C\ TYPE SYSTEM

WILL VEATCH

We define the System F_ω^{AL}, the basis for the C\ language. System F_ω^{AL} is system F_ω extended with affine and linear kinds. That is, we have have kinds $\star$ for unrestricted types, $\bullet$ for affine types, and $\circ$ for linear types, with the following subkinding relation: $\star \leq \bullet \leq \circ$. Alternatively, System F_ω^{AL} is System F°, described in [1], extended with type functions and an affine kind. We take inspiration from [3, 2].

$$\kappa ::= \star \mid \bullet \mid \circ \mid \kappa \Rightarrow \kappa \qquad \qquad \text{kinds}$$
$$\tau ::= \alpha \mid \tau \xrightarrow{\kappa} \tau \mid \forall\alpha{:}\kappa.\,\tau \mid \lambda(\alpha:\kappa).\,\tau \mid \tau\,\tau \qquad \qquad \text{types}$$
$$e ::= x \mid \lambda^\kappa(x:\tau).\,e \mid e\,e \mid \Lambda(\alpha:\kappa).\,v \mid e[\tau] \qquad \qquad \text{expressions}$$
$$v ::= \lambda^\kappa(x:\tau).\,e \mid \Lambda(\alpha:\kappa).\,v \qquad \qquad \text{values}$$

Date: May 23, 2024.

$$\frac{\Gamma \vdash \tau : \star}{\Gamma \vdash \tau : \bullet} \text{ [K-Sub}_1] \qquad \frac{\Gamma \vdash \tau : \bullet}{\Gamma \vdash \tau : \circ} \text{ [K-Sub}_2] \qquad \frac{}{\vdash \cdot} \text{ [Empty]}$$

$$\frac{\vdash \Gamma \qquad \alpha \notin \Gamma}{\vdash \Gamma, \alpha : \kappa} \text{ [U-TExt]} \qquad \frac{\alpha : \kappa \in \Gamma}{\Gamma \vdash \alpha : \kappa} \text{ [U-TVar]} \qquad \frac{\Gamma, \alpha : \kappa \vdash \tau : \kappa'}{\Gamma \vdash \forall \alpha : \kappa. \, \tau : \kappa'} \text{ [All]}$$

$$\frac{\Gamma \vdash \tau_1 : \kappa_1 \qquad \Gamma \vdash \tau_2 : \kappa_2}{\Gamma \vdash \tau_1 \xrightarrow{\kappa} \tau_2 : \kappa} \text{ [K-Arr]} \qquad \frac{\Gamma, \alpha : \kappa_1 \vdash \tau : \kappa_2}{\Gamma \vdash \lambda(\alpha : k_1). \, \tau : \kappa_1 \Rightarrow \kappa_2} \text{ [K-Lam]}$$

$$\frac{\Gamma \vdash \tau_1 : \kappa_2 \Rightarrow \kappa_1 \qquad \Gamma \vdash \tau_2 : \kappa_2}{\Gamma \vdash \tau_1 \tau_2 : \kappa_1} \text{ [K-App]}$$

$$\frac{\Gamma_1, \Gamma_2; \Delta; \Xi \vdash e : \tau' \qquad \Gamma_1 \vdash \tau : \star}{\Gamma_1, x : \tau, \Gamma_2; \Delta; \Xi \vdash e : \tau'} \text{ [U-Wkg]}$$

$$\frac{\Gamma; \Delta_1, \Delta_2; \Xi \vdash e : \tau' \qquad \Gamma; \Delta_1 \vdash \tau : \bullet}{\Gamma; \Delta_1, x : \tau, \Delta_2; \Xi \vdash e : \tau'} \text{ [A-Wkg]}$$

$$\cdot \uplus \cdot = \cdot \text{ [AU-Empty]} \qquad \frac{\Delta_1 \uplus \Delta_2 = \Delta \qquad x \notin \Delta}{\Delta_1, x : \tau \uplus \Delta_2 = \Delta, x : \tau} \text{ [AU-Left]}$$

$$\frac{\Delta_1 \uplus \Delta_2 = \Delta \qquad x \notin \Delta}{\Delta_1 \uplus \Delta_2, x : \tau = \Delta, x : \tau} \text{ [AU-Right]} \qquad \cdot \uplus \cdot = \cdot \text{ [LU-Empty]}$$

$$\frac{\Xi_1 \uplus \Xi_2 = \Xi \qquad x \notin \Xi}{\Xi_1, x : \tau \uplus \Xi_2 = \Xi, x : \tau} \text{ [LU-Left]} \qquad \frac{\Xi_1 \uplus \Xi_2 = \Xi \qquad x \notin \Xi}{\Xi_1 \uplus \Xi_2, x : \tau = \Xi, x : \tau} \text{ [LU-Right]}$$

$$\frac{x : \tau \in \Gamma}{\Gamma; \cdot; \cdot \vdash x : \tau} \text{ [U-Var]} \qquad \Gamma; x : \tau; \Xi \vdash x : \tau \text{ [A-Var]}$$

$$\Gamma; \Delta; x : \tau \vdash x : \tau \text{ [L-Var]}$$

$$\frac{\Gamma \vdash \tau : \star \qquad x \notin \Gamma; \Delta; \Xi}{[\Gamma; \Delta; \Xi], x : \tau \ni \Gamma, x : \tau; \Delta; \Xi} \text{ [U-Ext]}$$

$$\frac{\Gamma \vdash \tau : \bullet \qquad x \notin \Gamma; \Delta; \Xi}{[\Gamma; \Delta; \Xi], x : \tau \ni \Gamma; \Delta, x : \tau; \Xi} \text{ [A-Ext]}$$

$$\frac{\Gamma \vdash \tau : \circ \qquad x \notin \Gamma; \Delta; \Xi}{[\Gamma; \Delta; \Xi], x : \tau \ni \Gamma; \Delta; \Xi, x : \tau} \text{ [L-Ext]}$$

$$\frac{[\Gamma; \cdot; \cdot], x : \tau_1 \ni \Gamma'; \Delta'; \Xi' \qquad \Gamma'; \Delta'; \Xi' \vdash e : \tau_2}{\Gamma \vdash \lambda^\star(x : \tau_1). \, e : \tau_1 \xrightarrow{\star} \tau_2} \text{ [U-Lam]}$$

$$\frac{[\Gamma; \Delta; \cdot], x : \tau_1 \ni \Gamma'; \Delta'; \Xi' \qquad \Gamma'; \Delta'; \Xi' \vdash e : \tau_2}{\Gamma; \Delta \vdash \lambda^\bullet(x : \tau_1). \, e : \tau_1 \xrightarrow{\bullet} \tau_2} \text{ [A-Lam]}$$

$$\frac{[\Gamma; \Delta; \Xi], x : \tau_1 \ni \Gamma'; \Delta'; \Xi' \qquad \Gamma'; \Delta'; \Xi' \vdash e : \tau_2}{\Gamma; \Delta, \Xi \vdash \lambda^\circ(x : \tau_1). \, e : \tau_1 \xrightarrow{\circ} \tau_2} \text{ [L-Lam]}$$

$$\frac{\Gamma; \Delta_1; \Xi_1 \vdash e_1 : \tau_1 \xrightarrow{\kappa} \tau_2 \qquad \Gamma; \Delta_2; \Xi_2 \vdash e_2 : \tau_1 \qquad \Delta_1 \uplus \Delta_2 = \Delta \qquad \Xi_1 \uplus \Xi_2 = \Xi}{\Gamma; \Delta; \Xi \vdash e_1 e_2 : \tau_2} \text{ [App]}$$

$$\frac{\Gamma, \alpha : \kappa; \Delta; \Xi \vdash v : \tau}{\Gamma; \Delta; \Xi \vdash \Lambda(\alpha : \kappa). \, v : \forall \alpha{:}\kappa. \, \tau} \text{ [TLam]}$$

$$\frac{\Gamma; \Delta; \Xi \vdash e : \forall \alpha{:}\kappa. \, \tau' \qquad \Gamma \vdash \tau : \kappa}{\Gamma; \Delta; \Xi \vdash e[\tau] : \tau'[\tau/\alpha]} \text{ [TApp]}$$

Evaluation rules:

$$(\lambda^\kappa(x : \tau). \, e)v \mapsto e[v/x] \qquad (\Lambda(\alpha : \kappa). \, v)[\tau] \mapsto v[\tau/\alpha] \qquad \frac{e_1 \mapsto e_1'}{e_1 e_2 \mapsto e_1' e_2}$$

$$\frac{e \mapsto e'}{ve \mapsto ve'} \qquad\qquad \frac{e \mapsto e'}{e[\tau] \mapsto e'[\tau]}$$

Existential types may be encoded in the current system. We (admissibly) adopt existential types as primitives.

$$\tau ::= \dots \mid \exists \alpha{:}\kappa. \, \tau' \qquad\qquad \text{types}$$
$$e ::= \dots \mid \mathsf{pack}(\tau, e) \mid \mathsf{let} \, \mathsf{pack}(\tau, e) = e_1 \, \mathsf{in} \, e_2 \qquad \text{expressions}$$
$$v ::= \dots \mid \mathsf{pack}(\tau, v) \qquad\qquad \text{values}$$

$$\frac{\Gamma, \alpha : \kappa \vdash \tau' : \kappa' \qquad \kappa \leq \kappa'}{\Gamma \vdash \exists \alpha{:}\kappa.\, \tau' : \kappa'} \;\; [\mathsf{Exts}]$$

$$\frac{\Gamma \vdash \tau : \kappa \qquad \Gamma, \alpha : \kappa \vdash \tau' : \star \qquad \kappa \leq \star \qquad \Gamma, \Gamma'; \cdot; \cdot \vdash e : \tau'[\tau/\alpha]}{\Gamma, \Gamma'; \cdot; \cdot \vdash \mathsf{pack}(\tau, e) : \exists \alpha{:}\kappa.\, \tau'} \;\; []$$

$$\frac{\Gamma \vdash \tau : \kappa \qquad \Gamma, \alpha : \kappa \vdash \tau' : \bullet \qquad \kappa \leq \bullet \qquad \Gamma, \Gamma'; \Delta; \cdot \vdash e : \tau'[\tau/\alpha]}{\Gamma, \Gamma'; \Delta; \cdot \vdash \mathsf{pack}(\tau, e) : \exists \alpha{:}\kappa.\, \tau'} \;\; []$$

$$\frac{\Gamma \vdash \tau : \kappa \qquad \Gamma, \alpha : \kappa \vdash \tau' : \circ \qquad \kappa \leq \circ \qquad \Gamma, \Gamma'; \Delta; \Xi \vdash e : \tau'[\tau/\alpha]}{\Gamma, \Gamma'; \Delta; \Xi \vdash \mathsf{pack}(\tau, e) : \exists \alpha{:}\kappa.\, \tau'} \;\; []$$

$$\frac{\begin{array}{c}\Gamma, \Gamma'; \cdot; \cdot \vdash e_1 : \exists \alpha{:}\kappa.\, \tau' \qquad [\Gamma, \alpha : \kappa, \Gamma'; \cdot; \cdot], x : \tau' \vdash e_2 : \tau_2 \\ \Gamma \vdash \tau_2 : \star \qquad \Gamma, \alpha : \kappa \vdash \tau' : \kappa' \qquad \kappa' \leq \star\end{array}}{\Gamma, \Gamma'; \cdot; \cdot \vdash (\mathsf{let\,pack}(\alpha, x) = e_1 \mathsf{\,in\,} e_2) : \tau_2} \;\; []$$

$$\frac{\begin{array}{c}\Gamma, \Gamma'; \Delta; \cdot \vdash e_1 : \exists \alpha{:}\kappa.\, \tau' \qquad [\Gamma, \alpha : \kappa, \Gamma'; \Delta; \cdot], x : \tau' \vdash e_2 : \tau_2 \\ \Gamma \vdash \tau_2 : \bullet \qquad \Gamma, \alpha : \kappa \vdash \tau' : \kappa' \qquad \kappa' \leq \bullet\end{array}}{\Gamma, \Gamma'; \Delta; \cdot \vdash (\mathsf{let\,pack}(\alpha, x) = e_1 \mathsf{\,in\,} e_2) : \tau_2} \;\; []$$

$$\frac{\begin{array}{c}\Gamma, \Gamma'; \Delta; \Xi \vdash e_1 : \exists \alpha{:}\kappa.\, \tau' \qquad [\Gamma, \alpha : \kappa, \Gamma'; \Delta; \Xi], x : \tau' \vdash e_2 : \tau_2 \\ \Gamma \vdash \tau_2 : \circ \qquad \Gamma, \alpha : \kappa \vdash \tau' : \kappa' \qquad \kappa' \leq \circ\end{array}}{\Gamma, \Gamma'; \Delta; \Xi \vdash (\mathsf{let\,pack}(\alpha, x) = e_1 \mathsf{\,in\,} e_2) : \tau_2} \;\; []$$

Evaluation rules:

$$\mathsf{let\,pack}(\alpha, x) = (\mathsf{pack}(\tau, v)) \mathsf{\,in\,} e_2 \mapsto e_2[\tau, v/\alpha, x] \qquad \frac{e \mapsto e'}{\mathsf{pack}(\tau, e) \mapsto \mathsf{pack}(\tau, e')}$$

$$\frac{e_1 \mapsto e_1'}{\mathsf{let\,pack}(\alpha, x) = e_1 \mathsf{\,in\,} e_2 \mapsto \mathsf{let\,pack}(\alpha, x) = e_1' \mathsf{\,in\,} e_2}$$

References

[1] Karl Mazurak, Aa Bb, and S. Zdancewic. "Lightweight linear types in System F°". In: Jan. 2010, pp. 77–88. DOI: 10.1145/1708016.1708027.

[2] J. Garrett Morris. "The best of both worlds: linear functional programming without compromise". In: *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming.* ICFP'16. ACM, Sept. 2016. DOI: 10.1145/2951913.2951925.

[3] Jesse Tov and Riccardo Pucella. "Practical Affine Types". In: vol. 46. Jan. 2011. DOI: 10.1145/1926385.1926436.