

## Cover Letter

I have a specialization in Machine Learning and other related fields.

Also, I have developed a new **global optimization method that is better than Multistart** (without basins of attraction).

(<https://medium.com/@pushkarevvaleriyandreevich/gradient-descent-that-we-must-have-5a4542e218a0>

Read from What to do (with gradient descent)? )

That method performs **2-3 times better than Multistart**.

And if I add basins of attractions or other optimization techniques, I will outperform even GlobalSearch and others.

Also, I have developed **a communication technique that outperforms what existed in the Linux kernel** (with a few security issues at the time).

50 million calls per second versus 5 million calls. Or more than 10 times.

**(Part 1 - Library and main concepts:**

<https://medium.com/@pushkarevvaleriyandreevich/making-libs-drivers-verilog-endpoints-for-custom-hardware-for-windows-linux-f8cf2d1e8efe>)

**(Part 2 - Hardware endpoint**

<https://medium.com/@pushkarevvaleriyandreevich/making-libs-drivers-verilog-enpoints-for-custom-hardware-for-windows-linux-8a6f580aa3f3>)

Yes, I hear that **about 3–10% improvement in one discipline is enough for a PhD**, not 2-10 times).

Basically, if I have better Global Optimization method - I can grow less tissue variants and find optimal parameters for bioprinter faster. Also, I have some heuristics that allows to find myosin/actin-like proteins that works better.

### Why I'm out of programming.

The basis is simple - nowadays we (humans) even have software that can program better than us (DreamCoder (DARPA)).

Also, all translation from Lisp to any language (C#/C++/Rust or even C++) can be done with Transformer (ChatGPT, etc) or analogs.

So I can say that within 2-5 years there will be no any programmer vacancy (at least in non-specialized cases).

### Why i'm in computation biology

In other hand, all basic needs (manufacturing of molecules/proteins and medicine) lies inside computational biology.

But there is a trick - we cant add unlimited number of chemical compounds to one-cell organisms forever.

(because some compound can interact with each other out of optimal reaction paths).

In many cases, we need to grow a several layer tissue to accomplish biological production of some molecules.

Also, with AlphaFold we can make almost any proteins (and validate them in quantum chemistry simulators). But it's a quite limited approach.

So studying of a growing tissues will be actual for a long time (and even connected with some manufacturing needs).

It's a material part. (I don't need to end my education without any perspectives).

### What I plan to build?

In short - A global optimization pipeline with filters based on simulators and some text heuristics.

I have better global optimization algorithm so I can spend less time to grow a tissue variants.

Basically, project have directions:

1. Proteins inside cell - nowadays we can change and recalculate even chemical reactions/signal chains for artificial muscles, and change the "fuel", for example, and get most inadequate muscles :)

Or simply declare/find some set of rules to get Myosin that worked. And try to find Myosin analogs that can have **more force/length change during ATP binding/unbinding** in quantum chemistry simulator. We can **simply change all proteins to versions with lower size** (and spend less energy to move molecules nearby). Or even **change to bigger ones** (and move more molecules nearby with lower speed). And **modify trans-membrane proteins and binding sites of myosin (improve selectivity)** .

And of cause change proteins to versions that can resist to denaturation even with the lack of lipids (or even with presens harmful lipids that can change protein structure and properties). We can achieve that simply using **reactions between amino-acids of RNA**. And greatly decrease number of binding sites for external site of trans-membrane protein (or even **fix external structure**) **within wide temperature and pH range**.

Than we can build a simple 3-D model and set some percolate/don't percolate filters to know what exactly protein forms cant do some states of process (bind/unbind) and so on.

2. proteins that responsible for tissue connection - proteins between cells are important too, they can even define mechanical properties of tissue. The only question - what **proportions of myosin/actin/Z-disk protein is the best** for our goal. And how to get that proportions.
3. lipidation - proteins interact with other proteins/molecules, and this can change their properties too
4. higher abstraction level - how tissue grow, we have many variables, patterns to add chemical compounds/make mechanical stimulations and even change proteins outside the cells

And this is the field of Global Optimization and parametric model identification

First, simply running global optimization is cost-sensitive, we have too many solutions, and checking all of them is almost impossible. (Even for growing a tissue)

But we can:

1) Add some approximations of a solution (with a neural net that is trained to act like a bloom filter or even an SVM (that can act like a set of rules)).

In our case we can add simulations from Nvidia Microbes

2) Make many bloom filters to reduce the possible solution space.

Here we can take into consideration my another favorite part - NLU

3) Run developed global optimization within that area of parameters(that is already 2-3 times faster)

So we can simply grow 2-3 less tissues to find an optimum. We will save a lot of resources, time and money.

With Nvidia Microbes we can save even more.

And that will work better than GlobalSearch from Matlab (and other algorithms):

1) To find a basins of attractions we must compute derivatives (with 10-dimentional space of parameters we can take all space in laboratory :) )

2) Because we have not only basins of attractions but also approximations based on neural nets

3) We have a better global optimization algorithm beneath all of that

4) We can even add some NLU features to make more conditions/filters

Also, I must transfer all of that to OpenCL/Tensorflow to make it usable.

And I can solve any problem, even within computational biology, faster than others.

# **Resume**

## **Pushkarev Valeriy Andreevich**

Male, 32 years, born on 14 February 1990

Contacts +7 (951) 392-73-50

PushkarevValeriyAndreevich@proton.me — preferred means of communication

Novosibirsk, willing to relocate, not prepared for business trips

## **Middle C# Developer**

Specializations:Programmer, developer

Employment: full time

Work schedule: full day

## **Work experience 5 years**

January 2021 — currently

1 year 8 months

Freelance C# Backend developer WWF, FPW, .net core developer

October 2019 — april 2020

7 months

Siberian Networks

[www.sibset.ru](http://www.sibset.ru)/Telecommunications, Communications

Middle c# developer Development and testing of an ETL system for integrating a new BPM, development of a text template for generating classes according to Oracle tables (blToolkit). DeadLock-Free architecture due to diversification into only 2 consecutive tasks (Loading-preparing objects and validating-unloading)

Loaded data from all systems. CRM has not yet begun to be introduced.

Support for the old CRM (viewing and optimizing query plans, adding indexes, archiving obsolete data)

January 2018 — april 2018

4 months

Kaspersky Lab [www.kaspersky.ru](http://www.kaspersky.ru)

IT, System Integration

Middle C# Test Automatisatation Engineer Test refactoring. Connected Selenium. Brought Selenium and UI Automation entities to single classes. Made a single service, brought everything to a single interface. Increased the speed of tests by 2.5 times. Test support.

April 2016 — october 2016

7 months

GIS2gis.ru

IT, System Integration, Internet

Middle c# developer Refinement of the UI of the BuildMan system, adding

notifications via RabbitMQ and WCF (microservice), Adding a search for the state of builds by various criteria. Debugging the server side - transferring WWF from a custom provider to get workflow data to SQL, eliminating many Race-Conditions due to incorrect use of WWF.

February 2015 — march 2016

1 year 2 months

BCS Limited

Junior C# developer

Support for an ETL system for integrating and aggregating transaction data. Development of a new MDM solution (according to SOLID) and transfer of aggregation from SQL sheduler scripts to C#. Database size 1.5 TB, more than a million transactions per hour. Reporting development,

### **Key skills**

Quick learning

### **Driving experience**

Driver's license category B

### **About me**

Can solve problems like this: <https://github.com/m4rs-mt/ILGPU/issues/639> )If you have doubts about The Hardware/Software Interface or Washington University (from asm to C) - i can send pdf )

Have many certificates in AI, Electronics, and so from Washington University, Berkley и MIT with distinction. Even have Microsoft 98-361 certificate

### **Higher education**

2013 North-Eastern Federal University, NEFU, Yakutsk Electronics, Automatics and Electric drives design engineer

## **Electronic certificates**

Stanford Machine learning

Berkley CS169.1x: Software as a Service

MITx 6.002x: Circuits and Electronics

Stanford cryptography I

Washington university The Hardware/Software Interface

Berkeley CS188.1x: Artificial Intelligence



## Transcripts of all degrees

### Higher education -

#### Institute:

Federal State Autonomous Educational Institution of Higher Education "M. K. Ammosov North-Eastern Federal University"

#### Specialization:

Electric drive and automation of industrial installations and technological complexes  
(5 years total)

#### Disciplines:

Discipline Name:	Total Hours:	Final Grade(ECTS):
1. English language	342	A
2. Physical culture	408	A
3. Domestic history	125	B
4. Philosophy	125	B
5. Russian language and culture of speech	115	Pass
6. Jurisprudence	60	Pass
7. Economic	87	Pass
8. Mathematics	700	A
9. Computer science	300	Pass
10. Physics	500	A
11. Chemistry	150	B
12. Ecology	70	Pass
13. Physical foundations of electronics	172	Pass
14. Theoretical mechanics	180	Pass
15. Descriptive geometry. Engineering graphics	192	Pass
16. Materials science. Technology of structural materials	140	B
17. Mechanics	180	A

18. Theoretical foundations of electrical engineering	340	A
19. Electrical machines	170	B
20. Electrical and electronic devices	170	B
21. Metrology, standardization and certification	70	D
22. Life safety	180	Pass
23. Electric drive	150	B
24. Culturology	135	Pass
25. Sociology	135	Pass
26. Mathematical modeling in electrical engineering	80	Pass
27. Fundamentals of programming	70	Pass
28. Electronic and microprocessor technology	60	B
29. Circuit design of electric drive control systems	75	A
30. Introduction to the specialty	30	Pass
31. Introduction to electrical engineering	45	A
32. Psychology of business communication	70	Pass
33. Foreign language in the field of professional communications	200	Pass
34. Automation of a physical experiment	200	Pass
35. Theory of electric drive	220	A
36. Control systems of electric drives	300	A
37. Elements of automation systems	150	B
38. Automated electric	150	B

drive of standard production mechanisms		
39. Economics and organization of production of electric drives	100	B
40. Theory of automatic control	150	B
41. Converter technology (power electronics)	130	B
42. Power supply of industrial enterprises	150	A
43. Microprocessor tools in electric drives and technological complexes	130	A
44. Electrical equipment and automation of production facilities	130	A
45. Modeling of electric drives and automation systems	170	A

**Total hours: 7806**

## Online certificates

Course Name	Duration	Description	Grade	Link to certificate
6.002x: Circuits and Electronics	4 month	All basic electronics - from resistors to kirgoph polynomial, complex circuits and electromagnetic receivers/transfers	Pass	<a href="https://s3.amazonaws.com/verify.edx.org/downloads/ea9314d9749142d5951369866a692cb8/Certificate.pdf">https://s3.amazonaws.com/verify.edx.org/downloads/ea9314d9749142d5951369866a692cb8/Certificate.pdf</a>
Stanford Machine learning	4 month	Machine Learning By Andrew Ng This is reissued certificate. Now on Coursera there are about 3 courses each 3-4 week long. All basic thing about machine learning.	96.24%	<a href="https://coursera.org/share/4bc6e73fa9ecaa7fea29f01f0089c494">https://coursera.org/share/4bc6e73fa9ecaa7fea29f01f0089c494</a>
Stanford cryptography I	3 month	All basic cryptography from Caesar down to block ciphers, PGP, and all basic attacks.	100% with Distinction	<a href="https://coursera.org/share/fbeae640fad1de29171275777c6531db">https://coursera.org/share/fbeae640fad1de29171275777c6531db</a>
The Hardware/Software Interface	3 month	This undergraduate course covers basic principles of the hardware/software interface including hardware architecture, memory hierarchy, x86 assembly programming, and C vs. Java concepts	83.09% with Distinction	No link to certificate (
Berkley CS188.1x: Artificial Intelligence	2 month	MDP, BFS, DFS, A* and all other basic concepts of space exploration and agent building.	Pass	<a href="https://s3.amazonaws.com/verify.edx.org/downloads/f2ee5fa119164fbab3b1e97747b9d0d9/Certificate.pdf">https://s3.amazonaws.com/verify.edx.org/downloads/f2ee5fa119164fbab3b1e97747b9d0d9/Certificate.pdf</a>

## **Representative publications (on medium):**

<https://medium.com/@PushkarevValeriyAndreevich>

1. Global optimization better than multistart in MatLab (without basins and so on)
2. Making Libs, Drivers and verilog endpoint generator based on vDSO (SMO) - call without syscall, or 50M messages per second on 4 cores. (better than anything in Linux core)
3. Making Libs, Drivers and verilog endpoint - making data dependency graph and multi-issue on hardware endpoint, make levels of executions and methods endpoints

<https://medium.com/@PushkarevValeriyAndreevich2>

1. Making selector that takes for 20% less space than in ucdavis