How to Innovate

* reasons for lack of innovation in the last 113 years of aviation
  + large companies, large vehicles, each full vehicle costs a lot, any new design would bet the company
  + no money to experiment, hence no time
  + minimum cost rather than minimum –profit=cost-revenue (no time and money invested into finding out what the effect on revenue is, just see cost and discard idea)
  + no willingness to challenge assumptions, adherence to “rules”
  + schools and books teach status quo (i.e. assumptions) instead of just the basic theory, don’t let innovative ideas get a foothold
  + Uncertainty, don’t know how good this idea is, but current method is already 90% efficient, so why bother investigate? (Because 90% isn’t good enough. And investigation has led to new ideas with 200% improvement in performance)
  + wanting to know what vehicle will look like, not giving computer the design authority
  + Conceptual design group thinking they are superior compared to computers. No human has 16GB of RAM, and design requires a lot of RAM.
  + no perfectionism
  + people don’ t KNOW there is a better way of doing things. So they don’t even look for it. They don’t challenge assumptions, don’t question what they are being taught. Don’t read between the lines, looking for a way outside the box.
  + rigid mindset
    - people like to think in orthogonal coordinates (because they don’t have much RAM, they need to simplify)
      * e.g. optimize 2D airfoil, then apply that along wing (instead of thinking of wing as a 3D airfoil)
      * wing for lift, propeller for thrust, instead of wing for lift and thrust
  + people like to copy existing designs. Low risk. But also inside box. And if they copy incorrectly (e.g. bird wing) then it is also less efficient.
    - e.g. if you copy, thta means that you will never be better than what you are copying. You need to design vehicles from scratch. You can use existing designs as benchmarks, or initial conditions, but not as template.
  + long development times
    - only once every 15 years is a new plane developed -> only ask the question “what would an ideal aircraft look like?” once every 15 years
  + false sense of optimality (“birds fly that way, so it must be optimal”)
  + organizational inertia
    - engineers get older and older, will stick with what they did in 1950s. E.g. Boeing advanced development group still designing aircraft with Excel instead of Matlab.
    - Design tools have been developed to find the local minimum within specified constraints. Spend 10 years to develop a tool for finding an even better aircraft design inside the same box (= tube and wing aircraft). People don’t even challenge constraints, because they are not apparent, they are just there and accepted. Assumed to be a hard wall of box, even though constraint soft. Even if they think outside box (e.g. blended wing body), very costly to develop new tools for different aircraft configurations, large reluctance to pursue new design.
      * e.g. current optimization software takes as input taper ratio (constraint for Aluminium based aircraft which need ease of manufacture; carbon composite aircraft don’t need straight lines)
  + compartmentalization
    - if controls engineer realizes that there is less power consumed during a particular maneuver, he needs to mention it to the structural and aero engineers (and they need to run the numbers, instead of categorically saying no due to complexity) -> this communication likely has not happened because
      * the controls engineer did not know who to talk to (too large company, firewalls in between company, small company better, because shorter lines of communication, better communication, distributed intelligence working together, instead of as small silos)
      * did not talk to anyone because he thought he would be dismissed
      * he was dismissed
    - sometimes even in different companies. Jet engine manufacturers have been building wings with zero induced drag into their ducts for decades (possibly even without realizing it). And that advanced technology is ironically being bolted onto a wing trying to cancel the 7 tons of induced drag.
  + laziness, lack of ambition, easier to go with what has been done before
  + fear of failure
  + current method of running businesses:
    - Project manager’s performance is measured against cost and time (because they can be measured more easily), rather than maximize profit (i.e. which is more uncertain, but takes into account cost AND revenue increases from a superior, innovative product).
    - To make matters worse, this minimum-cost approach rewards a lack of innovation, which is the very foundation of profitability. The pressure to stay on time and on budget induces PMs and engineers to take the path of least resistance and least uncertainty, i.e. existing, proven technology that leads to the unavoidable 2-3% in performance and profitability (e.g. due to better materials, more streamlined production, and other incremental changes).
    - Often PMs or other people upstream in the organizational chart might also unnecessarily constrain the design by giving the engineers too narrow specifications (e.g. “build me a helicopter” instead of “build me a device that maximizes profitability of getting from A to B”). An example of that are the specifications for the Airbus Cargo Drone Challenge, which forbid the use of tilt rotors, and required one pusher propeller. Or the AHS human powered helicopter challenge, which required rotary wings.
    - If a radical new and innovative idea does emerge coincidentally (because it wasn’t asked for), it is often quickly dismissed as being “too complicated”, “crazy”, and “impractical”. The idea graveyard is littered with victims of the fear of running over budget and behind schedule. Since often appraising the impact on profitability is harder (read: more expensive and more time consuming) than judging the effect on cost and time (“too expensive” and “too time consuming”), the infant mortality rate is high.
    - If PMs’ performance had been measured by profitability in the past, there might have been a greater incentive to innovate and PMs would have been more inclined to invest additional time and money to establish an idea’s value. PMs and engineers might then have realized, that there is a method to innovation (see below).
    - The above mentioned coupling between determining profitability and cost is the hurdle to innovation. It is risky to invest money in establishing the profitability of an unproven, uncertain idea, because there is a perceived downside since it would depart from the current project plan (then, if profitability turns out to be zero of that particular idea (i.e. stick with previous design with 2% improvement), the extra research money was wasted).
    - With the right method, however, one can innovate without losing more money than one would lose otherwise. In this method, PMs are planning for innovation. This requires NOT knowing beforehand what the vehicle will look like. Instead, the engineers will tell the computer everything they know, and the computer will optimize the design globally. Such a method would require no extra research money, and would not change in the lower bound of profitability, because it contains all of the steps of the existing design process.
    - The method thus has the same lower bound on profitability increase of 2% as before. Crucially, however, it has NO DEFINITE UPPER BOUND. The lower bound is the same, because the current “box” of constraints are still included, and upper bound is uncertain because the “box” has been increased in size by the method (or stayed the same size in a worst case).
    - The key is that many different complete designs are considered nearly simultaneously (as long as program has to run), where the current state of the art is included in the design space (the 2%) as well as a lot of other designs we can’t even imagine (but the computer can).
    - In fact, there could be time and cost savings, because many processes can be carried out in parallel rather than in series. The series process is flawed in another sense, because each engineering group is tasked with defining the box within which the next group has to think. Any innovations made further downstream, even if justified and established as useful would be too costly to implement, because the upstream groups would need to repeat their calculations. A computer could do that effortlessly, however.
      * Current design process is in series:
        + conceptual design (30 people)

fuselage shape, wing configuration and location, engine size and type

final conceptual design layout

* + - * + preliminary design (100s of people)

wind tunnel

CFD

structures

control

instabilities

* + - * + detailed design (1000s of people)

assembly line

* + - * The new design process is massively parallel:
        + the preliminary and conceptual design are done in parallel, with every engineer writing part of a MDO program, and the optimization providing the output for the detailed design phase. The computer basically designs the whole aircraft, ultimately down to the last rivet.
* innovation method
  + Define the problem, go to the root of the problem, find out what the ONE physical mechanism is which is causing the problem in the first place. This has the added benefit of giving you focus, because usually there is just one problem that you need to fix
    - E.g. D0 is caused by adsorption of air molecules from a forward direction, and a desorption in a random direction -> 50% of particles have now stolen some of your forward kinetic energy -> solution: don’t let particles desorb in the wrong direction
    - E.g. aging is caused by damage to DNA -> build robots which know your DNA, and fix DNA in every single cell. Improve the body’s correction method, solve cancer and mortality at once
    - Aviation: problem is that you are paying for energy. Where is the energy going? Into the KE of the air. Well, let’s try to get it back for free. Harvest energy in air = molecular KE of air molecules.
    - Need to ask yourself what problem ALL machines in your field have in common. Remove those problems, rather than devising a machine that has the fewest problems, or is the most efficient. Devise a machine that has NO PROBLEMS.
  + Be a perfectionist. Don’t be satisfied with the state of the art. Even if you have already invented things that are much better than the publicly known state of the art. Try to find the Holy Grail. Because working on a shitty idea takes just as much time, just as much effort, as working on a great idea. So try to find the best idea.
    - E.g. what are other sources of energy that we can tap into? Make a list of energy sources, e.g. vacuum energy (E=mc^2), solar energy, air or water thermal energy. Obviously, the vacuum energy is the most useful to tap into, because we can also tap into it in space. Prioritize your efforts. First goal should be to harvest the vacuum energy. Second goal should be to harvest air energy (everywhere on planet). Etc.
    - “Always prioritize” also means prioritizing things that haven’t been invented yet. Work on the most important thing in the world. If that thing hasn’t been invented yet, then the first item on the agenda is to invent it.
    - I kept inventing, kept looking for solutions that my intuition told me was there (e.g. full wing flapping, macro kers). I was not happy with the solutions I had so far, didn’t give up when mathematics said it wouldn’t work (e.g. no performance gain, or no change in entropy in pot well), solution was right around the corner, but I was very close to not taking the extra step to look around the corner. NEVER STOP UNTIL YOU ARE SATISFIED. Also, NEVER BE SATISFIED. Be a perfectionist. It helps to know what you are looking for, and to know that there MUST be a solution out there. Then it is just a problem of “Show that …”. There is no uncertainty. Just plain old problem solving. You know there is a solution, and you know that you can find it. Be deeply upset by how things are done currently. Realize that this can’t be right (e.g. all humans will die eventually. That is just ridiculous.) This is how you will always be better than the competition. Think decades ahead of everyone else. Think about the big problems, think about the big picture. What really matters. Chasing the big problem will give you focus, it will give you time (because everyone else is chasing the tiny problems), and it will give you satisfaction. The satisfaction is the satisfaction of seeing a simple solution working (e.g. cp difference). Go for the clean solution, the simple solution. A solution that makes sense. Just like when solving a mathematics problem, you want the minimum amount of steps, the maximum clarity. Go for the deep insight (energy = temp + pot, change temp by changing pot). Go for the root of the problem. Get rid of the confusion, of the noise. See clearly. Clarity will give you satisfaction. Most problems are fundamentally simple. So solutions should be simple as well.
    - An optimum solution should be obviously optimal. This is a large part of what keeps me searching for better solutions. I’m not satisfied by the solutions that exist today (e.g. bird flight. Can’t be optimal. Does not agree with intuition (gravity force being vertically down does not favor rotation of lifting surface and lift force to an off-vertical direction)) or the solutions that I have found already. When you find an optimal solution, you will immediately know that it is optimal (e.g. light focusing) within the constraints you have set. Similarly, you will know that other solutions are not yet ready, not yet optimal.
      * e.g. for macro kers. Energy is there. It should be easy to access it. As simple as a wind turbine. Didn’t like idea of having to ionize gas.
    - Perfectionism is a result of a desire to not work on something shitty. e.g. cars got boring at age 14, quadrotors got boring at age 21, birds got boring at age 23.
  + Understand what you really need to do. Most advice is poorly worded.
    - E.g. “do things that don’t scale” should actually mean “think before you scale”, much clearer, and to the point. Not all things that don’t scale qualify as “thinking”.
    - E.g. “solve a problem from first principles” already implies a solution to a problem (i.e. implies a framework, and invites you to put a box around your solution space), but it should actually mean “**remove a problem**”.
      * It is only shitty solutions that introduce more problems than the ONE SINGLE PROBLEM you need to solve (e.g. Tesla battery dendrites during recharging. Solves sustainable fuel problem, but introduces a lot of other problems. That is a dead give-away that you are not removing the problem, merely solving it from first (or actually 1000th principles in the case of Tesla. Tesla is not solving problems from first principles. Eberhardt said himself that they didn’t want to do any research in Tesla. As a result, they had to solve a lot of engineering problems, without REMOVING the SINGLE, REAL problem of sourcing energy. They put themselves into a box from the beginning. Musk’s view of solving problems from first principles listing the raw materials that go into a car, find out what the cost are, and then estimating how cheap he can make it. That is not actually starting at 1st principles. That is starting at 1000th principle. Box already well defined)
      * you know when you have removed a problem. So this is better, more accurate, more precise advice than “solve a problem from first principles”
      * removing a problem does not require the introduction of new problems. E.g. hyperloop reduces (does not even remove) the problem of viscous drag, but introduces many more problems (tubes, cost, compressor disc, limited applicability (only works on tracks, between connected cities, pre-defined, and constrained by billions of dollars))
      * prior art has tried to reduce problem, e.g. reduce drag, reduce energy wasted. But energy is always conserved by definition. Energy is only “lost” “irreversibly” if you let it leave your control. The energy is still there, just in the atmosphere, or in outer space. So all you need to do it “get it back” from the atmosphere.
  + expose yourself to problems (read book, take lectures, e.g. m\_dot problem found when reading helicopter text book, looking at graph for induced power, was immediately apparent that we need large Vc, can make it artificial (looking at equation can be equally useful for finding out which parameters you need and can modify to get better performance), or bird flapping inefficient due to rotation, non-uniform velocity along wing etc.) and expose yourself to tools (save peculiar things you learn about (thermal radiation, osmosis, Casimir effect, atmosphere temperature gradient), save them in your “tools” or in your “peculiar, potentially useful physical mechanisms” folder in your brain) and then let your brain to the optimization, let your brain combine the tools which are aching to be applied to problems to real problems. DON’T LISTEN TO OTHER PEOPLE. THEY ARE, AS HUMANS, PROGRAMMED TO SHUT DOWN, DISCOURAGE NEW IDEAS, VIEW THEM AS A THREAT.
    - I was very close to inventing a couple of things in undergrad which I invented later. Mostly because my intuition was right, but I didn’t think one step further, too busy working on homework problems, and when I shared intuition with friends, they were the opposite of encouraging (as humans tend to be when hearing about new ideas, and don’t want them to be true)
      * thermal radiation modified using optics
      * osmosis used for pressure
      * fixed wing flapping (glider ladder)
      * bounding flight in grad school (also be careful who you share your intuition with – intuition may be right, friends may be willing to risk friendship and steal your idea)
  + pick the right problem
    - should be a big problem (>$1tr in revenue), like energy, transportation, tourism, health care, communications, real estate, computing
    - should have many free variables that you can control (problem should be inside a big box, lots of freedom to innovate, large design space)
    - should be a complex problem (i.e. in the past, without computers, people have made unnecessary assumptions to simplify problem and shrink the box to think inside -> using computing you can find better optimum), e.g. aerodynamics
    - you should be able to contribute (match your education, but not that important, because all you need are the basics, and the less you know about the status quo and the existing assumptions, the better)
  + be a perfectionist and be an optimist, which also provides the required stubbornness or tenacity
    - perfectionist to be unhappy about the status quo
      * bored of cars at age 12, bored of quadrotors at age 21
      * dissatisfied with solutions that are not simple, clean, and easy to build (e.g. can build in garage) and easy to explain and understand, obviously globally optimal
      * also helps with being good in school and learning the skills you need to understand problems, and gathering the tools you need to remove problems. Education VERY important. Don’t drop out of school. Facebook, Theranos, Microsoft terrible at innovating. Can of course learn skills to innovate later, but they rarely actually do that.
    - optimist to know that there is a solution, to know that you can find a solution
      * second law is wrong because microscopically it is no problem to harvest kinetic energy of gas molecules -> so now it’s just a matter of trying to find a more practical solution, preferably macroscopic solution so that it is easier to build (e.g. avoid ionization, because impractical, inefficient)
      * in exam or on homework it is much easier to solve a problem that says “show that \_\_\_\_”, i.e. you know there is a solution, and you know what it is. Then you can just brainstorm knowing that there is a solution and knowing that you will know when you have found it. Much easier to commit the time and money to such a problem, e.g. many years
  + write your own independent claim
    - that way you force yourself to define the problem and describe the solution, and explain why solution works,
    - find what the crux or the core of the solution is, i.e. the key part of your invention which is part of (i.e. the common denominator) of all embodiments of the invention
    - find what the problem is in the first place (ideally start with the problem, though)
  + it’s about asking the right questions
    - Where the fuck is all the energy going, and how the fuck do I get it back?
    - Where on earth is all the energy going, and how on earth do I get it back?
  + What you need to do is pose any problem you have as an engineering problem. All engineering problems can be solved.
  + use simplest mathematical model first (e.g. idealized case, instead of calculus of variations)
    - e.g. induced drag heli equation, look at terms, get intuition, optimize
    - e.g. wing drag equation, look at terms, get intuition, optimize
    - e.g. infinite wing span 2D case -> then try to make that work in 3D
    - e.g. try to understand what is actually happening on the molecular model, MODEL the system in your head (first idea started 2014 by modeling bird flight in head – was annoyed at variable speed along wing), program your intuition, let it do the work, and spit out better ideas
      * Create equations on your own, even if they already exist, derive them on your own from first principles -> understand model, system, understand what the problem is, what is causing inefficiencies?
    - e.g. realize that aircraft in flight are similar to aircraft under water (similarity rules) -> model of aircraft with a large viscous drag and plenty of mass to move around -> try to overcome viscous drag first
    - maximize insight per unit time; a simple mathematical model will give you 80% of the results, if they are not worthwhile, you can look somewhere else
    - later, use simple model optimum as initial condition for more advanced theory to refine global optimum rather than some local optimum (multi-fidelity analysis)
    - Simple theories help you build an intuition. Intuition is crucial, because that is the programming language of your supercomputer, your brain. You MUST program your supercomputer. It will work in the background, and then output a fully formed solution when you think about the problem next. Programming entails
      * internalizing the tools you need (e.g. physical laws and trends e.g. “want to maximize mass flow” “want to maximize area”, constraint equations (but careful, constraints not always required, or necessary), e.g. irrotationality => understand Helmholtz laws)
      * **reading patent classes incredibly useful** method to GIVE YOUR BRAIN THE TOOLS TO INNOVATE
        + read entire CPC patent classification scheme (and other schemes) learn about tools such as winglet, annular wings, buoyancy, Magnus effect, etc.
        + need to maintain that database of tools (e.g. read them once a month)
        + then make sure YOU and your optimization software can use those tools (find out how they work, what they do, what they can be used for)

make or find in literature simple mathematical models of each tool (e.g. momentum theory of ducted propeller)

find free parameters

plot cost vs those parameters

find where cost is zero, and intuitively understand why, then apply that to all problems in aviation

* + - * also need to internalize problem you are having (induced drag ⬄ induced power ⬄ closed vortex loop)
    - if it is a complicated system, start at the heart of the problem e.g. before trying to optimize trajectories, try to understand how lift works intuitively
      * e.g. if you want to maximize range, look at Breguet range equation
      * then look at L/D
      * then look at D
        + lift induced drag (e.g. vortex theory better than conventional momentum theory)
        + skin friction drag

try to understand them, why they are there, and under what circumstances these can be made zero, take the most practical one

* + use as many different mathematical models to describe the same problem as you can
    - e.g.
      * momentum theory
      * energy conservation
      * potential flow
    - then find the optimum in each (using your brain (i.e. intuition), or using optimization software) -> compare the optima -> one model might then give you an insight into other models -> helps in finding global optimum
    - this has worked beautifully, with one model finding completely different optima than the other one, and both giving insights when implemented in the other models
    - works, because it is similar to a coordinate change or mapping to different basis vectors => in some cases it reduces the complexity of the model (e.g. momentum theory far wake instead of induced drag using vorticity tells us we need to maximize m\_dot -> several new designs that do that) and in other cases it allows the generation of a new optimum (i.g. don’t need to move any fluid downwards from rest, rather use potential flow vortex theory to move it up first and then down -> no induced losses)
  + explore the limits
    - similar to idealized case above, but ask “what if wingspan is infinite”, then make it happen using other tools in model
  + READ to learn the theory, the equations for the simple model
    - Most textbooks will be wrong, or contain wrong assumptions (otherwise your invention would have been made decades ago). Use books and experts only to learn the basics. Only takes a couple of days. Don’t have too much respect for a field you currently know nothing about.
    - Read books carefully
      * in pursuit of a simple model, i.e. only relevant pages
      * building intuition and understanding, rather than memorizing, or referencing a paragraph’s location in your brain for future reference (like a pointer)
      * with perfection in mind
    - Read between the lines. Careful, most assumptions are not stated explicitly. They are just, you know, assumed.
      * What if we removed this constraint?
      * Is this assumption necessary?
      * How can I explain this with intuition?
  + find optimum using mathematical model (for optimization you need a cost function, a constraint function, and an optimization algorithm)
    1. Find the cost function. It should always be profit ($) at the end, because you are running a business. But it can be anything suitable temporarily for simplicity, e.g. energy consumption/unit distance travelled.
    2. Find out what the most lenient constraints are.
* This should be the FIRST QUESTION YOU ASK.
* This is crucial. Make only the minimum amount of assumptions. This is how you think outside the box.
* This should give you a HUGE design space (biggest possible box)
  + e.g. have airbus A320, Chinook Helicopter, a fish, and cyclogyro in same design space -> find the global optimum
  + need to make sure that ALL possible designs are in the design space. Can double check, by reading patent classes (e.g. CPC classification scheme)
* Pretend you are the first person to try to derive XY or design a plane or helicopter. Pretend there is no trodden path.
* don’t make ANY assumptions or add any constraints based on feasibility or practicality
  + Those decisions can only be made AFTER you know what the theoretical optimum is, because you MUST know what benefits you would get in return from the cost of complexity. You can’t say from the get go: no XY feature, because it’s too complicated to implement. If it turns out that you get a 700% improvement in performance, then you might still make a profit despite the cost of complexity. And THEN you can even find methods to make XY less complicated. If you fail at that, you still have the option of not doing it, but at least you know what the cost-benefit trade-off is.
* The only constraint should be the pursuit of perfection.
* It helps to find the theoretical optimum, to realize what happens when constraints are removed
* Helps to pack as many variables into equations as possible. Make them time and space dependent. In the limit, could go down to atomic level, give each atomic particle the 6 DOF (translation, rotation) in inertial space.
  + e.g. free wingspan, free aspect ratio (what helped me find P1 -> let algorithm solve and optimize over high DOF system, found new optimum, thought it was a software bug at first)
  + to get more DOF, it helps to write down existing equation, then put an x in front of every parameter, where x can be any number between 0 and 1 -> make parameters that people conventionally think are fixed free. That also applies to “constants”, e.g. Planck constant, gravitational “constant”, heat capacity cp. Increase the design space that way. Note that x can be time and or space dependent.
    - e.g. Lift is now x\*Lift -> variable Lift
    - e.g. number density is now x\*n -> variable flow rate through filter due to repulsion of filtered objects
    - e.g. find equation, find ACTUAL parameters that govern equation. Then free up some parameters (gain ability to go outside the design box), then use new nullspace, move through nullspace to go to a point in the performance space that was previously unreachable, e.g. rate of diffusion through a hole depends on temperature, number density, and many other parameters. Change temperature locally to change density without changing the rate of diffusion. Or change the rate of diffusion by changing the temperature locally at entrance to channel, and changing it back within channel.
* Don’t compartmentalize. In the old days of aircraft design, a team of conceptual designers would freeze an overall vehicle configuration, and then all the other groups of engineers would have to design their part of the vehicle (e.g. wing) within the given shape and weight constraints. This is not ideal. Instead, let every group design a detailed, parametric model of their area (e.g. aerodynamics), and let an algorithm optimize and return the final design. Need to model EVERYTHING at once. Profitability depends on everything, so don’t start with a budget (you can apply one later, after you know the effect it has on profitability), don’t start with a preconceived vehicle design, let the optimization algorithm find the best combination overall. The computer needs to know everything to find the global optimum.
* Look at equations for insight and intuition
* Examples
  + for transportation, the theoretical optimum is ZERO energy/distance travelled (e.g. satellite in space) -> this tells us that the constraint or assumption of there being an atmosphere can be ignored -> and leads to interesting new designs, such as a vacuum tube shuttle on earth (but final model should have the profit as a cost function)
  + internal forces should cancel on average, not necessarily at every instant
  + there can be relative motion between body parts that were originally assumed fixed to each other
  + irrationality constraint leads to there always being a closed vortex loop -> all you need to define is the optimum shape and motion of that loop
    1. Use (or write) an optimization algorithm that can find the optima in that space. There are many off-the-shelf algorithms in software packages like Matlab (e.g. fmincon)
    2. Find the global optimum (that maximizes profit) and learn from it (beware local optima, try several different initial conditions)
       - does it agree with intuition?
       - what happens when constraints added?
         1. useful for finding new local minima
         2. work backwards, reduce the cost (and complexity) while trying to keep the benefit high -> maximize profit, find global maximum
    3. Improve the mathematical model (more detailed design)
       - add a cost model
       - add a structures model
       - better aerodynamics etc.
    4. Find the final, detailed global optimum
    5. Patent all optima (license locally optimum designs, that are still better than the status quo, build the best one yourself)