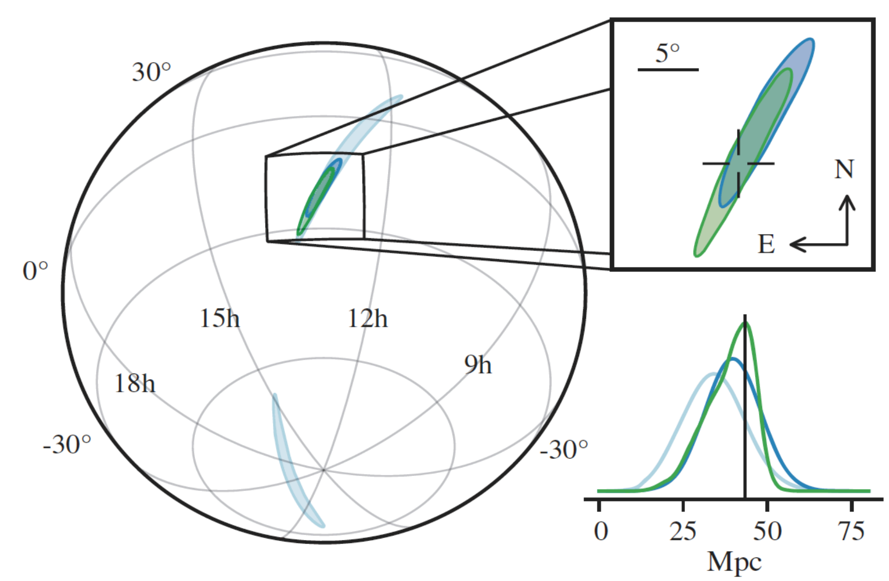


credit: Ed Porter

Parameter Estimation (PE)



- Bayesian Inference
- Extract astrophysical parameters
- Provide posterior distributions…
- …and confidence intervals



GW170817, B. Abbott et al, PRL 119, 161101 (2017)

credit: Ed Porter

Primer on GW data analysis

Detection

- Get the data
- Generate a bank of N templates
- Cross-correlate the templates with the data
- Find template with parameters closest to signal

Bayesian Inference

- Get the data
- Start with parameters of best-match template
- Use a stochastic sampler requiring M templates
- Extract posterior distributions

credit: Ed Porter

Primer on GW data analysis

Detection

- **Get the data**
- **Generate a bank of N templates**
- **Cross-correlate the templates with the data**
- **Find template with parameters closest to signal**

Bayesian Inference

- **Get the data**
- **Start with parameters of best-match template**
- **Use a stochastic sampler requiring M templates**
- **Extract posterior distributions**

The problem is that N and M are big!!!

credit: Ed Porter

Template Length

- Assume we are are going to target BBH, BNS and NSBH systems separately
- Assume a minimum mass of 1 $M_{\odot}$ for a NS and 5 $M_{\odot}$ for a BH
- Assume the longest template in any search/PE is equal mass
- Use a 3.5PN chirp time to calculate template duration
- Assume each template has a sampling frequency of $2f_{Nyq}$, where the Nyquist frequency assumes minimum total masses of 2, 6 and 10 $M_{\odot}$ for each search

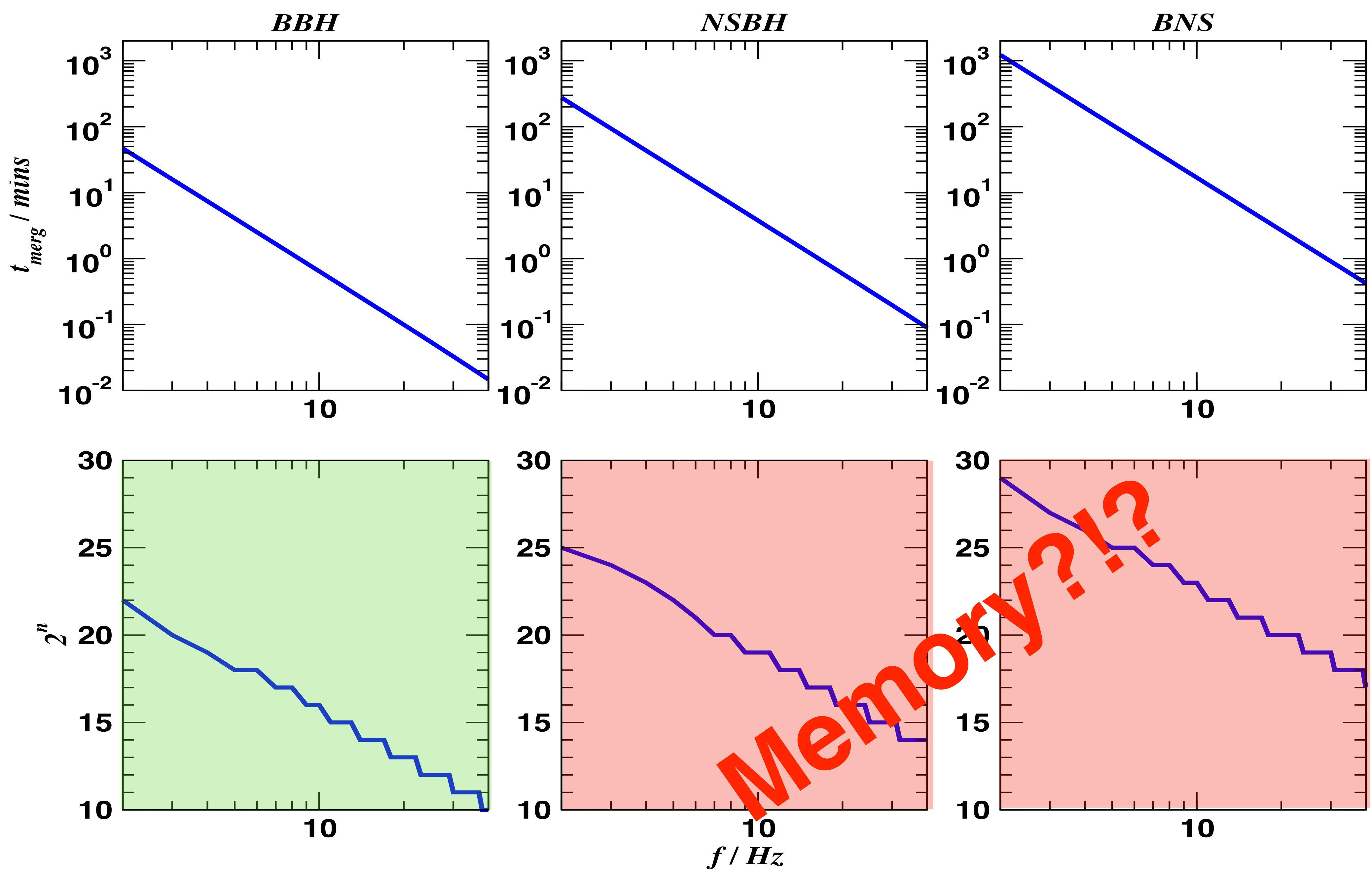
credit: Ed Porter

Template Duration

f / Hz	BBH	NSBH	BNS
30	2 secs	12 secs	1 min
20	6 secs	36 secs	3 min
15	13 secs	76 secs	6 min
10	40 secs	4 min	17 min
9	50 secs	5 min	22 min
8	70 secs	7 min	30 min
7	1.5 min	10 min	45 min
6	2.5 min	15 min	1 hr
5	4 min	25 min	1.75 hr
4	8 min	45 min	3 hr
3	16 min	1.5 hr	7 hr
2	45 min	5 hr	20 hr

credit: Ed Porter

Template Duration



credit: Ed Porter

Computational Power

- To get an idea of the necessary computational power (Schutz, 1989)
- To filter N templates of length F through the data, where F is given by

$$F = \frac{5}{32} f_{Nyq} \left(\pi f_{low}\right)^{-8/3} \left(2GM_{min}/c^3\right)^{-5/3}$$

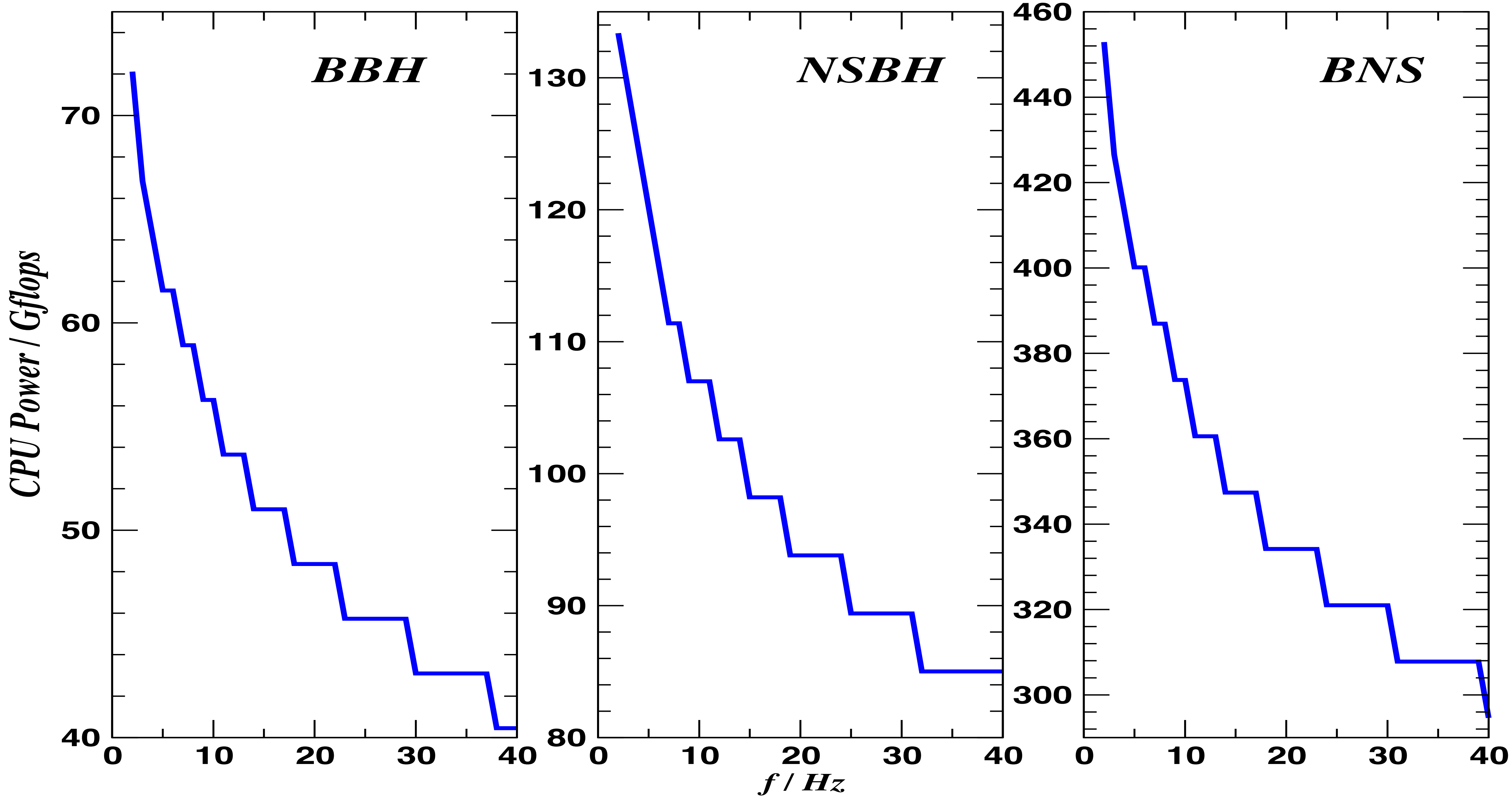
- requires a computational power, in flops, of

$$P \approx N f_{Nyq} \left(32 + 6 \log_2 F\right)$$

- assuming M_{min} has η= 1/4, and $f_{samp} = 2\, f_{Nyq}$
- *Assume my template bank requires 10^6 templates*

credit: Ed Porter

Template Duration



N.B. : CPU power scales linearly with template number

credit: Ed Porter

Other complications…

- From the GWIC 3G Science Case draft:
  - “With filters that are a day long, one needs to consider the fact that the detector antenna responses are not constant as the detector rotates with respect to the source. This means that the standard technique for maximizing over sky location will no longer apply. The most obvious solution to this problem would be to include the sky location in the list of parameters over which the template bank is constructed. However, it has not yet been possible to even estimate the size of such a template bank, other than to demonstrate that it would be at least two orders of magnitude larger than the template banks currently being used to analyse GW data.”

3G Parameter Estimation

- It’s not yet clear what our scientific requirements will be for PE in the 3G era.
- In O2, the LVC ran multiple, “deep” PE analyses on every viable CBC candidate, and we were constrained by available human effort as much as by computing. In additional, the computing costs of development, testing, and exploration of PE codes were greater than that of the final PE runs used for publication.
  - O(10M) CPU core-hours of computing needed for O(10) signals in O2. O(10k) CBC signals per year expected for 3G IFOs. **3 orders of magnitude increase.**

3G Parameter Estimation

- In an era of multiple signals per day, do we want and expect to run equally-precise PE on every CBC candidate, or will we have different degrees of PE precision (and therefore cost) for certain candidates?
- This will obviously be driven by the computing efficiency of PE codes (and the degree of automation possible in the detection->PE process), but the range of scientific scenarios is not obvious due to the many remaining uncertainties:
    - What is the lower limit of PE precision needed to realize our most basic 3G scientific goals, and what is its present computing cost?
    - What is the upper limit to PE precision we can currently achieve by applying more computing power, given the expected signal quality?
    - Where between these two limits do we get the maximum scientific benefit as a function of cost? Do we get diminishing returns for additional computing after a certain degree of precision?
    - Are there obvious “tiers” of precision and cost which we should target for different kinds of candidate signals? How should we define those tiers given the science we hope to extract from the data?
- These questions need to be explored iteratively by GW scientists over the next decade in close collaboration with computing experts, as new computing efficiencies are realized and new scientific goals are identified and translated into PE requirements.

3G Hardware Resources: Bottom Line

- A simple scaling of current matched-filter CBC searches and PE to 3G design sensitivities will require many (3+) orders of magnitude more CPU and RAM than 2G.
- **Moore's Law alone will not save us.**
  - Moore's Law is slowing already, and — barring a breakthrough — may slow further as we reach the physical limits of silicon transistors.
  - The performance improvements that *are* being delivered by CPUs and/or co-processors are becoming more complicated to exploit in software (with task and/or data parallelism). Many existing codes won't benefit fully.

3G Hardware Resources: Bottom Line

- **Something has to give**: barring an unexpected breakthrough, we may need to be clever about how and where in parameter space our searches can be less sensitive, to not miss what we care about most.
    - E.g., matched-filtering with sparser template banks and/or less faithful waveforms in the most expensive and/or less scientifically valuable regions of the parameter space, reduced-order modeling, unmodeled burst searches, machine learning, etc. **We don't yet know the right mix of approaches for 3G-scale analysis.**
- We have to identify new "sweet spots" of sensitivity vs. cost given our science goals.
- If the GW science we can realize will be limited by the efficiency of search and PE codes, targeted investments in data analysis development and optimization in advance of the 3G era may pay outsized scientific dividends.
- These investments should be driven by our understanding the detection and PE requirements needed for specific discoveries.

Software Infrastructure

- So the hardware resources for 3G will be orders of magnitude larger than 2G…
- How will these resources need to be organized and managed?
    - A centralized "walled garden" of homogeneous, dedicated GW clusters is unlikely to be funded.
    - a diverse worldwide GW collaboration is unlikely to select a single external computing provider (commercial or otherwise).
    - 3G computing optimization may require specialized hardware for different searches.
- Software infrastructure (aka "cyberinfrastructure") will be needed to make a more complex, dynamic network of resources to scientists usable and robust.
    - distributed scheduling, identity management and access control infrastructure, data analysis software development and testing tools, resource accounting and reporting tools, low-latency alert and coordination infrastructure, public data and code release infrastructure, etc.

Long-Term Challenge #1

- Increasing heterogeneity, complexity of computing platforms:
  - of **processing hardware** (CPU generations, GPUs, MICs) — due to the opportunities for cost savings, we MUST support multiple generations of CPUs, GPUs, MIC platforms and treat them each as distinct platforms — lowest common denominator code not good enough
  - of **providers** — internal to project, partners & collaborators, institutional, regional/national, commercial, volunteer
  - of target operating systems and **software environments** — containerization, etc. are tools to mitigate but aren't a silver bullet
  - of **batch/queueing** systems
  - of **storage and network** interfaces and capabilities
  - of **policies** for identity and access management, workflow prioritization
  - of **accounting** models and accounting systems
  - of **motivations and expectations** — mutual scientific/strategic interest, public or scientific recognition, financial or other compensation, etc. — and not everything is in a MOU, SLA, or contract

Long-Term Challenge #2

- Uncertain and discontinuous funding streams for **computing labor embedded in the collaboration groups** outside the IFO Laboratories:
    - The need to professionalize software development/engineering and to support increasingly complex computing environments demands more full-time professional computing expertise side-by-side with collaboration scientists (vs. part-time volunteer/service work by scientists).
    - Many of these are not strict software development or IT roles that can be outsourced beyond the project — they are hybrids of research computing, consulting, software engineering, and distributed systems development and administration roles.
    - These roles benefit enormously from institutional (project) "memory" — we pay dearly in time, money and quality when experience and relationships are lost.
    - Hard to recruit and retain career professionals on overlapping 1-3 year awards. Hard to find funding for this work; not always "transformative" science in and of itself, but needed to *enable* transformative science.
    - **This is an old problem** but is becoming more acute with the increasing need for these computing roles.

Implications

- Increasing heterogeneity, complexity of computing platforms drives:
  - need for better software engineering and testing
  - need for additional organizational expertise+effort in optimization, distributed computing (architecture, engineering, support), and computing management
  - need for better tools, services, and processes for sustainable optimization
    - how do we help scientists write code that can be run efficiently on multiple platforms?
    - compilers aren’t there yet, some higher level libraries can help
    - education and consulting for scientist/developers who are not first and foremost software engineers — provide value, avoid mandates
  - need to automate build and test for diverse h/w platforms — cloud testing is not there yet — not commoditized
  - need for more complex deployment, orchestration, instrumentation, and accounting of DA workflows

Implications (cont)

- Increasing heterogeneity, complexity of computing platforms drives (cont):
  - need for funding — and budgeting — models for “metered” computing by the minute or watt, in addition to fixed capital investments every few years…
    - what information (and safeguards, and levers) do funders, project leaders, and computing users need to plan, execute, and abort metered computing, and on what timescales?
    - as a community, we have no idea how to do this — different world
    - the technology to enable metered computing will be easy compared to the policy, management, and scientific workflow changes needed to manage the spending of money on such a profoundly different timescale
  - need for aggregate accounting of work on disparate resources — very tricky, simplifications and approximations needed but each one makes someone unhappy — resources are tightly coupled to issues of money, scientific capacity, and recognition, things people care about enormously
  - need for management of the complex scientific, social, political, and financial aspects of shared computing

The Good News

- The high-luminosity LHC experiments face similarly-daunting computational scaling problems over the same timeframe, but they have an order of magnitude larger starting requirements — they are helping to blaze a trail in optimization and distributed computing infrastructure which we can and should follow and collaborate on.
- More efficient techniques for background estimation may reduce CBC search costs by a factor of a few; machine learning-based searches may live up to their promise in non-Gaussian noise; etc.
- Even a slowed Moore's law could help improve the performance of sequential codes by a factor of a few over the next decade. Parallel algorithms will track Moore's Law closer than sequential algorithms.
- LIGO's recent experience has shown that the return-on-investment of dedicated, "full-stack" computing optimization by a team of GW+computing experts is overwhelmingly positive, in term of saved costs for expensive data analysis codes. We should plan on making this investment for 3G.
- A factor of a few here, a factor of a few there, pretty soon you're talking an order of magnitude more computing/$…

The Bad News

- More efficient data analysis codes needed for 3G will require more software development effort. This will require time from busy GW scientists and/or hiring more computing specialists. An investment in computing specialists can free up scientist to do science.
- Long-term software development costs may be higher for rapidly-evolving parallel hardware platforms (GPU, MIC, AVX512, etc.) than for traditional CPUs, given that parallel programming interfaces are less stable targets. Single-threaded CPU codes have worked on new hardware with minimal modifications for 30+ years.
- Data analysis on more distributed, non-dedicated computing grid/cloud platforms will require ongoing computing infrastructure investment. This will require time from busy GW scientists and/or hiring more computing specialists.
- **2G-3G computing labor costs may go up by a factor of a few.**

3G Optimization Approach: "The Whole Patient"

Iterate

- Scientific Prioritization and Scoping
- Estimation and Benchmarking of Computational Costs
- Optimization of Data Analysis Methods and Algorithms
- Optimization of Code Implementation and Libraries
- Compiler Optimizations
- Workflow Management Optimizations
- Development, Testing, and Simulation Process Optimizations
- 3G Computing Network Scheduling Optimizations
- Resource Supply Optimizations (make more cycles available)
- Workflow Portability Optimizations (expand usable resources)
- Hardware Procurement
- Pipeline Reviews including Computational Efficiency
- Documentation, Training, Collaboration and External Engagement

What are other large science projects doing in each of these areas?

Neglect nothing, focus on "bang for the buck" and where optimization effort can be most effective. Avoid adding burden where the payback is small.

Opportunities for non-GW Collaboration

- Better coordination of Distributed Scheduling, Computing Security, and Identity and Access Management efforts between LHC-HL and LIGO
  - Some of this interaction already happening, but the GW community should expand those efforts and have a stronger voice in forums where these matters are discussed.
- More collaboration on optimization tools and technologies.
- More collaboration on software engineering tools and technologies.

Conclusion

- The sky is not falling but we need to make smart investments in advance of, and during, the 3G era.
- Without major innovation, 3G data analysis (search and PE) naively requiring a 1000x increase in computing demand may need to be implemented on 10-100x faster hardware. Data analysis innovations, and expert code optimization effort for the most expensive search and PE codes will be needed bridge this gap.
- We need to plan for an increased investment in software development and computing infrastructure effort compared to the 2G era, or pay the price in science opportunity cost.
- Where current methods cannot scale, we need to understand computing cost vs. scientific benefit and make smart tradeoffs to achieve our goals with available computing resources.