Coastal Virginia

5115 Hampton Boulevard
Norfolk, Virginia 23529

Cybersecurity Research and Innovation FY 2022

RFP #: COVACCI-21-02

Project Title: Towards Trustworthiness in Autonomous Vehicles

Project Abstract (no more than 250 words):

Autonomous vehicles (AVs) are one of the most complex software-intensive Cyber-Physical Systems (CPS). In addition to the basic car machinery, they are equipped with driving assistance mechanisms that use smart sensors and machine learning (ML) for environment perception, pathfinding, and navigation.

Even though tremendous progress has been made in advancing the safety and security of AVs, they are shown to be vulnerable to accidental and malicious faults that negatively affect their perception and control functionality and result in safety incidents.

Recent works have highlighted two major challenges in safety validation and assurance of AVs: (i) With the increasing use of specialized hardware accelerators (GPUs) for running ML-based perception algorithms, AV control systems have become susceptible to transient faults (soft errors) that can result in erroneous ML inference and unsafe decision making and control. (ii) Safety assurance for AVs requires testing their resilience by identifying and simulating realistic safety-critical fault and attack scenarios by mining a tremendous fault space. To address these challenges, this project brings together a team of experts in GPU and CPS resilience from two CCI nodes to develop a holistic approach for end-to-end resilience assessment of AVs. We combine strategic fault injection at both hardware accelerator and controller software levels to assess the sensitivity of the ML components and control system to accidental or malicious faults and identify critical components and system states. The results from this project will make a firm step towards achieving trustworthiness in autonomous vehicles.

Total Requested Amount: 150,000.00

Project Investigators:

PI Name	Institution	Department	Phone Number	Email
Evgenia Smirni	William & Mary	Computer Science	757 221 3580	esmirni@cs.wm.edu
Homa Alemzadeh	UVA	ECE	(434) 924-6739	ha4d@virginia.edu

Towards Trustworthiness in Autonomous Vehicles

1. Rationale

Autonomous vehicles are one of the most complex software-intensive cyber-physical systems (CPS). In addition to the basic car machinery (e.g., gas/brake system, steering system), they are equipped with driving assistance mechanisms such as Adaptive Cruise Control (ACC), Lane Keeping Assist System (LKAS), and Assisted Lane Change. AVs use smart sensors (e.g., camera, RADAR, LIDAR) and artificial intelligence/machine learning (AI/ML) algorithms for perception of the surrounding environment, pathfinding, and navigation. With the increasing deployment of AVs on the road and the goal of moving towards full autonomy in the near future, reliability and safety of these systems are of highest importance.

The safe operation of an AV depends not only on the proper functioning of the sensors, actuators, and mechanical components, but also on the proper operation of the autonomous control software and its interactions with other components, human driver, and environment. Accidental or malicious faults targeting the AV controller may become activated at a critical operational time, further propagate into the system, and potentially result in safety hazards. Recent studies have highlighted the importance of end-to-end resilience assessment of AVs by considering the impact of faults and attacks that originate in sensing and perception components on the safe operation of control and decision-making algorithms [1, 2]. An analysis of data from the California Department of Motor Vehicles concluded that 64% of disengagements were the result of incorrect or untimely decisions made by the machine learning (ML) systems [3]. Several recent works have also shown the vulnerability of ML-based perception systems in AVs to adversarial attacks [4]. These studies highlight the fact that while individual AV components may have matured, AV safety hinges on the robustness of ML systems for perception and control [5].

ML systems on AVs rely on specialized hardware such as GPUs that pack a large number of computational units to provide magnitudes of higher throughput compared to traditional CPUs. Designed specifically for AVs, the NVIDIA Drive AGX Xavier and Pegasus systems on chip (SoC) deliver 30 and 320 trillion operations per second, respectively, and provide an environment for high throughput, low latency, and high availability. A recent study of catastrophic failures in AVs points at transient faults (soft errors) that are caused by cosmic radiation and low voltage settings and can result in erroneous AI/ML inference [5].

The overall goal of this project is to enhance the resilience and trustworthiness of AVs by addressing two intertwined challenges in safety and security validation and assurance:

Trustworthiness of AI/ML GPU Accelerators: As GPUs are becoming increasingly susceptible to transient hardware faults (soft errors) their reliable operation is of critical importance. Transient hardware faults can lead to bit flips in storage devices including the register file and DRAM. If bit flips occur during application execution, they may result in application crashes/hangs. It has been recorded that even a single-bit flip in vehicle software can have catastrophic effects: the Bookout v Toyota Motor Corp. case established that a bit flip caused Toyota vehicles to inadvertently accelerate, the driver to lose control of the electronic throttle control system resulting in at least one case in wrongful death [6].

The operation of the object recognition systems in AVs[1] is of critical importance as the output from this component is fed as input into the control and decision making AV components. To assess the vulnerability of the object recognition system to bit-flips, we will develop a software fault injection methodology with a focus on AI/ML models based on Tensorflow/Pytorch. We will evaluate their sensitivity to single and multi-bit hardware faults and input errors. We will specifically

[1] In this work, we only consider faults (i.e., bit flips) during the inference phase of the ML applications and not during training. We assume that training is a one-time process and that no faults occur during training.

focus on popular ML model optimizations such as weight compression, weight pruning, and weight quantization, and identify the critical software and hardware components that hamper the trustworthiness of ML models executing on GPUs. In addition, we will explore low-overhead techniques to improve the reliability and trust of ML inference models.

End-to-end Resilience Assessment of AV Control Systems: Current practice in safety validation of AVs involves simulation testing using realistic scenarios in virtual environments or real-world testing on the road. Besides the tremendous cost for developing high-fidelity testbeds, safety-critical faults resulting in significant hazards are rare and, thus, such test experiments require hundreds of millions of miles of driving to discover safety-critical problems [7]. Besides, the real-world road testing has the high risk of causing harm to drivers and pedestrians as reported in the recent fatal incidents involving AVs by Tesla and Uber [8]. International safety standards (e.g., ISO 26262) and previous works [1,9] have emphasized the importance of fault-injection and fuzz testing to evaluate the performance of safety mechanisms in AVs. Traditional fault injection techniques do not focus on end-to-end resilience and their impact on safety, and often require significant amount of time under test [1,2]. So, an open challenge is identifying and simulating the specific safety-critical fault and attack scenarios that can accurately represent real-world settings and, if happen, can get propagated into the system and lead to safety hazards and incidents.

We will study the impact of accidental and malicious faults in sensors, actuators, hardware, and software on the safe operation of AVs by identifying and simulating the most salient safety-critical AV scenarios. To reduce the fault injection space while maintaining high coverage of hazards, we will develop a novel fault injection technique that combines ML and program analysis with the domain knowledge of AV operation from functional and safety specifications to identify the most critical targets and system conditions in which the occurrence of faults might lead to incidents.

1.1. Intellectual Merit

The intellectual merit of this work lies in introducing a holistic methodology for improving AV resilience by combining strategic fault injection at both hardware accelerator (Task 1) and controller software (Task 2) levels.We will identify how ML optimizations affect the robustness of object recognition in the presence of single and multi-bit faults and under what conditions the effect of such faults can impact the performance of the controller software. This resilience analysis will lead to defining new, improved optimizations for image recognition that remain fast without compromising accuracy. More broadly, this project will assess the sensitivity of the ML components and end-to-end control system to accidental or malicious faults to identify the critical components and system states and gain a better understanding of effects of faults and their likely propagation paths within the system. This will guide the design of mechanisms for timely detection and mitigation of hazards, resulting in prevention of catastrophic accidents and improving trust in AVs.

1.2. Broader Impact

As AVs that operate in fully autonomous or semi-autonomous modes are slowly becoming mainstream, there is a clear need to provide the steps towards achieving the standards for functional safety of AVs, especially given the high throughput, low latency, and high reliability requirements of their individual components but also their smooth and highly reliable integration. The vulnerability analysis and safety assessment methods proposed by this research will directly apply to the critical AV technologies in the transportation sector which is the focus of this COVA CCI call. The tools, methods, testbeds, and datasets from this research will be made publicly available to the research community. While the focus in this work is on AVs, similar fault and attack models as well as safety assessment methods and tools may apply to cyber-physical and autonomous systems in other safety-critical domains such as health care. The fortification of object detection inference will have broader applicability in areas such as image recognition in robotics or for security.

Commonwealth Workforce Development: Through this research, the PIs will make a tangible impact on workforce development in two CCI nodes: William and Mary (Coastal Virginia) and UVA (Central Virginia). The PIs have a strong track record of integrating research into undergraduate and graduate classes. This project will provide case studies and project ideas for their classes. PI Smirni teaches an undergraduate class on simulation and a graduate class on data analysis, that provide the basics of reliability. She has an excellent history advising female M.S. and Ph.D. students (50% of her graduated Ph.D.s are female, she currently advises two female Ph.D. students). PI Alemzadeh teaches both undergraduate and graduate courses on "Dependable Computing Systems" and "Real-time Embedding Systems" at UVA. Alemzadeh has been very active in mentoring undergraduate students and promoting undergraduate research experiences. She has worked with over 24 undergraduate researchers, with the majority of them from minority and underrepresented groups, which resulted in publications and best poster awards. The PIs plan to continue their mentoring and outreach activities as part of this project.

2. Proposed Research Plan

2.1. Task 1: Assessing Trustworthiness of GPU-Accelerated ML Inference

The effect of single bit flips in application execution may have three outcomes: correct application output (i.e., the bit flip is *masked*), application hang or crash where the bit flip becomes instantly discernible, or can result in silent data corruption (SDC) where the application successfully completes execution but its output is incorrect. SDC outcomes are most undesirable as they erroneously provide the user with the illusion of correct output. For ML object recognition, an SDC may result in misclassification (e.g., a STOP sign may be recognized as some other sign, a truck may be recognized as something as innocuous as a bird), resulting in catastrophic decisions. In addition, bit flips may happen as a result of malicious attacks that use a side channel to find a critical time to deliberately cause same catastrophic outcomes. In HPC environments, protection/correction techniques (i.e., duplication/triplication) have been proposed to ensure application execution with output that is always correct [10] but typically suffer from high overhead and performance cost, making them impractical for the high-throughput, low latency requirements of image inference within the AV domain.

Smirni has developed a state-of-the-art mechanism for evaluating the resilience of HPC applications by dramatically pruning the fault space (from billions to a few hundred/thousand sites) without compromising prediction accuracy [11]. This pruning is based on the fact that thread resilience strongly depends on its dynamic instruction (DI) count. As multiple threads within an application have the same DI count, a representative thread can be used as a proxy. This technique has been applied (with superbe results) to multi-bit faults [12], resilient GPGPU application execution [10], and extrapolation of application resilience for different input sizes [13].

In this work, we propose to leverage the fault-site pruning methodology for investigating the reliability of ML inference models. Specifically, we will focus on the following:

- We will evaluate existing fault injection frameworks for ML applications and specifically the PyTorch framework [14] and the BinFI framework [5] and evaluate whether our fault-site pruning methodology can be used for fault injection in ML models. We will explore the effect of single-bit and multi-bit faults as well as the effect of intermittent faults in ML inference. This will be critical to evaluate the effect of *timing* of fault occurrence as the output of ML inference becomes the input to other AV components.
- Our ongoing work on the effectivenss of matrix-matrix/matrix-vector multiplication (note that such operations are in the heart of NNs that are used for image inference). In our ongoing work we have established that the range of values in the input (specifically, the norm of the input matrices) greatly affects application vulnerability to single-bit flips as they influence

the percentage of outputs that are SDC. We will explore the range of values in trained ML systems affect the norm of matrices and how these values propage across the various layers.
- Using the ImageNet data set (a popular image inference data set), GTSRB (a real world traffic sign data set), and Driving (a data set specifically for AVs), we will evaluate the classification accuracy of several ML models including VGG19, ResNet50, and InceptionV3. We will focus on the effect of *weight quantization*, where for weight parameters are *quantized* from the 32-bit floating point representation to a smaller data type such as 8-bit integer.
- The above analysis will highlight where NNs are more vulnerable and whether it is possible to "fortify" them using techniques such as those reported by [10] for generic GPU applications.

Evaluation: We will conduct experiments using NVBitFI[2], a fault injection framework by NVIDIA. We will evaluate the effects of single and multi bit faults on inference by comparing them to a golden output. A major part of our focus would be on resilience fortification methods that do not compromise on the quality of inference. The vulnerable locations within the ML model identified in this task will be used as one of the fault injection scenarios (faulty perception output) in Task 2.

2.2. Task 2: Strategic Fault Injection for Safety and Security Validation

Fault injection is the process of deliberately introducing faults in the system by "corrupting values of variables" or "corrupting software or hardware components" to analyze the system behavior and assess the performance of fault-tolerance and safety mechanisms in presence of faults and attacks [15]. Each injection experiment requires specification of the target *locations* (specific file, function, and variable), values, and *triggers* (time or system states or conditions) for injecting the fault. Because of the complexity of the control software/hardware and Ml models, and the unexpected inputs and environment, the test space for fault injection testing of AVs is often huge. For example, an open-source AV control software (OpenPilot by Comma.Ai) which we analyzed in our previous work [1] consists of a repository of over 150 python files, with over 22,000 lines of code. It will take enormous amount of time (e.g., over 600 days for about 100K injections [2]) for a traditional random fault injection approach to exhaustively cover all locations and conditions for simulation of all possible fault scenarios. Further, a large portion of randomly injected faults do not get activated, do not manifest as error because of getting masked by the system logic or existing safety mechanisms, or if manifested, do not lead to any unsafe system states or accidents [1].

To address this challenge, Alemzadeh has developed a strategic fault injection method where the *times or conditions* for injection are determined based on the unsafe control actions and critical system context identified from the hazard analysis process. This is driven by the observation that safety hazards and accidents are context-dependent [16] and occur as a result of specific control actions issued by the AV controller in a given system state. For example, if the AV controller provides an unintended acceleration command when the distance to the lead vehicle is less than a safety threshold, then safety distance will be violated and a crash incident might happen. The strategic fault injection method uses a systems-theoretic hazard analysis approach based on the AV control structure to identify the system contexts or the combinations of controller state variable values under which the unsafe control actions can occur and use those contexts as the triggers for fault injection [15]. The experiments on an open-source AV testbed have shown that using this approach compared to random fault injection, the probability of generating unsafe scenarios and identifying hazards increases and, thus, the fault injection space is reduced [1].

We plan to build upon this previous experience and further extend the strategic fault injection methodology with the ability to identify the most suitable target *locations* within the AV software for activating the faults and attacks and *values and durations* of faults that will most likely lead to safety hazards and accidents. We will specifically do the following:

[2]https://github.com/NVlabs/nvbitfi

- **Program analysis for identifying critical system targets:** Based on analysis of past real incidents, the most critical targets for accidental faults and malicious attacks in cyber-physical system are the inputs (sensors and perception system outputs (provided by Task 1), outputs (actuators), and the controller software/hardware stack. We will adopt static and dynamic program analysis techniques for automated mining of AV software repositories and reverse engineering of AV controller code to identify the location of controller command and feedback variables. Specifically, program slicing combined with natural language processing and system call profiling will be used to analyze the syntax and semantics of the code, and to identify the locations within the code (files, functions, variables, and memory addresses) that implement the controller logic and its interface with the sensors and actuators. We will use data clustering to group similar variables and statements in our analysis.
- **Machine learning for generating critical fault scenarios:** Having identified the critical software variables for emulating the effect of faults and attacks, we will develop an ML-based fault injection engine capable of finding the fault scenarios (represented by the values and duration of faults to be injected into one or more variables) that will most likely lead to safety violations. The proposed engine will rely on probabilistic reasoning about the behavior of the AV under a fault scenario through integrating domain knowledge and safety models (from functional specifications and hazard analysis) with pre-collected simulation traces from system and ML models (e.g., Neural Networks (NN) or Reinforcement Learning (RL))

Evaluation: We will use an open-source testbed for AV safety validation developed by Alemzadeh. This testbed integrates the OpenPilot PC simulator (by Comma.ai) with a computer vision based lane detection component and a compile-time software fault injection framework that mimics the effect of faults on the RADAR and car sensors and real-world environmental conditions impacting camera input (such as rain, fog, snow, occlusion, and blur) [1]. We will compare the performance of the proposed strategic fault injection approach in finding safety-critical faults to traditional random fault injection and recently proposed ML-fault injection [2, 4]. We will use metrics such as fault activation rate, hazard coverage (percentage of injected faults that resulted in hazard), and percentage of undetected hazards by the AV controller.

2.3. Outcomes and Milestones

Table 1 shows the division of tasks and milestones. Smirni and one graduate student will focus on Task 1. Alemzadeh and her students will work on Task 2. All participants will collaborate on the evaluation studies and the demonstration of results on the testbed.

Table 1: Project Timeline and Milestones

Milestone	Q1	Q2	Q3	Q4
Task 1: Trustworthiness of ML Inference Models	Smirni (S)	S	A, S	
Task 2: Strategic Fault Injection for AV Safety and Security	Alemzadeh (A)	A	A, S	
Data Collection and Writing Reports			A, S	A, S
Writing Proposal for External Grants				A, S

2.4. Alignment with CCI Objectives/Long-term Sustainability of the Collaboration

This proposal leverages the expertise of the two PIs on GPU reliability and GPU software resilience (Smirni, William and Mary, Coastal Virginia Node) and resilient cyber-physical systems (Alemzadeh, University of Virginia, Central Virginia Node). Both PIs are active in the general dependability area. They routinely serve on the program committee of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), the premier conference in the field. They have both been elected to the IFIP W.G. 10.4 on Dependable Computing and Fault Tolerance. Smirni serves on the steering committee of DSN and has served as technical program co-chair of DSN 2017. Alemzadeh received the 2017 William C. Carter PhD Dissertation Award in Dependability from the IEEE Technical Committee and the IFIP W.G. 10.4. They both routinely

publish at DSN with their graduate students. While the PIs work in relatively close domains, they have never collaborated before. This COVA CCI call provided the momentum for the PIs to collaborate on the intersection of GPU and CPS resilience with a focus on the security and safety challenges in the AV domain. We anticipate that the collaboration of the two CCI nodes will eventually culminate in a NSF proposal submission, either to the CCF CORE or the SaTC program.

2.5. Collaboration and Management Plan

PI Smirni will lead the coordination of the overall project. PI Alemzadeh will lead the demonstration exercises. Each PI will work closely with and mentor their graduate and undergraduate students, in addition to the collaborative efforts. The PIs and their graduate students will have weekly meetings via Zoom. Secure cloud services (e.g., Slack and Box) will be used for communication and sharing of project material. Upon lifting of COVID restrictions, the project personnel anticipate to meet on site (W&M/UVA) over the year.

3. Project Assessment

The overall goal of the proposed research is the development of a holistic approach for assessing the end-to-end resilience of AVs against faults and attacks. We plan to develop two fault injection methodologies and tools for simulating the effects of accidental/malicious faults at the hardware (GPU accelerators) level (Task 1) and controller software and input/output (Task 2).

Performance Metrics: We will evaluate the performance of the proposed fault-injection methods compared to traditional random fault-injection and previously proposed ML fault-injection methods on an AV testbed, by simulating representative driving scenarios and using metrics such as fault activation and manifestation rates, hazard coverage, detection coverage, and reaction time.

Success Criteria: Our functional testbeds and fault injection tools demonstrating the proposed methods will be disseminated to the community. Our findings will be also published at refereed conferences (e.g., IEEE DSN, SRDS, PRDC) and journals (e.g., IEEE TDSC).

References

[1] A. H. M. Rubaiyat, Y. Qin, and H. Alemzadeh, "Experimental resilience assessment of an open-source driving agent," in *23rd IEEE Pacific Rim International Symposium on Dependable Computing, PRDC 2018, December 4-7, 2018*, pp. 54–63, 2018.

[2] S. Jha *et al.*, "Ml-based fault injection for autonomous vehicles: A case for bayesian fault injection," in *49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2019*, pp. 112–124, 2019.

[3] S. S. Banerjee *et al.*, "Hands off the wheel in autonomous vehicles?: A systems perspective on over a million miles of field data," in *48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2018*, pp. 586–597, 2018.

[4] S. Jha *et al.*, "Ml-driven malware that targets av safety," in *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pp. 113–124, 2020.

[5] Z. Chen *et al.*, "Binfi: an efficient fault injector for safety-critical machine learning systems," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–23, 2019.

[6] "Toyota case: Single bit flip that killed." https://www.eetimes.com/toyota-case-single-bit-flip-that-killed/. Accessed: 2021-05-05.

[7] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?," *Transportation Research Part A: Policy and Practice*, vol. 94, pp. 182–193, 2016.

[8] B. Canis, "Issues in autonomous vehicle testing and deployment," *Congressional Research Service, Tech. Rep. R45985*, 2021.

[9] C. Hutchison *et al.*, "Robustness testing of autonomy software," in *2018 IEEE/ACM 40th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*, pp. 276–285, IEEE, 2018.

[10] L. Yang, B. Nie, A. Jog, and E. Smirni, "Enabling software resilience in gpgpu applications via partial thread protection," in *43rd International Conference on Software Engineering, 23-29 May 2021 (to appear)*, 2021.

[11] L. Nie, Bin Yang, A. Jog, and E. Smirni, "Fault site pruning for practical reliability analysis of gpgpu applications," in *2018 51st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pp. 749–761, IEEE, 2018.

[12] L. Yang, B. Nie, A. Jog, and E. Smirni, "Practical resilience analysis of gpgpu applications in the presence of single- and multi-bit faults," *IEEE Transactions on Computers*, vol. 70, no. 1, pp. 30–44, 2021.

[13] L. Yang, B. Nie, A. Jog, and E. Smirni, "Sugar: Speeding up gpgpu application resilience estimation with input sizing," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 5, no. 1, pp. 1:1–1:29, 2021.

[14] A. Mahmoud *et al.*, "Pytorchfi: A runtime perturbation tool for dnns," in *50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops, DSN Workshops*, pp. 25–31, 2020.

[15] H. Alemzadeh *et al.*, "Systems-theoretic safety assessment of robotic telesurgical systems," in *International conference on computer safety, reliability, and security*, pp. 213–227, Springer, 2014.

[16] X. Zhou, B. Ahmed, J. H. Aylor, P. Asare, and H. Alemzadeh, "Data-driven design of context-aware monitors for hazard prediction in artificial pancreas systems," *arXiv preprint arXiv:2104.02545*, 2021.

Coastal Virginia

Project Budget Worksheet

I. Proposal Information

a. Sponsor Name:	COVA CCI
b. RFP Name:	Cybersecurity Research and Innovation Funding
RFP Number:	COVACCI-21-02
c. Sponsor Deadline:	May 15, 2021
d. Project Name:	Towards Trustworthiness in Autonomous Vehicles
e. Overall PI Name/Institution:	Evgenia Smirni, William & Mary
f. Institutional PI/Co-PI Name/Institution:	Homa Alemzadeh, UVA
g. Project Start Date:	07/01/2021
h. Project End Date:	06/30/2022
i. Total Project Budget: Auto calculates from costs identified below.	150000

II. Labor Costs

a. Faculty: List total cost for all faculty supporting the project by academic institution. Include salary based on level of effort for each faculty.

Institution	Total Salary
William & Mary	35249
Total	$ 35249

b. Support Staff / Post-Docs / Research Scientists / Lab Managers / Hourly

List total cost for all support staff, etc., supporting the project by academic institution. Include salary and fringe benefit based on level of effort for each faculty. Each institution has their own fringe rate and this rate will be used in calculating total fringe.

Institution	Total Salary
Total	$ 0

c. Graduate Research Assistants (GRAs)

List total cost for graduate research assistances by institution.

Institution	Total
William & Mary	33750
Total	$ 33750

Include tuition in budget? (Select Yes or No) Yes No

d. Fringe Benefits: Include fringe benefit based on level of effort for each faculty and staff. Each institution has their own fringe rate and this rate will be used in calculating total fringe. ODU Fringe rate for CCI funded projects is 23.25% + 12,000

Calculations: Annual Salary x Fringe Rate (.2325 + 12000) x % of effort = Total Fringe

Institution	Total
William & Mary	3730
Total	$ 3730

III. Other Direct Costs

Use the table below to plan other expenses are needed to accomplish the project. Include institution receiving funding in the description along with a brief description of the item requested.

Category/Description	**Total Budget**
a. **Equipment**	
b. **Materials & Supplies**	
c. **Travel**	
	2000
d. **Subcontract & Consultant**	
UVA	69993
e. **Other Expenses**	
	5278
Total Other Direct Costs	$ 77271

IV. Subcontractors / Consultants

If you will need any subcontractors or consultants for your project, please provide the following information.

Company Name:	Contact Name/e-mail:
UVA	PI Homa Alemzadeh/ha4d@virginia.edu

V. Budget Summary

Include a budget summary showing total costs by category as calculated above. All cells in this table will auto calculate.

Category	**Amount**
II. Labor Costs	
a. Faculty	$ 35249
b. Support Staff	$ 0
c. Graduate Research Assistants	$ 33750
d. Fringe Benefits	$ 3730
Total Labor Costs	**$** 72729
III. Other Direct Costs	
a. Equipment	$ 0
b. Materials & Supplies	$ 0
c. Travel	$ 2000
d. Subcontract/Consultant	$ 69993
e. Other Expenses	$ 5278
Total Other Direct Costs	**$** 77271
Total Budget	**$** 150000

VI. Indirect Costs or Facilities & Administrative (F&A) Costs

CCI funds CAN NOT be used for indirect costs or facilities and administration costs. Nodes may offer unrecovered F&A as the basis for matching funds, separately calculated for each of the four broad categories: Research, Regional Innovation Ecosystem, Talent Pipeline, and Operations .

Indirect costs or F&A costs are real expenses that cannot be readily identified with certain activities. Examples include library costs, utility costs, costs to operate and maintain buildings and grounds, and costs of administering sponsored projects. Indirect costs are determined by each institution and these rates will be used to calculate total IDC for the project. The IDC rate is applied to the total direct costs of the budget minus individual equipment costing $5,000 or more; subcontract costs in excess of $25,000; graduate student tuition; rental costs; and participant support costs.

Current IDC rate for Norfolk State
Current IDC rate for Old Dominion University
Current IDC rate for William & Mary

VII. Matching Funds/Cost Sharing

Does the RFP or solicitation require matching YES
NO

Include indirect costs as part of the Matching Funds/Cost Sharing.
Identify Funding Source for match.

Category/Funding Source	Amount
Cash Match	
IDC/F&A costs	97711
Faculty Release Time	52289
Federal Grant	
Other	
Other	
Other	
Total	**$** 150000

IX. Contact Information

Send this form to jcostanz@odu.edu

BUDGET JUSTIFICATION

"Towards Trustworthiness in Autonomous Vehicles"

PI, Evgenia Smirni, W&M

Labor Costs

W&M PI Smirni, 1.84 summer month $35,249

Dr. Evgenia Smirni, PI, (AY @ $164,080 * expected 5% increase) will contribute 1 month summer (CCI request), as well as 1.35 months academic year (match).

Graduate Student, 12 months $33,750

Graduate Student salary is calculated on current rate established by the Department of Computer Science; effort is 50% during academic year and 100% during summer.

Fringe

FICA, 7.65%, on summer PI and student salaries $3,730

Other

Travel $2,000

Trips to related technical conferences, workshops, seminars, etc. Trips to collaborator for technical discussions and presentation of results.

Subcontract, UVA $69,993

Budget and justification are attached.

Graduate student tuition/fees $2,663

Tuition is $1,855; fees are $808

Graduate student health $2,615

William & Mary's federally negotiated indirect cost rate is 44% MTDC (Office of Naval Research, date of agreement July 1, 2020). Indirect costs are not eligible costs for this RFP; unrecovered F&A will be used to partially fulfill the CCI required contribution detailed in the CCI budget form and commitment letter.

Cost-share letters of commitment are attached.

University of Virginia Budget Detail

	CCI Funds 07/01/2021 06/30/2022	UVA Cost Match 07/01/2021 06/30/2022	Total Budget
Homa Alemzadeh -9 month	**11,289**	**16,582**	**27,871**
fringe benefits @ 6.5% Smr & 27.7% AY	734	4,593	5,327
AY effort - 1.47 AY Mos.	0.00%	16.32%	
Summer effort	33.33%	0.00%	
Graduate Research Assistant (ECE)	**31,538**	0	31,538
CY effort	100.00%	0.00%	
Undergraduate Research Assistant			
40 hrs. mo. x 9 mos. AY x $15.00/hr.	5,400	0	5,400
80 hrs. mo. x 3 mos. smr. x $15.00/hr.	3,600	0	3,600
Fringe Benefits - 6.5% of Summer Only	234	0	234
Subtotal Personnel	**51,827**	**16,582**	**68,409**
Subtotal Benefits	**968**	**4,593**	**5,561**
Travel			
Travel to Conferences and Collaborator	2,280	0	2,280
Other Costs			
Tuition Remission - Research Only	11,789	0	11,789
Health Insurance	3,129	0	3,129
Total Direct Costs	**69,993**	**21,175**	**91,168**
Indirect Cost Contribution (Unrecovered IDC)	0	35,795	35,795
**While indirect costs (IDC) are NOT eligible costs in the proposals submitted in response to this RFP, unrecovered IDC can be used to fulfill this contribution requirement.*			
Indirect Cost Contribution (Match IDC)	0	13,023	13,023
Total	**69,993**	**69,993**	**139,986**

Budget Justification

Personnel - Dr. Homa Alemzadeh, PI, (AY@$101,600 plus increases) will contribute 1 month summer in (CCI Request) and 1.47 Academic Year months (UVA Match) during this twelve month project.

Graduate Research Assistants (GRAs) and Undergraduate Research Assistants (URAs) - Costs are estimated based on the minimum and maximum payments for the academic year established by the University Office of the Vice-President and Provost. All compensation in SEAS proposals are within these guidelines. Per UVa policy, GRAs and URAs are limited in the number of hours they can work while taking classes, therefore to calculate hourly rates conversions are made by applying 1056 (GRA) and 840 (URA) hours per calendar year. The support provided for GRAs also includes tuition and health insurance shown below as Other costs. 1 GRA (CY@$31,538) @ 100% (88 hrs. mo. x 12 months). 1 UGRA @ 40 hrs. mo. x 9 mos. AY and 80 hrs. mo. x 3 mos. summer x $15.00/hr.

Fringe Benefits - The University of Virginia's fringe benefits rates as they apply to sponsored programs are as follows: 27.7% for faculty, research staff, postdoctoral fellows, 37.4% for classified staff, university staff and professional staff, 6.5% for hourly staff, temporary employees and wage employees. Fringe Benefits can include: FICA/Medicare, Retirement, Disability Insurance, Life Insurance, TIAA/CREF, Workers' Compensation, Unemployment Insurance and Health Insurance.

Travel - Trips to related technical conferences, workshops, seminars, etc. Trips to collaborator for technical discussions and presentation of results.

Other –

a. Tuition Remission - Effective September 1, 1990, it is the policy of the University of Virginia to provide tuition for graduate research assistants as partial compensation for services. Research Only Tution - $11,789.

b. Graduate Research Assistant Health Insurance – Effective July 1, 2005, it is the policy of the University of Virginia to provide health insurance for graduate research assistants as partial compensation for services. Health insurance is increased 5% each year to cover future rate increases. $3,129.

Facilities and Administrative (F&A) (Indirect/Overhead) Costs - The University of Virginia's negotiated (Modified Total Direct Costs (MTDC) F&A rates with DHHS, per agreement of January 29, 2021, is 61.5% "on campus", 26% "off-campus", and 38% for Other Sponsored Activities. (Note: The MTDC base consists of total direct costs less equipment, capital expenditures, charges for patient care, rental costs, tuition remission, scholarships and fellowships, participant support costs and the portion of each subaward in excess of $25,000. Includes F&A on the first $25,000 of subcontracts. Indirect costs (IDC) are NOT eligible costs in the proposals submitted in response to this RFP, unrecovered IDC can be used to fulfill this contribution requirement.

WILLIAM & MARY
CHARTERED 1693

OFFICE OF SPONSORED PROGRAMS

May 13, 2021

Mr. John Costanzo
Coastal Virginia Center for Cyber Innovation
Commonwealth Cyber Initiative
5115 Hampton Boulevard
Norfolk, VA 23529
jcostanz@odu.edu

Subject: Cost Share Commitment Letter, COVA CCI_21-02

Dear Mr. Costanzo,

The letter is to confirm and document our cost share for proposed project entitled, "Towards Trustworthiness in Autonomous Vehicles," PI Evgenia Smirni. The total cost share is valued at $80,007 and is detailed below:

- Academic year PI compensation: $25,952, 1.35 academic year months
- Academic year PI benefits: $5,162, 19.89% full benefit rate
- Unrecovered F&A, 44%: $48,893

The cost share will be used for this project for the period of performance from July 1, 2021 to June 30, 2022.

We are aware that if this proposal is awarded, we will be required to:

- Maintain internal records to document this cost share commitment; and
- Provide confirmation that the cost share commitment was met upon completion of the project.

We confirm this cost share commitment was not counted or documented for any other federal awards.

Sincerely,

Katherine Davis Small
Director, Office of Sponsored Programs

UNIVERSITY of VIRGINIA

SCHOOL of ENGINEERING & APPLIED SCIENCE

Office of Preaward Research Administration
ena-opra@virginia.edu

May 13, 2021

Elizabeth A. Montalvo (Liz), CRA
Senior Sponsored Programs Administrator
Office of Sponsored Programs
William & Mary
757-221-3901
Email: eamont@wm.edu

Dear Ms. Montalvo,

Dr. Homa Alemzadeh, Department of Electrical and Computer Engineering, School of Engineering and Applied Science at the University of Virginia, will be available to provide support to project entitled "Towards Trustworthiness in Autonomous Vehicles" if selected for funding. We intend to collaborate and/or commit resources as detailed in the SOW of the proposal. The appropriate University officials have administratively approved the proposal.

The budget for the University of Virginia project is $69,993. The inclusive dates are expected to be July 1, 2021 – June 30 2022.

The University will provide $69,993 cost match in order to satisfy the required 1:1 cost match required per the Coastal Virginia Center for Cyber Innovation; Cybersecurity Research and Innovation Funding FY 2022 per the details in the budget.

The University of Virginia reserves the right to negotiate any terms and conditions as appropriate for a public educational institution conducting research as part of its core mission. For additional assistance or questions on this submission, contact Stephen Cornelison, Sr. Director of Engineering Research Administration, at (434) 297-7402 or e-mail ena-opra@virginia.edu. **NOTE: Contractual information and negotiations should be addressed to, Stewart P. Craig, Executive Director, Office of Sponsored Programs, P.O. Box 400195, Charlottesville, Virginia, 22904-4195 or phone (434) 924-4270 or e-mail (ospnoa@virginia.edu).**

Sincerely,

Stephen Cornelison 8
Digitally signed by Stephen Cornelison 8
Date: 2021.05.13 14:15:52 -04'00'

Stephen Cornelison
Sr. Director, Engineering Research Administration
Legal Name: The Rector and Visitors of the University of Virginia
DUNS No. 065391526
TIN No. 54-6001796
CAGE Code: 9B982
District: VA-005

Arif Karim, Acting Director
Division of Cost Allocation
7700 Wisconsin Avenue, Suite 2300
Bethesda, MD 20857
Phone: 301-492-4855

Evgenia Smirni

Department of Computer Science,
William and Mary,
Williamsburg, VA 23185
(757) 221-3580
esmirni@cs.wm.edu

Professional Preparation:

1. Vanderbilt University, Nashville, TN, Computer Science, Ph.D., May 1995.
2. Vanderbilt University, Nashville, TN, Computer Science, M.Sc., May 1993.
3. University of Patras, Patras, Greece, Computer Engineering and Informatics, B.Sc., Jan. 1987.

Appointments:

1. Sidney P. Chockley Professor of Computer Science, William and Mary, Williamsburg, VA, August 2008 to present.
2. Visiting Professor, Monash University, Melbourne, Australia, December 2019-February 2020
3. Visiting Scientist, IBM Research, Zurich Lab, Rueschlikon, Switzerland, 2010-2011.
4. Associate Professor, Computer Science, William and Mary, Williamsburg, VA, 2002-2008.
5. Assistant Professor, Computer Science, William and Mary, Williamsburg, VA, 1997-2002.

Five Closely Related Publications:

1. G. Kadam, E. Smirni, A. Jog, “Data-centric Reliability Management on GPUs.”, in Proceedings of the 51th IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2021, (Taipei, Taiwn, online event), June 2021. Acceptance Rate: 16.5%. (to appear)
2. L. Yang, B. Nie, A. Jog, and E. Smirni, “SUGAR: Speeding-up GPGPU Application Resilience Estimation with Input Sizing”, Proceedings of the ACM on Measurement and Analysis of Computing Systems (SIGMETRICS 2021), 5(1) (2021): 129. Fall Deadline, Acceptance Rate: 12%. (to appear)
3. L. Yang, B. Nie, A. Jog, and E. Smirni, “Enabling Software Resilience in GPGPU Applications via Partial Thread Protection”, to appear in ICSE 2021, Acceptance Rate: 23%.
4. L. Yang, B. Nie, A. Jog, E. Smirni, “Practical Resilience Analysis of GPGPU Applications in the Presence of Single- and Multi-bit Faults”, in *IEEE Transactions on Computers*, 70(1): 30-44 (2021)
5. B. Nie, L. Yang, A. Jog, E. Smirni, “Error Site Pruning for Practical Reliability Analysis of GPGPU Applications”, in Proceedings of the 51st Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2018, Fukuoka City, Japan, October 2018, pp. 749-761. Acceptance Rate: 21%.

Five Other Significant Publications:

1. B. Nie, J. Xu, J. Alter, H. Chen, E. Smirni: "Mining Multivariate Discrete Event Sequences for Knowledge Discovery and Anomaly Detection", in Proceedings of the 50th IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2020: 552-563, Valencia, Spain, June 2020. Acceptance Rate: 16.5%.
2. J. Alter, J. Xu, A. Riska, E. Smirni, "SSD Failures in the Field: Symptoms, Causes, and Prediction Models", in Proceedings of Supercomputing 2019, Denver, CO, November 2019, pp.75:1-75:14. Acceptance Rate: 20%.
3. B. Nie, J. Xue, S. Gupta, T. Patel, C. Engelmann, E. Smirni, D. Tiwari, "Machine Learning Models for GPU Error Prediction in a Large Scale HPC System", in Proceedings of the 48th International Conference on Dependable Systems and Networks (DSN), Luxembourg City, Luxembourg, June 2018, pp. 96-106. Acceptance rate: 25%.
4. B. Nie, J. Xue, S. Gupta, C. Engelmann, E. Smirni, D. Tiwari, "Characterizing Temperature, Power, and Soft-Error Behaviors in Data Center Systems: Insights, Challenges, and Opportunities", in Proceedings of the *2017 IEEE 25th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Banff, Canada, September 2017, pp: 22-31.
5. B. Nie, D. Tiwari, S. Gupta, E. Smirni, J. H. Rogers, "A Large-Scale Study of Soft-Errors on GPUs in the Field", in Proceedings of the *22nd IEEE Symposium on High Performance Computer Architecture (HPCA 2016)*, Barcelona, Spain, March 2016, pp. 519-530. Acceptance Rate 22%.

Synergistic Activities:

Professor Smirni has been working in the area of Performance and Reliability for the past 25 years. She develops analytic techniques, explores the simulation or existing analytic techniques for the modeling of complex computer systems, and uses extensive experimentation to validate the proposed prediction models on real systems.

She has been named IEEE Fellow (January 2020) and ACM Distinguished Scientist (August 2013), she has been elected to IFIP W.G. 7.3 (October 2010) and to IFIP W.G. 10.4 (October 2020). She served on the ACM SIGMETRICS Board (elected position) from June 2011 to May 2015. She currently serves on the steering committee of DSN and ACM HPDC. For her work, she has received several best paper awards: IEEE Cloud 2015, International Teletraffic Congress 2010, ACM/IFIP/Usenix 9th International Middleware Conference (Middleware'08), 5th International Conference on Quantitative Evaluation of Systems (QEST'08), Internet Performance Symposium, IEEE GlobeCom 2002, and 9^{th} *International Conference on Modeling Techniques and Tools for Computer Performance Evaluation, TOOLS'97.*

Smirni served as the Program Vice Chair for the Cloud Computing and Data Center track for ICDCS 2021, as TPC co-chair for HPDC 2019 (the ACM 28th International Symposium on High-Performance Parallel and Distributed Computing), SRDS 2019 (the 38th IEEE International Symposium on Reliable Distributed Systems (SRDS 2019), Dependable Systems and Networks (DSN) 2017, the International Conference of Performance Engineering (ICPE) 2017, QEST'05, SIGMETRICS/Performance 2006, and HotMetrics'10. She has served as a general co-chair for QEST'10 and Numerical Solutions of Markov Chains (NSMC 2010).

Together with her students at William and Mary she has developed tools that have been made available to the community: the KPC-Toolbox is available at
`https://github.com/kpctoolboxteam/kpc-toolbox`

Homa Alemzadeh

Department of Electrical and Computer Engineering, University of Virginia
351 McCormick Road, PO Box 400743, Charlottesville, VA 22904-4743
Phone: (434) 924-6739, Email: alemzadeh@virginia.edu

Professional Preparation

University of Tehran, Iran	Computer Engineering	B.Sc.	2005
University of Tehran, Iran	Computer Engineering	M.Sc.	2008
University of Illinois at Urbana-Champaign	Electrical & Computer Engineering	Ph.D.	2016

Appointments

2017-present	Assistant Professor, Electrical and Computer Engineering, University of Virginia
2016-2017	Research Staff Member, IBM Research, Yorktown Heights, NY
2009-2016	Research Assistant, Coordinated Science Laboratory, University of Illinois (UIUC)

Most Relevant Publications:

- X. Zhou, B. Ahmed, J. H. Aylor, P. Asare, H. Alemzadeh, "Data-driven Design of Context-aware Monitors for Hazard Prediction in Artificial Pancreas Systems," To appear in: *Proc. 51st IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN)*, June 2021. arXiv preprint arXiv:2104.02545. (Acceptance Rate ~ 16.3%)
- M. S. Yasar, H. Alemzadeh, "Real-Time Context-aware Detection of Unsafe Events in Robotic Surgery", *Proc. 50th IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN)*, pp. 385-397, 2020. DOI: 10.1109/DSN48063.2020.00054. (Acceptance Rate ~ 16.5%)
- H. Lin, H. Alemzadeh, Z. Kalbarczyk, R. K. Iyer, "Challenges and Opportunities in Detection of Safety-Critical Cyber-Physical Attacks," *IEEE Computer Magazine*, vol. 53, no. 3, pp. 26–37, 2020. DOI: 10.1109/MC.2019.2915045. (Impact Factor: 1.94 - 2017).
- A. H. M. Rubaiyat, Y. Qin, H. Alemzadeh, "Experimental Resilience Assessment of An Open-Source Driving Agent", *Proc. 23rd IEEE Pacific Rim Int. Symp. on Dependable Computing (PRDC)*, pp. 54-63, 2018. DOI: 10.1109/PRDC.2018.00016.
- A. Wu, A. H. M. Rubaiyat, C. Anton, H. Alemzadeh, "Model Fusion: Weighted N-Version Programming for Resilient Autonomous Vehicle Steering Control," (Fast Abstract) *Proc. 29th IEEE Int. Symp. on Software Reliability Engineering Workshops (ISSREW)*, pp. 144-145, 2018. DIO: 10.1109/ISSREW.2018.00-11.

Most Significant Publications:

- K. Varshney, H. Alemzadeh, "On the Safety of Machine Learning: Cyber-Physical Systems, Decision Sciences, and Data Products," *Big Data Journal 5:3, 246–255,* 2017. DOI: 10.1089/big.2016.0051. (Impact Factor: 2.11 - 2018)
- H. Alemzadeh, R. K. Iyer, Z. T. Kalbarczyk, N. Leveson, J. Raman, "Adverse Events in Robotic Surgery: A Retrospective Study of 14 Years of FDA Data," *PLOS ONE*, vol. 11, no. 4: e0151470, 2016. DOI: 10.1371/journal.pone.0151470. (Impact Factor: 2.77 - 2018).
- H. Alemzadeh, D. Chen, X. Li, T. Kesavadas, Z. T. Kalbarczyk, R. K. Iyer, "Targeted Attacks on Teleoperated Surgical Robots: Dynamic Model-based Detection and Mitigation," *Proc. 46th IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN)*, 2016. DOI: 10.1109/DSN.2016.43. (Acceptance Rate ~ 20.5%)
- H. Alemzadeh, D. Chen, A. Lewis, Z. Kalbarczyk, and R. K. Iyer, "Systems-theoretic Safety Assessment of Robotic Telesurgical Systems," *Proc. of 34th International Conference on*

Computer Safety, Reliability, and Security (SAFECOMP), 2015. DOI: 10.1007/978-3-319-24255-2_16. (Acceptance rate ~ 32%)

- H. Alemzadeh, R. K. Iyer, Z. Kalbarczyk, J. Raman, "Analysis of Safety-Critical Computer Failures in Medical Devices," *IEEE Security & Privacy*, vol. 11, no. 4, pp. 14-26, July-Aug. 2013. DOI: 10.1109/MSP.2013.49. (Impact Factor: 2.04)

Complete list of published work:

https://scholar.google.com/citations?user=sXpmLxUAAAAJ&hl=en.

Synergistic Activities

- **Mentoring:** Mentor and research adviser to several undergraduate students, advising capstone and independent undergraduate research projects as part of the *UNLEASH Undergraduate Research Program* at the University of Virginia and *Promoting Undergraduate Research in Engineering (PURE)* and *Women in Electrical and Computer Engineering (WECE)* programs at the University of Illinois.
- **Teaching:** Developed and taught a new graduate/senior undergraduate course at the University of Virginia, covering the fundamentals of dependable computing and fault-tolerance, covering topics on system reliability modeling and analysis, hardware, software, and network fault-tolerance, information redundancy, error detection and recovery in processors and distributed systems, and experimental dependability evaluation.
- **Outreach:** Student forum co-Chair of the *IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN) 2019* and U.S. publicity chair of the *ACM/IEEE Int. Conf. on Cyber-Physical Systems (ICCPS) 2019*.
- **Conference Organization:** Program co-chair and organizer of the *IEEE Int. Workshop on Dependable and Secure Machine Learning (DSN-DSML),* attached to *DSN 2018-2020,* and *IEEE Int. Workshop on Software Certification (WoSoCer),* attached to *ISSRE 2016-2019.*
- **Reviewer:** Technical program committee member of "Dependability" and "Cyber Physical Systems" conferences and workshops, including: *IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN) 2018-2020*, *ACM/IEEE Int. Conf. on Cyber-Physical Systems (ICCPS) 2019-2021*, *IEEE ICCPS Medical Cyber Physical Systems Workshop 2018-2019, IEEE Int. Symp. on Software Reliability Engineering (ISSRE) 2017.* Technical reviewer of several conferences and journals, including: *IEEE Transactions on Dependable and Secure Computing (TDSC), ACM Transactions on Cyber-Physical Systems (TCPS), IEEE Embedded Systems Letters (IEEE-ESL),* and *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS).*

Identification of Potential Reviewers:

Kun Sun, Associate Professor, CIS, George Mason, ksun3@gmu.edu
Haining Wang, Professor, ECE, Virginia Tech, hnw@vt.edu
Dimitrios Nikolopoulos, Professor, CS, Virginia Tech, dsn@vt.edu